Regularity-aware Routability-driven Placement Prototyping Algorithm for Hierarchical Mixed-size Circuits

Jai-Ming Lin, Bo-Heng Yu, Li-Yen Chang

Speaker: Li-Yen Chang

From: NCKU SEDA LAB

Date: 2017/01

Outline

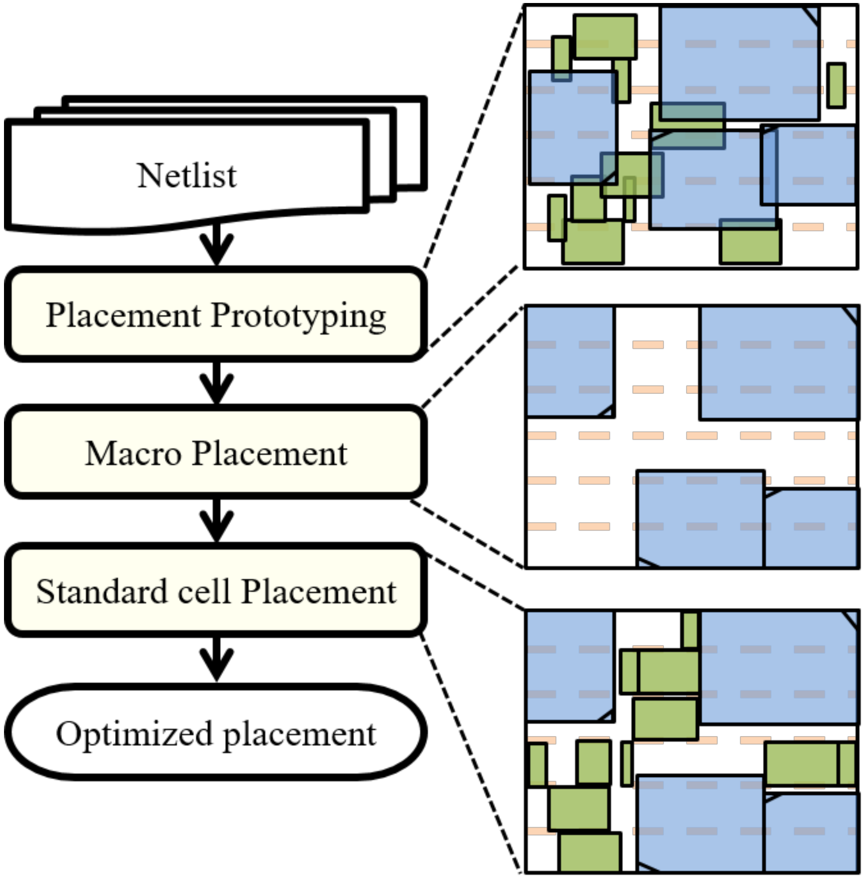
- Introduction
- Hierarchy Aware Clustering Algorithm for Mixed-Size Cells
  - Cost function for clustering large objects
  - Constraint for clustering two objects
- Overview of Our Methodology
  - Placement flow
  - Pre-coarsening
  - Post-processing
- Experimental Result
- Conclusion

Introduction

- Placement is a critical stage in today’s physical synthesis flow with tremendous impact on final performance of VLSI design
- Macro placement stills is the most time consuming step in the industry
  - Engineers still have to spend a lot of time to get feasible results of macros even though commercial tools already can obtain acceptable results for standard cells
- Consideration of macros placement includes
  - Pre-placed macros
  - Routability
  - Powerplanning

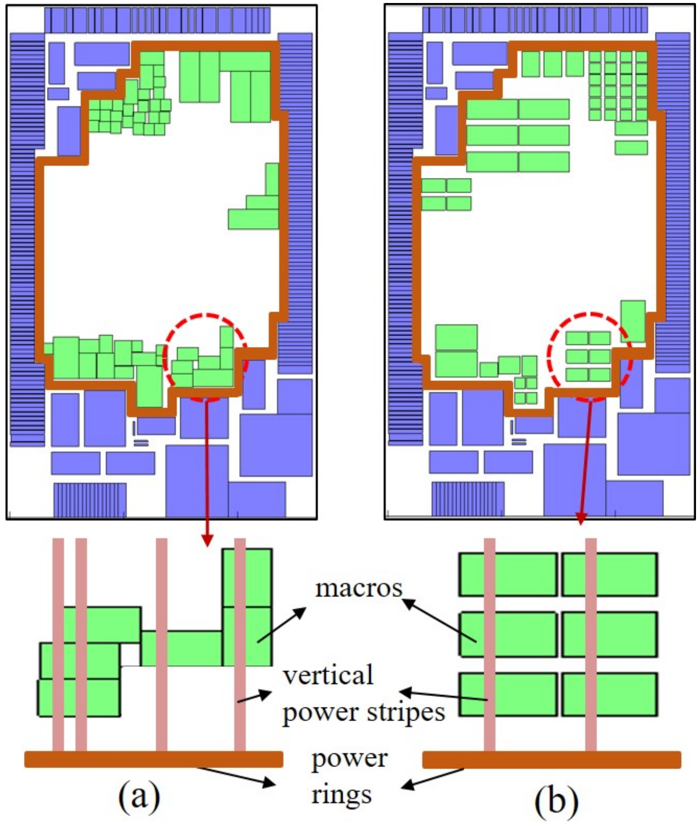
Three Stage Mixed-size Placement

- The three stage mixed-size placement algorithm is most suitable for commercial design flows since it divides placement into three steps:
  - Placement prototyping: distributes standard cells and macros over a placement regions while optimizing wirelength and routability
  - Macro placement: determine locations of macros
  - Standard cell placement: determine locations of standard cells
- Focus on placement prototyping with considering macro regularity



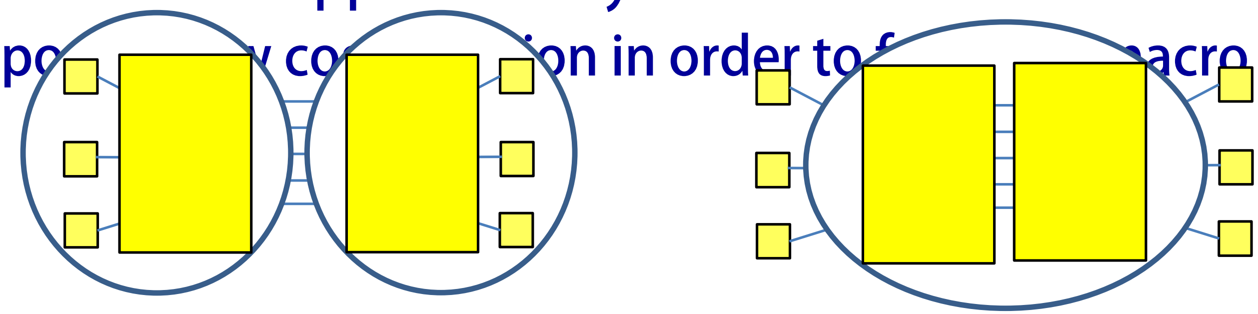
Placement Prototyping with Macro Regularity

- Placement prototyping is the most important step because
  - Its result determines the distribution of macros
  - The locations of macros directly influence standard cell placement
- Macro regularity is an important issue but is ignored by previous works
  - Regular macros: macros which are in the same types and have strong connections
- Place macro regularity can simplify powerplanning and improve routability



Contribution in Placement Prototyping

- Consider the regularity placement of macros
- Propose a three step clustering method which includes macro clustering, standard cell clustering, and mixed-cell clustering
  - The traditional approach only clusters standard cells
  - Propo[…] co[…]on in order to f[…]acro clustering



(a) Classic clustering result　　(b) Desirable result

- Consider nets of two objects which connect to pre-placed macros during clustering the two objects
  - We do not cluster two objects even thought they have strong connections if associated nets connect to fixed-pins which have longer distance

Cluster Score Function

- Let $\mathrm{m}(o_i, o_j)$ represent that two objects o_i and o_j are going to be merged
- A new cluster score function $s(o_i, o_j)$ for $\mathrm{m}(o_i, o_j)$ when o_i and o_j have larger sizes

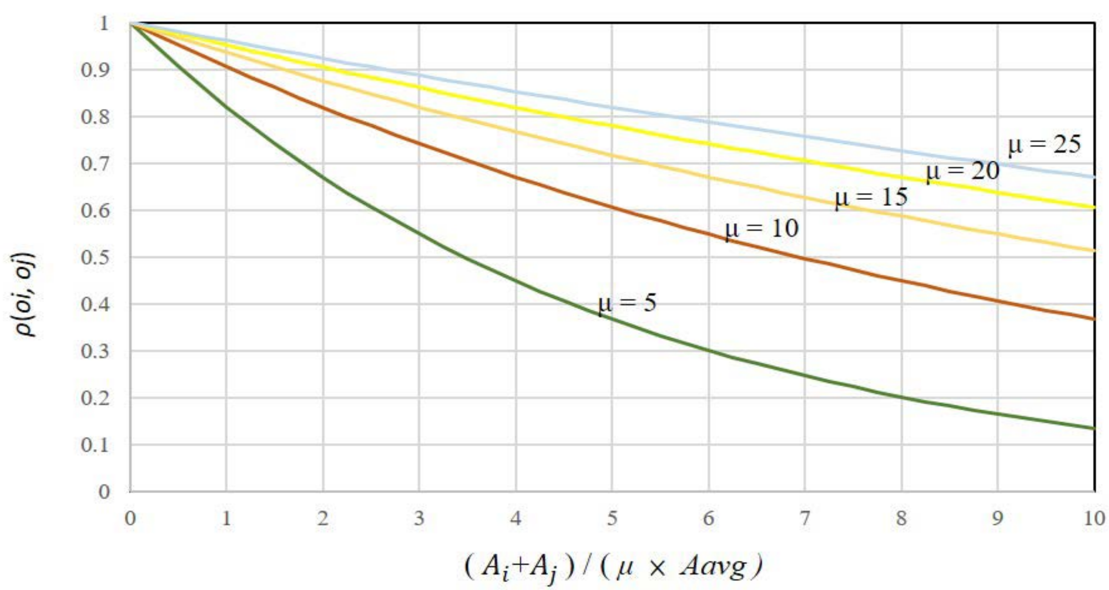
$$s(o_i, o_j) = \rho(o_i, o_j) \times [\alpha \times \sum_e w_e + \beta \times \Gamma(o_i, o_j)]$$

  - w_e denotes the weight of a hyper-edge *e*
    - $W_e = 1 / (\mathrm{d}(e)-1)$, d(*e*) denotes degree of a net
  - $\Gamma(o_i, o_j)$: the affinity of o_i and o_j which denotes the logical relation between the two objects
    - $\Gamma(o_i, o_j) = \varepsilon \times \omega_{i,j}$, where $\omega_{i,j}$ is the number of common parts of name between the o_i and o_j and ε is user-defined parameter
  - $\rho(o_i, o_j)$ is a penalty function about combined area which adjusts their clustering score

- Use a penalty function to decline the impact of large combine area

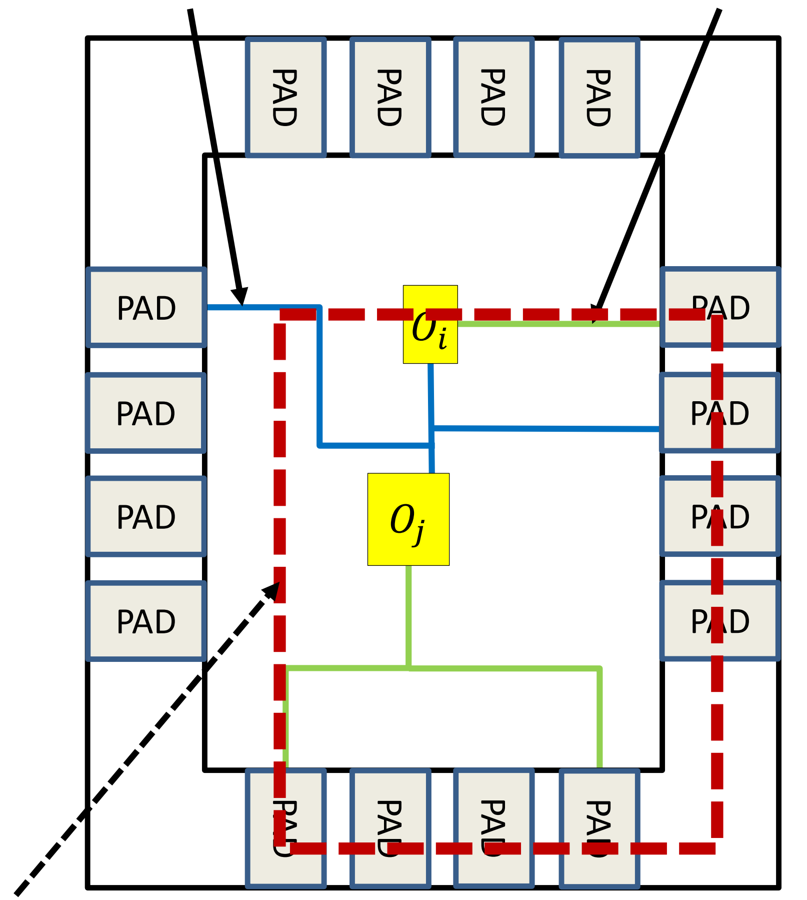
$$\text{Penalty:}\quad \rho = e^{\frac{-(A_i+A_j)}{A_{avg}*\mu}}$$

  - A_i (A_j) denotes area of o_i(o_j)
  - A_{avg} denotes average area in each cluster in current iteration
  - μ is a pa

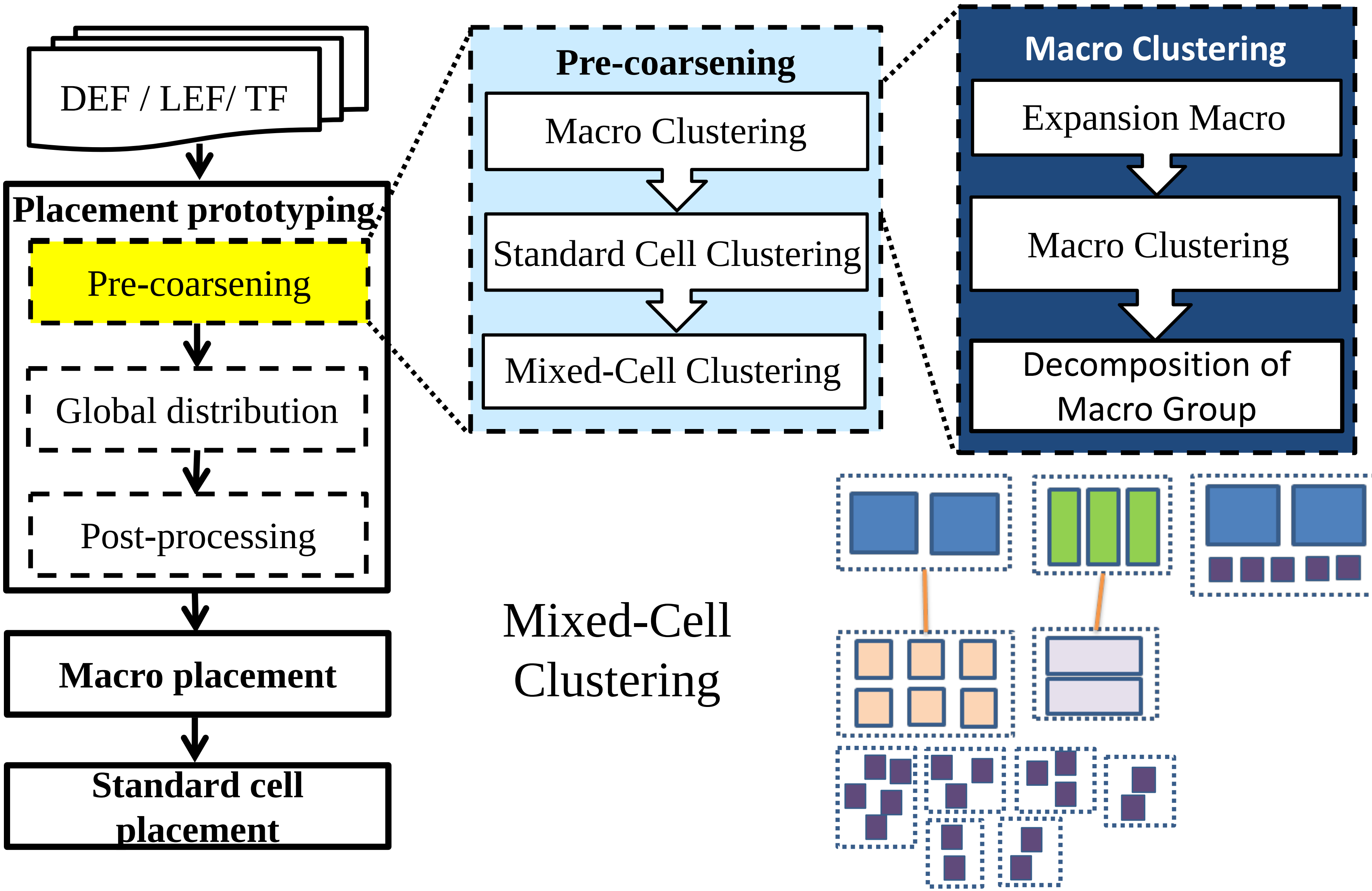


Bounding Box Constraint (B.B.C) for Clustering Objects

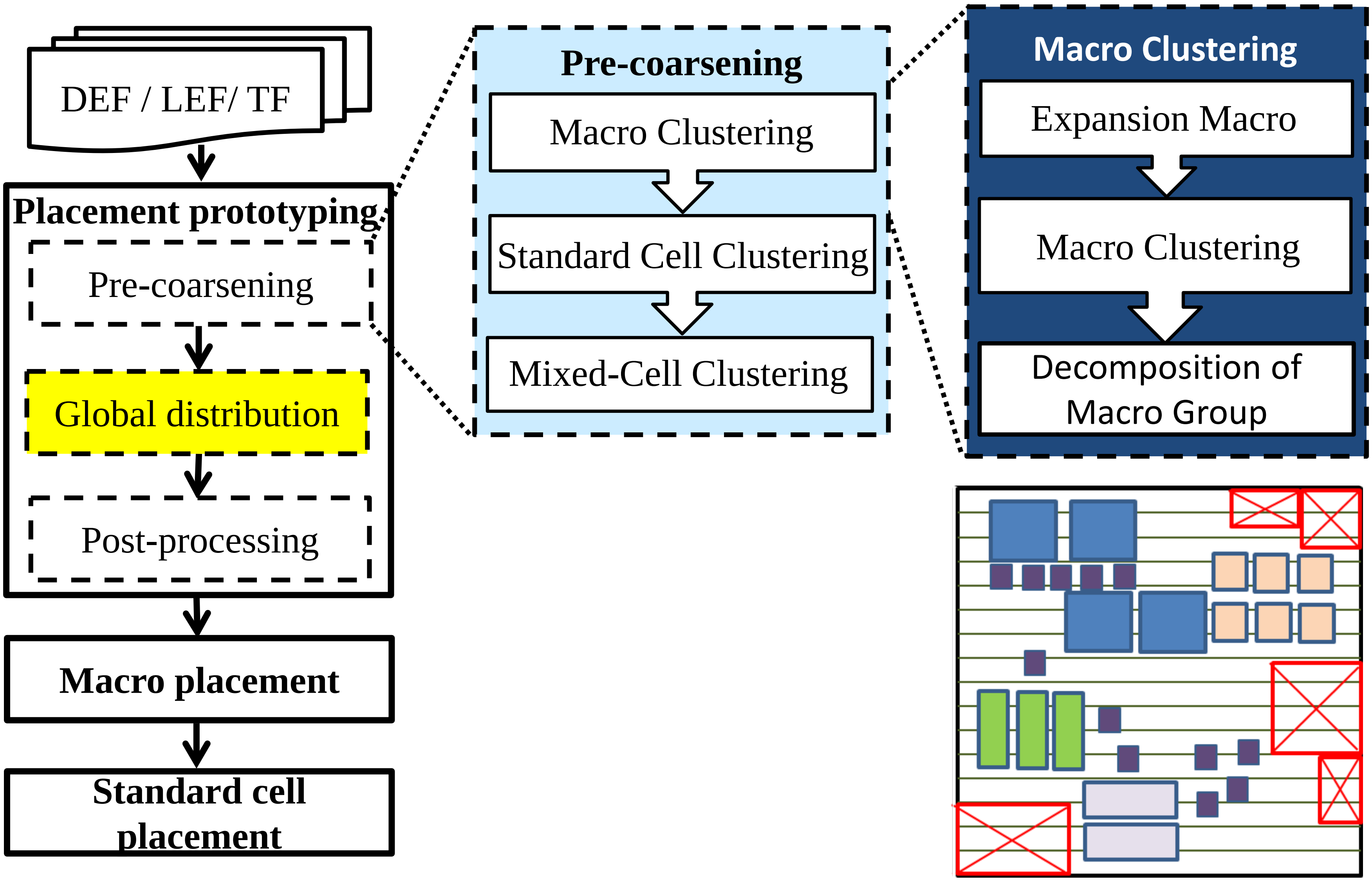
- Use a bounding box constraint to improve wirelength
  - A net which connects to fixed pins is called a bounding net
  - An related net for $\mathrm{m}(o_i, o_j)$ which has fixed pins is called a related bounding net
  - An unrelated net for $\mathrm{m}(o_i, o_j)$ which has fixed pins is called a unrelated bounding net
  - Two objects o_i and o_j cannot be merged if half parameter of unrelated bounding net bounding box nets is large than L/λ
    - L: chip outline
    - λ: user-specified value



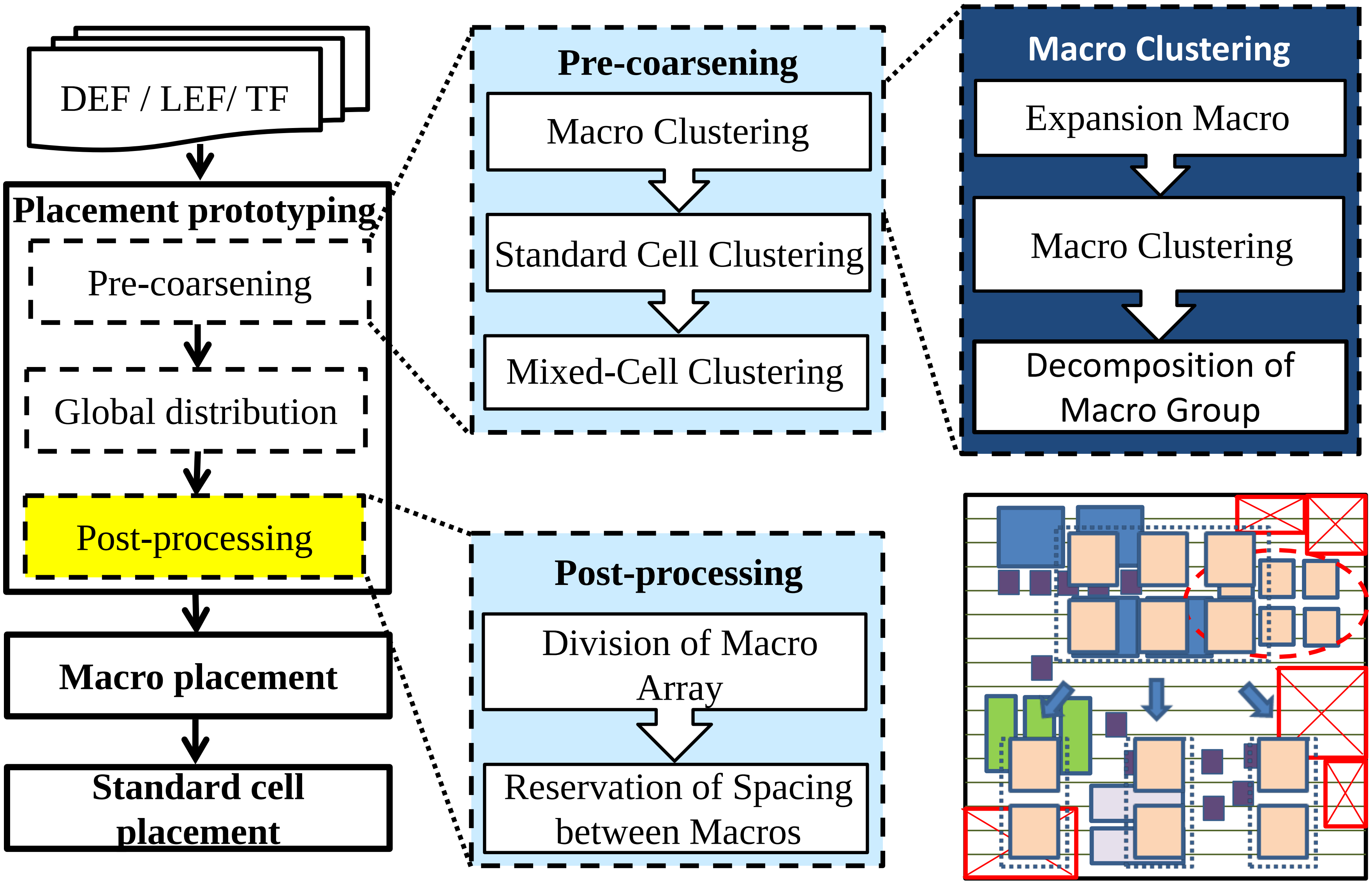
Placement Prototyping Flow



Placement Prototyping Flow

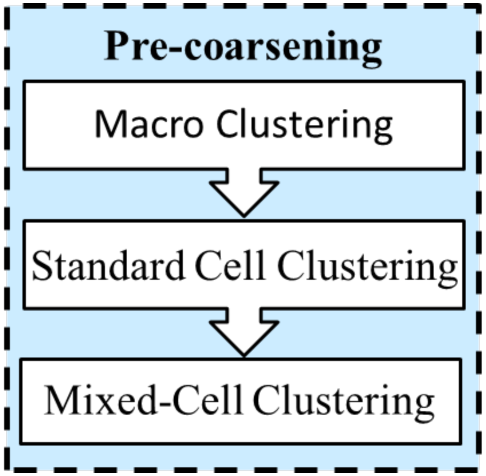


Placement Prototyping Flow



Pre-coarsening

- Pre-coarsening is to enhance the speed of global distribution
  - Step 1: Cluster macros to generate macro groups (MGs) for considering regular macros
    - Apply best choice
    - Use our score function with B.B.C
  - Step 2: Merged standard cell to form standard cell groups (SGs) before clustering standard cells with MGs
    - Apply first choice
    - Use traditional score function with B.B.C
  - Step 3: MGs and SGs are clustered to generate mixed-cell groups (XGs)
    - Apply first choice
    - Use our score function with B.B.C

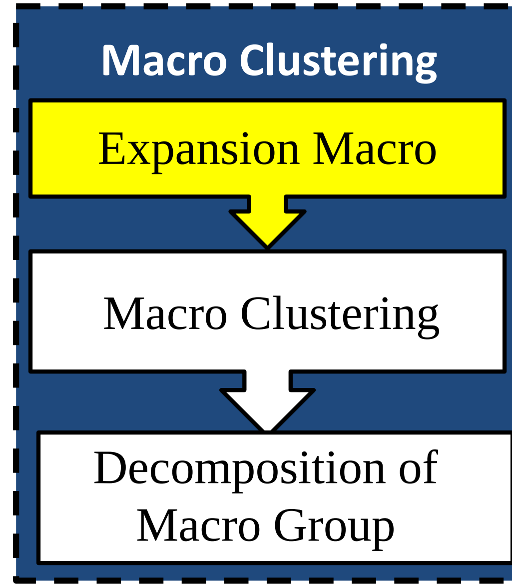


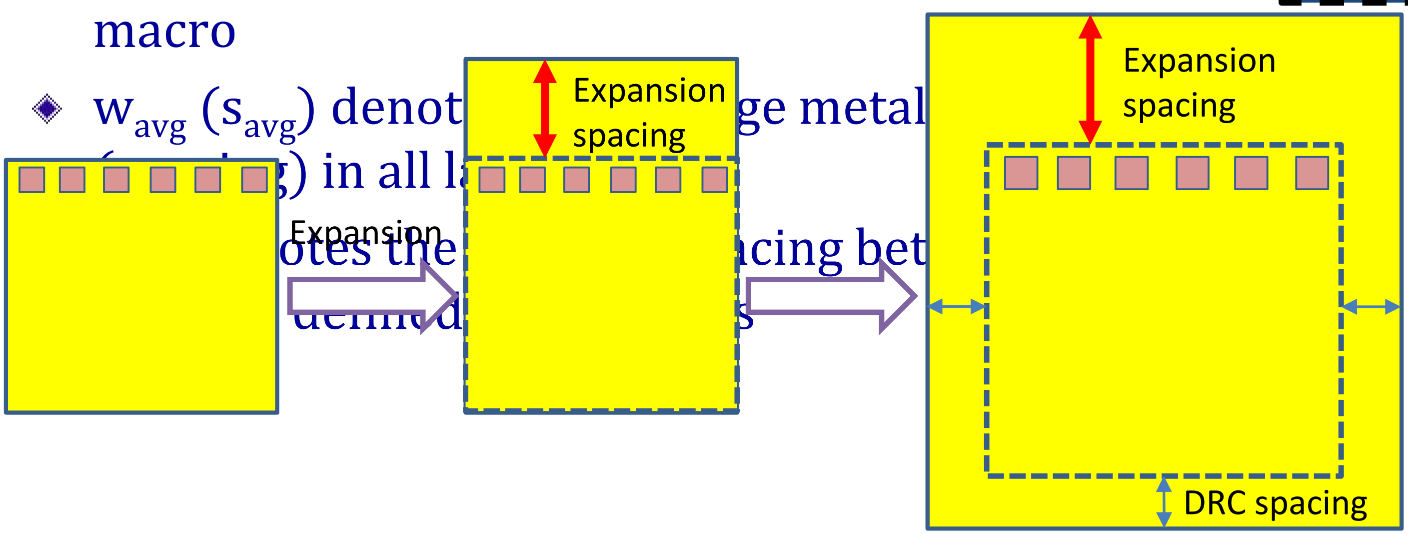
Expansion Macro

- Reserve spacing around macros to improve routability of macros
- e(i) determins the spacing in the i-th side of a macro, where i= {left, right, bottom, top}

$$e(i) = \gamma \times \left(N_p^i \times \left(w_{avg} + s_{avg}\right) + s_{avg} + s_{\mathrm{DRC}}\right)$$

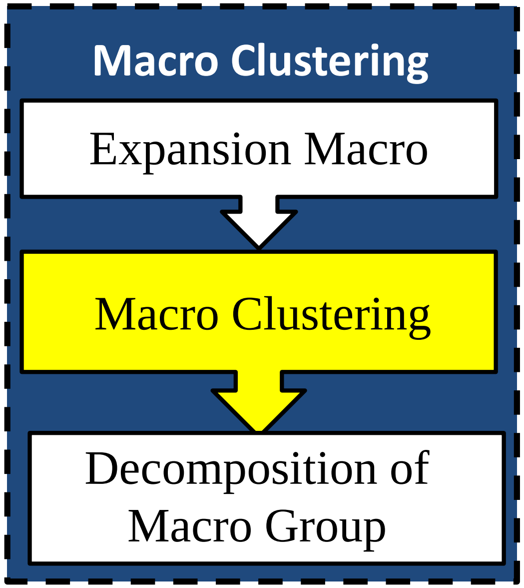
  - N_p^i denotes number of pins in the i-th side of the macro
  - w_{avg} (s_{avg}) denot ... ge metal ... g) in all l...
  - ... otes the ... acing bet... de... d... s





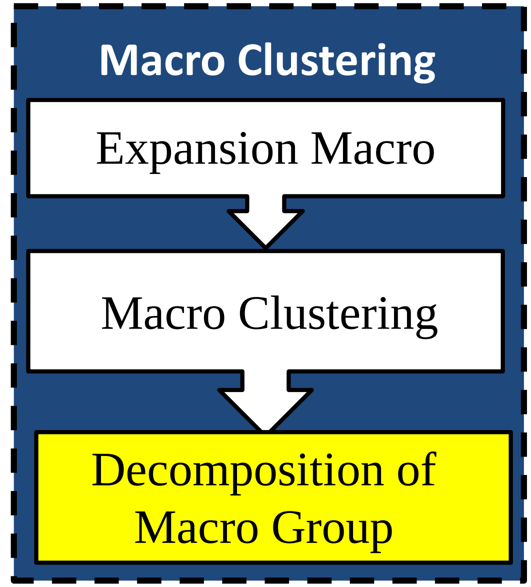
Macro Clustering

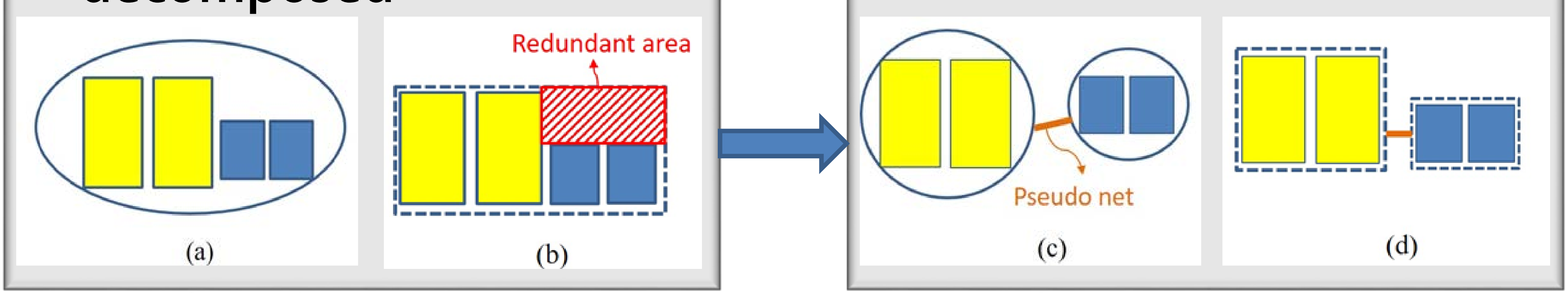
- Apply Best Choice combined with our score function to handle macro clustering
  - Best Choice can provide a high quality clustering solution, but waste time.
    - Since number of macros is significantly less than that of standard cells, its runtime is still acceptable.
  - Our score function is able to cluster two macros with large combined area when they have strong connectives and similar design hierarchies.



Decomposition of Macro Group

- Decompose a macro cluster into sub-clusters according to their types if the cluster has different type macros
  - The actual area of macros in a cluster cannot be calculated correctly which is required during global placement
- Add a pseudo net to connect sub-clusters if a macro cluster is decomposed



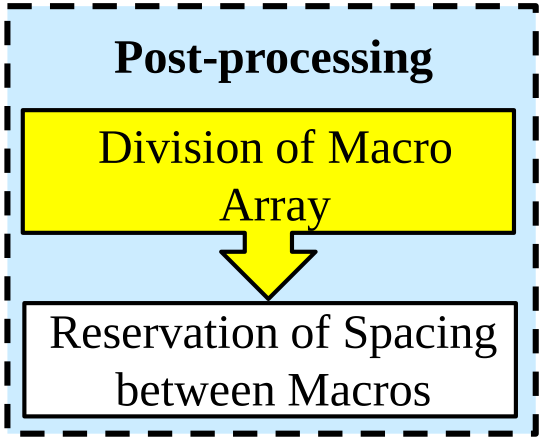


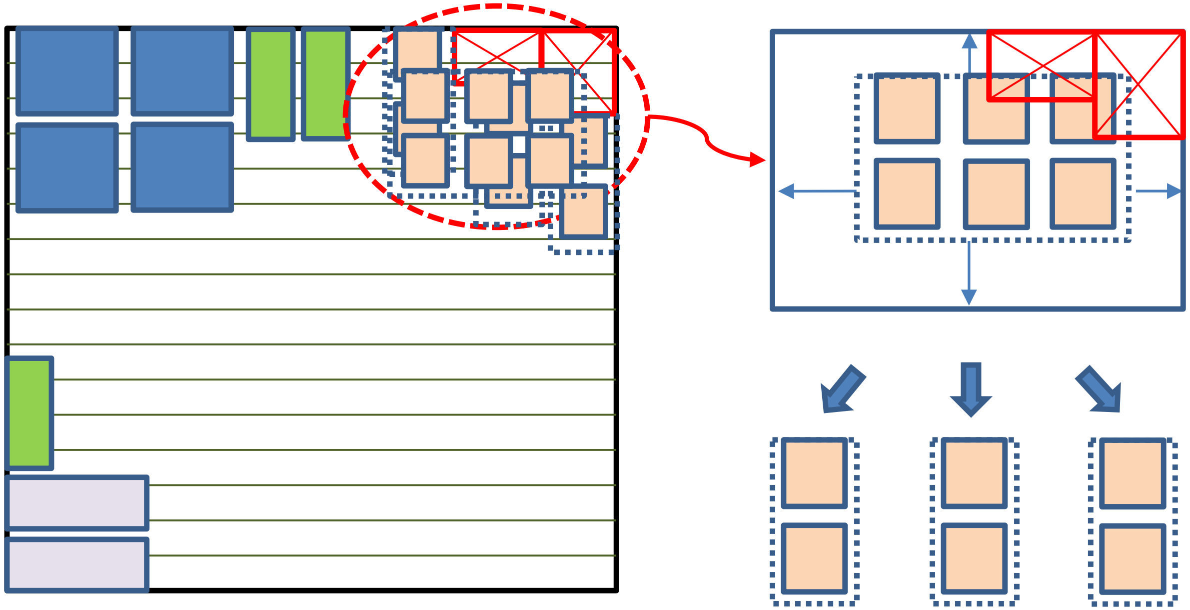
Before decomposition

After decomposition

Division of Macro Array

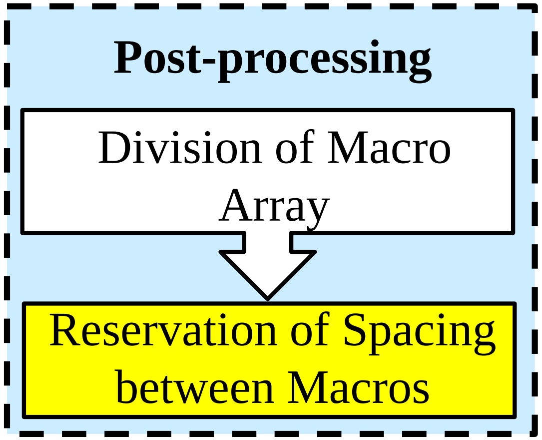
- Determine whether a macro array should be divided according the following conditions:
  - The number of macros in an array is larger than a user-specified value λ_1
  - The number of pre-placed macros around an array is larger than a user-specified value λ_2





Reservation of Spacing between Macros

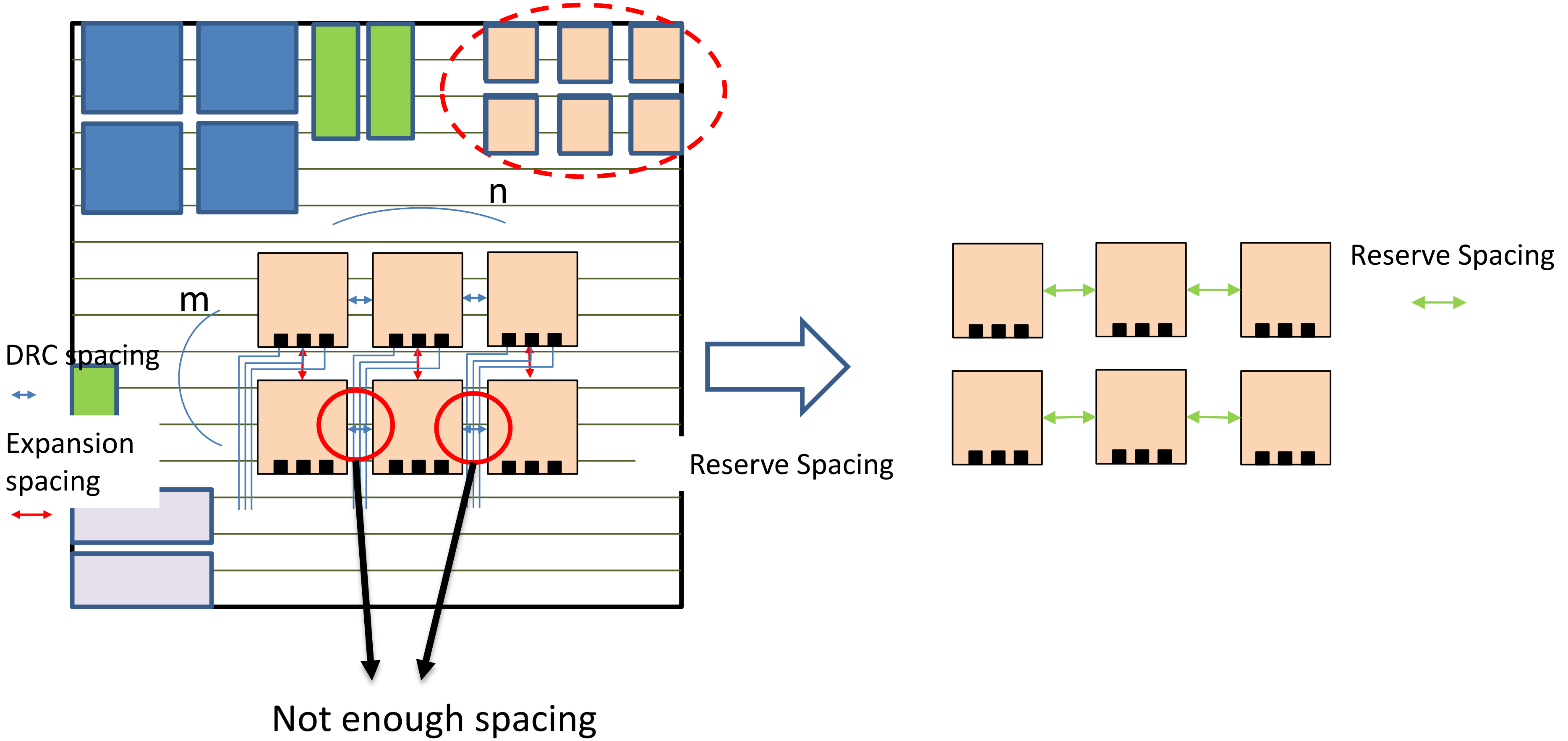
- Given a macro array whose dimension is $m \times n$, we will keep $n+1$ $(m+1)$ vertical (horizontal) channels, where m (n) is number of rows (columns)
- The width w_h of a vertical channel is computed by the following equation:



$$w_h = \{\frac{m \times n \times (N_p^{top} + N_p^{bottom})}{n+1} \times (w_{avg} + s_{avg}) + s_{avg}\} + \delta \times C$$

  - N_p^{top} and N_p^{bottom} respectively denote number of pins in the top and bottom sides of a macro in the cluster
  - w_{avg} (s_{avg}) denotes the average metal width (spacing) in all layers
  - C denote the congestion of the bin of the array where it belongs to and the value is calculated by FLUTE

Reservation of Spacing between Macros



Experimental Environment

- C/C++ language
- IBM x3250 M2, Linux server with Intel Xeon 2.27GHz
- 90GB Memory
- Four industrial benchmark
- Routed wirelength and overflow are evaluated by IC Compiler

Circuit	# Movable macros	# Pre-placed macros	# IO pads	# Standard cells	# Nets
Circuit 1	30	13	130	157K	181K
Circuit 2	55	15	219	232K	235K
Circuit 3	71	47	365	1098K	1126K
Circuit 4	38	15	169	321K	327K

Experimental Result

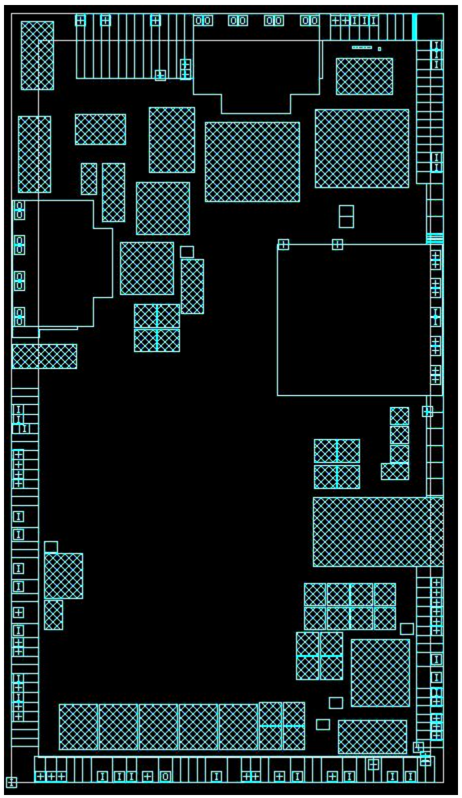
- Our experiment can be divided into two parts
  - Demonstrate the impact of the macro regularity
  - Show the effect of bounding box constraint

Circuit	w/o Regularity Constraint					w/ Regularity Constraint					
	Tool		w/o Our Methodology			w/o B.B. Constraint			w/ B.B. Constraint		
	# of O.F	WL ($10^7 \mu m$)	# of O.F	WL ($10^7 \mu m$)	T (s)	# of O.F	WL ($10^7 \mu m$)	T (s)	# of O.F	WL ($10^7 \mu m$)	T (s)
Circuit 1	636	1.31	31	1.15	1243	31	1.18	1260	31	1.06	1260
Circuit 2	1086	1.40	775	1.31	22.7	727	1.27	24.2	746	1.24	24.2
Circuit 3	9755	7.41	293	6.56	120	264	6.77	123	267	6.32	133
Circuit 4	1071	1.84	3225	1.74	62.7	1090	1.73	61.3	293	1.70	65.5
Normalized	9.39	1.15	3.23	1.04	0.97	1.58	1.06	0.98	1	1	1

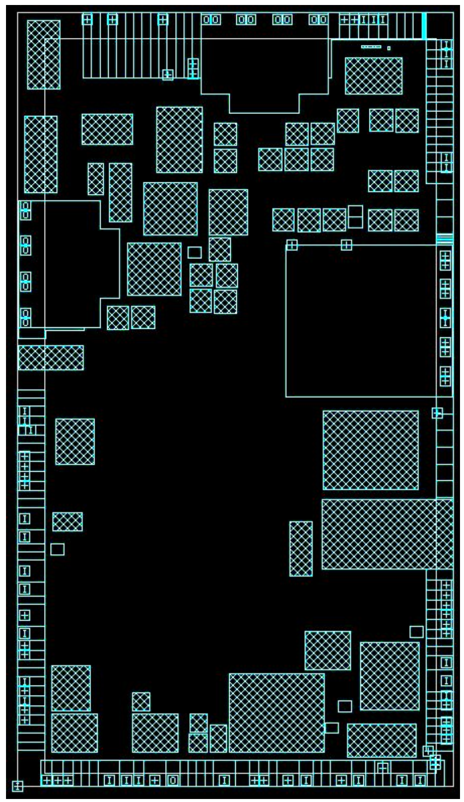
**w/o our methodology* just cluster the standard cell by first choice

Experimental Result

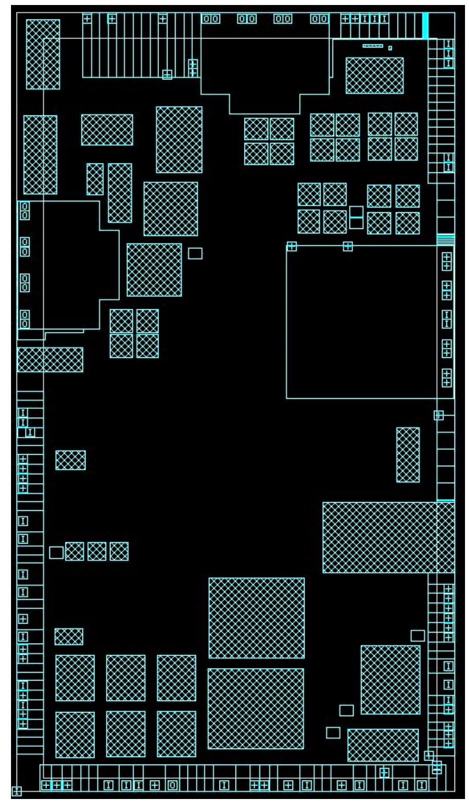
- Circuit 4 : Final layouts of macro placement



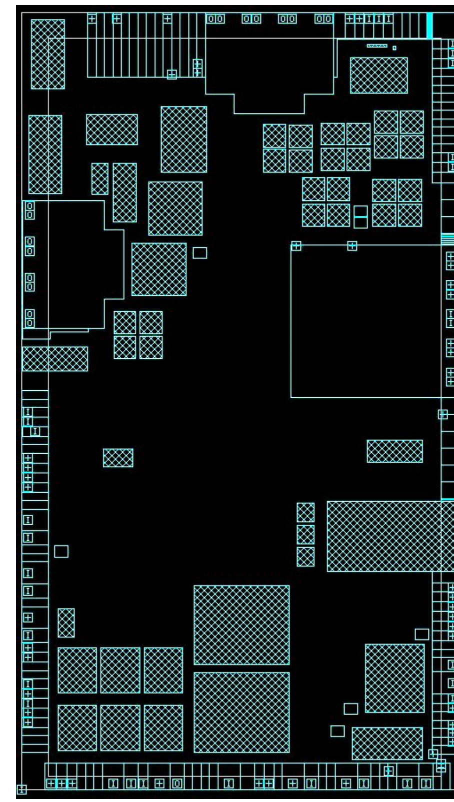
(a) Tool
WL: 1.84 ($10^7 \mu$m)



(b) w/o Our Methodology
WL: 1.74 ($10^7 \mu$m)

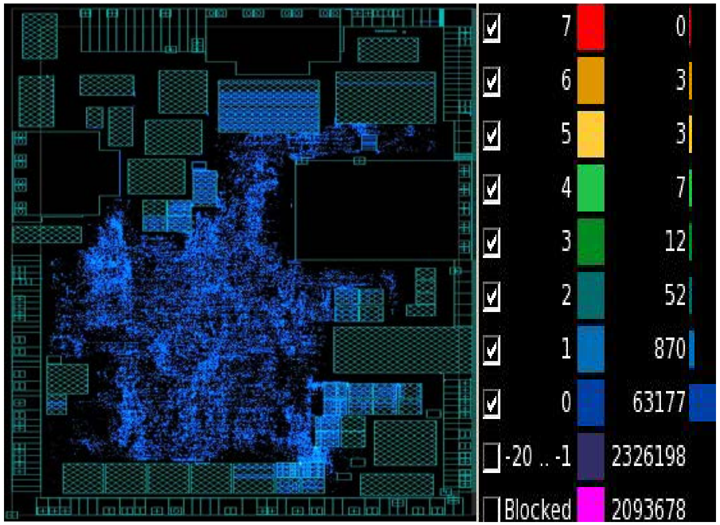


(c) Our w/o B.B.C
WL: 1.73 ($10^7 \mu$m)

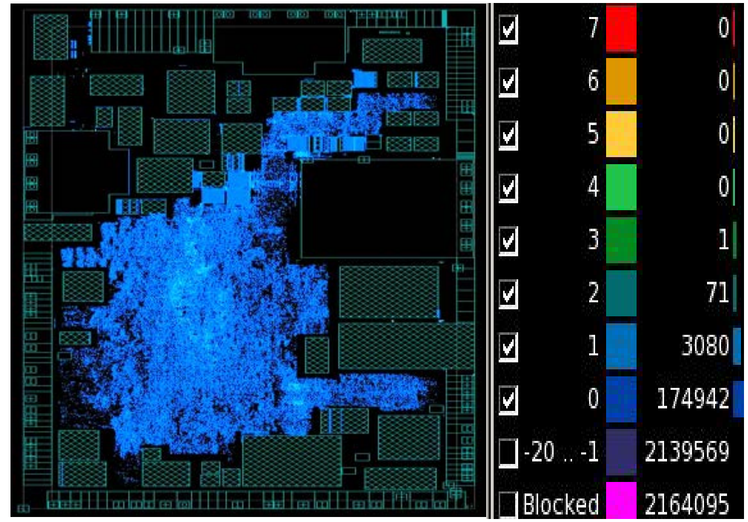


(d) Our w/ B.B.C
WL: 1.70 ($10^7 \mu$m)

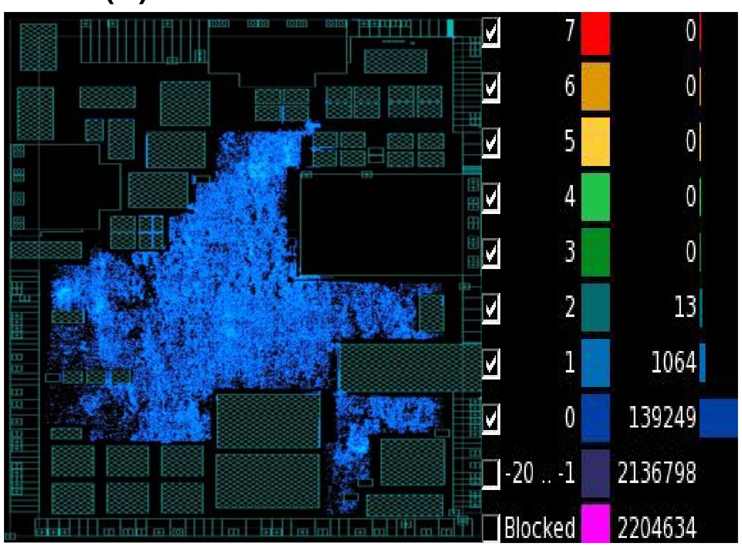
Global Congestion Map



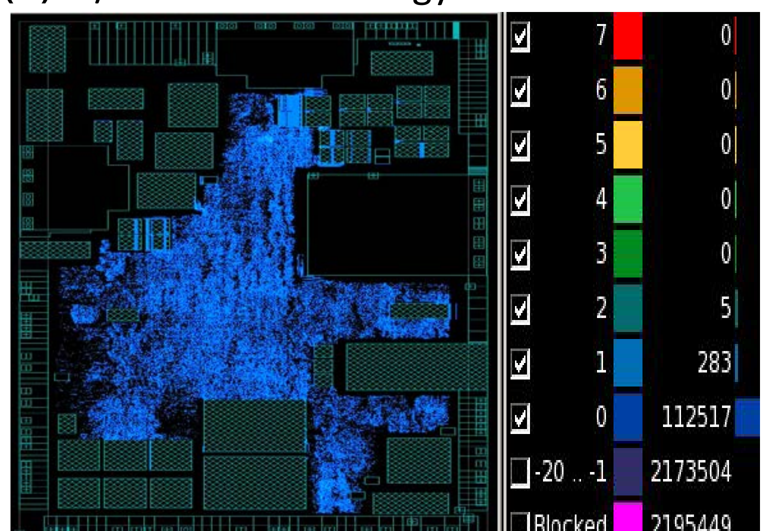
(a) **Tool with** 1071 overflow



(b) w/o Our Methodology with 3225 overflow



(c) Our w/o B.B.C with 1090 overflow



(d) Our w/ B.B.C with 293 overflow

Conclusion

- Propose a routability-driven placement prototyping methodology which considers macro regularity in order to facilitate powerplanning
  - Use a three step coarsening methodology to cluster mixed-size cells
  - Propose a new clustering score function to facilitate cluster mixed-size cells
  - Propose a bounding box constraint to avoid cluster improper objects which may increase wirelength

- The experimental results show that the proposed algorithm can further improve the routing overflow and routing wirelength