PROS: A Plug-in for Routability Optimization applied in the State-of-the-art commercial EDA tool using deep learning

Jingsong Chen[1], Jian Kuang[2], Guowei Zhao[2], Dennis J.-H. Huang[2], Evangeline F. Y. Young[1]

[1]CSE Dept., The Chinese University of Hong Kong

[2]Cadence Design Systems

香港中文大學 The Chinese University of Hong Kong

cadence

Speaker Bio

Jingsong Chen is a fourth-year Ph.D. candidate at the Department of Computer Science and Engineering, The Chinese University of Hong Kong, under the supervision of Prof. Evangeline F.Y. Young since Fall 2017.

Before joining CUHK, he received his Bachelor Degree of Computer Science and Technology from Zhejiang University.

His research interests include physical design of VLSI circuits and related machine learning-based problems.

His homepage: https://jingsongchen.github.io/.

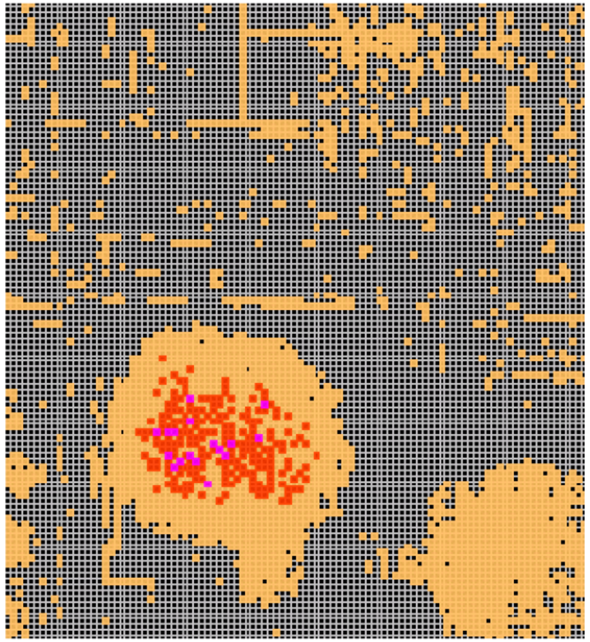
Outline

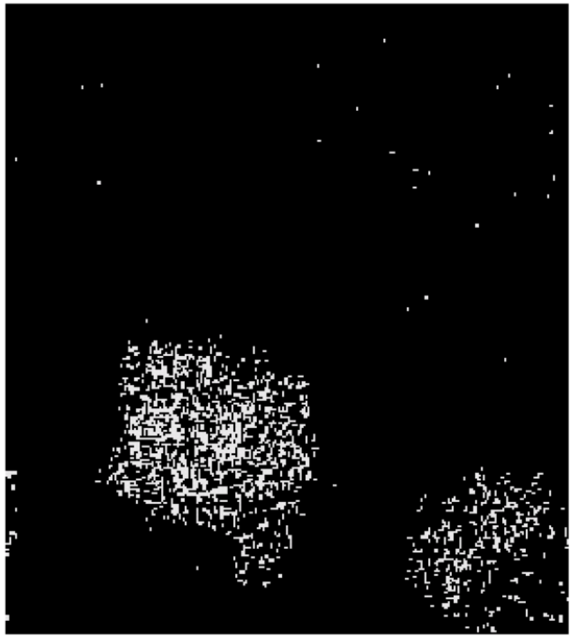
- Problem Background & Motivation
- Overall Flow of PROS
- FCN-based Predictor for GR Congestion
- Optimizer for GR Cost Parameters
- Experimental Results
- Conclusion

Outline

- Problem Background & Motivation
- Overall Flow of PROS
- FCN-based Predictor for GR Congestion
- Optimizer for GR Cost Parameters
- Experimental Results
- Conclusion

Prior Knowledge Is Useful for P&R

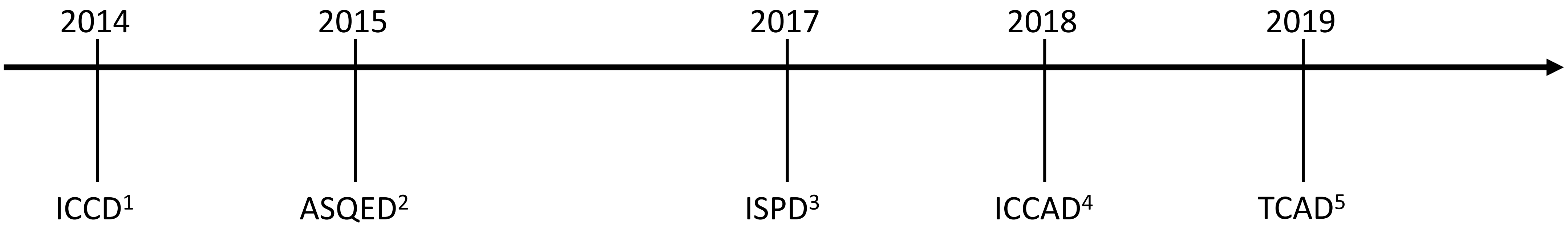




- Routing congestion:
  - Obtained after global routing
  - Affect DRC violation distribution, power, timing, and etc.
- Design rule check (DRC) violation:
  - Obtained after detailed routing
  - Decide whether the design can be taped out successfully or not.

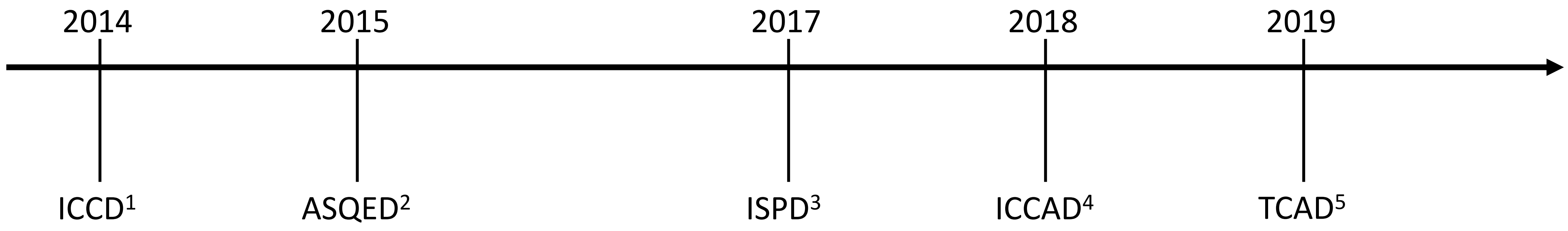
*These figures are from "Accurate Prediction of Detailed Routing Congestion using Supervised Data Learning", in Proc. ICCD, 2014.

Acquire Prior Knowledge by ML Techniques



Work	Applied in an EDA tool?	Avoid extra runtime overhead for feature preparation?
[1]"Accurate prediction of detailed routing…"	Yes	No
[2]"An accurate detailed routing routability…"	No	N/A
[3]"Routability optimization for industrial…"	Yes	No
[4]"Routenet: Routability prediction for…"	No	N/A
[5]"Eh? predictor: A deep learning framework…"	No	N/A

Acquire Prior Knowledge by ML Techniques



Work	Applied in an EDA tool?	Avoid extra runtime overhead for feature preparation?
[1]"Accurate prediction of detailed routing…"	Yes	No
[2]"An accurate detailed routing routability…"	No	N/A
[3]"Routability optimization for industrial…"	Yes	No
[4]"Routenet: Routability prediction for…"	No	N/A
[5]"Eh? predictor: A deep learning framework…"	No	N/A
WANTED	Yes	Yes

PROS: A Plug-in for Routability Optimization

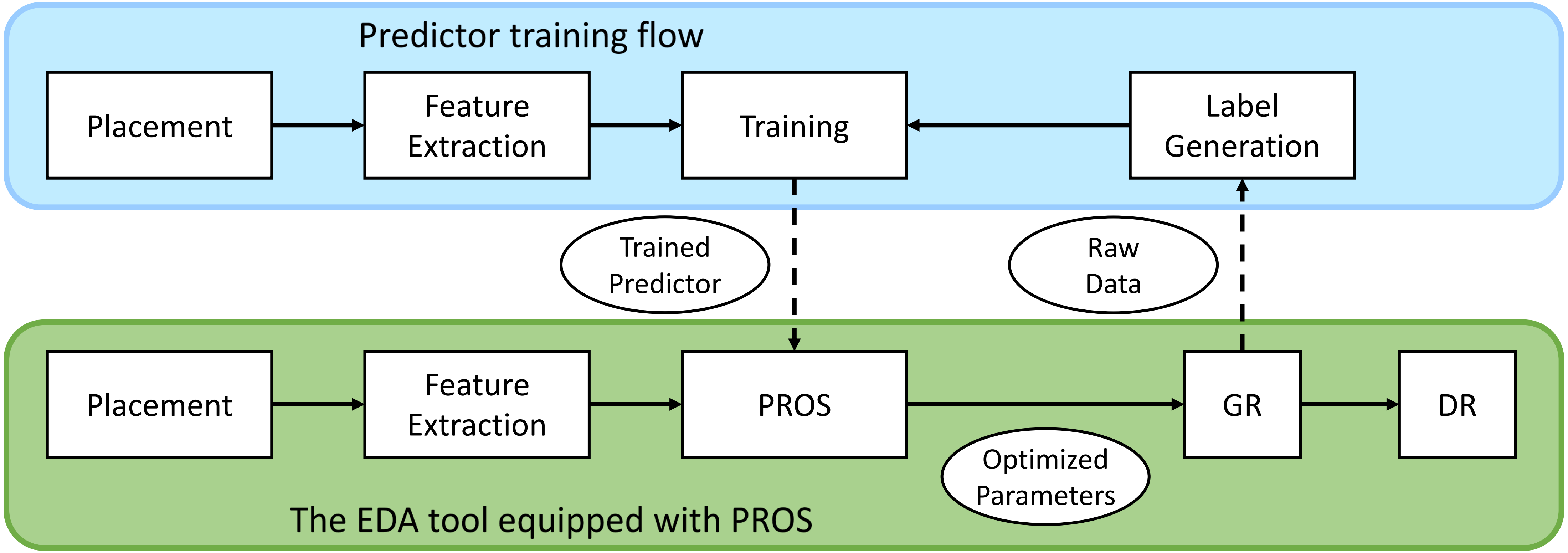
- Main features of PROS:
  - Predict routing congestion by deep learning
    - PROS can learn the behavior of a specific router.
  - Require only data from the placement result
    - Runtime overhead of PROS is negligible.
  - Just optimize the cost parameters of global routing
    - PROS can be easily embedded into any other routers as a plug-in.
  - Work well when integrated into the State-of-the-art commercial EDA tool*
    - PROS is the first ML framework which demonstrates its practicality.
- Target of PROS:
  - Reduce routing congestion and thus improve routability optimization (#DRC)

* Cadence Innovus v20.1

Outline

- Problem Background & Motivation
- Overall Flow of PROS
- FCN-based Predictor for GR Congestion
- Optimizer for GR Cost Parameters
- Experimental Results
- Conclusion

Overall Flow of PROS



- For one technology node, the predictor only needs to be trained once.
- Feature extraction, prediction, and optimization of GR cost parameters can be performed very quickly.
- The original parts of the EDA tool are not changed a lot.

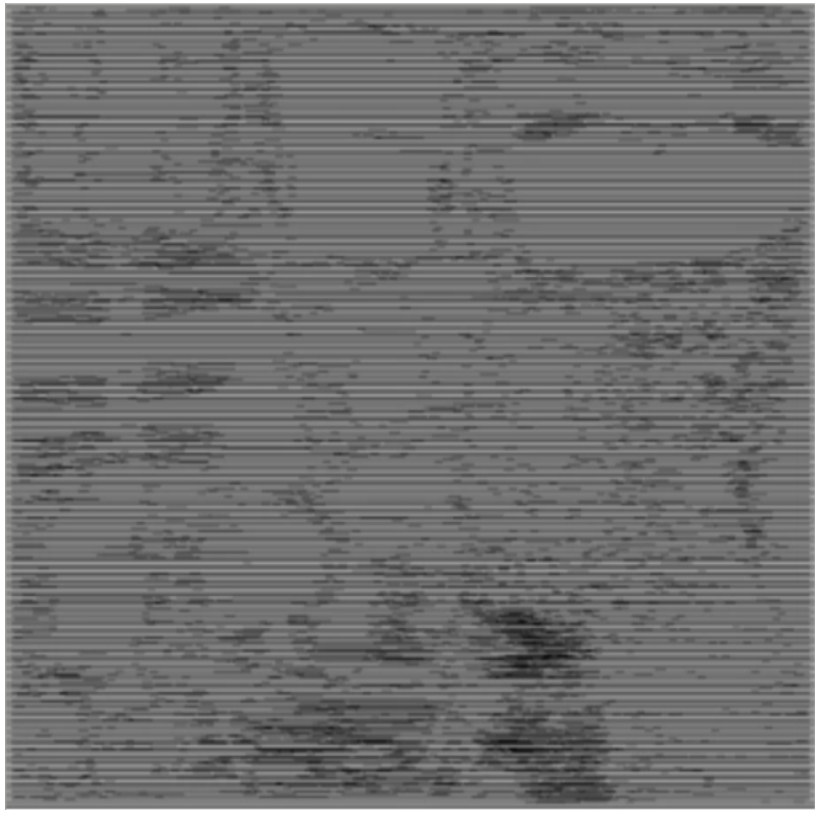
Outline

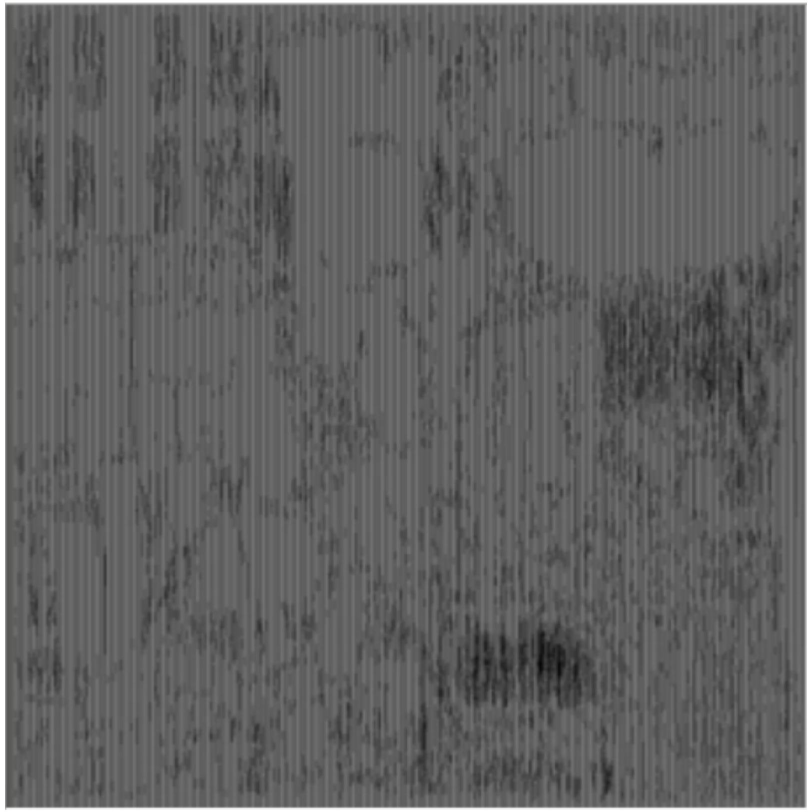
- Problem Background & Motivation
- Overall Flow of PROS
- FCN-based Predictor for GR Congestion
- Optimizer for GR Cost Parameters
- Experimental Results
- Conclusion

Features & Label Summary

- Features:
  - Routing capacity map (horizontal, vertical)
  - Standard cell density map
  - Standard cell pin density map
  - Pin accessibility density map
  - Cross net density map (horizontal, vertical)
  - Flip-flop cell density map
  - Fixed cell density map
  - RUDY map (small nets, large nets)
  - Pin RUDY map
- Label:
  - A 2-D binary map reflecting GR routing congestion:
    - 1 → congestion
    - 0 → no congestion

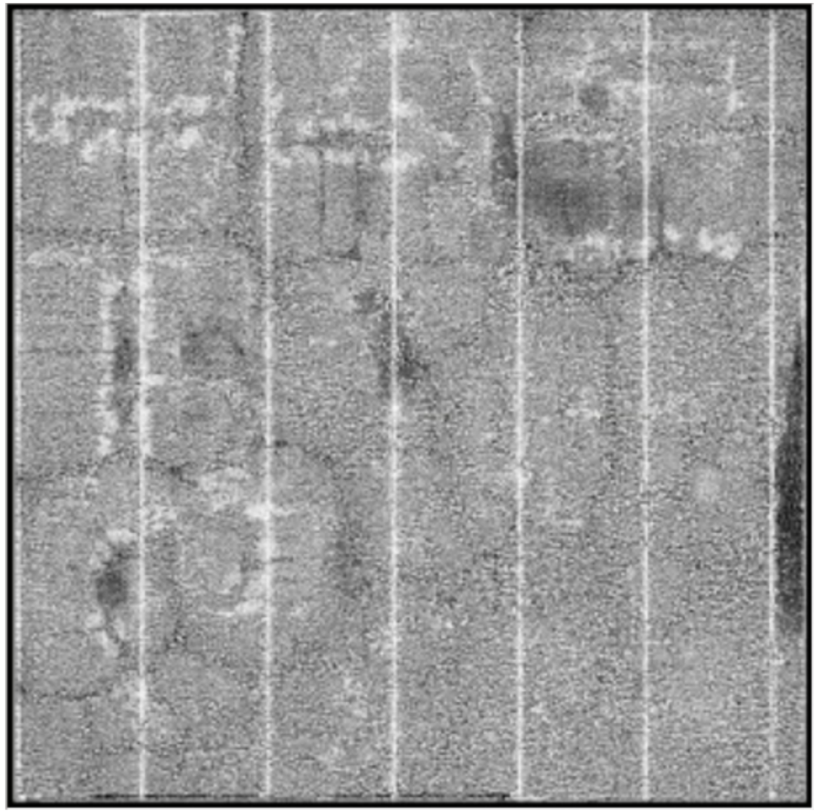
Routing capacity map


Horizontal capacity map

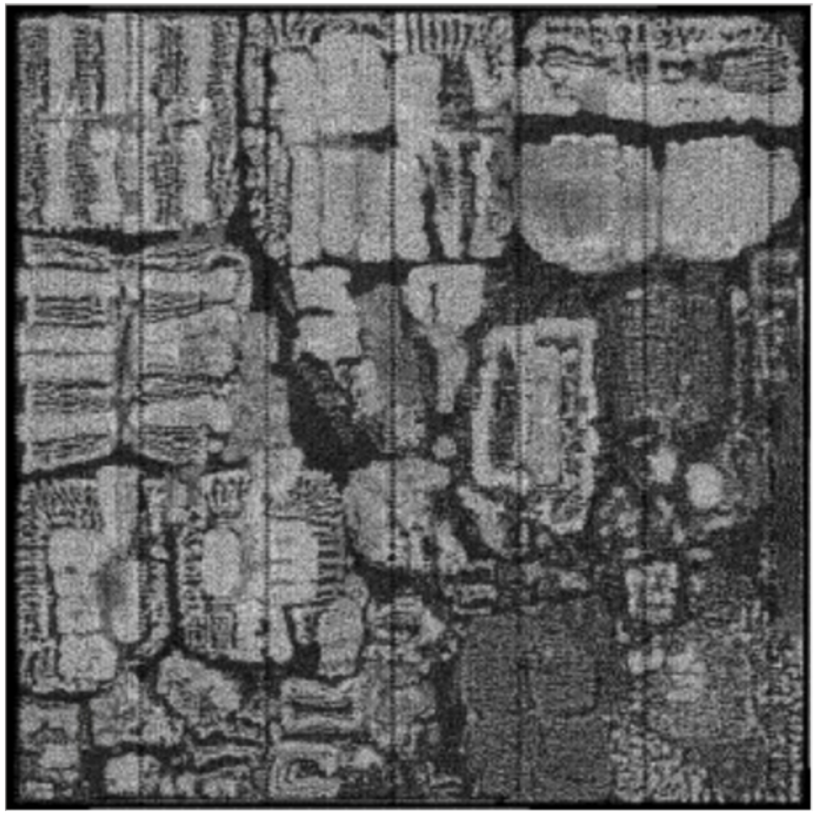

Vertical capacity map

- The capacity value: low → high; the color in image: black → white.
- 3-D features → 2-D image: accumulate all the layers.

Standard cell & cell pin density map



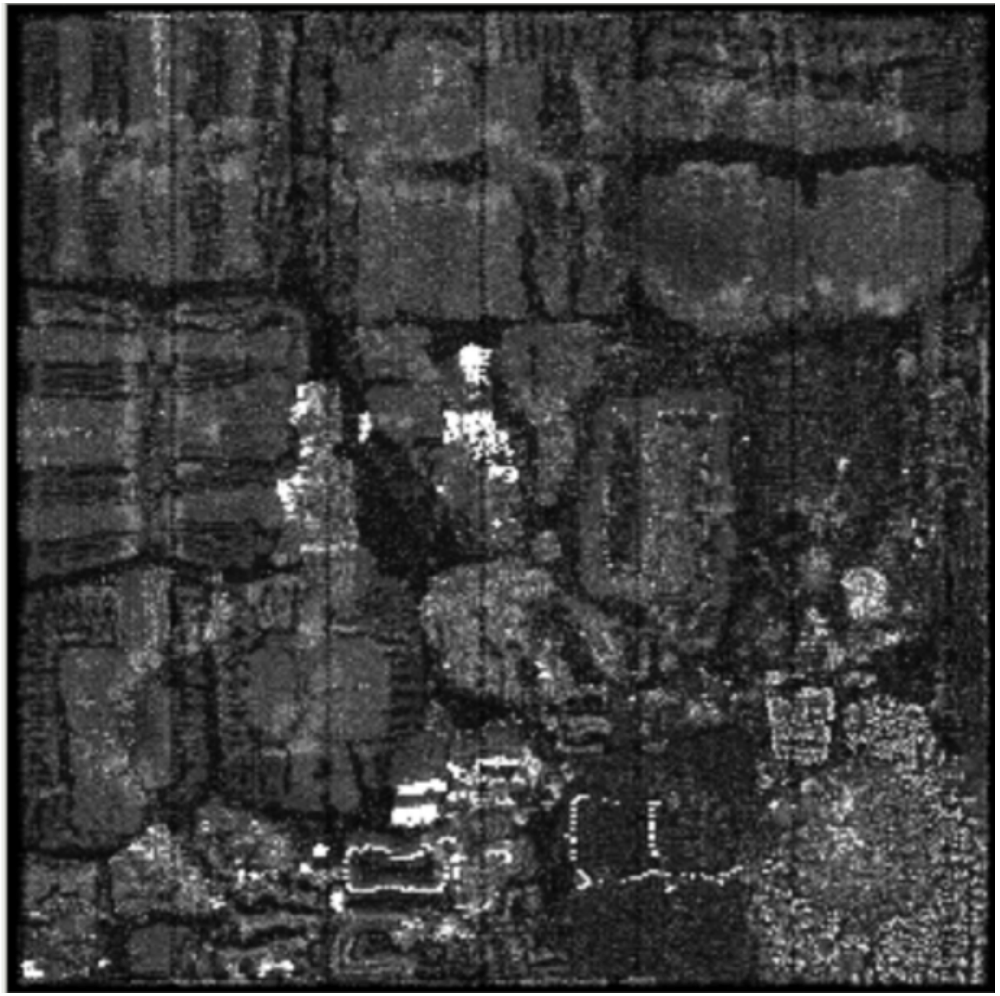
Cell density map



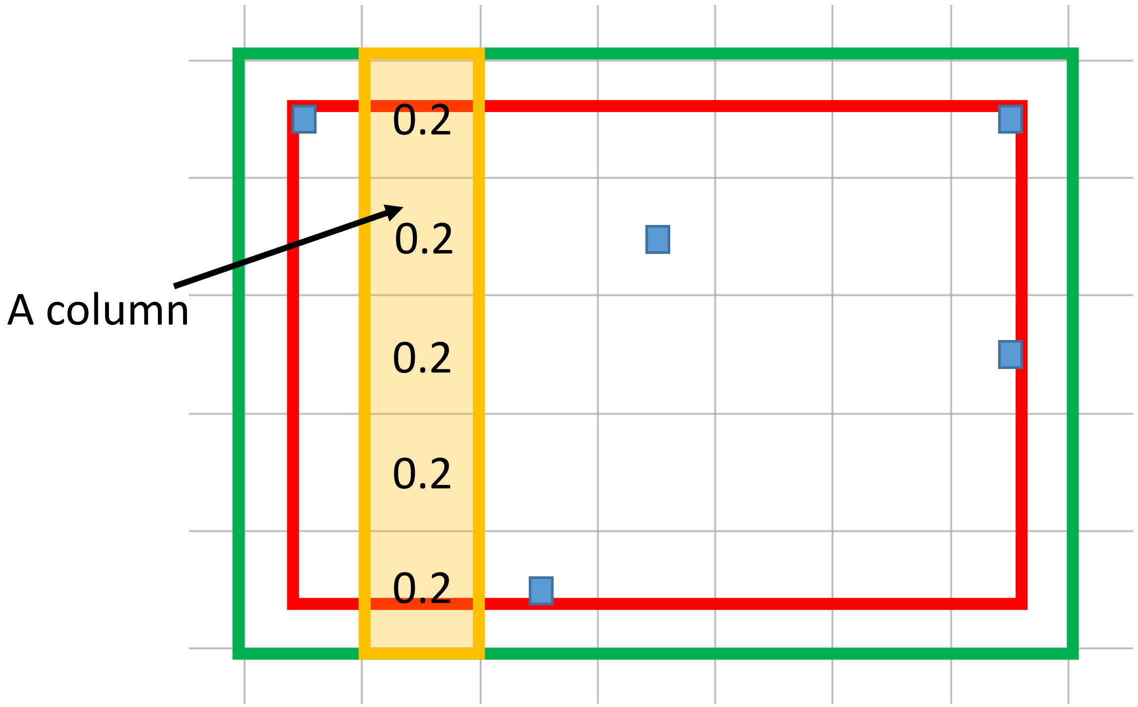
Cell pin density map

Pin accessibility density map

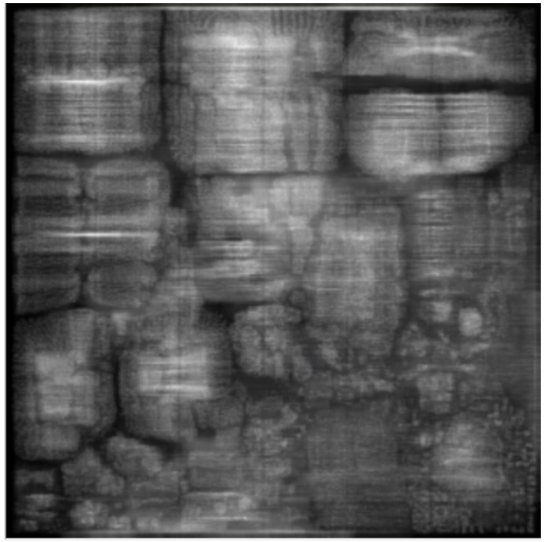
- Pin accessibility density map (F) ：
  - For each cell C:
    - For each pin p of C:
      - p location: x, y,
      - $npat$ = #pin accessing patterns of C,
      - $npin$ = #pins of C,
      - $F(x,y) \mathrel{+}= \frac{1.5^{npin}}{(npat+1)\times npin}$.
  - Higher value → More difficult to route



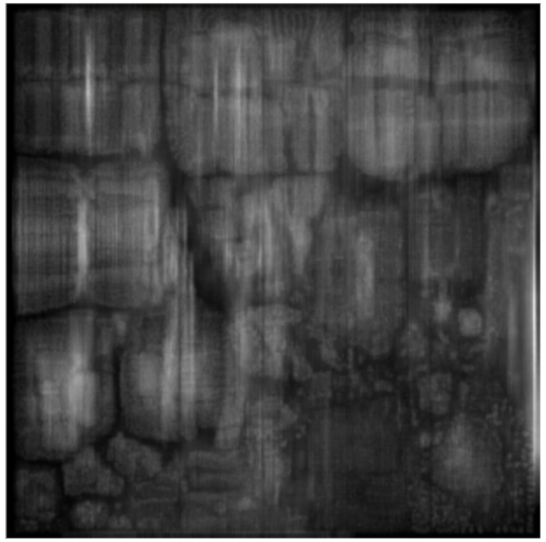
Cross net density map



- Blue boxes: a 5-pin net.
- Red box: BBox of the net.
- Green box: g-cells which will get net density.
- Horizontal (vertical) cross net density = 1 / #g-cells in a column (row).

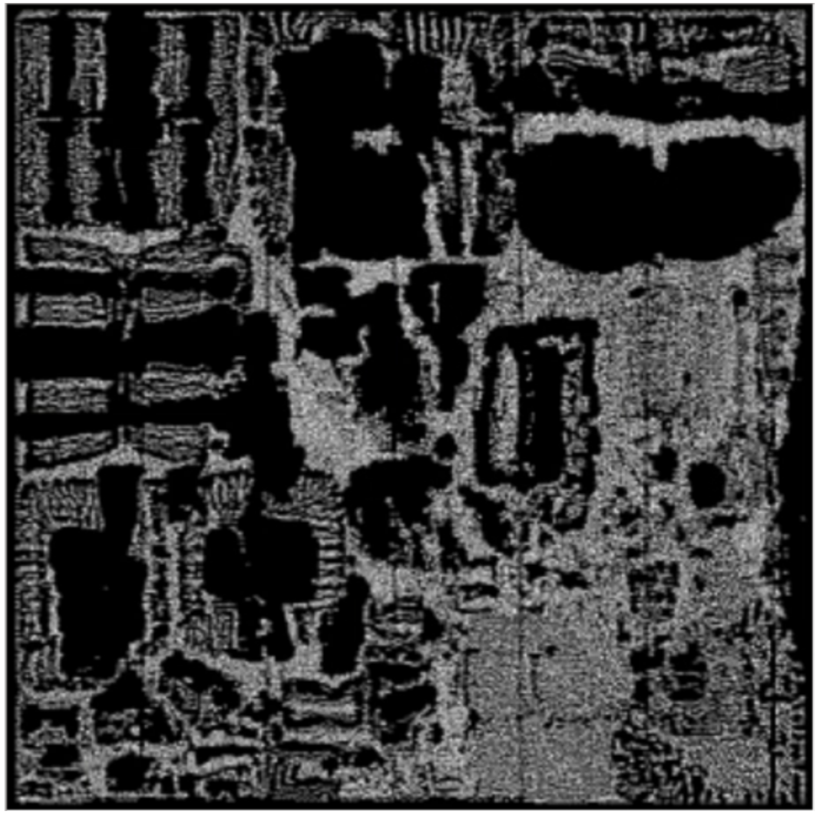


Horizontal cross net density map

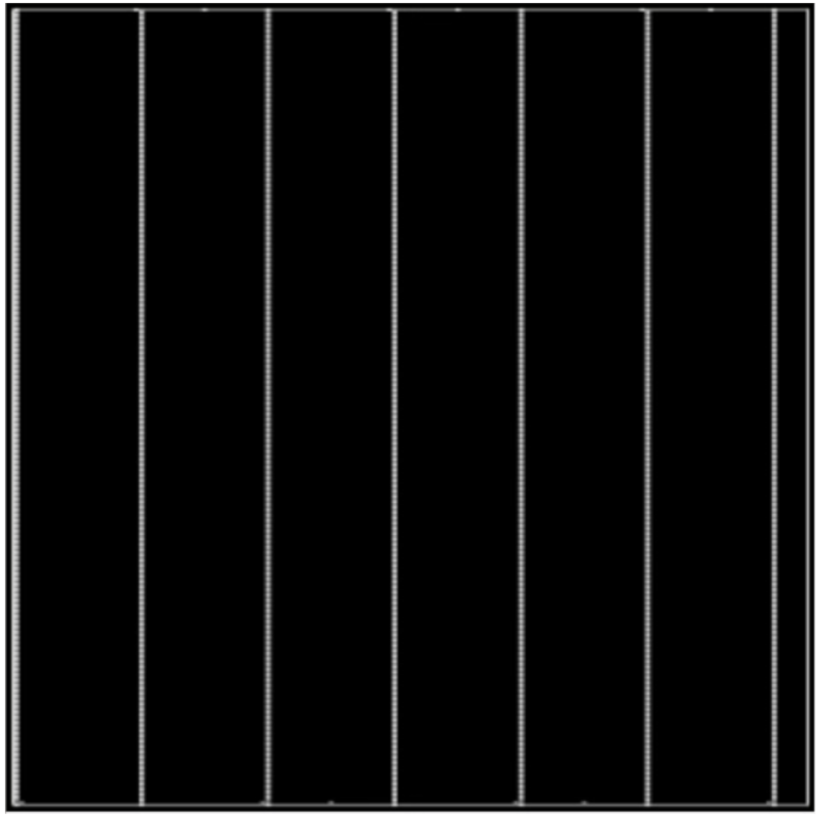


Vertical cross net density map

Flip-flop cell & fixed cell density map



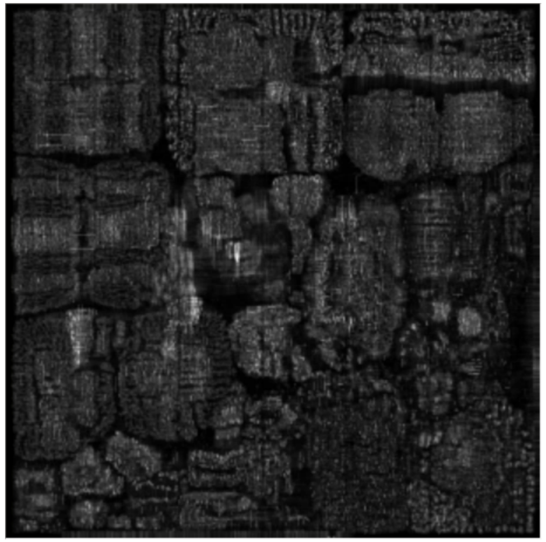
Flip-flop cell density map



Fixed cell density map

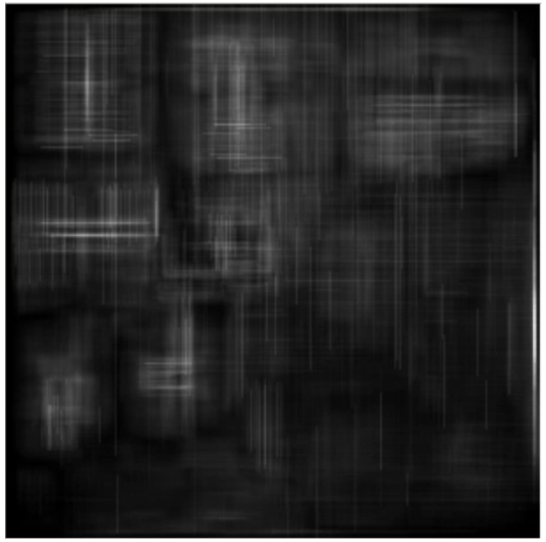
RUDY map

- Large net: HPWL of the net BBox >= 15 * g-cell size.
- For each net:
  - RUDY value of each g-cell in the net BBox = wire length / #g-cells in the net BBox.
  - Wire length = HPWL of the net BBox * Ratio_#pins.



RUDY map of small nets

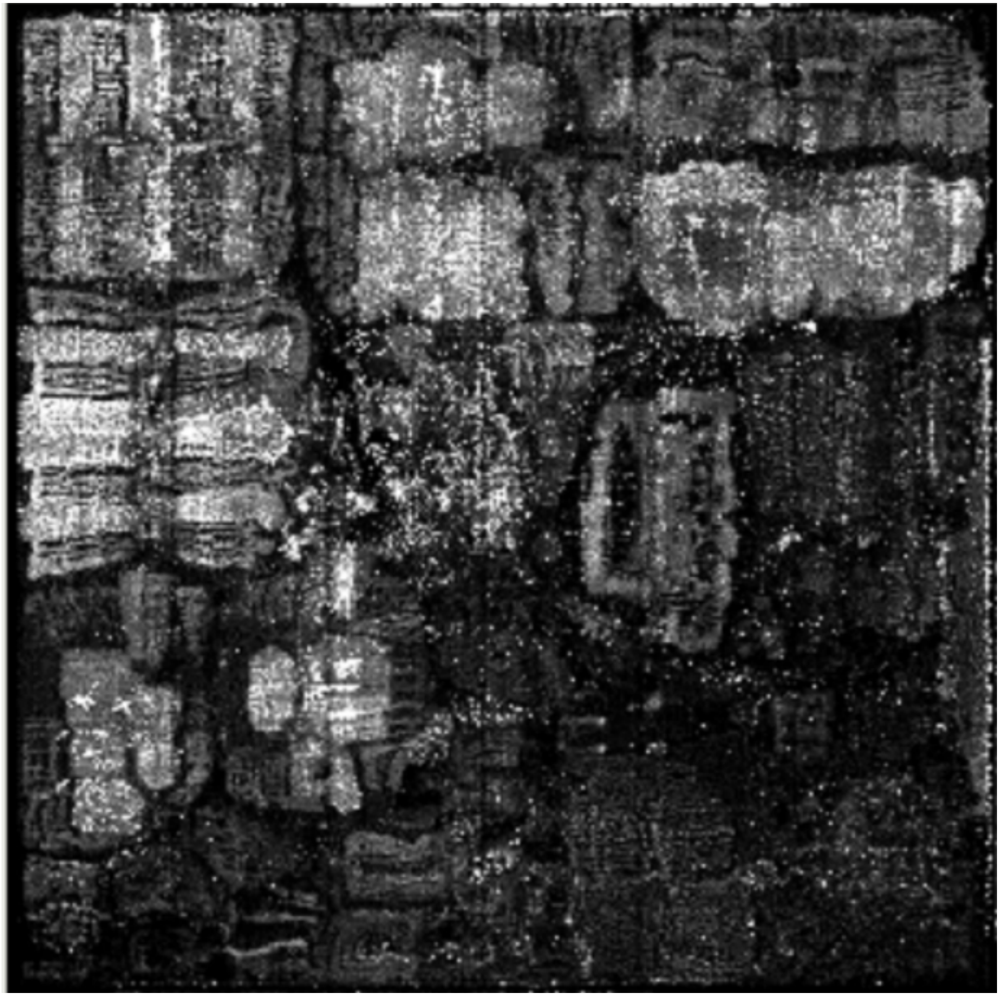
#pins	4	5	6	<=8	<=10	<=15	<=20	>20
Ratio_#pins	1.06	1.13	1.19	1.31	1.42	1.66	1.87	2.22



RUDY map of large nets

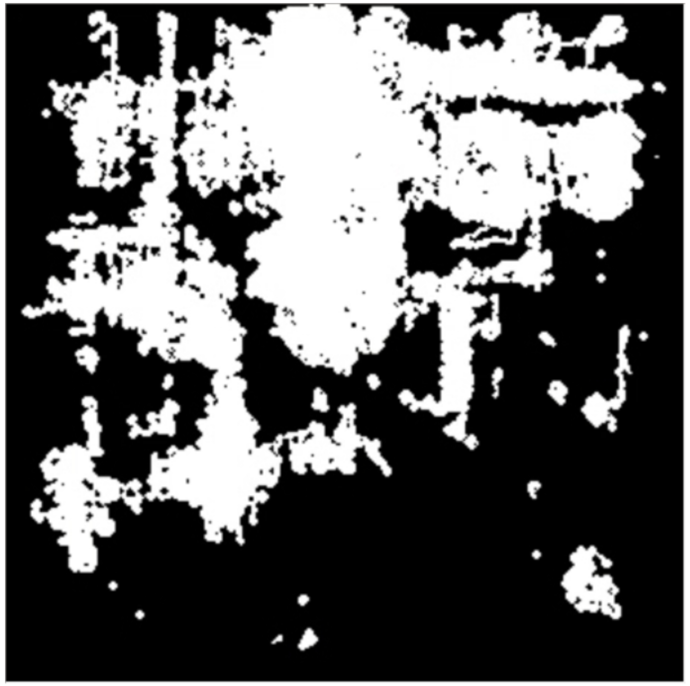
Pin RUDY map

- Highlight the role of large nets on congestion prediction.
- A combination of cell pin density map and large-net RUDY map.
- The contribution of each pin on its location equals to the wire density of the large net it belongs to.



Label Generation: Smoothening Process

Raw

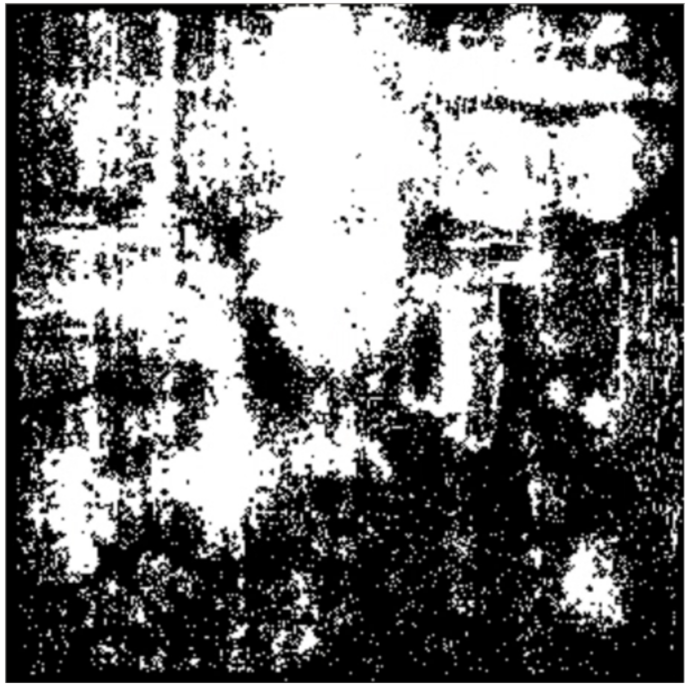


Label

- Purpose: generate more clear label to improve prediction accuracy.

Label Generation: Smoothening Process

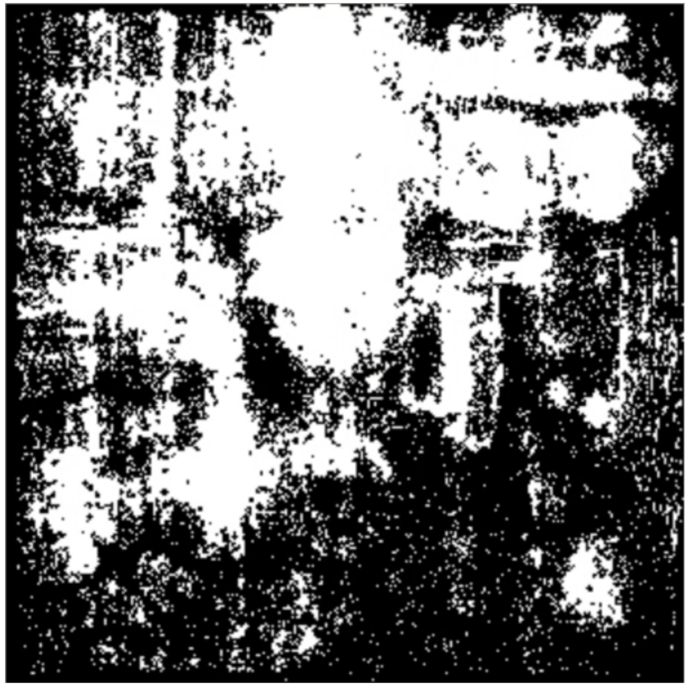
Raw



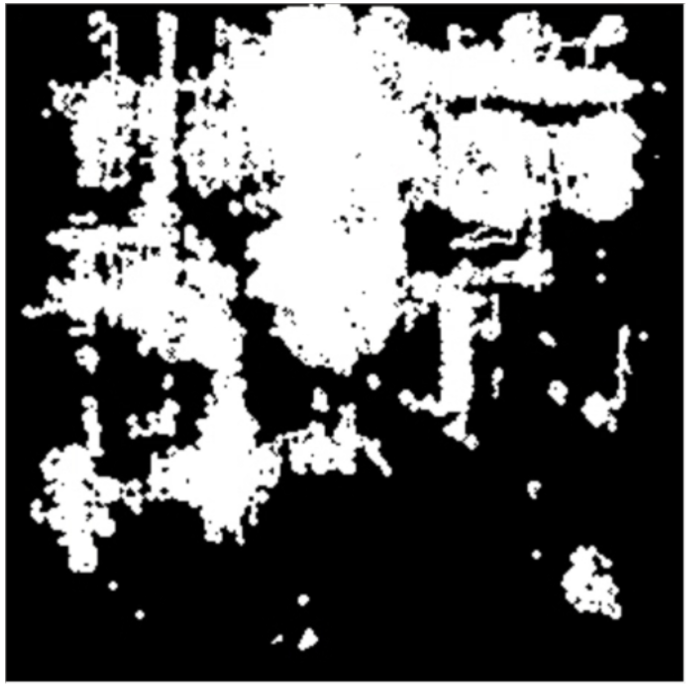
After INSERT

- Smoothening process from raw GR congestion map to congestion label:
  1. INSERT: If there are >= 6 congested surrounding g-cells, the center non-congested g-cell will be relabeled as congested.

Label Generation: Smoothening Process



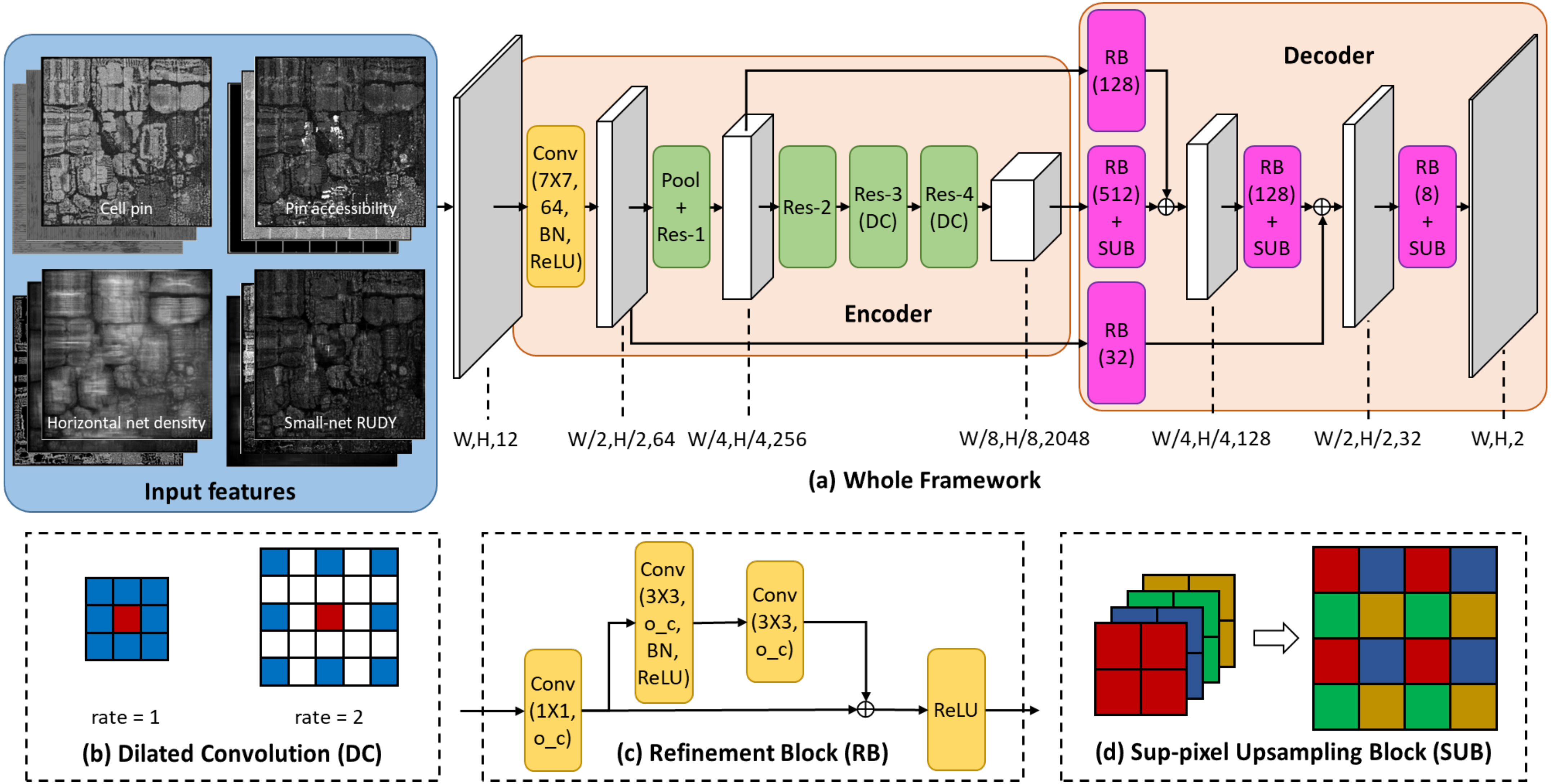
After INSERT



After CLEAN

- Smoothening process from raw GR congestion map to congestion label:
  1. INSERT: If there are >= 6 congested surrounding g-cells, the center non-congested g-cell will be relabeled as congested.
  2. CLEAN (10 iters): If there are <= 3 congested surrounding g-cells, the center congested g-cell will be relabeled as non-congested.

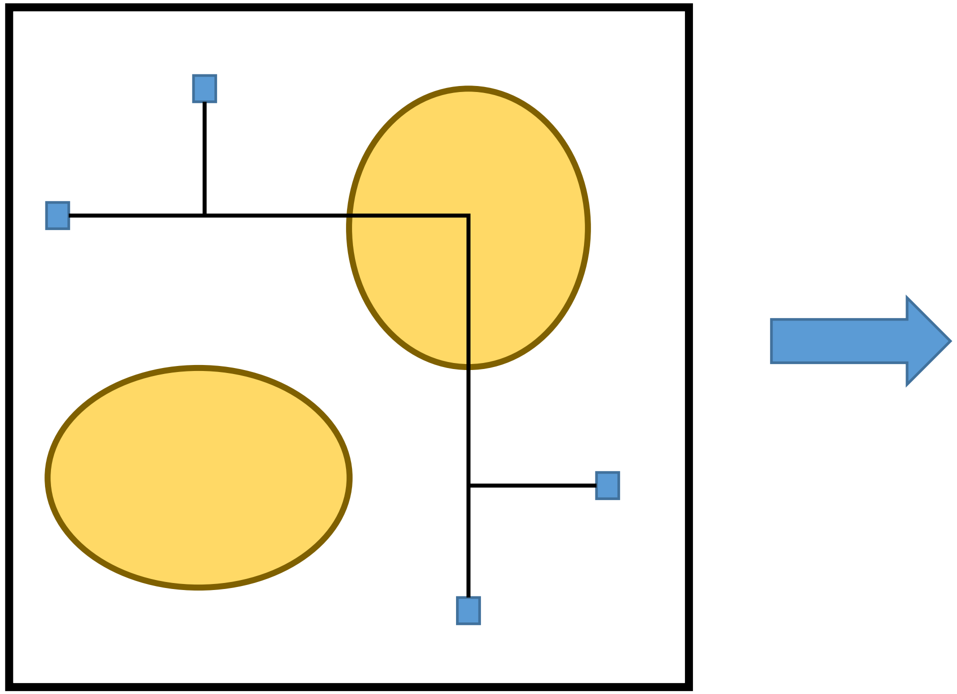
Prediction Model



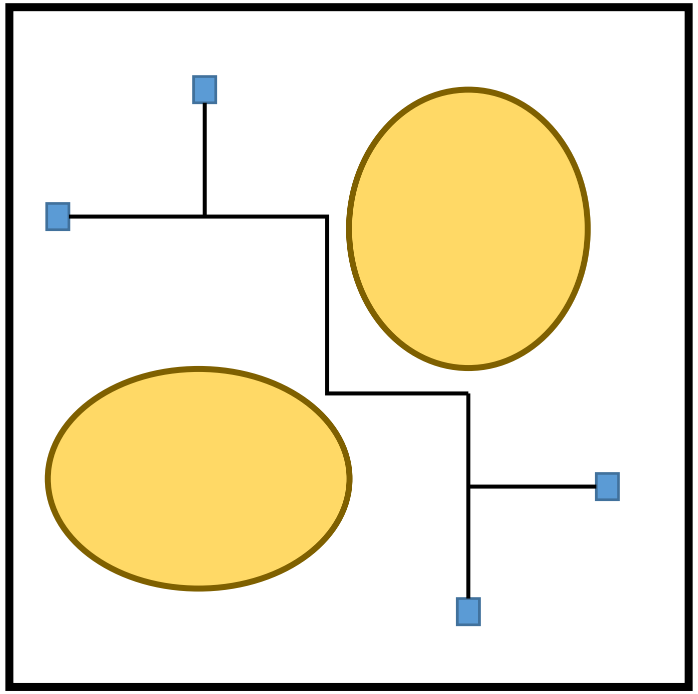
Outline

- Problem Background & Motivation
- Overall Flow of PROS
- FCN-based Predictor for GR Congestion
- Optimizer for GR Cost Parameters
- Experimental Results
- Conclusion

Optimizer for GR Cost Parameters



Before optimization



After optimization

- For each congested grid cell:
  - Increase the overflow cost
  - Increase the wire/via cost for the nets with large BBox

Outline

- Problem Background & Motivation
- Overall Flow of PROS
- FCN-based Predictor for GR Congestion
- Optimizer for GR Cost Parameters
- Experimental Results
- Conclusion

Experimental Settings

- Dataset:
  - 19 designs: each design has ~80 different placements. Totally, we have ~1600 design cases.
- Experiment:
  - Divide 19 designs into 5 groups: 4, 4, 4, 4, 3. When testing one group, the remaining four groups will be used for training. Repeat the round of training and testing for 5 times.
- Evaluation:
  - Positive: congested in label; Negative: no congested in label.
  - True positive rate (TPR) = #True positive / #Positive,
  - Precision (PRE) = #True positive / (#True positive + #False positive),
  - False positive rate (FPR) = #False positive / #Negative,
  - F1 score (F1) = (2 * TPR * PRE) / (TPR + PRE),
  - Accuracy (ACC) = (#True positive + #True negative) / (#Positive + #Negative).

Results of Congestion Prediction

- Baselines for congestion prediction:
  - **LR1X1**: Logistic regression.
  - **LR9X9**: Enhanced **LR1X1** with a window size of 9×9 g-cells to capture neighboring features.
  - **OneSUB**: Replace three cascaded SUBs by one RB and one SUB.
  - **ThreeSUB-NoSkipAdd**: Remove all the skip connections and addition operators.

Model	F1 (%)	FPR (%)	ACC (%)
LR1X1	63.86	25.39	75.25
LR9X9	66.38	16.70	79.67
OneSUB	70.09	10.45	84.23
ThreeSUB-NoSkipAdd	70.87	8.74	85.32
PROS	73.34	8.92	86.15

Results of Routability Optimization

Designs	Congested G-cell Ratio			#DRC Violations				Wire Length Diff (%)	Via Count Diff (%)
	Orig (%)	PROS (%)	Diff* (%)	Orig	PROS	PROS - Orig	Diff (%)		
Design1	3.93	3.74	-6.27	40	6	-34	-85.00	0.29	0.20
Design2	3.74	3.78	1.07	62	71	9	14.52	0.03	-0.09
Design3	2.91	2.78	-4.47	266	171	-95	-35.71	-0.04	-0.05
Design4	3.62	3.59	-0.8	59	48	-11	-18.64	0.12	0.03
Design5	7.40	7.25	-2.03	30	39	9	30.00	0.01	0.00
Design6	5.32	5.28	-0.75	31	34	3	9.68	-0.04	-0.09
Design7	11.41	10.99	-3.68	2350	2157	-193	-8.21	0.23	0.05
Design8	8.83	8.04	-8.95	1251	1306	55	4.40	0.29	0.19
Design9	3.01	2.91	-3.32	1422	1367	-55	-3.87	0.07	-0.11
Design10	4.52	4.25	-5.97	453	355	-98	-21.63	0.02	-0.15
Design11	5.48	5.11	-6.75	1105	1022	-83	-7.51	0.01	-0.08
Design12	6.05	6.18	2.15	628	603	-25	-3.98	0.11	0.01
Design13	5.40	5.31	-1.67	65	57	-8	-12.31	-0.01	0.04
Design14	5.35	5.13	-4.11	793	813	20	2.52	-0.01	-0.07
Design15	4.57	4.44	-2.84	870	722	-148	-17.01	-0.04	0.01
Design16	4.45	4.19	-5.84	636	502	-134	-21.07	-0.01	0.10
Design17	4.28	3.94	-7.94	581	556	-25	-4.30	0.10	0.16
Design18	8.38	7.91	-5.61	267	157	-110	-41.20	0.63	0.13
Design19	6.09	5.83	-4.27	879	862	-17	-1.93	0.14	0.03
Average			-3.79			-49.47	-11.65	0.10	0.02

* Diff = 100% * (PROS – Orig) / Orig

Outline

- Problem Background & Motivation
- Overall Flow of PROS
- FCN-based Predictor for GR Congestion
- Optimizer for GR Cost Parameters
- Experimental Results
- Conclusion

Conclusion

- Propose a prediction model used in PROS to predict routing congestion:
  - Based on fully convolutional network (FCN),
  - Only use data collected from placement,
  - Achieve a high prediction accuracy.
- By utilizing prediction results, PROS can improve routability:
  - Effectively reduce routing congestion ratio (-3.79%) and DRC violation number (-11.65%) by optimizing cost parameters of GR,
  - Maintain wire length (+0.10%) and via count (+0.02%).

Thank you for your attention!

Q & A