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

05/11/2025

- Motivation
- Proposed Algorithm
  - Divergence Analysis for Preconditioning
  - Net Weighting Scheme
- Experimental Results & Conclusion

Conventional Global Placement

1 Motivation

- **Relaxed Objective Function**:

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

- **Density Penalty Multiplier λ**:

$$\lambda \leftarrow \lambda \times cof,$$

ePlace-MS fixes $cof \in [0.95, 1.05]$[a].

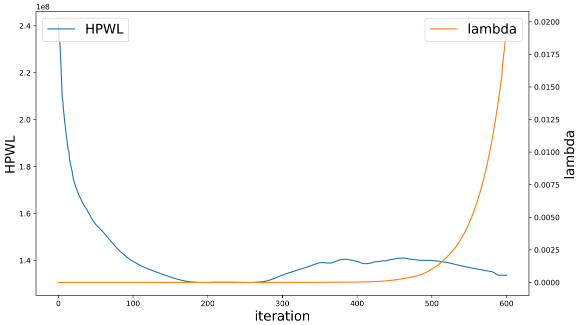


Figure: Evolution of HPWL and λ over global placement iterations in superblue19 benchmark.

[a] Jingwei Lu et al. "ePlace-MS: Electrostatics-Based Placement for Mixed-Size Circuits". In: *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems* 34.5 (2015), pp. 685–698.

Clustering in Placement

1 Motivation

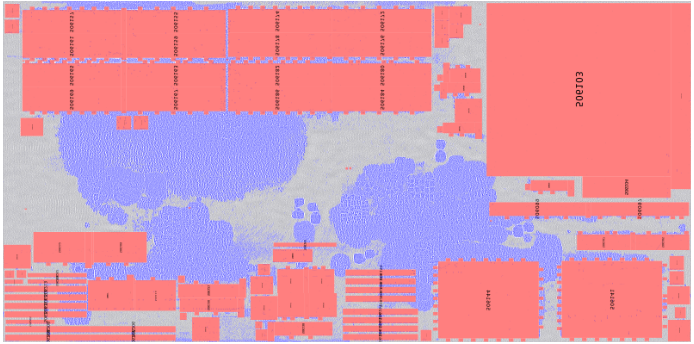
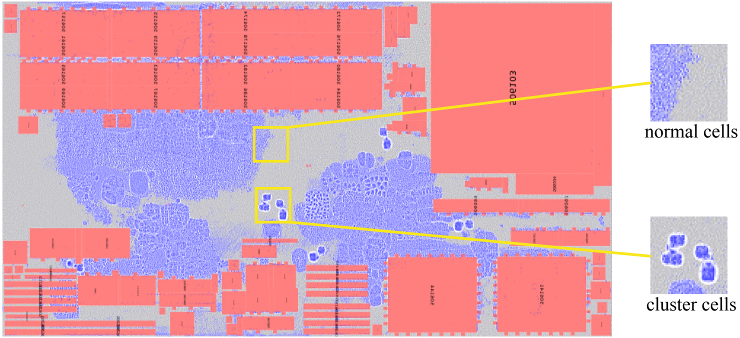


Figure: Final global placement result in superblue19 benchmark.



Placers are limited by their "global" nature.[1]

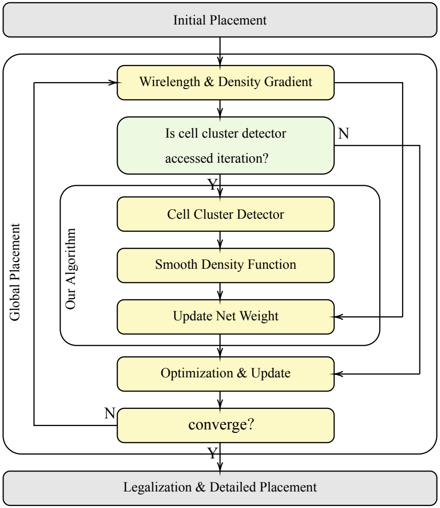
[1] Chung-Kuan Cheng et al. "RePlAce: Advancing Solution Quality and Routability Validation in Global Placement". In: *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems* 38.9 (2019), pp. 1717–1730.

Outline

2 Proposed Algorithm

- Motivation
- Proposed Algorithm
  - Divergence Analysis for Preconditioning
  - Net Weighting Scheme
- Experimental Results & Conclusion

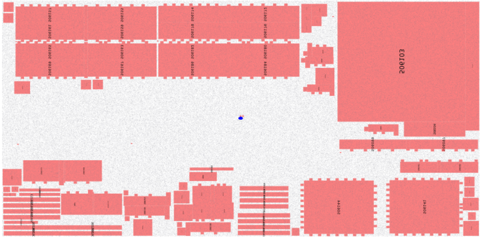
Overall Flow



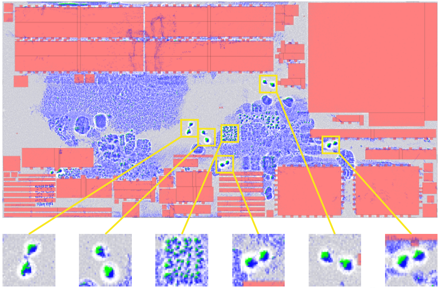
- **Divergence-guided cell cluster detector**
  — using the spectral methods;
- **Diffusion-based smooth density function**
  — locally adaptive adjustment of clusters.
- **Dynamic net weighting scheme**
  — incorporating the wirelength and density information of every cluster into Backpropagation;

Physical Perspective

Initial placement diffuses from a **SOURCE** point.
Close-knit nets form **SINK** progressively.



(a) source



(b) sink

- **Cell Cluster Detector**

Poisson’ s Equation in ePlace:

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

$$\Downarrow$$

Electric Field Function:

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

$$\Rightarrow$$

Divergence Function:

$$\nabla \cdot \xi = \frac{\partial \xi_x}{\partial x} + \frac{\partial \xi_y}{\partial y},$$

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

- **Cell Cluster Detector**

Poisson's Equation in ePlace:

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

$$\Downarrow$$

Electric Field Function:

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

$$\Rightarrow$$

Divergence Function:

$$\nabla \cdot \xi = \frac{\partial \xi_x}{\partial x} + \frac{\partial \xi_y}{\partial y},$$

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

- **Diffusion-based Smooth Density Function**

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

$$\Downarrow$$

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

where K is the diffusion coefficient, its value depends on the degree of cluster in a design.

- **Diffusion-based Smooth Density Function**

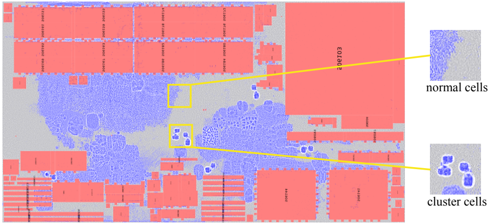
$$\hat{\rho}(x,y) = \rho(x,y) + div$$
$$\Downarrow$$
$$\hat{\rho}(x,y) = \rho(x,y) + K \times (\nabla \cdot \xi),$$

where K is the diffusion coefficient, its value depends on the degree of cluster in a design.

Net Weighting Scheme

2 Proposed Algorithm

normal cells & cluster cells



- Ensure fair force contribution.
- Normalize gradient.

Algorithm 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**

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

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

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

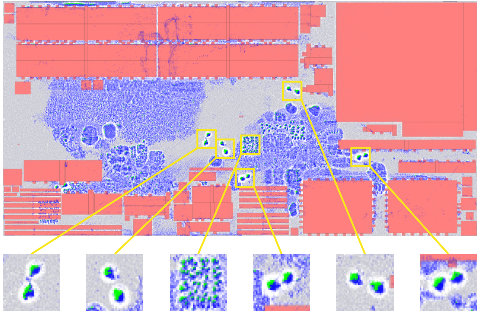
  $w_{cluster} \leftarrow w_{cluster} \times s$; ▷ Update the net weights

end for

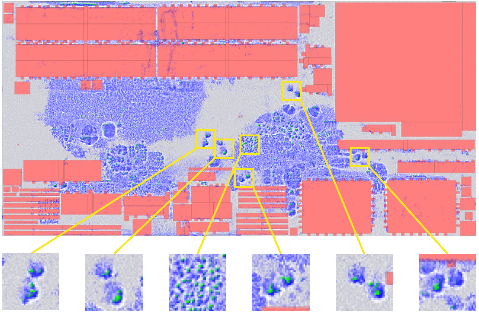
Outline

Cluster Detection and Smooth

3 Experimental Results & Conclusion



(a) iteration=500



(b) iteration=515

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

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[2]

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

[2] Yibo Lin et al. "DREAMPlace: Deep Learning Toolkit-Enabled GPU Acceleration for Modern VLSI Placement". In: *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems* 40.4 (2021), pp. 748–761.

Conclusion

- We propose a adaptive preconditioning algorithm for global placement guided by divergence analysis.
- Experimental results on the DAC-2012 benchmarks suite show that our adaptive preconditioning algorithm can make the best tradeoff between HPWL and iteration.
- This work provides new ideas and methodological support for solving placement challenges in complex integrated circuit design.

THANK YOU!