Are Analytical Techniques Worthwhile for Analog IC Placement?

Yishuang Lin*, Yaguang Li*, Donghao Fang*, Meghna Madhusudan†
Sachin S. Sapatnekar†, Ramesh Harjani†, Jiang Hu*
*Texas A&M University †University of Minnesota
{lionlin, liyg, donghao, jianghu}@tamu.edu; {madhu028, sachin, harjani}@umn.edu

Abstract—**Analytical techniques have long been a prevailing approach to digital IC placement due to their advantage in handling large-sized problems. Recently, they have been adopted for analog IC placement, an area where prior methods were mostly based on simulated annealing. However, a comparative study between the two classes of approaches is lacking. Moreover, the effectiveness of different analytical techniques is not clear. This work attempts to shed light on both issues by studying existing methods and developing a new analytical technique. Since prior analytical methods have not addressed circuit performance, a critical concern for automated analog layout, this work also extends the new analytical placer for performance-driven placement. Experiments on various test circuits show that for a conventional performance-oblivious formulation, the proposed analytical technique achieves $55\times$ speedup and 12% wirelength reduction compared to simulated annealing. For performance-driven placement, the proposed technique outperforms simulated annealing in terms of circuit performance, area, and runtime. Moreover, the proposed technique generally provides better solution quality than an alternative analytical technique.**

I. Introduction

Automated analog IC layout has been pursued by numerous research groups for decades. Undoubtedly, placement is a critical layout step that greatly affects circuit performance, area, power, etc. Placement rules and constraints can lead to better circuit performance [1]–[3]. Historically, most analog placement methods were based on simulated annealing and focused on handling geometrical constraints such as symmetry that are specific to analog ICs [4]–[8]. A few years ago, an analytical technique [9] based on digital placement [10] was proposed for addressing layout effects such as well proximity. Later, a similar analytical technique was introduced in [11] for exploiting overlap among different device layers. The same analytical framework was adopted in the MAGICAL open source project [12], [13]. However, these techniques do not explicitly include circuit performance in their optimization objective functions.

Analytical techniques have a much longer history of applications in digital IC placement, from quadratic placement in 1980's to later nonlinear programming (NLP) [14], [15]. The NLP-based approach has a widely recognized advantage – the capability of dealing with huge problem sizes, e.g., placement of millions of objects, which are often seen in digital designs. Prior efforts that adopt analytical placement for analog ICs seem to take for granted, without much justification, that the choice of this placement paradigm is desirable, even though analog circuits are usually much smaller than digital ones and may not specifically need analytical methods. Are analytical techniques superior to simulated annealing and necessary for analog placement?

An early work on analytical analog placement [9] performed a comparison with simulated annealing [16]. However, the comparison mainly serves to demonstrate the importance of including layout effects in objective functions and does not compare the effectiveness of analytic placement vs. simulated annealing. In [17], an Mixed Integer Linear Programming (MILP) based placement method was developed for emphasizing hierarchical designs in analog ICs. When compared with simulated annealing [5], it shows a few percent wirelength and area reduction with several times of runtime increase. Although MILP can be claimed as an analytical approach, it is significantly different from the mainstream nonlinear programming (NLP) based analytical techniques [14], [15]. In [11], the NLP-based analytical technique was compared with MILP [17] but the difference from simulated annealing is not clearly delineated. To the best of our knowledge, there has been almost no study that justifies why analytical NLP techniques may be useful for general analog placement or performs a comparison with simulated annealing based methods.

We attempt to fill the void by providing an introspective study on analytical analog placement. The *first part* of this study is focused on a conventional formulation that minimizes area and wirelength subject to non-overlap and analog geometric constraints. The main contributions in this regard include:

- An analytical analog placement method based on [11] is compared with simulated annealing (it is reasonable to treat [11] as a representative approach that covers [9] as both follow the same digital placement framework [10]). Our results indicate that not every analytical technique is superior to simulated annealing in solution quality.[1]
- A new analog placer, ePlace-A is developed, extending a state-of-the-art digital placer, ePlace [15]. This method is significantly different from [10]. The extension also includes a new detailed placement technique different from most previous methods [9], [11], [15], [16].
- ePlace-A is compared with both simulated annealing and [11] so that the effectiveness of different analytical techniques can be observed. Results show that ePlace-A outperforms both simulated annealing and [11].
- Parameters of the different methods are varied to demonstrate area-wirelength tradeoff and it is shown that the

[1] In theory, simulated annealing can always achieve the optimal solution if its runtime is sufficiently long. Our experiments use practical runtime limits for simulated annealing, which result in suboptimal solutions from the method.

advantage of ePlace-A is available for multiple Pareto-optimal points and is not limited to a specific setting.

The *second part* of this work develops an analytical approach to *performance-driven* analog placement. Although it is widely recognized that analog circuit performance can be seriously affected by the quality of placement [18], [19], most existing placement techniques are unable to directly optimize performance. Early efforts relied on simple performance models [18], transforming performance constraints to geometric constraints [20], or using geometric constraints correlated with performance [6], [21]. Recently, machine learning models [19], [22] have been explored for performance-driven analog placement, but so far, almost all of these methods are based on simulated annealing. The contributions of this part include:

- Graph Neural Network (GNN) guided performance-driven analytical analog placement techniques: to the best of our knowledge, this is the first research on performance-driven analytical analog placement.
- Comparisons with the latest performance-driven work [19], based on simulated annealing, demonstrating that our technique, built upon ePlace-A, shows significant improvement on area, circuit performance, runtime, and area-performance tradeoffs.

This work provides a better understanding of analytical techniques for analog placement, and advances the state of the art of analog placement for both the conventional performance-oblivious formulation and the performance-driven formulation.

II. Background on Analytical Placement

A modern analytical placement method typically consists of global placement based on NLP and a stage of legalization and detailed placement. The input to placement includes a set of n movable cells/devices V and a set of nets E. The location of the i-th cell is designated by its coordinates (x_i, y_i). Then, the placement decision variables form a vector $\boldsymbol{v} = (\boldsymbol{x}, \boldsymbol{y})^T = (x_1, \ldots, x_n, y_1, \ldots, y_n)^T$.

Global placement minimizes wirelength and overlap among cells, and can be formulated as the NLP problem

$$\min_{\boldsymbol{v}} \text{HPWL}(\boldsymbol{v}) + \lambda \cdot \text{Overlap}(\boldsymbol{v}) \qquad (1)$$

where λ is a weighting factor. The wirelength is estimated by Half-Perimeter Wire Length, $\text{HPWL}(\boldsymbol{v}) = \sum_{e \in E} \text{HPWL}_e(\boldsymbol{v})$, where HPWL of a net e is obtained by $\text{HPWL}_e(\boldsymbol{v}) = \max_{i,j \in e} |x_i - x_j| + \max_{i,j \in e} |y_i - y_j|$. The function $\text{Overlap}(\boldsymbol{v})$ measures the overlap area among all cells.

In their original forms, neither $\text{HPWL}(\boldsymbol{v})$ nor $\text{Overlap}(\boldsymbol{v})$ is differentiable, and a key ingredient of analytical placement is to approximate them with smooth and differentiable functions. $\text{HPWL}(\boldsymbol{v})$ can be smoothed by either Log-Sum-Exponential (LSE) function [10] or Weighted-Average (WA) function [15]. We adopt the WA function for smoothing $\text{HPWL}(\boldsymbol{v})$ in our analog placement: as shown in [23], it has smaller estimation error. Specifically, $\max_{i,j \in e} |x_i - x_j|$ is approximated by

$$\text{WA}_{ex}(\boldsymbol{v}) = \frac{\sum_{i \in e} x_i \exp(\frac{x_i}{\gamma})}{\sum_{i \in e} \exp(\frac{x_i}{\gamma})} - \frac{\sum_{i \in e} x_i \exp(\frac{-x_i}{\gamma})}{\sum_{i \in e} \exp(\frac{-x_i}{\gamma})} \qquad (2)$$

where γ is a parameter controlling the accuracy.

The $\text{Overlap}(\boldsymbol{v})$ function is smoothed by a bell-shaped function in NTUplace3 [10] and a potential energy function in ePlace [15]. Distinguished from most earlier approaches, ePlace performs a Fourier transform to obtain frequency-domain information in computing gradients of the potential energy function. Another distinction of ePlace is its use of Nesterov's method [24] for solving the NLP problem. While other variants of analytical global placement have been proposed, our review here is focused on the most prominent methods, NTUplace3 [10] and ePlace [15], as NTUplace3 is the basis of previous analog placement methods [9], [11] while our new development is based on ePlace, which is the state of the art for analytical placement.

Global placement is followed by legalization, which completely removes overlap among cells, and detailed placement, which fine-tunes the placement for further optimizing certain objectives. These steps can be performed either separately or in an integrated manner. In this regard, techniques for analog placement diverge from digital placement. The work of [9] developed an integrated technique based on network flow, while in [11], legalization is performed with area compaction followed by detailed placement for minimizing wirelength, and both steps are realized through linear programming (LP).

III. Overview of This Work

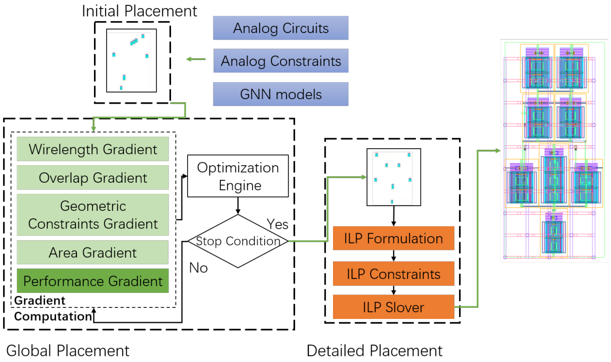


Fig. 1: Overview of ePlace-A and ePlace-AP.

This work is composed by two parts: (1) a study for **conventional analog placement** without explicitly considering performance, and (2) development for **performance-driven analog placement**. Both parts are focused on analytical techniques.

The problem formulation for the first part is to minimize total area and wirelength subject to geometric constraints specific to analog ICs. We extend ePlace [15], a state-of-the-art digital placer, to **ePlace-A** for analog IC designs. A major difference from ePlace [15] is that the legalization and detailed placement in ePlace-A are realized through Integer Linear Programming (ILP), whose formulation is very specific to analog circuits. In part (1), ePlace-A is compared with both simulated annealing and a recent previous work on analytical analog placement [11]. As [11] is based on a different digital placement method [10] and takes a significantly different approach to legalization and detailed placement from ePlace-A, the comparison covers multiple analytical techniques.

In part (2), we develop **ePlace-AP**, a performance-driven placement based on ePlace-A. Its objective function includes a performance term estimated by a GNN model [19]. A key element in this technique is the gradient computation for the GNN-based performance model. The legalization and detailed placement of ePlace-AP are the same as ePlace-A. An overview of ePlace-A and ePlace-AP is depicted in Figure 1.

IV. Analytical Techniques for Conventional Analog Placement

A. Global Placement of ePlace-A

The global placement (GP) step in ePlace-A is similar to ePlace [15], except that terms related to the analog geometric constraints and total layout area are added into the objective function. Using symmetry as a representative geometric constraint, the objective is formulated as

$$\min_{\boldsymbol{v}} W(\boldsymbol{v}) + \lambda N(\boldsymbol{v}) + \tau Sym(\boldsymbol{v}) + \eta Area(\boldsymbol{v}) \tag{3}$$

where $W(\boldsymbol{v})$ is the WA function for smoothed approximation of HPWL and $N(\boldsymbol{v})$ is the energy potential function for smoothed approximation of device overlap area (both defined in Section II), and λ, τ and η are weighting factors. Function $Sym(\boldsymbol{v})$ adds a penalty for violating symmetry constraints, e.g., for two devices i and j symmetric to a vertical axis at $x_{i,j}$, its corresponding term is $(y_i - y_j)^2 + (x_i + x_j - 2x_{i,j})^2$. As in [11], the symmetry constraints in global placement are soft. The post-detailed-placement results in Table I indicate that enforcing symmetry as hard constraints ($y_i = y_j; x_i + x_j = 2x_{i,j}$) in global placement increases both area and wirelength as compared to a solution using soft constraints. Other geometric constraints, such as device alignment and ordering, are also included in ePlace-A.

TABLE I: Soft vs. hard symmetry constraints in GP.

Design	Area(μm²)		HPWL(μm)		Runtime(s)	
	Soft	Hard	Soft	Hard	Soft	Hard
CC-OTA	100.3	117.5	31.4	34.3	0.22	0.28
Comp2	130.9	141.8	80.8	114.6	2.73	3.02
VCO2	516.4	535.7	304.1	320.2	0.94	1.15

Function $Area(\boldsymbol{v})$ in (3) is the total area estimated by $\text{WA}_{V,x}(\boldsymbol{v}) \cdot \text{WA}_{V,y}(\boldsymbol{v})$, where V is the set of all devices and $\text{WA}_{V,x}(\boldsymbol{v})$ ($\text{WA}_{V,y}(\boldsymbol{v})$) is the WA function for smoothed approximation of $(\max_{i,j \in V} |x_i - x_j|) \cdot (\max_{i,j \in V} |y_i - y_j|)$. While area is usually ignored in digital placement objective functions, it is explicitly considered for analog placement. The reason is that the number of circuit elements in an analog circuit is much smaller than a digital circuit so that the placement area has a greater impact on parasitics, which in turn significantly affect circuit performance. The post-detailed-placement results in Figure 2 show that neglecting the area term causes over 20% increase in area and wirelength. The analytical formulation in Problem (3) is solved in the same way as [15].

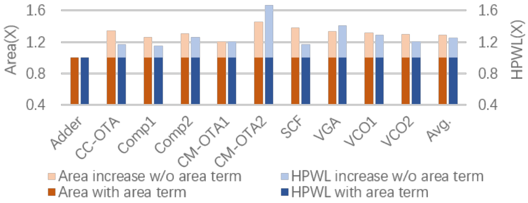


Fig. 2: Area and HPWL comparison for with and without the area term in the objective function.

B. Legalization and Detailed Placement of ePlace-A

We propose an integrated legalization and detailed placement method based on ILP and refer it as detailed placement for brevity. Although the detailed placement of [11] is based on LP (Linear Programming), the difference between our detailed placement and this previous work extends far beyond the integer constraint. The method of [11] is a two-stage approach, consisting of an area minimization stage followed by a wire-length minimization stage. In contrast, our approach is a single-stage integrated area and wirelength minimization. Moreover, our method supports the option of device flipping, which is not considered in [11]. Since our layout system is built on discrete grids, integer solutions are preferred. It is important to mention that although ILP does not scale well for large problems, the problem sizes of analog circuits are generally small, making an ILP solution tractable.

TABLE II: Notations for describing our detailed placer.

Notations	Descriptions
$\hat{x}_i, \hat{y}_i$	Coordinate of the pin of device i connected to net e.
$(\underline{x}_e, \underline{y}_e) - (\bar{x}_e, \bar{y}_e)$	Min bounding box of net e.
$(0,0) - (\mathcal{W}, \mathcal{H})$	Min bounding box of all devices.
w_i, h_i, s_i	Width, height and area of the i-th device.
$f_{xi}, f_{yi} \in \{0,1\}$	Horizontal and vertical flipping of the i-th device.
μ	Weighting factor between HPWL and area.
ζ	User-defined chip area utilization factor.
x_{pin_i}, y_{pin_i}	Pin offset from the lower-left corner of i-th device.

Some notations used for describing our method are listed in Table II. In addition, we define $\tilde{W} = \tilde{H} = \sqrt{\frac{\sum_{i=1}^{n} s_i}{\zeta}}$ as approximated constant width and height of the layout area.

Our ILP formulation is summarized below. If the same constraint exists for both x and y coordinates, only one of them is shown here for brevity. Device overlap and symmetry are handled in constraints instead of the objective function, as in the global placement formulation.

$$\min_{\boldsymbol{v}} \sum_{e \in E} \text{HPWL}_{De}(\boldsymbol{v}) + \mu \cdot \text{Area}_D(\boldsymbol{v}) \tag{4a}$$

$$= \sum_{e \in E} ((\bar{x}_e - \underline{x}_e) + (\bar{y}_e - \underline{y}_e)) + \mu \cdot \frac{\tilde{H} \cdot \mathcal{W} + \tilde{W} \cdot \mathcal{H}}{2}$$

$$\text{s.t. } \underline{x}_e \le \hat{x}_i \le \bar{x}_e, \underline{y}_e \le \hat{y}_i \le \bar{y}_e, \forall i \in e, \forall e \in E \tag{4b}$$

$$\frac{w_i}{2} \le x_i \le \mathcal{W} - \frac{w_i}{2}, \forall i \in V \tag{4c}$$

$$\hat{x}_i = x_i - \frac{w_i}{2} + x_{pin_i} \cdot (1 - f_{xi}) + (w_i - x_{pin_i}) \cdot f_{xi}, \forall i \in e, \forall e \in E \tag{4d}$$

$$x_j + \frac{w_j}{2} \le x_k - \frac{w_k}{2}, \forall (j,k) \in P^H \tag{4e}$$

$$\frac{x_{q_1} + x_{q_2}}{2} = x_r = x_m, \quad \forall (q_1, q_2) \in S_m^p, r \in S_m^s, \forall (S_m^p, S_m^s) \in S \tag{4f}$$

$$y_{b_1} - \frac{h_{b_1}}{2} = y_{b_2} - \frac{h_{b_2}}{2}, \forall (b_1, b_2) \in P^B \tag{4g}$$

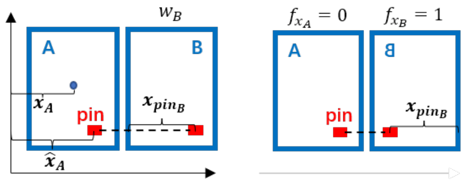
$$x_{vc_1} = x_{vc_2}, \forall (vc_1, vc_2) \in P^{VC} \tag{4h}$$

$$x_{o_1} + \frac{w_{o_1}}{2} \le x_{o_2} - \frac{w_{o_2}}{2}, \forall (o_1, o_2) \in O^H \tag{4i}$$

$$x_i \in \mathbb{N}, \forall i \in V, \quad \bar{x}_e, \underline{x}_e \in \mathbb{N}, \forall e \in E, \quad \mathcal{W}, \mathcal{H} \in \mathbb{N} \tag{4j}$$

The objective function (4a) covers area and HPWL. Since their estimations here are different from global placement, we use subscript D to distinguish them. Note that $\mathcal{W}$ and $\mathcal{H}$ are variables to be minimized. Constraints (4b) define the HPWL bounding boxes to be minimized for each net. Since coordinate x_i is at the center of the i-th device, constraints (4c) defines the upper bound $\mathcal{W}$ for layout width.

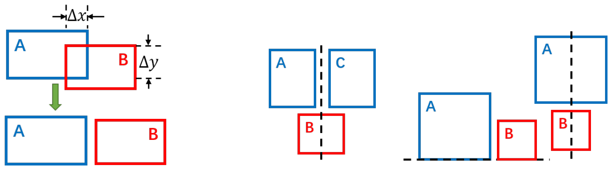
Constraint (4d) allows horizontal flipping, as decided by binary variable f_{x_i}. When $f_{x_i} = 0$, which implies no flipping, (4d) becomes $\hat{x}_i = x_i - \frac{w_i}{2} + x_{pin_i}$ and pin location $\hat{x}_i$ is x_{pin_i} away from the left boundary of device i. When $f_{x_i} = 1$, which implies flipping, (4d) becomes $\hat{x}_i = x_i + \frac{w_i}{2} - x_{pin_i}$ so that pin coordinate $\hat{x}_i$ is x_{pin_i} away from the right boundary. The example in Figure 3 compares device placements with and without flipping. Vertical flipping is realized in a similar way.



(a) No flipping for B. (b) Flipping device B.

Fig. 3: Flipping device B reduces wirelength between red pins.

If a pair of devices have overlap from global placement, width Δx and height Δy of the overlapping area shown in Figure 4(a) are examined. P^H indicates a set of overlapping device pairs with $\Delta x < \Delta y$ after global placement. For a pair in P^H, constraint (4e) forces one device to be at the left of the other according to their x-coordinates as shown in Figure 4(a). If $\Delta x > \Delta y$, one device will be forced to be above the other.



(a) Separate overlapping devices. (b) Symmetry. (c) Alignment.

Fig. 4: Legalization and geometry constraints.

Unlike global placement, hard symmetry constraints are used in detailed placement. These are represented by (4f), where S_m^p denotes a set of device pairs symmetric to the same axis and S_m^s is a set of self-symmetric devices in the same group as S_m^p. All symmetry groups form the set S. For example, in Figure 4(b) devices A and C are in S_m^p, B is in S_m^s, and they share the same symmetry axis. The constraints state that the center of a symmetric pair $(x_{q_1} + x_{q_2})/2$ and the center of a self-symmetric device x_r should equal x_m of the symmetry axis.

Bottom alignment and central alignment constraints are handled by constraints (4g) and (4h), respectively, where P^B represents a set of bottom alignment pairs and P^{VC} denotes a set of vertically central alignment pairs. Examples of the alignment constraints are shown in figure 4(c). Device ordering constraints are to realize monotone path for certain critical signals [16] and enforced by constraints (4i), where O^H is a set of devices with a specific horizontal order.

C. Experimental Evaluation

We conduct experiments on a Linux machine with a Xeon (R) E5-2680 V2 processor, 2.8GHz frequency and 256G memory. The testcases include three Operational Transconductance Amplifier (OTA) designs, two comparator designs, two Voltage Controlled Oscillator (VCO) designs, an analog adder, a Variable Gain Amplifier (VGA) and an Switched Capacitor Filter (SCF). Each circuit has dozens of devices.

Comparisons are made among simulated annealing, recent previous analytical work [11][2] and our ePlace-A method. The main results are listed in Table III. The runtime advantage from analytical techniques is obvious as both the previous work and ePlace-A are more than 50× faster than simulated annealing. In terms of solution quality, ePlace-A and the previous work behave differently. While ePlace-A achieves significant reductions on both area and wirelength compared to simulated annealing, the previous analytical technique results in solution degradation. In theory, simulated annealing can approach the optimal solution if the number of iterations is sufficiently large. In practice, 50× runtime only allows it to outperform [11] but is not adequate for catching up ePlace-A. There are three reasons contributing to the improvement of ePlace-A over the previous work: (1) area is explicitly optimized in ePlace-A but not in [11]; (2) the WA-based HPWL smoothing in ePlace-A is better than the LSE-based on in [11] according to [23]; (3) device flipping is considered in ePlace-A but not in [11].

Parameter values of the three methods are varied for an experiment on circuit CM-OTA1 and the results are plotted in Figure 5. Compared to simulated annealing and [11], most solutions from ePlace-A are closer to the lower-left corner, corresponding to smaller area and HPWL. Hence, ePlace-A

[2] Although the placement of open source tool [12] is based on [11], it does not support GF12nm PDK used in our testcases. Thus, we implemented the analytical placement of [11] for the experiment.

TABLE III: Main comparison results for conventional performance oblivious formulation.

Design	Simulated annealing			Previous analytical work [11]			Our ePlace-A		
	Area(μm²)	HPWL(μm)	Runtime(s)	Area(μm²)	HPWL(μm)	Runtime(s)	Area(μm²)	HPWL(μm)	Runtime(s)
Adder	49.8	10.2	1.43	49.8	10.2	0.02	49.8	10.2	0.02
CC-OTA	84.8	37.2	17.12	100.3	37.4	0.16	81.6	34.1	0.22
Comp1	124.2	43.2	26.07	130.0	53.5	0.54	102.1	41.9	1.49
Comp2	141.4	87.9	71.87	251.3	110.1	1.60	130.9	80.8	2.73
CM-OTA1	139.9	37.7	27.52	139.3	36.4	0.51	114.1	28.1	0.19
CM-OTA2	165.9	66.6	52.12	229.0	93.5	0.18	161.4	61.2	0.75
SCF	2735.9	429.4	52.06	2158.9	486.0	10.87	1873.9	416.0	10.44
VGA	120.4	131.2	15.66	155.4	119.8	1.24	116.4	85.2	3.64
VCO1	315.7	202.3	126.65	315.7	201.1	1.27	315.7	181.7	3.12
VCO2	516.4	327.0	88.71	516.4	344.2	0.61	516.4	304.1	0.94
Avg. (X)	1.11	1.14	55.20	1.25	1.24	0.80	1.00	1.00	1.00

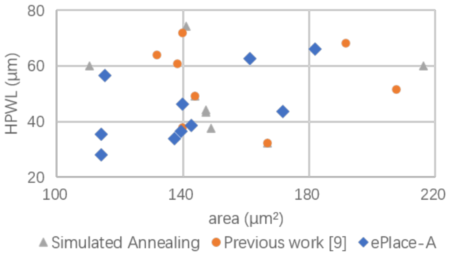


Fig. 5: HPWL-area tradeoff by varying placement parameters for CM-OTA1.

provides an advantage for any tradeoff point, not just for a specific setting.

We compare the detailed placement of [11] and ePlace-A using the same global placement solutions. The results in Table IV show that ePlace-A leads to smaller wirelength than [11] mainly due to its consideration of device flipping.

TABLE IV: Comparison between detailed placement of ePlace-A and [11]. Runtime only covers detailed placement.

Design	Previous work [11]			ePlace-A		
	Area(μm²)	HPWL(μm)	Runtime(s)	Area(μm²)	HPWL(μm)	Runtime(s)
VCO1	315.7	188.1	0.95	315.7	181.7	1.07
Comp1	102.1	45.3	0.42	102.1	41.9	0.75
SCF	1873.9	436.7	1.91	1873.9	416.0	2.32

V. Performance-Driven Analytical Placement

Our performance-driven analytical analog placer, ePlace-AP, is based on ePlace-A. Its approach to detailed placement, including legalization, is the same as ePlace-A.

A. Global Placement of ePlace-AP

In order to consider circuit performance during global placement, we employ a GNN (Graph Neural Network)-based performance model [19]. Its input is a circuit graph $\mathcal{G}$, which covers device types, locations, connections, etc. Its output Φ is the probability that circuit performance is unsatisfactory. The performance model Φ is included in the objective function and the NLP problem formulation becomes:

$$\min_{\boldsymbol{v}} W(\boldsymbol{v}) + \lambda N(\boldsymbol{v}) + \tau Sym(\boldsymbol{v}) + \eta Area(\boldsymbol{v}) + \alpha\Phi(\mathcal{G}) \quad (5)$$

where α is a weighting factor. Note that the first four terms of (5) are the same as (3) and $\mathcal{G}$ contains all information of $\boldsymbol{v}$.

There is key difference between the application of GNN model $\Phi(\mathcal{G})$ in ePlace-AP and [19], which is a simulated annealing-based performance-driven placement. In [19], inference of $\Phi(\mathcal{G})$ is conducted to directly assess circuit performance, which is a part of its objective function. In ePlace-AP, however, the NLP is solved using the gradient of the objective function. Hence, ePlace-AP needs to compute gradient $-\frac{\partial\Phi(\mathcal{G})}{\partial \boldsymbol{v}}$ instead of $\Phi(\mathcal{G})$ itself. Fortunately, TensorFlow has a built-in function for computing this gradient. Once the gradient is obtained, problem (5) is solved in the same way as ePlace-A.

B. Performance Metrics

The performance of an analog circuit is usually evaluated by multiple metrics $z_1, z_2, ..., z_M$, such as bandwidth, unity gain frequency and phase margin. Each metric z_i has a corresponding specification ψ_i. We partition the metrics of a design into two sets: Π^+ (Π^-) is the set of metrics that are preferred to be greater (less) than ψ_i, such as gain and bandwidth (delay and offset). Then, we normalize each performance metric as

$$\tilde{z}_i = \begin{cases} \min(\frac{z_i}{\psi_i}, 1), & \text{for } z_i \in \Pi^+ \\ \min(\frac{\psi_i}{z_i}, 1), & \text{for } z_i \in \Pi^- \end{cases} \quad (6)$$

so that $\tilde{z}_i \in [0, 1]$ and is preferred to be near 1. Like the previous work [19], the overall performance of a circuit is evaluated by a composite metric FOM (Figure of Merit): $FOM = \sum_{i=1}^{M} \beta_i \cdot \tilde{z}_i$, where β indicates weighting factors and $\sum_{i=1}^{M} \beta_i = 1$. The GNN model $\Phi(\mathcal{G})$ output is the probability that FOM is below a user-specified performance threshold.

C. Experimental Evaluation

Placement solutions were routed using an open source router from [25]. Next, parasitic extraction and SPICE simulations were performed using GlobalFoundries 12nm technology. The GNN models employed here have the same configuration as [19]. By varying parameters, over 1000 training samples were generated. Each sample has label 0 (1) for satisfactory (unsatisfactory) circuit performance. The cross entropy error metric is used during training.

TABLE V: Comparison of FOM results among three placement methods, each with variants of conventional and performance-driven formulation. Here, Perf* is our performance-driven extension of [11].

Design	Simulated annealing		Previous work [11]		ePlace-A	ePlace-AP
	Conv	Perf	Conv	Perf*	Conv	Perf
Adder	0.85	0.93	0.85	0.93	0.85	0.96
CC-OTA	0.86	0.94	0.83	0.93	0.86	0.96
Comp1	0.77	0.79	0.77	0.81	0.77	0.84
Comp2	0.72	0.78	0.73	0.80	0.72	0.83
CM-OTA1	0.90	0.97	0.87	0.97	0.86	0.99
CM-OTA2	0.87	0.91	0.87	0.92	0.87	0.94
SCF	0.83	0.84	0.83	0.84	0.83	0.86
VGA	0.77	0.87	0.82	0.89	0.77	0.91
VCO1	0.79	0.83	0.79	0.84	0.76	0.84
VCO2	0.77	0.84	0.78	0.84	0.78	0.85
Avg.	0.81	0.87	0.81	0.88	0.81	0.90

We compared circuit performance results in terms of FOM among simulated annealing [19], previous analytical work [11] and our ePlaceA/ePlace-AP methods. For each approach, we show results from two different formulations: a conventional one that is performance-oblivious and one that is performance-driven. Note that [11] is not performance-driven, but we extend it in the same way as ePlace-AP. The results are summarized in Table V. One can see that performance-driven techniques indeed improve FOM. The two analytical techniques provide greater improvement than simulated annealing. ePlace-AP achieves about 11% improvement over the performance-oblivious formulation and is the best among the three methods. Due to space limitations, we only show the detailed performance of CC-OTA from real simulations in Table VI. In this case, ePlace-A only satisfies the gain specification while ePlace-AP meets the specifications of both gain and unity gain frequency. ePlace-AP also improves bandwidth by 43% at the expense of 8% degradation on phase margin. Similar levels of improvement are seen on other testcases.

Results of the three performance-driven methods are listed in Table VII. While ePlace-AP reduces both area and wirelength

TABLE VI: Detailed performance results of CC-OTA.

Metric	Gain (dB)	UGF (MHz)	BW (MHz)	PM (°)	FOM
Specification	25.0	1200	70.0	90.0	
ePlace-A	26.2 (100%)	975 (81%)	48.2 (69%)	84.4 (94%)	0.86
ePlace-AP	25.5 (100%)	1244 (100%)	69.0 (99%)	78.6 (87%)	0.96

compared to simulated annealing, the performance extension to [11] increases these metrics. The runtime advantage of analytical techniques decreases in performance-driven placement as the computation of gradient $-\frac{\partial \Phi(\mathcal{G})}{\partial \boldsymbol{v}}$ in analytical techniques is much more expensive than computing $\Phi(\mathcal{G})$ in simulated annealing. Nevertheless, analytical techniques are about 3× faster than simulated annealing. The absolute runtimes are all less than a minute.

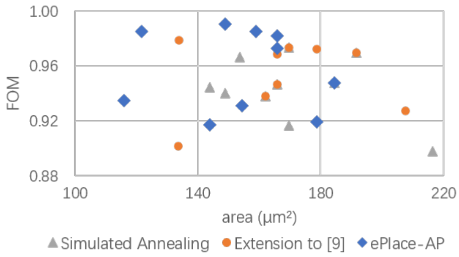


Fig. 6: FOM-area tradeoff of CM-OTA1 by varying parameters.

The parameters of the three methods are varied to obtain tradeoff points in Figure 6. Solutions with the best FOM-area tradeoff (near the upper-left corner) are from ePlace-AP, which demonstrates an overall advantage over competing methods.

VI. Conclusions

This work provides an introspective study on analytical techniques for analog placement. For conventional performance-oblivious analog placement, analytical techniques are 55× faster than simulated annealing. However, not all analytical techniques can reduce area and wirelength compared to simulated annealing. We propose an analytical technique based on ePlace and achieves 10% and 12% reductions on area and wirelength, respectively. We develop a performance-driven analytical technique. It obtains 11% improvement on overall circuit performance. Further, it results in less area/wirelength than performance-driven simulated annealing with 3× speedup and runtimes of under a minute for all testcases.

Acknowledgement

This work is supported by the DARPA ERI IDEA program.

References

[1] M. Eick *et al.*, "Comprehensive generation of hierarchical placement rules for analog integrated circuits," *IEEE TCAD*, vol. 30, no. 2, pp. 180–193, 2011.

[2] M. Strasser *et al.*, "Deterministic analog circuit placement using hierarchically bounded enumeration and enhanced shape functions," in *Proc. ICCAD*, 2008, pp. 306–313.

[3] H. E. Graeb, *Analog Layout Synthesis: A Survey of Topological Approaches*. Springer, 2010.

[4] J. Liu *et al.*, "Thermal-driven symmetry constraint for analog layout with CBL representation," in *Proc. ASPDAC*, 2007, pp. 191–196.

[5] P.-H. Lin *et al.*, "Analog placement based on symmetry-island formulation," *IEEE TCAD*, vol. 28, no. 6, pp. 791–804, 2009.

[6] C.-W. Lin *et al.*, "Performance-driven analog placement considering boundary constraint," in *Proc. DAC*, 2010, pp. 292–297.

[7] L. Xiao *et al.*, "Analog placement with common centroid and 1-D symmetry constraints," in *Proc. ASPDAC*, 2009, pp. 353–360.

[8] Q. Ma *et al.*, "Simultaneous handling of symmetry, common centroid, and general placement constraints," *IEEE TCAD*, vol. 30, no. 1, pp. 85–95, 2011.

[9] H.-C. Ou *et al.*, "Layout-dependent effects-aware analytical analog placement," *IEEE TCAD*, vol. 35, no. 8, pp. 1243–1254, 2016.

[10] T.-C. Chen *et al.*, "NTUplace3: An analytical placer for large-scale mixed-size designs with preplaced blocks and density constraints," *IEEE TCAD*, vol. 27, no. 7, pp. 1228–1240, 2008.

[11] B. Xu *et al.*, "Device layer-aware analytical placement for analog circuits," in *Proc. ISPD*, 2019, pp. 19–26.

[12] H. Chen *et al.*, "MAGICAL: An open-source fully automated analog IC layout system from netlist to GDSII," *IEEE D&T*, vol. 38, no. 2, pp. 19–26, 2020.

[13] K. Zhu *et al.*, "Effective analog/mixed-signal circuit placement considering system signal flow," in *Proc. ICCAD*, 2020, pp. 1–9.

[14] A. Kahng *et al.*, "Implementation and extensibility of an analytic placer," *IEEE TCAD*, vol. 24, no. 5, pp. 734–747, 2005.

[15] J. Lu *et al.*, "ePlace-MS: Electrostatics-based placement for mixed-size circuits," *IEEE TCAD*, vol. 34, no. 5, pp. 685–698, 2015.

[16] H.-C. Ou *et al.*, "Simultaneous analog placement and routing with current flow and current density considerations," in *Proc. DAC*. IEEE, 2013, pp. 1–6.

[17] B. Xu *et al.*, "Hierarchical and analytical placement techniques for high-performance analog circuits," in *Proc. ISPD*, 2017, pp. 55–62.

[18] K. Lampaert *et al.*, "A performance-driven placement tool for analog integrated circuits," *IEEE JSSC*, vol. 30, no. 7, pp. 773–780, 1995.

[19] Y. Li *et al.*, "A customized graph neural network model for guiding analog IC placement," in *Proc. ICCAD*, 2020, pp. 1–9.

[20] U. Choudhury *et al.*, "Automatic generation of parasitic constraints for performance-constrained physical design of analog circuits," *IEEE TCAD*, vol. 12, no. 2, pp. 208–224, 1993.

[21] P.-H. Wu *et al.*, "Performance-driven analog placement considering monotonic current paths," in *Proc. ICCAD*, 2012, pp. 613–619.

[22] Y. Li *et al.*, "Exploring a machine learning approach to performance driven analog IC placement," in *Proc. ISVLSI*, 2020, pp. 24–29.

[23] M.-K. Hsu *et al.*, "TSV-aware analytical placement for 3D IC designs," in *Proc. DAC*, 2011, pp. 664–669.

[24] Y. E. Nesterov, "A method of solving a convex programming problem with convergence rate $O(1/k^2)$," *Soviet Math*, vol. 27, no. 2, pp. 372–376, 1983.

[25] T. Dhar *et al.*, "ALIGN: A system for automating analog layout," *IEEE D&T*, vol. 38, no. 2, pp. 8–18, 2021.

TABLE VII: Area, wirelength and runtime comparison among performance-driven methods.

Design	Performance-driven simulated annealing [19]			Performance extension to [11]			ePlace-AP		
	Area(μm²)	HPWL(μm)	Runtime(s)	Area(μm²)	HPWL(μm)	Runtime(s)	Area(μm²)	HPWL(μm)	Runtime(s)
Adder	49.0	21.2	15.17	49.0	21.2	15.10	49.5	21.2	15.89
CC-OTA	117.5	34.5	39.67	158.8	47.4	17.54	117.5	36.8	16.37
Comp1	141.6	66.4	48.53	136.0	67.4	30.90	110.3	52.8	31.00
Comp2	209.4	85.7	88.44	287.4	96.9	37.10	198.9	84.5	38.00
CM-OTA1	191.7	43.2	34.13	169.6	43.9	10.68	148.8	34.9	9.19
CM-OTA2	225.2	72.7	65.72	229.0	93.5	20.58	215.6	62.0	19.89
SCF	2425.3	465.5	73.61	2360.1	537.7	23.48	2402.7	644.2	22.49
VGA	233.9	108.0	53.48	225.4	109.3	7.16	191.8	107.3	8.49
VCO1	347.2	186.8	157.92	315.7	224.6	35.26	347.2	208.5	34.19
VCO2	535.7	363.6	138.60	574.0	362.3	57.20	535.7	391.0	56.40
Avg. (X)	1.09	1.02	3.09	1.14	1.13	1.01	1.00	1.00	1.00