《芯片设计自动化与智能优化》
Legalization

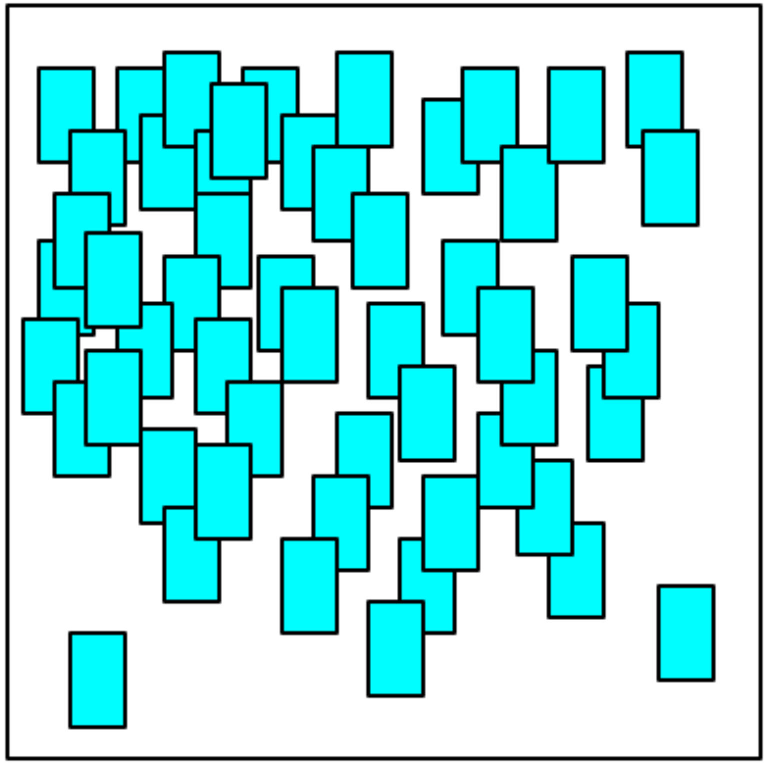
The slides are partly based on Prof. David Z. Pan's lecture notes at UT Austin.

Yibo Lin

Peking University

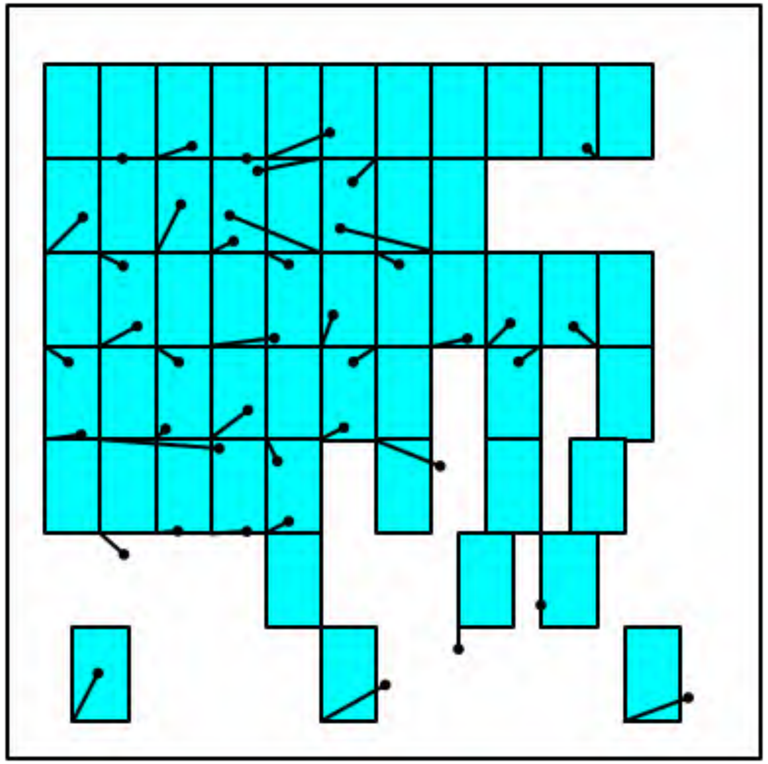
Typical Placement Flow

WL: 1.00e+6



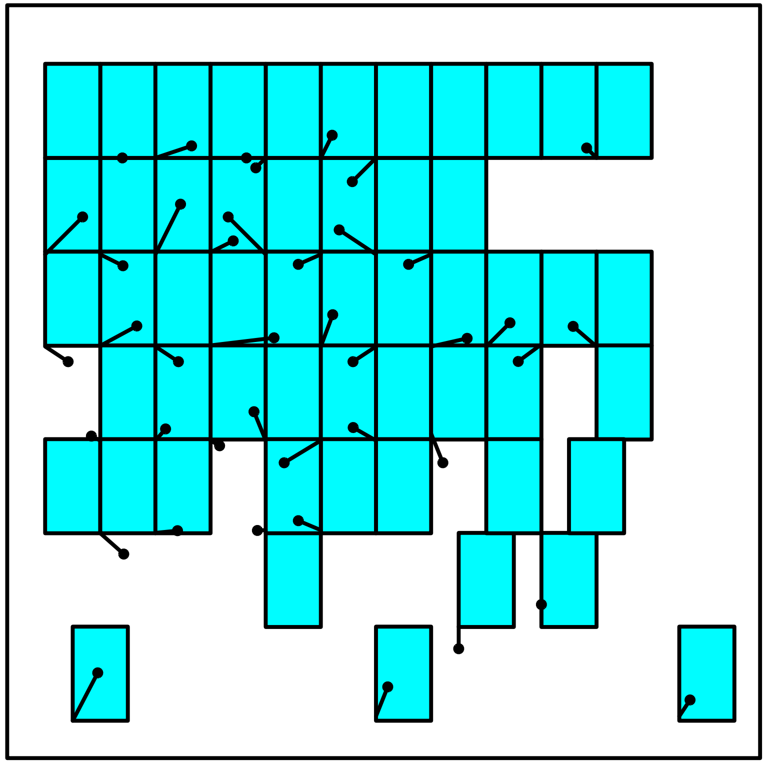
Global placement

WL: 1.05e+6



Legalization

WL: 1.02e+6



Detailed Placement

Outline

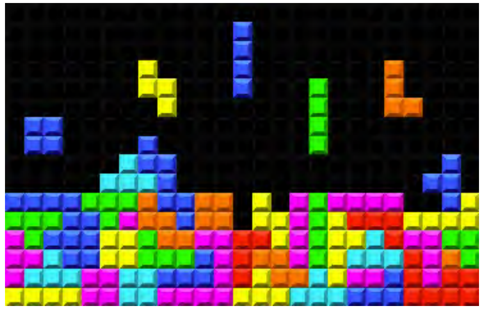
- What is placement
- History of placement algorithms
- Global placement
  - Quadratic placement: FastPlace & SimPL
  - Nonlinear placement: NTUplace & ePlace
- **Legalization**
  - Tetris
  - Row-based algorithms: Abacus, DP, LP, MCF
  - Integer linear programming
- Detailed placement
  - Global move & swap
  - Independent set matching
  - Local reordering
  - Row-based algorithms: DP, LP, MCF
- Other topics
  - Routability-driven placement
  - Timing-driven placement
  - Macro placement

Legalization – Problem Formulation

- Input
  - Global placement solution
- Output
  - Determine block locations
  - Remove overlaps between blocks
  - Satisfy all design rules
- Objective
  - Minimize total displacement

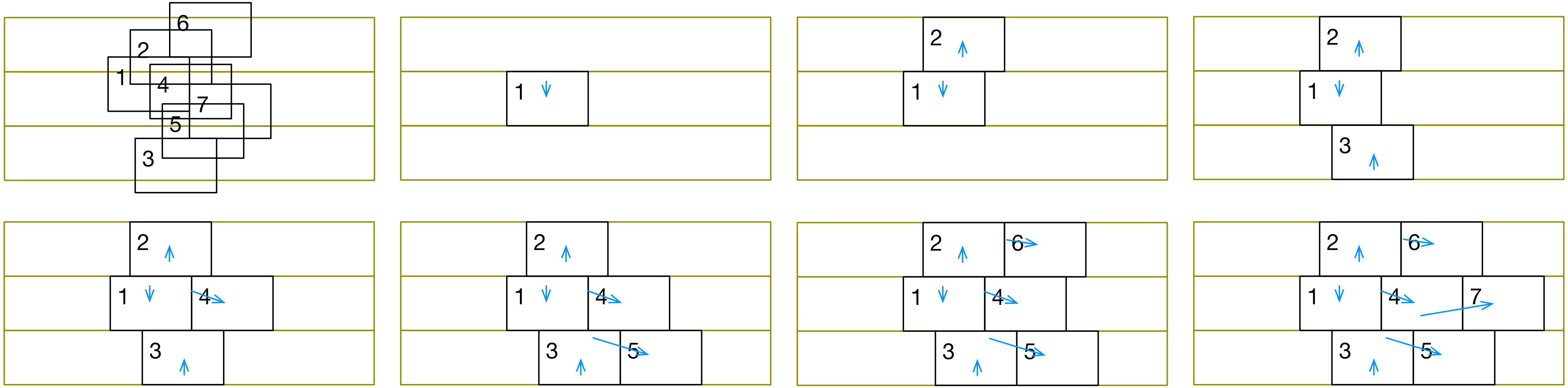
Tetris – Greedy Approach

- Sort cells from left to right
- Place cells one-by-one

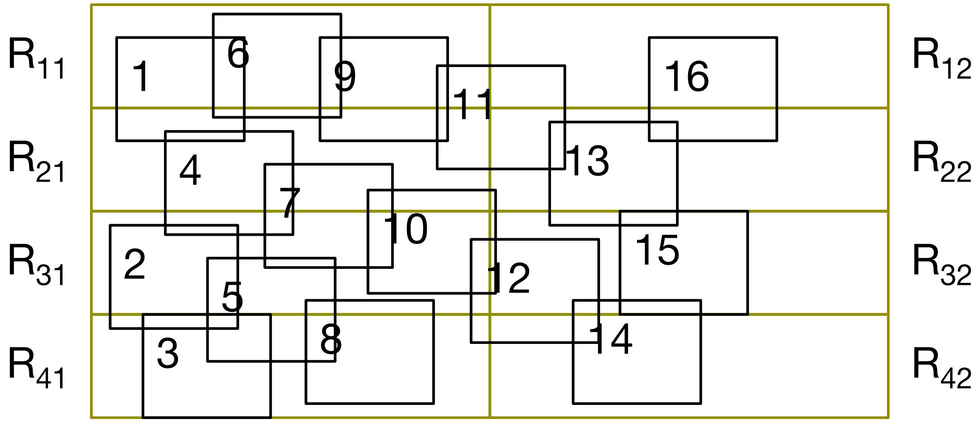


Simple, but effective
[NTUplace3, TCAD2018]

Global placement

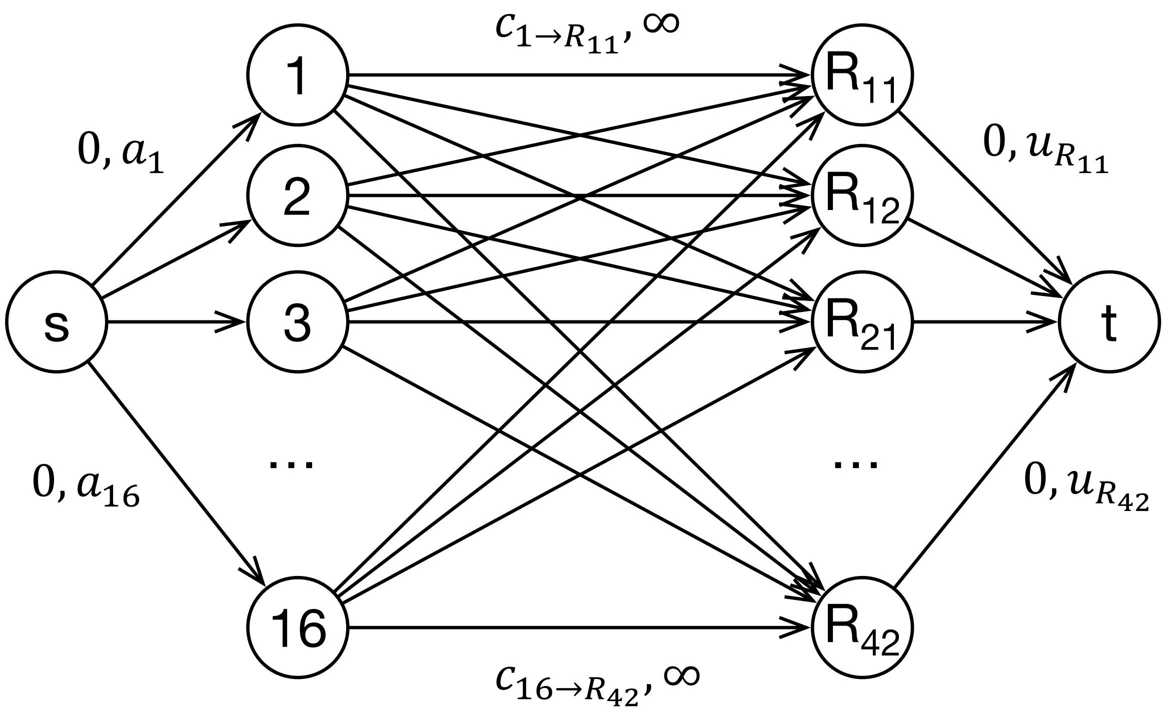


Region Assignment with Min-Cost Flow



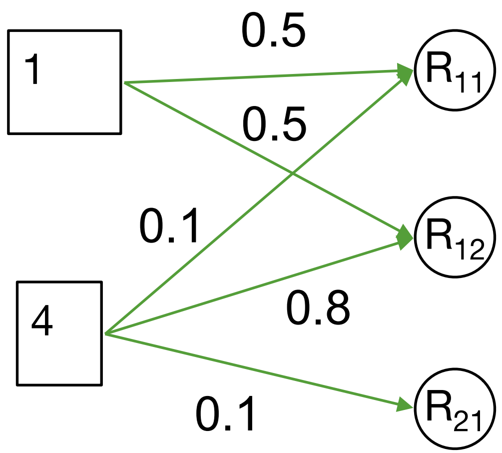
Edge cost
$c = \widehat{disp}$

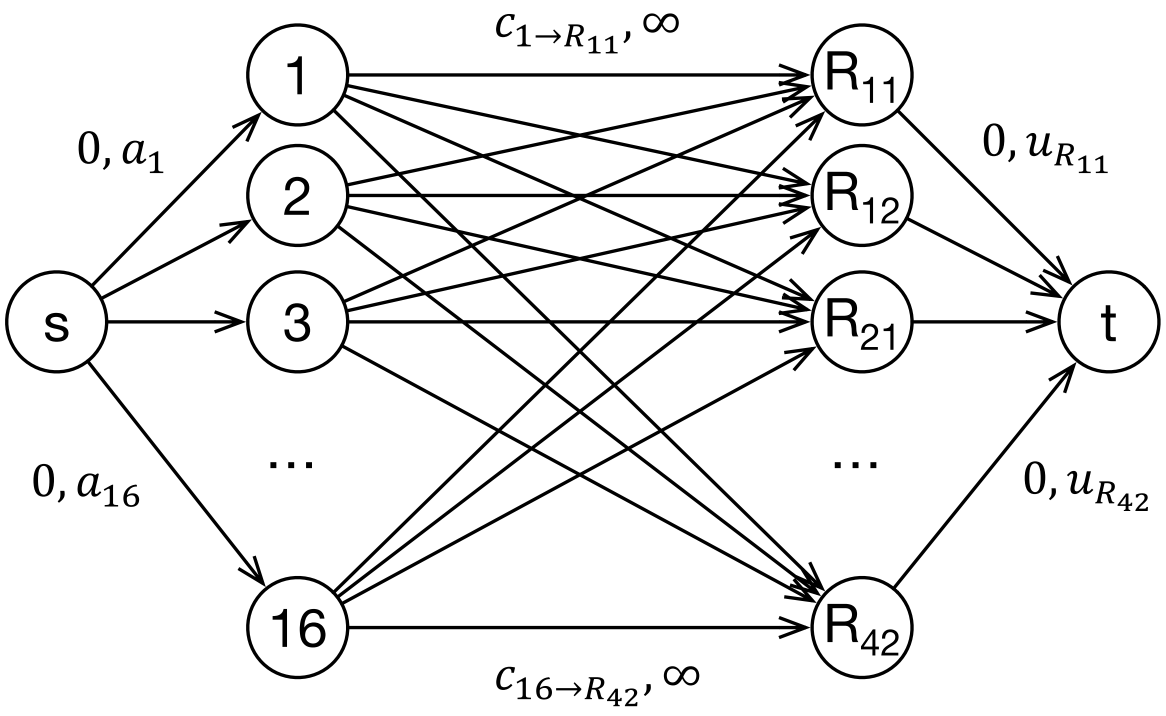
Center-to-center distance?
Sensitive to region sizes?



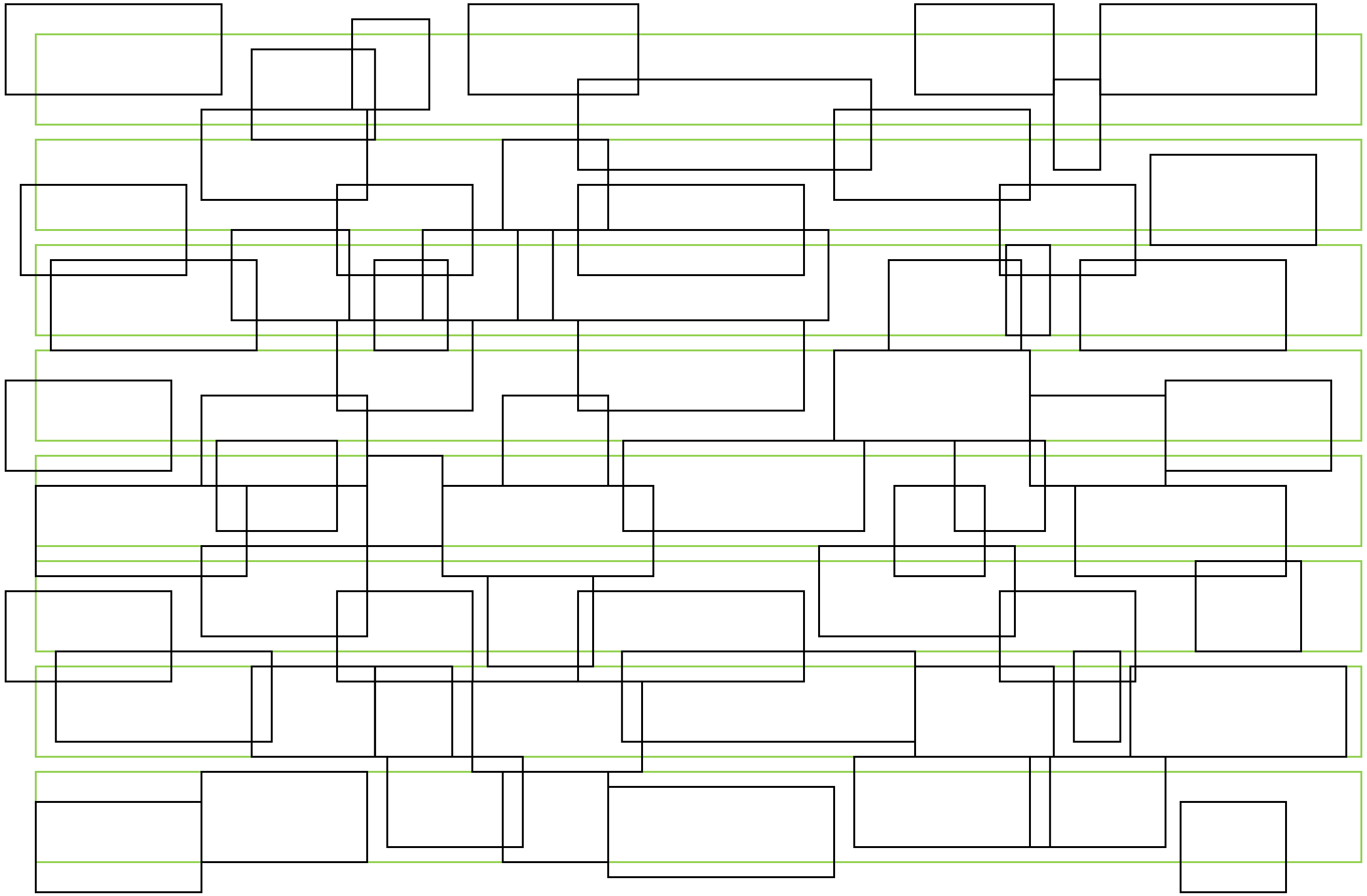
Region Assignment with Min-Cost Flow

A cell may be assigned to multiple regions

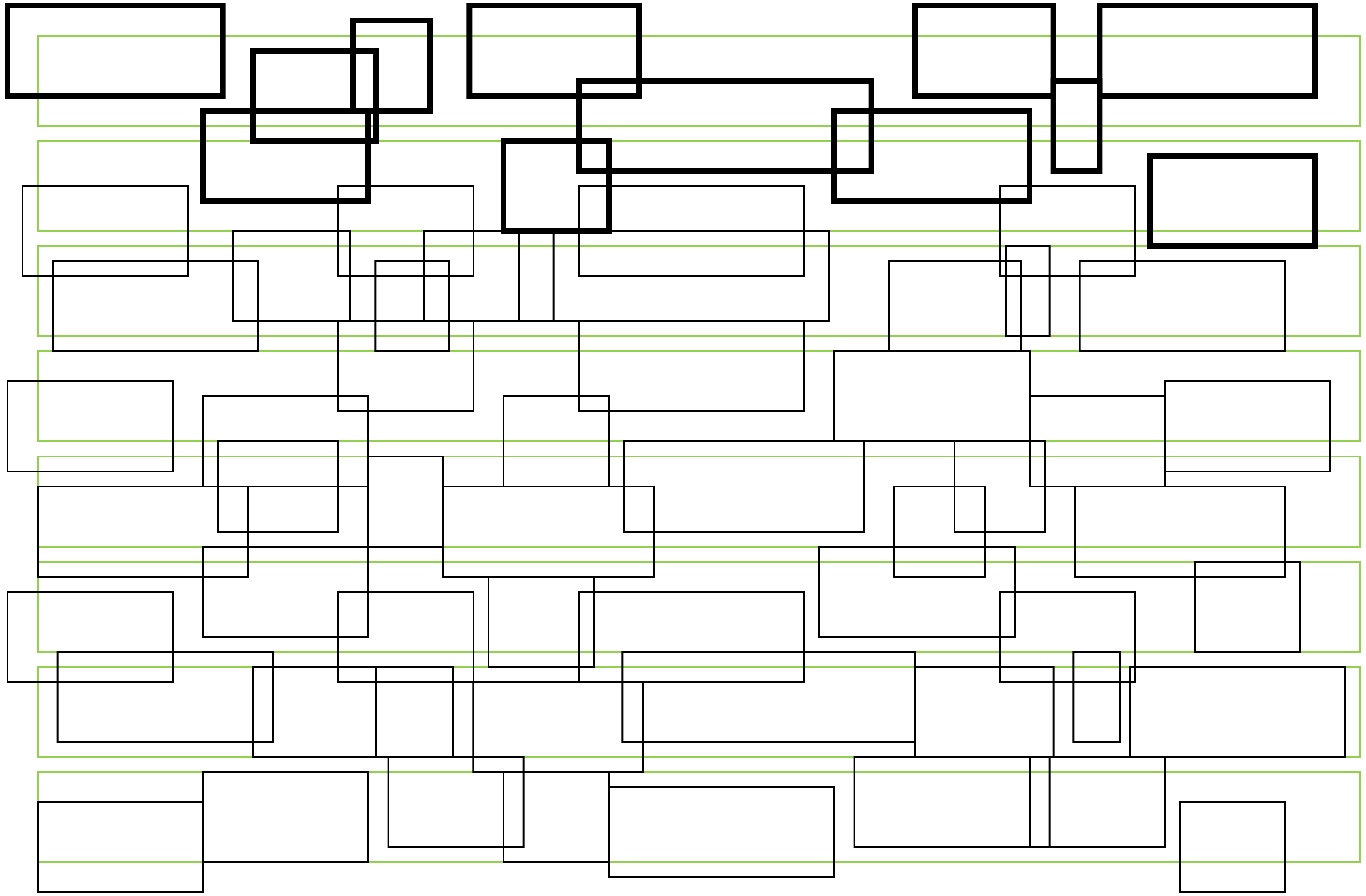




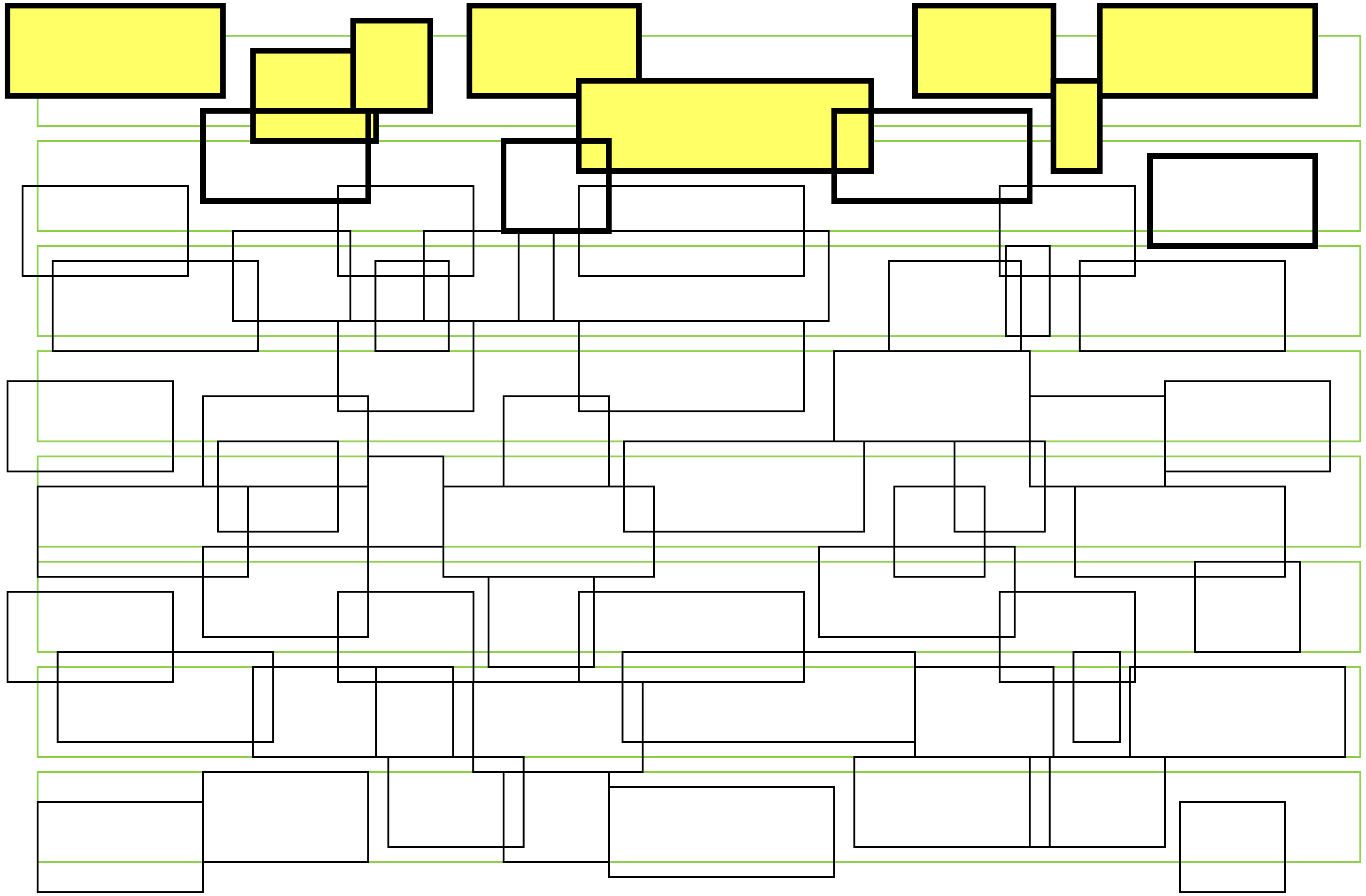
Row-by-Row Legalization



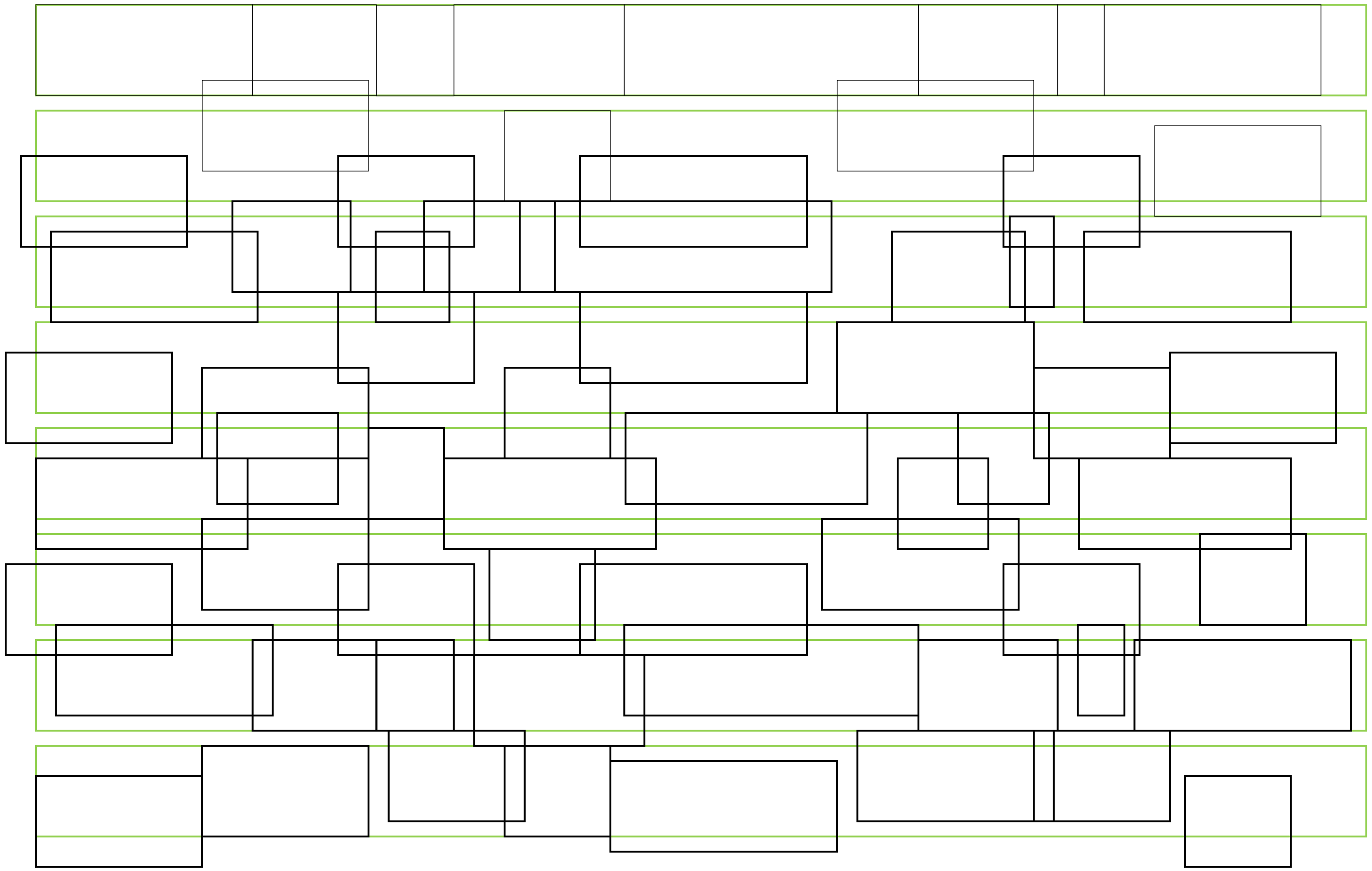
Row-by-Row Legalization



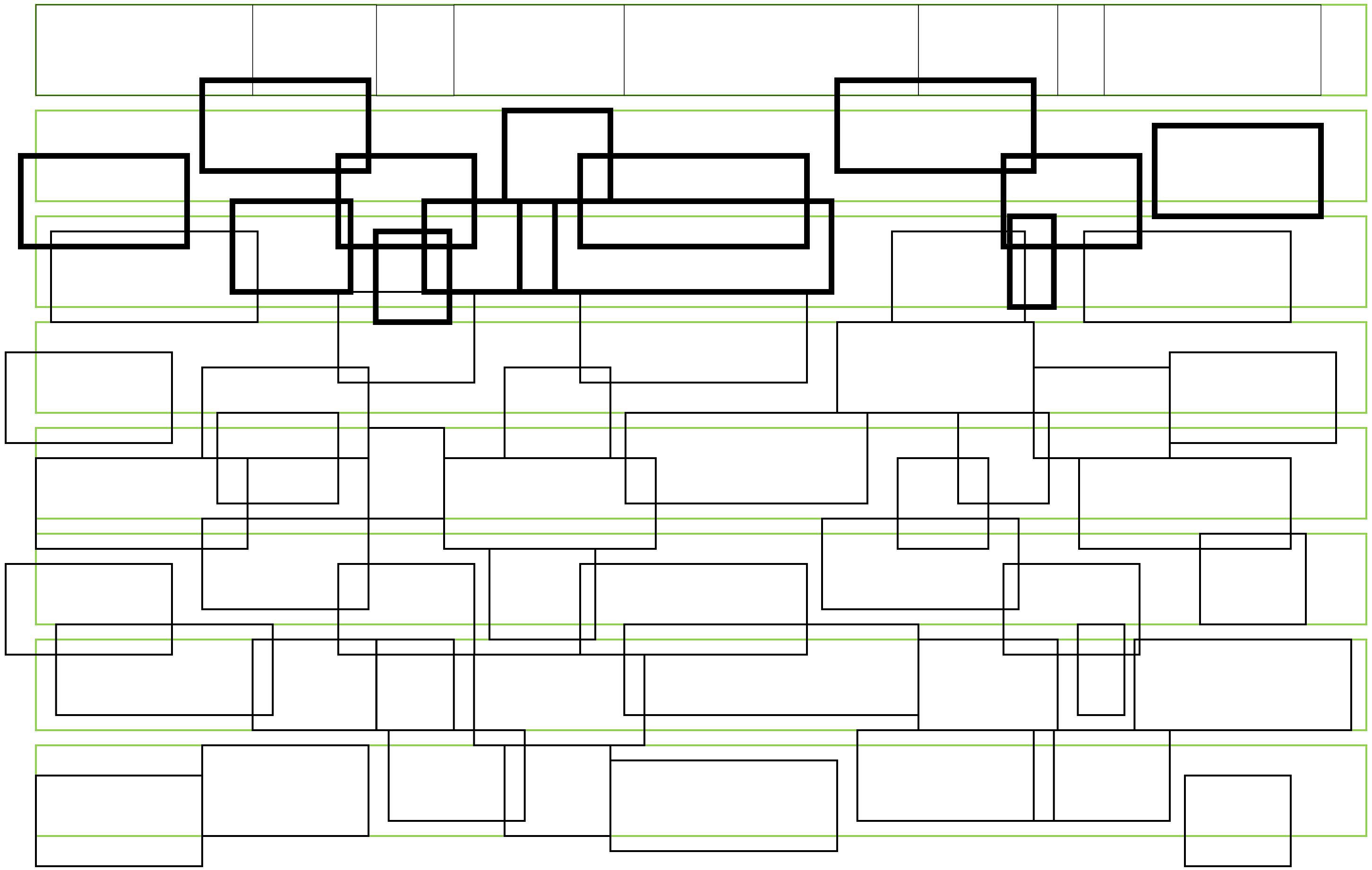
Row-by-Row Legalization



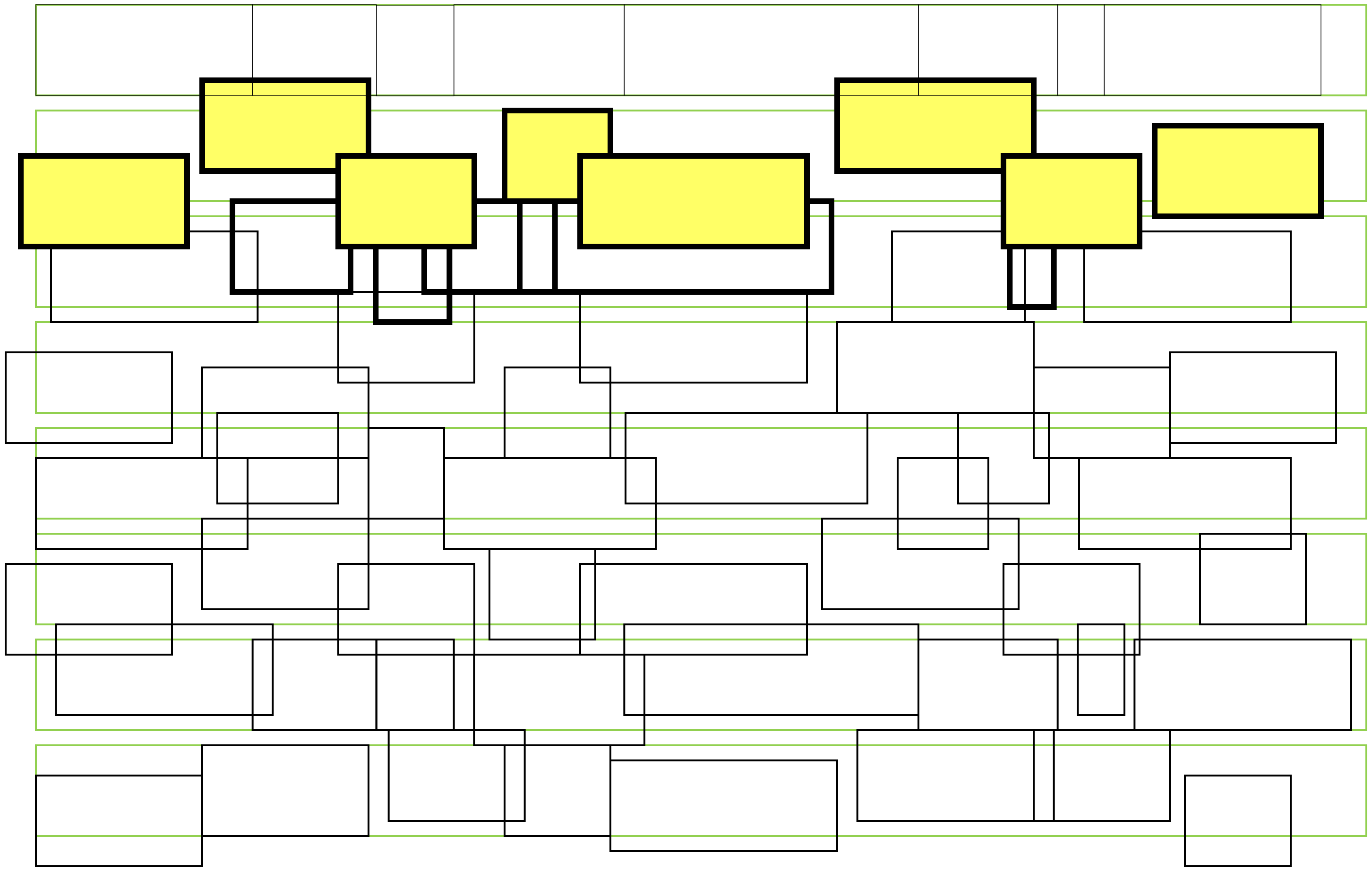
Row-by-Row Legalization



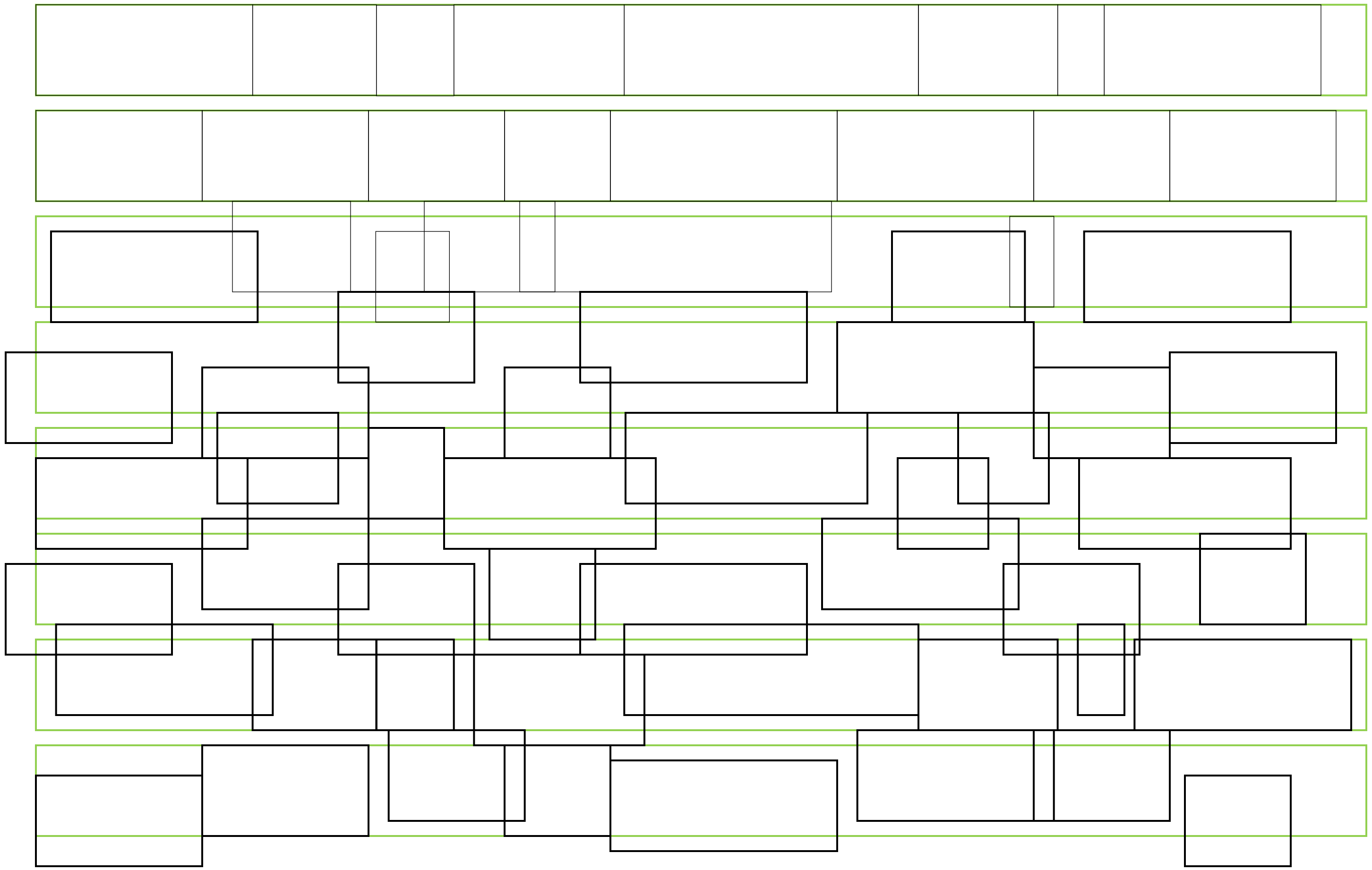
Row-by-Row Legalization



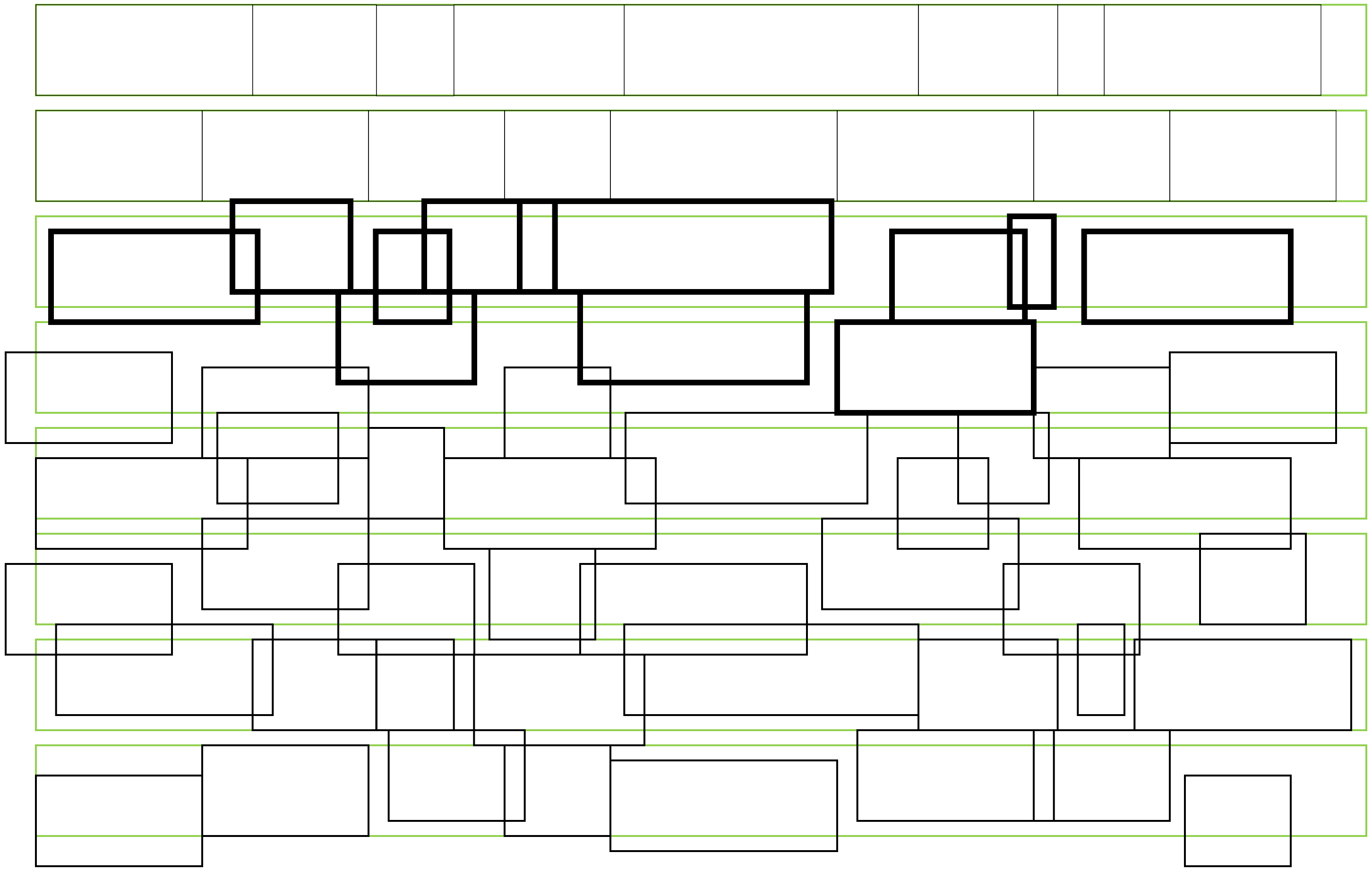
Row-by-Row Legalization



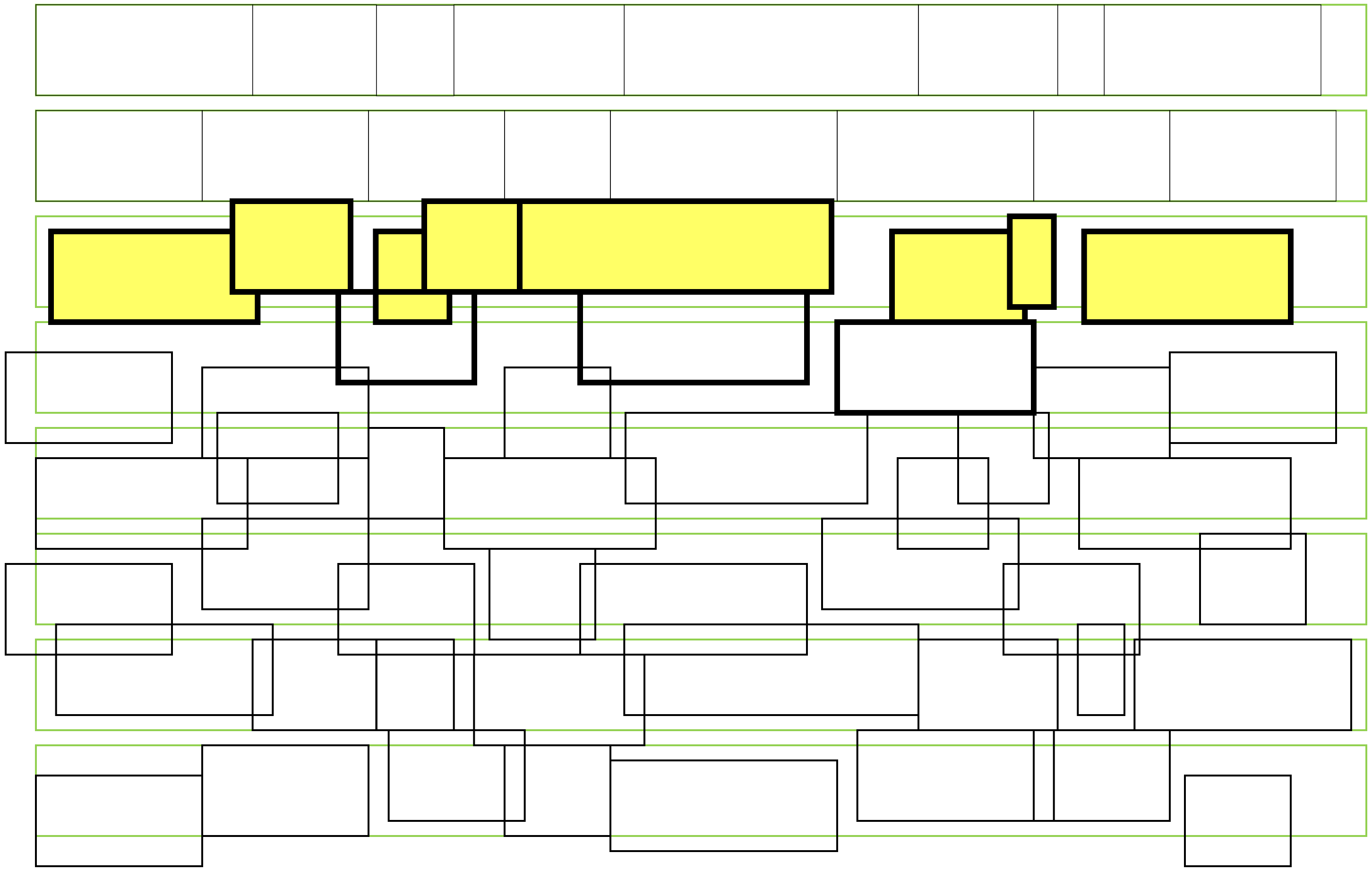
Row-by-Row Legalization



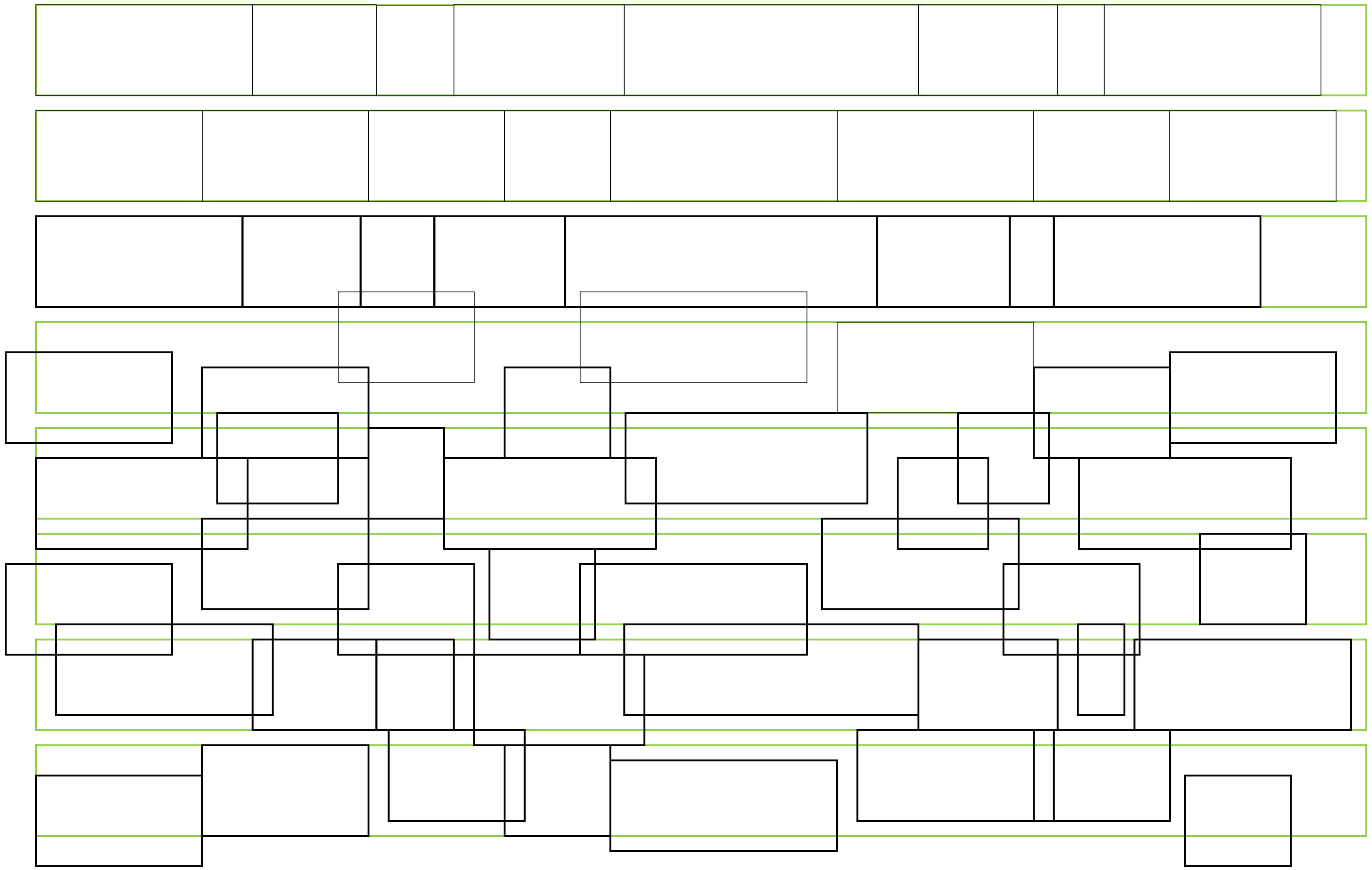
Row-by-Row Legalization



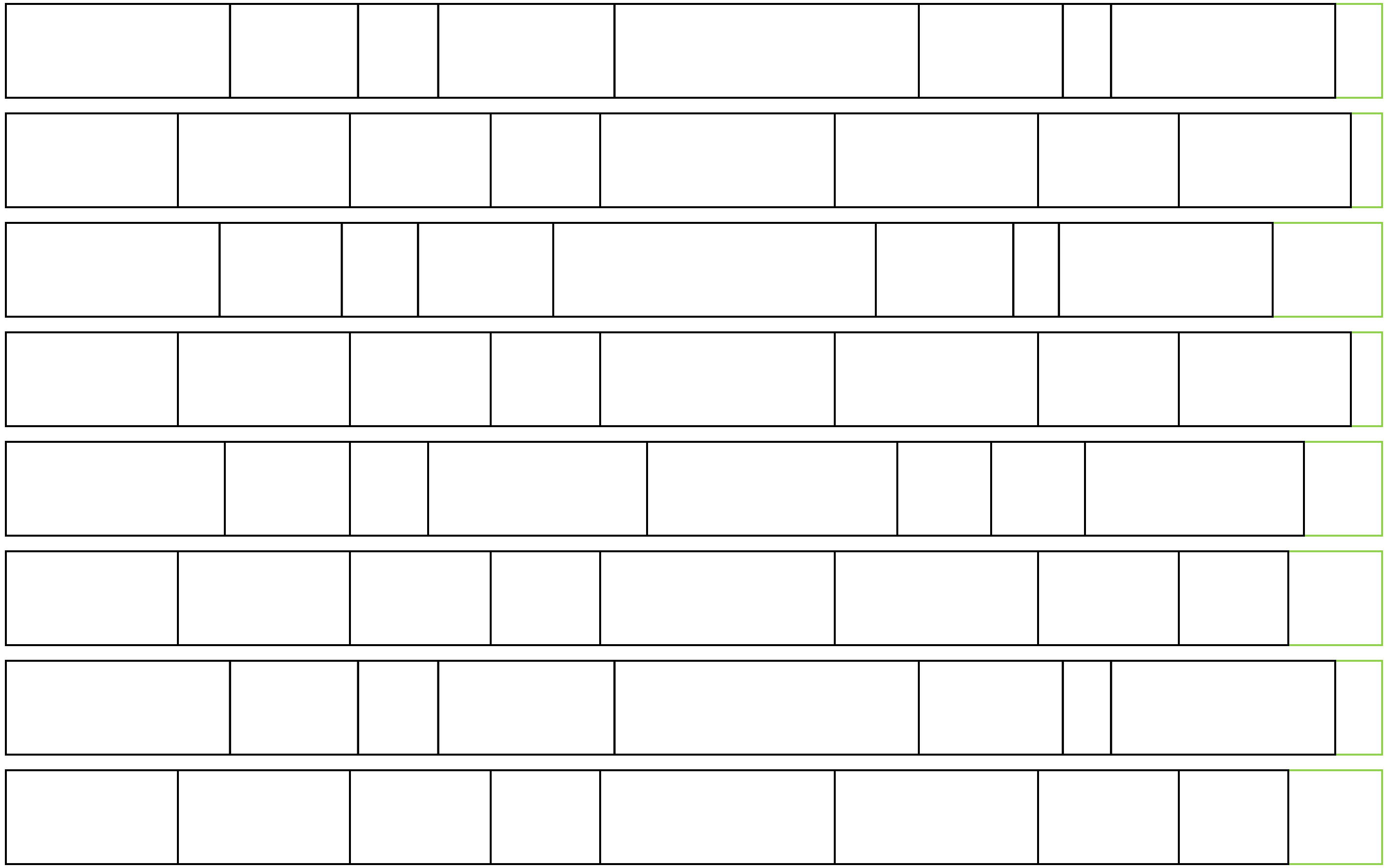
Row-by-Row Legalization



Row-by-Row Legalization

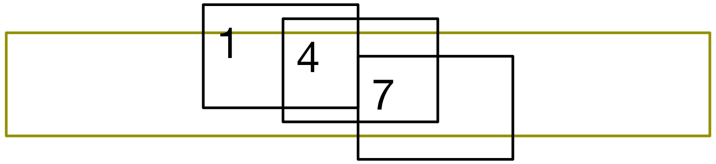


Row-by-Row Legalization



Row-based Algorithms – Abacus

- Assume the y coordinates of cells have been determined
  - Keep the **relative order** of cells
  - Only compute the x coordinates of cells
- Consider a row of cells (sorted from left to right)
  - $\min_{\boldsymbol{x}} \sum_{i=1}^{N_r} e_i \left(x_i - x_i^{gp}\right)^2,$
  - $s.t.\quad x_i - x_{i-1} \geq w_{i-1}, \quad i = 2, \cdots, N_r$
- Further simplify $\geq$ to $=$
  - $x_i = x_1 + \sum_{k=1}^{i-1} w_k$
  - $\underbrace{\sum_{i=1}^{N_r} e_i}_{e_c} x_1 - \underbrace{\left[e_1 x_1^{gp} + \sum_{i=2}^{N_r} e_i \left(x_i^{gp} - \sum_{k=1}^{i-1} w_k\right)\right]}_{q_c} = 0$
  - $x_1 e_c - q_c = 0 \Leftrightarrow x_1 = \frac{q_c}{e_c}$

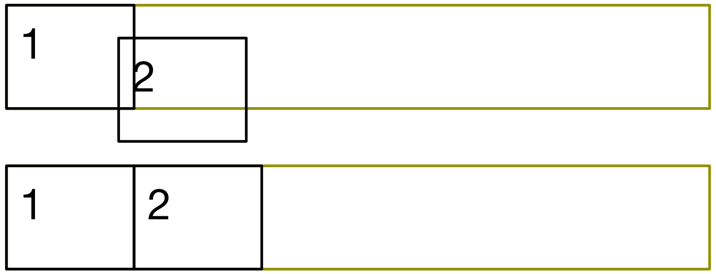


Init	Iteration ($i = 1, 2, \cdots, N_r$)
$e_c = 0$	$e_c \leftarrow e_c + e_i$
$q_c = 0$	$q_c \leftarrow q_c + e_i(x_i^{gp} - w_c)$
$w_c = 0$	$w_c \leftarrow w_c + w_i$

Row-based Algorithms – Abacus

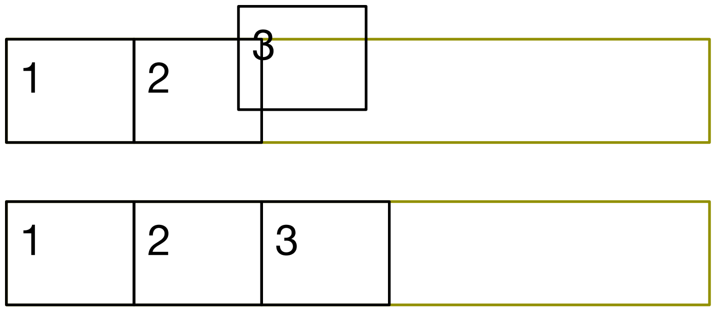
- Handling inequality constraints with clustering

Place cell 1 at $x_1^* = \frac{q_c}{e_c}$



x_2^* overlaps with x_1^*
Merge 1 and 2 as one cluster

$$x_{\{1,2\}}^* = \frac{q_c}{e_c}$$

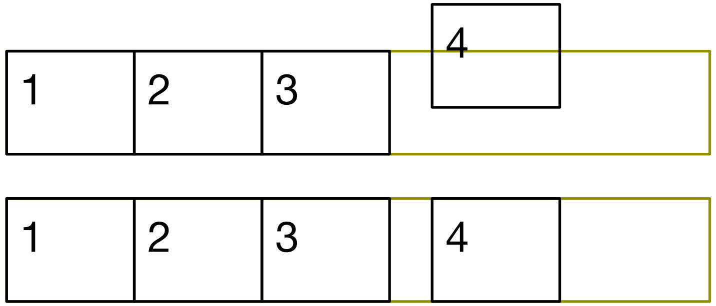


x_3^* overlaps with $x_{\{1,2\}}^*$
Merge 3 and {1,2} as one cluster

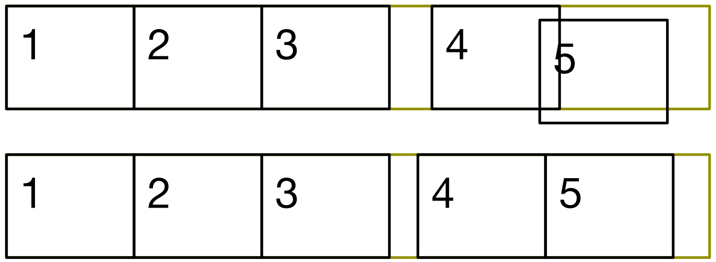
$$x_{\{1,2,3\}}^* = \frac{q_c}{e_c}$$

Row-based Algorithms – Abacus

- Handling inequality constraints with clustering

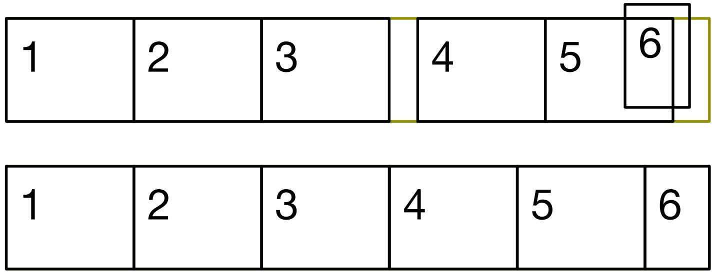


x_4^* does not overlap with $x_{\{1,2,3\}}^*$
Put cell 4 at x_4^*



x_5^* overlaps with x_4^*
Merge 5 and 4 as one cluster

$$x_{\{4,5\}}^* = \frac{q_c}{e_c}$$



x_6^* overlaps with $x_{\{4,5\}}^*$
Merge 6 and {4,5} as one cluster

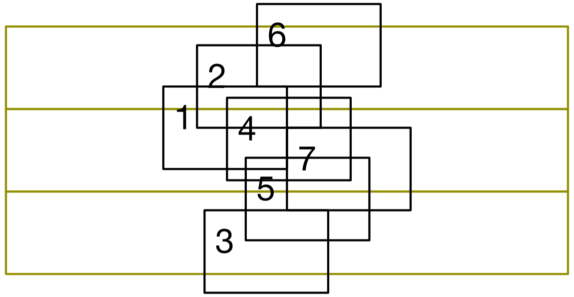
$$x_{\{4,5,6\}}^* = \frac{q_c}{e_c}$$

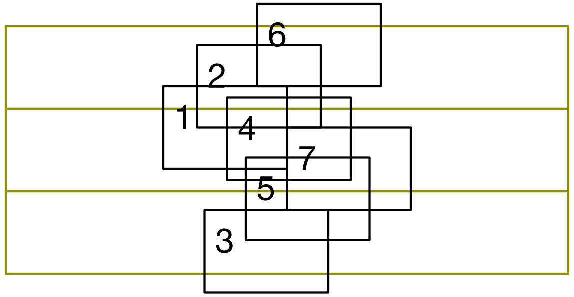
$x_{\{4,5,6\}}^*$ overlaps with $x_{\{1,2,3\}}^*$
Merge {4,5,6} and {1,2,3} as one cluster

$$x_{\{1,2,3,4,5,6\}}^* = \frac{q_c}{e_c}$$

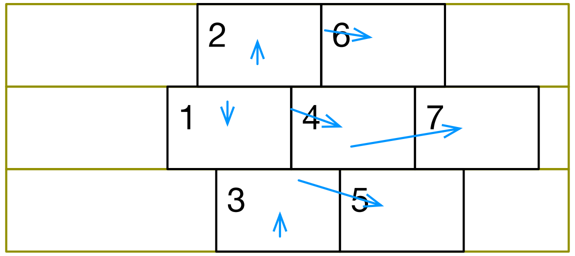
Row-based Algorithms – Abacus

- Extend single-row legalization to multiple rows
- Sort all cells from left to right
- For each cell
  - Try assigning it to each row and legalize that row
  - Choose the best row to assign

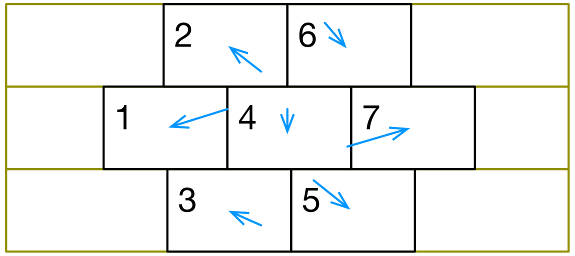




Global placement



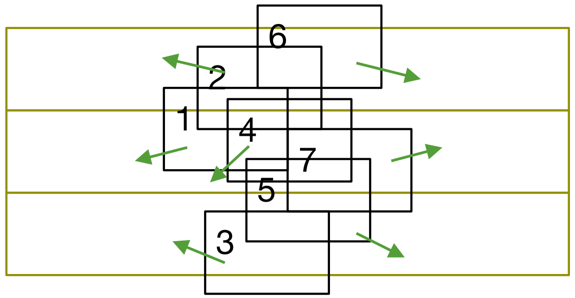
Tetris
Displacement = 3.752



Abacus
Displacement = 2.915

Row-based Algorithms – Linear Programming (LP)

- Pre-assign cells to rows
  - Assign {2, 6} to the 1st row
  - Assign {1, 4, 7} to the 2nd row
  - Assign {3, 5} to the 3rd row



- Mathematical programming
  - Keep the **relative order** of cells
  - $\min_{\boldsymbol{x}} \sum_i |x_i - x_i^{gp}|$
  - $s.t. \quad x_6 - x_2 \geq w_2,$
  - $x_4 - x_1 \geq w_1, x_7 - x_4 \geq w_4,$
  - $x_5 - x_3 \geq w_3,$
  - $x_1 \geq L, x_2 \geq L, x_3 \geq L, x_5 \leq R, x_7 \leq R, x_6 \leq R$

Convert |·| to linear constraints
Introduce additional variables d_i

$$\min_{\boldsymbol{x},\boldsymbol{d}} \sum_i d_i$$

$$s.t. \quad x_i - x_i^{gp} \leq d_i,$$
$$x_i^{gp} - x_i \leq d_i$$

Row-based Algorithms – Dual Min-Cost Flow

- LP with differentiable constraints only
  - The duality of this LP is a min-cost flow problem (strong duality)
  - Min-cost flow can be solved with network simplex, much faster than regular simplex solvers for LP

$$
\begin{aligned}
\mathcal{P}: \min \quad & \sum_{i \in N} b_i \pi_i + \sum_{(i,j) \in E} u_{ij} \alpha_{ij}, & \\
\text{s.t.} \quad & \pi_i - \pi_j - \alpha_{ij} \le c_{ij}, & \forall (i,j) \in E, \\
& \alpha_{ij} \ge 0, & \forall (i,j) \in E,
\end{aligned}
$$

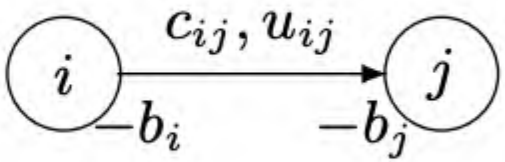
Slack variable

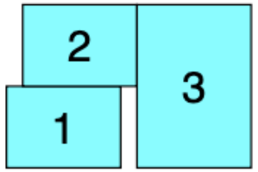
$$
\begin{aligned}
\mathcal{D}: \min \quad & \sum_{(i,j) \in E} c_{ij} f_{ij}, \\
\text{s.t.} \quad & \sum_{j:(i,j) \in E} f_{ij} - \sum_{j:(j,i) \in E} f_{ji} = -b_i, \forall i \in N, \\
& 0 \le f_{ij} \le u_{ij}, \forall (i,j) \in E,
\end{aligned}
$$

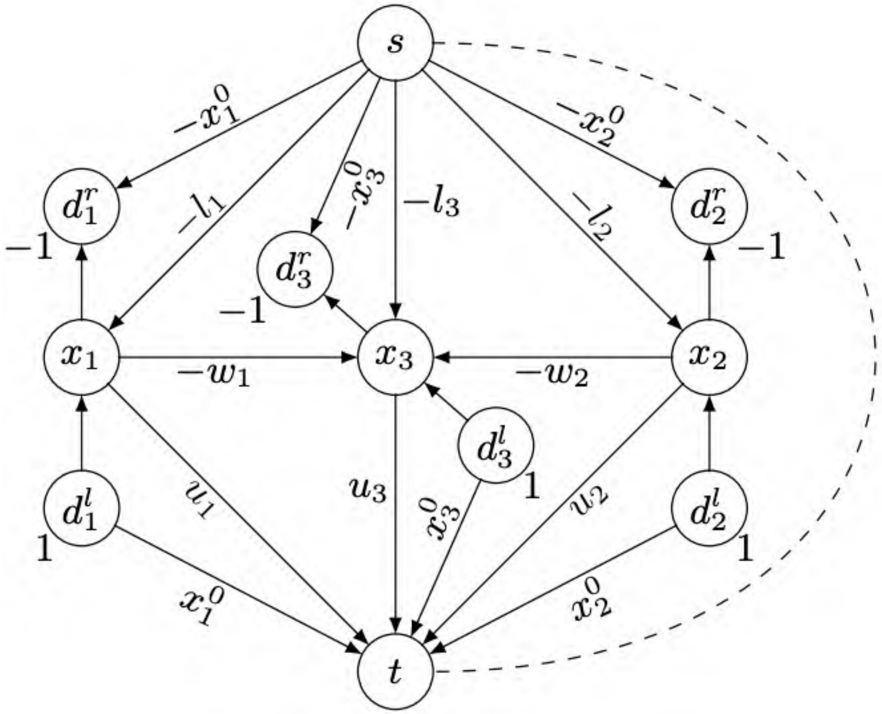
Brenner, Ulrich, and Jens Vygen. "Legalizing a placement with minimum total movement." IEEE TCAD 2004.
Lin, Yibo, et al. "MrDP: Multiple-row detailed placement of heterogeneous-sized cells for advanced nodes." IEEE TCAD 2017.

Row-based Algorithms – Dual Min-Cost Flow

$$
\begin{aligned}
\mathcal{P}: \min \quad & \sum_{i \in N} b_i \pi_i + \sum_{(i,j) \in E} u_{ij} \alpha_{ij}, \\
\text{s.t.} \quad & \pi_i - \pi_j - \alpha_{ij} \le c_{ij}, && \forall (i,j) \in E, \\
& \alpha_{ij} \ge 0, && \forall (i,j) \in E,
\end{aligned}
$$





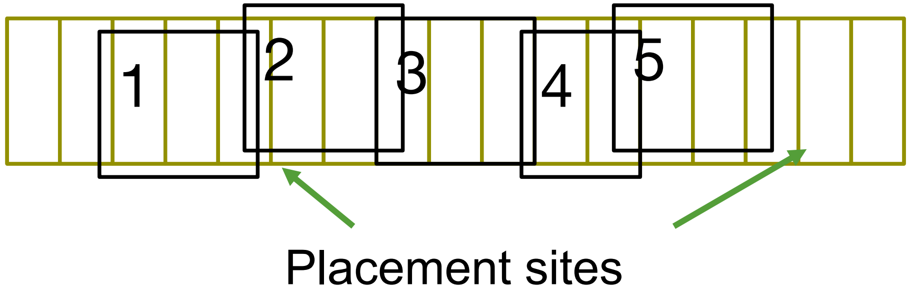


More extension with quadratic objective
Linear complementary problem (LCP)
[Chen+, DAC2017] Best Paper

Lin, Yibo, et al. "MrDP: Multiple-row detailed placement of heterogeneous-sized cells for advanced nodes." IEEE TCAD 2017.

Site-based Algorithms

- Dynamic programming
  - Taghavi, Taraneh, et al. "New placement prediction and mitigation techniques for local routing congestion." ICCAD 2010.
  - Lin, Yibo, et al. "Stitch aware detailed placement for multiple e-beam lithography." Integration 2017. (Best Paper Award)
- Integer linear programming
  - Li, Shuai, and Cheng-Kok Koh. "Mixed integer programming models for detailed placement." ISPD 2012.



Legalization – Summary

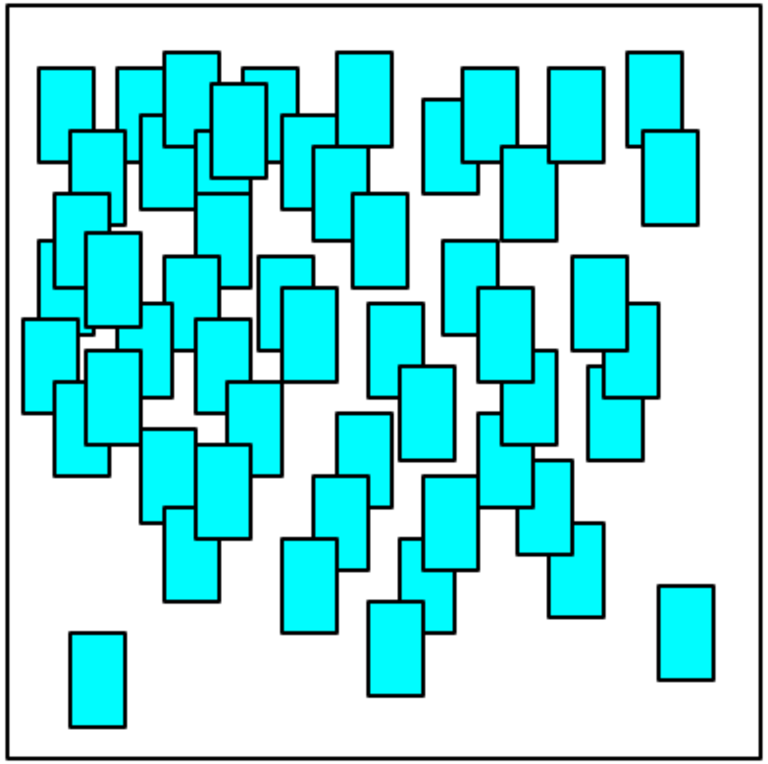
- Tackle legalization problem from different angles
- Tetris
- Region assignment
- Row-based legalization
  - Abacus
  - Linear programming
  - Min-cost flow
- Site-based legalization
  - Dynamic programming
  - Integer linear programming

Continuous Problem

Discrete Problem

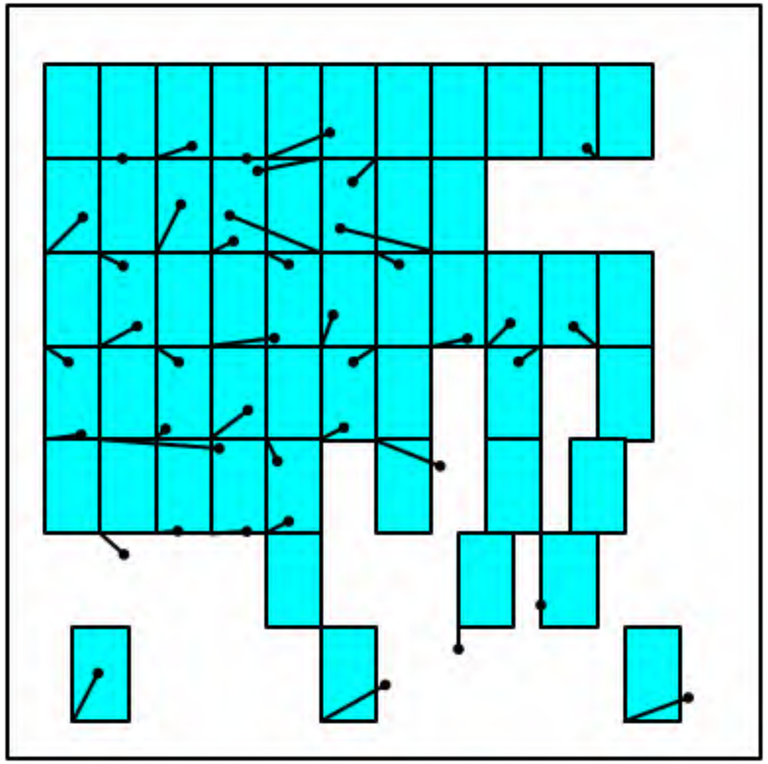
Typical Placement Flow

WL: 1.00e+6



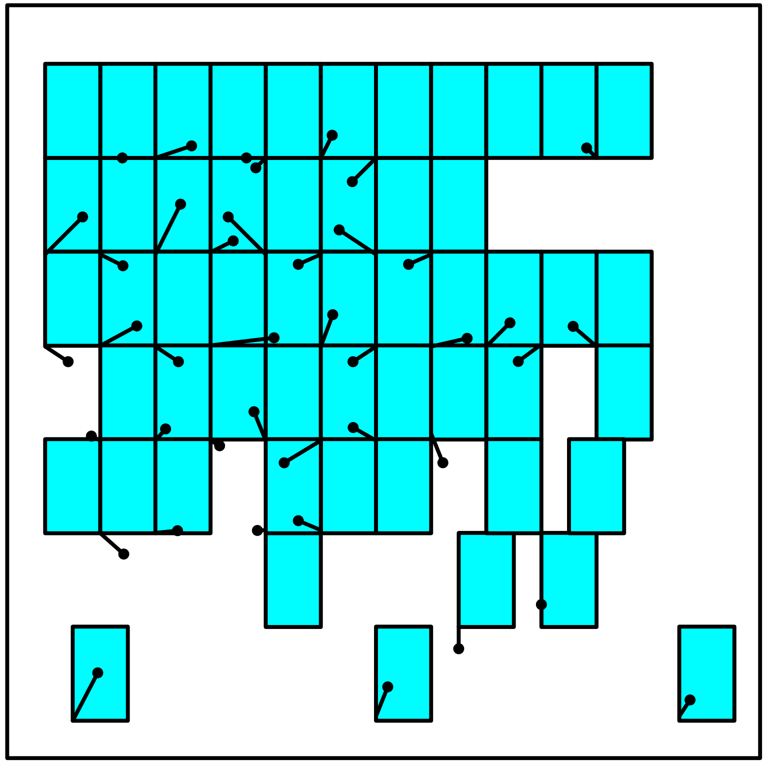
Global placement

WL: 1.05e+6



Legalization

WL: 1.02e+6



Detailed Placement