Adaptive Preconditioning Guided by Divergence Analysis for Enhanced VLSI Global Placement

Liwen Jiang, Ruiyu Lyu, Fan Yang, and Keren Zhu*
State Key Laboratory of Integrated Chips and Systems, College of Integrated Circuits and Micro-Nano Electronics
Fudan University, Shanghai, China
*Corresponding author: krzhu@fudan.edu.cn

***Abstract*—Global placement is a critical step in modern VLSI physical design. Traditional electrostatic-based analytical placement algorithms, such as ePlace, employ a two-layer optimization loop: an inner loop using Nesterov's gradient descent, and an outer loop updating the density penalty multiplier to progressively enhance the density penalty. However, these conventional update methods often fail to balance the optimization efforts across different nets, leading to suboptimal solutions. In this paper, we introduce an adaptive preconditioning algorithm that addresses these limitations. Our approach utilizes divergence analysis to identify clusters within the placement region, followed by net preconditioning to adjust the optimization efforts accordingly. This ensures a more balanced distribution of optimization efforts across all nets, leading to improved placement quality. Experimental results on the DAC-2012 benchmarks suite demonstrate that our algorithm achieves a reduction in final placement wirelength compared to DREAMPlace, while obtaining an optimal tradeoff between half-perimeter wirelength (HPWL) and iteration count.**

***Index Terms*—global placement, divergence analysis, net weighting.**

I. Introduction

Conventional global placement methodologies in VLSI physical design primarily optimize wirelength minimization while incorporating density constraints to mitigate instance overlap. The emergence of electrostatic-based placers in the ePlace series [1–4] has revolutionized placement techniques by establishing a novel paradigm for simultaneous wirelength optimization and uniform spatial distribution of heterogeneous circuit components, including standard cells, macros, and other movable instances. These advanced implementations employ fast Fourier transform (FFT)-accelerated numerical solvers for real-time density potential evaluation, achieving superior solution quality and computational efficiency through three key innovations: (1) continuous differentiable system formulation enabling gradient-based optimization, (2) electrostatic charge analogy ensuring smooth density gradient propagation, and (3) spectral methods for lower complexity in potential computation.

However, the electrostatic-based analytical placers are limited by their "global" nature of their iterations [5, 6], which may fail to capture the fine-grained spatial and temporal dynamics of the placement process. Recently, researchers have developed various dynamic processes to improve the performance of placer [7–10]. Since further improvement on HPWL is challenging, many dynamic methods for global placement are designed to enhance algorithm robustness or guide timing and routability. The dynamic adaptive scheme can continue following placement status and promptly adjust the optimization result we expect, which is suitable for guiding the fine-grained placement process.

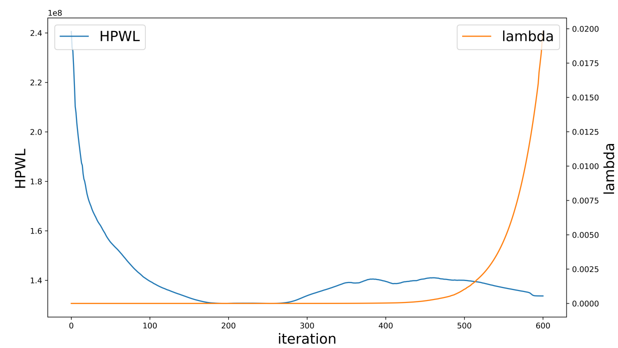


Fig. 1: Evolution of HPWL and λ over global placement iterations in superblue19 benchmark. The density penalty dominate optimization due to exponential growing λ to spread the few clusters. Such behavior empirically degradate the optimization results with wirelength overhead induced.

We observe that some designs have close-knit nets which cause cell clusters during placement flow, with the result that the global density penalty multiplier cannot smoothly reach the overflow target in such extreme cases. Figure 1 shows the evolution of HPWL and density penalty multiplier λ throughout the global placement procedure. While the density constraints can be satisfied by gradually increasing lambda, the placer always use a huge density penalty to wait for these clusters reaching the overflow target, resulting in an unnecessary wirelength increase in other nets. This situation leads other cells, which have already reached the overflow target be subject to a higher density penalty, thereby affecting the final placement quality. RePlAce [3] sets local density function at per-bin granularity to spreading cells at a fine-

This work is supported partly by Science and Technology Commission of Shanghai Municipality Project (24JD1400800), and partly by the State Key Laboratory of Integrated Chips and Systems, Fudan University (No. SKLICS-Z202404).

grained level, but it cannot explain and solve the root cause of the cell spreading being not smooth. Observing the shortfall in local smoothing, we develop a novel algorithm to dynamically detect and adjust imbalanced cell distribution, ultimately improving global placement quality.

In this paper, we propose a global placement algorithm with preconditioning adaptation, and optimize the local smoothness problem in the gloabl placement flow with net weighting. The major contributions are summarized as follows.

- We design a cell cluster detector using the spectral methods, which can accurately identify clusters affecting normal placement behavior.
- We propose a divergence-guided smoothing method for the density function, enabling locally adaptive adjustment of clusters.
- We propose a dynamic net weighting scheme to address the imbalance density map. By incorporating the wirelength and density information of every cluster into the backward propagation process, we refined and optimized the placement of cluster cells.
- Experimental results on the DAC-2012 benchmarks suite [11] show that on average we achieve HPWL and iteration improvements compared to the state-of-the-art placer after legalization and detailed placement.

The rest of the paper is organized as follows. Section II provides some preliminaries including objective of placement and Poisson's equation. Section III presents the overall flow of our proposed algorithm and the detailed explanations. Section IV demonstrates the experimental results and some related analysis, followed by Section V summarizing the whole paper.

II. Preliminaries

A. Problem Statement

For wirelength-driven placement problem, we minimize the wirelength with density constraints. By using the penalty factor λ, we can produce an unconstrained optimization problem as equation (1) shows

$$\min_{\mathbf{v}} \quad f(\mathbf{v}) = W(\mathbf{v}) + \lambda D(\mathbf{v}), \tag{1}$$

where the wirelength objective $W(\mathbf{v})$ takes every net instance and returns the wirelength modeled by weighted-average wirelength (WA) [12, 13], while the density penalty $D(\mathbf{v})$ addresses instance overlap. During nonlinear optimization, a density penalty multiplier λ is gradually increased to reduce overlap [2, 14–16], and the density constraints can be satisfied by suitable λ to reach target overflow.

B. Poisson's Equation in ePlace

The core idea of ePlace is using the electrostatic analogy to transform a complex density penalty function into a solvable Poisson's equation problem [2, 17, 18]. In ePlace, the DCT and DST are used in spectral methods to generate the solution to the partial differential equations, where density penalty and gradient are modeled as system potential energy and electric force. Poisson's equation has been used to compute the potential field induced by a given charge density distribution. Based on Gauss's law, a partial differential equation with Poisson's equation, boundary condition, and the compatibility condition in ePlace is used:

$$\begin{cases} \nabla \cdot \nabla \psi(x, y) = -\rho(x, y), & \text{(2a)} \\ \hat{\mathbf{n}} \cdot \nabla \psi(x, y) = 0, \quad (x, y) \in \partial \mathrm{R}, & \text{(2b)} \\ \iint_R \rho(x, y)\, dx\, dy = \iint_R \psi(x, y)\, dx\, dy = 0. & \text{(2c)} \end{cases}$$

Equation (2a) represents the Poisson equation, relating the Laplacian of the potential $\psi(x, y)$ to the density function $\rho(x, y)$. Equation (2b) is the Neumann boundary condition to prevent a block from running out of the boundary, where $\hat{\mathbf{n}}$ is the outward unit normal vector on the boundary $\partial \mathrm{R}$. Equation (2c) is the compatibility condition, which makes the system of equations a unique solution.

Let u and p denote integral indexes, and the frequency components are defined as $\omega_u = 2\pi u/m$ and $\omega_v = 2\pi v/m$, respectively. The coefficient of each wave function of a discrete cosine transform is denoted by $a_{u,v}$ as follows:

$$a_{u,v} = \frac{1}{m^2} \sum_{x=0}^{m-1} \sum_{y=0}^{m-1} \rho(x, y) \cos(\omega_u x) \cos(\omega_v y). \tag{3}$$

Then take the Fourier transform of Equation(2a) to get the Fourier series relationship, the potential value is easily calculated by

$$\psi_{DCT}(x, y) = \sum_{u=0}^{m-1} \sum_{v=0}^{m-1} \frac{a_{u,v}}{\omega_u^2 + \omega_v^2} \cos(\omega_u x) \cos(\omega_v y), \tag{4}$$

where $\psi(x, y)$ is the electric potential at the center of bin b_{lj}. Based on the solution to the potential function in Equation(4), we can obtain the solution to the electric field $\xi(x, y)$ as follows:

$$\begin{cases} \xi_{X_{DCT}} = \sum_{u=0}^{m-1} \sum_{v=0}^{m-1} \frac{a_{u,v}\omega_u}{\omega_u^2 + \omega_v^2} \sin(\omega_u x) \cos(\omega_v y) \\ \xi_{Y_{DCT}} = \sum_{u=0}^{m-1} \sum_{v=0}^{m-1} \frac{a_{u,v}\omega_v}{\omega_u^2 + \omega_v^2} \cos(\omega_u x) \sin(\omega_v y). \end{cases} \tag{5}$$

The electric potential and field distribution can be rapidly determined by solving Poisson's equation using spectral methods.

III. Proposed Algorithm

Figure 2 shows the overview of our placement algorithm. Compared with other classic placer, our placement algorithm introduces an adaptive preconditioning step after a suitable iteration to further improve the quality of placement solutions. Cell cluster detector obtains the net and node sets for dynamic density smoothing and net weighting to dispersed cluster close to the behavior of normal cells.

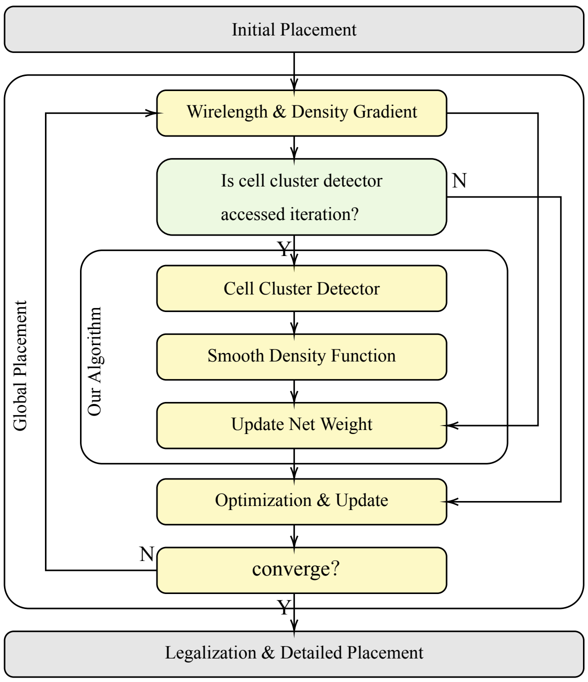


Fig. 2: Our placement algorithm flow.

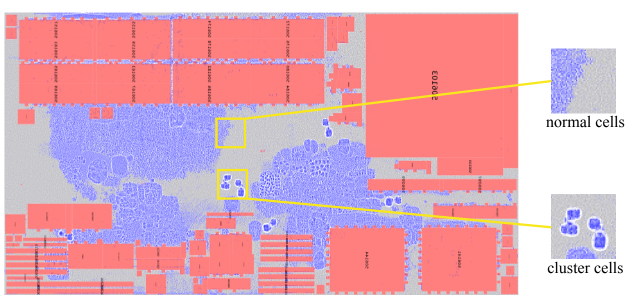


Fig. 3: Cell distribution at the 550 iteration of the superblue19 benchmark. The unsuited density force pushes the cells away from clusters, caused under-filled whitespace around cluster and density overhead.

A. Divergence Analysis for Preconditioning

Close-knit nets are clustered together due to the entanglement between wirelength and density. Such premature over-optimization in their objectives often leads to accelerated local convergence in the early stages of global placement, which may caused the inefficiency of placer in the later stage for solving local density overhead. Figure 3 shows clustering phenomenon in the later stage of placement.

1) Cell Cluster Detector: Since the divergence of an electric field is a measure of how much the electric field spreads out from a point, the tendency of cells to cluster in each iteration can be expressed by divergence. By dividing the placement region into $m \times m$ bins, we obtain an $m \times m$ density matrix ρ, where each element represents the total area occupied by the blocks within the corresponding bin. Then, the divergence is calculated by

$$\nabla \cdot \xi = \frac{\partial \xi_x}{\partial x} + \frac{\partial \xi_y}{\partial y}. \tag{6}$$

Based on the solution to the electric field function, we can obtain the solution to divergence in the form of DCT and DST in the horizontal and vertical directions respectively as follows:

$$\begin{cases} \frac{\partial \xi_x}{\partial x} = \sum_{u=0}^{m-1} \sum_{v=0}^{m-1} \frac{a_{u,v} {\omega_u}^2}{\omega_u^2 + \omega_v^2} \sin(\omega_u x) \cos(\omega_v y) \\ \frac{\partial \xi_y}{\partial y} = \sum_{u=0}^{m-1} \sum_{v=0}^{m-1} \frac{a_{u,v} {\omega_v}^2}{\omega_u^2 + \omega_v^2} \cos(\omega_u x) \sin(\omega_v y). \end{cases} \tag{7}$$

At the stage of solving equation (2a)-(2c), we use discrete spectral transformation leveraging fast Fourier transform engine in DREAMPlace [4]. After solving the Poisson equation, we can also get the divergence of current iteration by Equation (6) and Equation (7). The detector we designed identifies the bin reaching the divergence threshold as a cluster and extracts the associated net and node sets in it.

2) Diffusion-based Smooth Density Function: These clusters found by our divergence detector can be considered as a sink in the electrostatic field. From a physical perspective, the initial placement diffuses from a source point, and extreme close-knit nets form sinks in the placement flow. We aim to eliminate all sinks in the density field so that the diffusion flux reaches dynamic equilibrium in the calculation domain. When the gradient curl of the density field disappears, forming a field without sink and source, marking the dynamic equilibrium state of the diffusion process.

$$\hat{\rho}(x, y) = \rho(x, y) + div \tag{8}$$

Specifically, if the divergence of density field $div > 0$, it indicates that the point is in the local concentration minimum region, and a concentration increment is introduced to promote regional smoothing; if $div < 0$, it indicates that the point is in the local concentration maximum region, and concentration attenuation is applied to eliminate gradient mutations.

In the electrostatic-based analytical placement, what guides cells to spread is not cell density, but electric field force. Hence, we use the electric field divergence get by Equation (6) to guide the correction of the density map at the cluster. The modified diffusion equation is

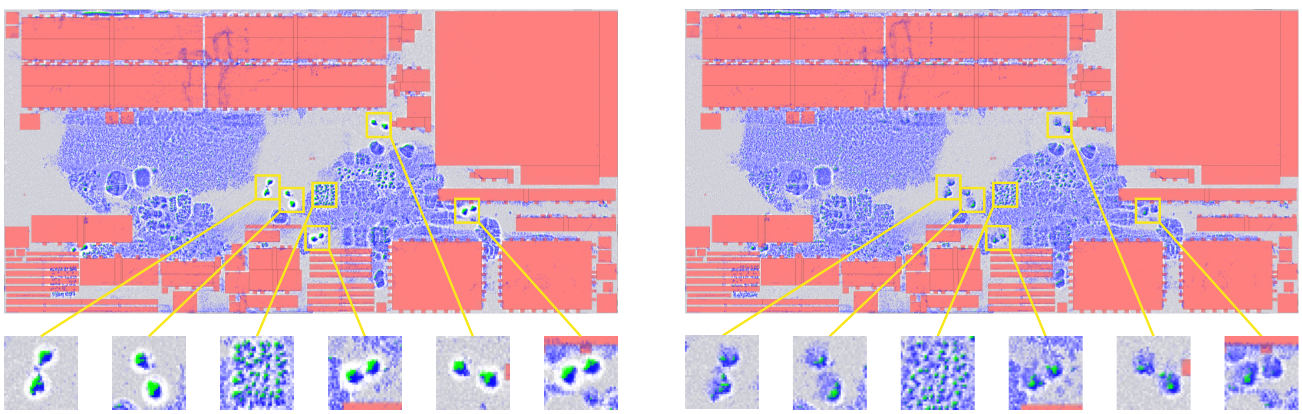
$$\hat{\rho}(x, y) = \rho(x, y) + K \times (\nabla \cdot \xi), \tag{9}$$

where K is the diffusion coefficient, its value depends on the degree of cluster in a design. After obtaining the new density function, we apply it to compute the electric potential and field again.

TABLE I: The HPWL and iteration comparison on the DAC-2012 benchmarks suite.

Circuits	# Movable Cells	# Nets	DREAMPlace-1 *		DREAMPlace-2 *		Ours	
			HPWL	iteration	HPWL	iteration	HPWL	iteration
SUPERBLUE2	921273	990899	5.669E+08	694	5.667E+08	811	**5.667E+08**	802
SUPERBLUE3	833370	898001	2.880E+08	710	2.874E+08	825	**2.865E+08**	825
SUPERBLUE6	919093	1006629	3.036E+08	720	3.036E+08	743	**3.035E+08**	755
SUPERBLUE7	1271887	1340418	3.634E+08	757	3.637E+08	774	**3.634E+08**	763
SUPERBLUE9	789064	833808	2.086E+08	744	2.098E+08	889	**2.069E+08**	869
SUPERBLUE11	859771	935731	3.211E+08	710	**3.193E+08**	827	3.197E+08	843
SUPERBLUE12	1278084	1293436	2.235E+08	953	2.259E+08	1093	**2.218E+08**	1066
SUPERBLUE14	567840	619815	2.105E+08	646	2.101E+08	702	**2.091E+08**	704
SUPERBLUE16	680450	697458	2.416E+08	685	2.411E+08	798	**2.403E+08**	784
SUPERBLUE19	506097	511685	1.363E+08	605	1.360E+08	703	**1.348E+08**	723
ratio			1.005	0.889	1.005	1.003	1.000	1.000

DREAMPlace-1: released DREAMPlace
DREAMPlace-2: smaller-step-size DREAMPlace with similar iteration to ours



(a) iteration=500 (b) iteration=515

Fig. 4: Detected clusters during our global placement for the superblue19 benchmark. Standard cells, macros and clusters are denoted by blue, pink and green, respectively.

Algorithm 1 Update Net Weights

Require: $cluster\_net\_set : CNets$, $cluster\_node\_set : CNodes$

$\mathbf{N} \leftarrow AllNodes - CNodes$; ▷ Get normal node set

$$\mathbf{N}_s \leftarrow \texttt{compute\_scale}\,\left(\frac{\sum_{\mathbf{N}} \nabla W}{\sum_{\mathbf{N}} \nabla D}\right);$$

for each $cluster$ in $CNets$ **do**

$\quad \mathbf{C} \leftarrow CNodes[cluster]$;

$$\mathbf{C}_s \leftarrow \texttt{compute\_scale}\,\left(\frac{\sum_{\mathbf{C}} \nabla W}{\sum_{\mathbf{C}} \nabla D}\right);$$

$\quad s \leftarrow \mathbf{C}_s \;/\; \mathbf{N}_s$; ▷ Compute scaling factor

$\quad w_{cluster} \leftarrow w_{cluster} \times s$; ▷ Update the net weights

end for

B. Net Weighting Scheme

The local density of the cluster cells exceed that of the normal cells, thus the global placement flow have to wait for them until the layout reaching the target overflow in the later stage with the density penalty multiplier λ increasing. For normal cells (divergence is less than threshold we set), their wirelength and density gradient norm in backward propagation represent their optimization degree. The local density of cluster cells is larger than normal cells, so their optimization degree of wirelength should be stronger to ensure that they do not continue being cluster.

Net-based approaches always used to optimize timing in placement [19–23], we apply it to adjust the cluster spreading based on the same principle of ignoring critical paths. The standard wirelength-density scale depends on normal cells, which ensures the net weight of cells in clusters is adjusted in proportion to the wirelength and density scaling factor. For each cluster, the wirelength-density scale is normalized by the standard scale. This step ensures that the placement of cells in cluster is adjusted in proportion to the wirelength and density scaling factors.

To efficiently help clusters spreading smoothly, we propose an improved dynamic net weight adaptation strategy that dynamically adjusts the net weight of every cluster and the scaling factor with respect to the wirelength and density gradient norm per cycle iteration. The net and node sets returned by the cell cluster detector is the target object we need to adjust. Algorithm 1 illustrates detailed implementation steps.

IV. Experimental Results

A. Experimental Setup

We build our global placement algorithm on an open-source VLSI placer DREAMPlace [4]. All the experiments run on a Linux server with 2.10GHz Intel Xeon Gold 6230 CPU using 8 threads and use the datatype of single-precision floating-point for evaluation.

B. HPWL and Iteration Improvement

We conduct experiments on the DAC-2012 benchmarks suite [11]. Local cluster adjustment may slow down the convergence speed, and have impact on the behavior of the global placement. To ensure a fair comparison, we specially slow down the optimization process of DREAMPlcae by reducing the step size [5, 24] to compare with ours so that the total iterations of the two remain close.

Cell positions are highly overlapped at the earlier stages, and conducted divergence detector in this situation will consume a lot of runtime and will not have a significant direct benefit on the final placement quality. Thus detector is typically conducted once the cells are approximately distributed by density forces. In our experiments, we assess detector and adjust net weights every 5 iterations, starting after about half of the total iterations of the baseline.

Table I gives the crucial benchmark statistics and compares the placement quality of our algorithm with released DREAMPlace (DREAMPlace-1), smaller-step-size DREAMPlace with similar iteration to ours (DREAMPlace-2). The results show a detailed comparison that includes the HPWL wirelength of the final placement (HPWL) and the total number of iterations (iteration) for each placer. We use boldface to emphasize the best HPWL among the three results, and the bottom row gives the normalized wirelength and iteration ratios based on our results.

The results demonstrate that our algorithm outperforms DREAMPlace by achieving a 0.5% improvement in average final placement HPWL. Moreover, compared with smaller-step-size DREAMPlace, our algorithm achieves a 0.5% improvement in HPWL while reducing the number of iterations by 0.3%. It is clear that our analytical method is the most accurate, and our adaptive preconditioning algorithm can make the best tradeoff between HPWL and iteration.

C. Cluster Detection and Smooth

Figure 4a illustrates detected clusters on superblue19 benchmark at iteration 500 in normal flow. Standard cells, macros, and detected clusters are marked in blue, pink, and green, respectively. After adjusted twice, the result at the 515th iteration is shown in Figure 4b. By observing the six dominant clusters, we can find that the clustering phenomenon is effectively solved after two dynamic net weighting steps. The experimental results show that our method can effectively detect clusters and locally smooth them.

V. Conclusion

In this paper, we propose a adaptive preconditioning algorithm for global placement guided by divergence analysis. The distribution of the clusters snapshot result shows that our algorithm can effectively capture the spatial correlation of instances in the global placement stage. Experimental results show that ours can produce better placement solutions achieving 0.5% HPWL improvement on average compared to the most widely-used analytical placer. This work provides new ideas and methodological support for solving placement challenges in complex integrated circuit design.

References

[1] J. Lu, H. Zhuang, P. Chen, H. Chang, C.-C. Chang, Y.-C. Wong, L. Sha, D. Huang, Y. Luo, C.-C. Teng, and C.-K. Cheng, "eplace-ms: Electrostatics-based placement for mixed-size circuits," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 34, no. 5, pp. 685–698, 2015.

[2] J. Lu, P. Chen, C.-C. Chang, L. Sha, D. J.-H. Huang, C.-C. Teng, and C.-K. Cheng, "eplace: Electrostatics-based placement using fast fourier transform and nesterov's method," *ACM Trans. Des. Autom. Electron. Syst.*, vol. 20, no. 2, Mar. 2015.

[3] C.-K. Cheng, A. B. Kahng, I. Kang, and L. Wang, "Replace: Advancing solution quality and routability validation in global placement," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 38, no. 9, pp. 1717–1730, 2019.

[4] Y. Lin, Z. Jiang, J. Gu, W. Li, S. Dhar, H. Ren, B. Khailany, and D. Z. Pan, "Dreamplace: Deep learning toolkit-enabled gpu acceleration for modern vlsi placement," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 40, no. 4, pp. 748–761, 2021.

[5] T.-C. Chen, Z.-W. Jiang, T.-C. Hsu, H.-C. Chen, and Y.-W. Chang, "Ntuplace3: An analytical placer for large-scale mixed-size designs with preplaced blocks and density constraints," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 27, no. 7, pp. 1228–1240, 2008.

[6] A. Kahng and Q. Wang, "Implementation and extensibility of an analytic placer," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 24, no. 5, pp. 734–747, 2005.

[7] J. Gu, Z. Jiang, Y. Lin, and D. Z. Pan, "Dreamplace 3.0: multi-electrostatics based robust vlsi placement with region constraints," ser. ICCAD '20. New York, NY, USA: Association for Computing Machinery, 2020.

[8] P. Liao, S. Liu, Z. Chen, W. Lv, Y. Lin, and B. Yu, "Dreamplace 4.0: Timing-driven global placement with momentum-based net weighting," in *2022 Design, Automation & Test in Europe Conference & Exhibition (DATE)*, 2022, pp. 939–944.

[9] X. He, T. Huang, L. Xiao, H. Tian, G. Cui, and E. F. Young, "Ripple: An effective routability-driven placer by iterative cell movement," in *2011 IEEE/ACM International Conference on Computer-Aided Design (ICCAD)*, 2011, pp. 74–79.

[10] M.-K. Hsu, Y.-F. Chen, C.-C. Huang, S. Chou, T.-H. Lin, T.-C. Chen, and Y.-W. Chang, "Ntuplace4h: A novel routability-driven placement algorithm for hierarchical mixed-size circuit designs," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 33, no. 12, pp. 1914–1927, 2014.

[11] N. Viswanathan, C. Alpert, C. Sze, Z. Li, and Y. Wei, "The dac 2012 routability-driven placement contest and benchmark suite," in *DAC Design Automation Conference 2012*, 2012, pp. 774–782.

[12] M.-K. Hsu, Y.-W. Chang, and V. Balabanov, "TSV-aware analytical placement for 3D IC designs," in *Proceedings of the 48th Design Automation Conference*, ser. DAC '11, Jun. 2011, pp. 664–669.

[13] M.-K. Hsu, V. Balabanov, and Y.-W. Chang, "TSV-Aware Analytical Placement for 3-D IC Designs Based on a Novel Weighted-Average Wirelength Model," *IEEE Trans. Comput.-Aided Des. Integr. Circuits Syst.*, vol. 32, no. 4, pp. 497–509, Apr. 2013.

[14] A. Kahng and Qinke Wang, "Implementation and extensibility of an analytic placer," *IEEE Trans. Comput.-Aided Des. Integr. Circuits Syst.*, vol. 24, no. 5, pp. 734–747, May 2005.

[15] T.-C. Chen, Z.-W. Jiang, T.-C. Hsu, H.-C. Chen, and Y.-W. Chang, "NTUplace3: An Analytical Placer for Large-Scale Mixed-Size Designs With Preplaced Blocks and Density Constraints," *IEEE Trans. Comput.-Aided Des. Integr. Circuits Syst.*, vol. 27, no. 7, pp. 1228–1240, Jul. 2008.

[16] M.-K. Hsu and Y.-W. Chang, "Unified Analytical Global Placement for Large-Scale Mixed-Size Circuit Designs," *IEEE Trans. Comput.-Aided Des. Integr. Circuits Syst.*, vol. 31, no. 9, pp. 1366–1378, Sep. 2012.

[17] T. Chan, J. Cong, and K. Sze, "Multilevel generalized force-directed method for circuit placement," in *Proceedings of the 2005 International Symposium on Physical Design*, ser. ISPD '05. New York, NY, USA: Association for Computing Machinery, 2005, p. 185–192.

[18] W. Zhu, Z. Huang, J. Chen, and Y.-W. Chang, "Analytical solution of Poisson's equation and its application to VLSI global placement," in *Proceedings of the International Conference on Computer-Aided Design*, Nov. 2018, pp. 1–8.

[19] H. Eisenmann and F. Johannes, "Generic global placement and floorplanning," in *Proceedings 1998 Design and Automation Conference. 35th DAC*, 1998, pp. 269–274.

[20] T. Kong, "A novel net weighting algorithm for timing-driven placement," in *IEEE/ACM International Conference on Computer Aided Design, 2002. ICCAD 2002.*, 2002, pp. 172–176.

[21] B. Riess and G. Ettelt, "SPEED: Fast and efficient timing driven placement," in *Proceedings of ISCAS'95 - International Symposium on Circuits and Systems*, vol. 1, 1995, pp. 377–380.

[22] M. Burstein and M. Youssef, "Timing Influenced Layout Design," in *22nd ACM/IEEE Design Automation Conference*, 1985, pp. 124–130.

[23] B. Obermeier and F. M. Johannes, "Quadratic placement using an improved timing model," in *Proceedings of the 41st Annual Design Automation Conference*, Jun. 2004, pp. 705–710.

[24] T.-c. Chen, Z.-w. Jiang, T.-c. Hsu, H.-c. Chen, and Y.-w. Chang, "A high-quality mixed-size analytical placer considering preplaced blocks and density constraints," in *2006 IEEE/ACM International Conference on Computer Aided Design*, 2006, pp. 187–192.