Congestion and Timing Aware Macro Placement Using Machine Learning Predictions from Different Data Sources: Cross-design Model Applicability and the Discerning Ensemble*

Xiang Gao† #, Yi-Min Jiang† #, Lixin Shao #, Pedja Raspopovic #, Menno E. Verbeek #, Manish Sharma#, Vineet Rashingkar #, Amit Jalota#

#Synopsys Inc. Mountain View, CA, USA. {xiangg, jym, lixin, pedja, menno, smanish, vineet, jalota} @synopsys.com

Presenter:

Xiang Gao and Yi-Min Jiang

*Part of this work is included in a US patent provisional filing.
†Equal contribution.

ISPD '22, March 27–30, 2022, Banff, AB, Canada

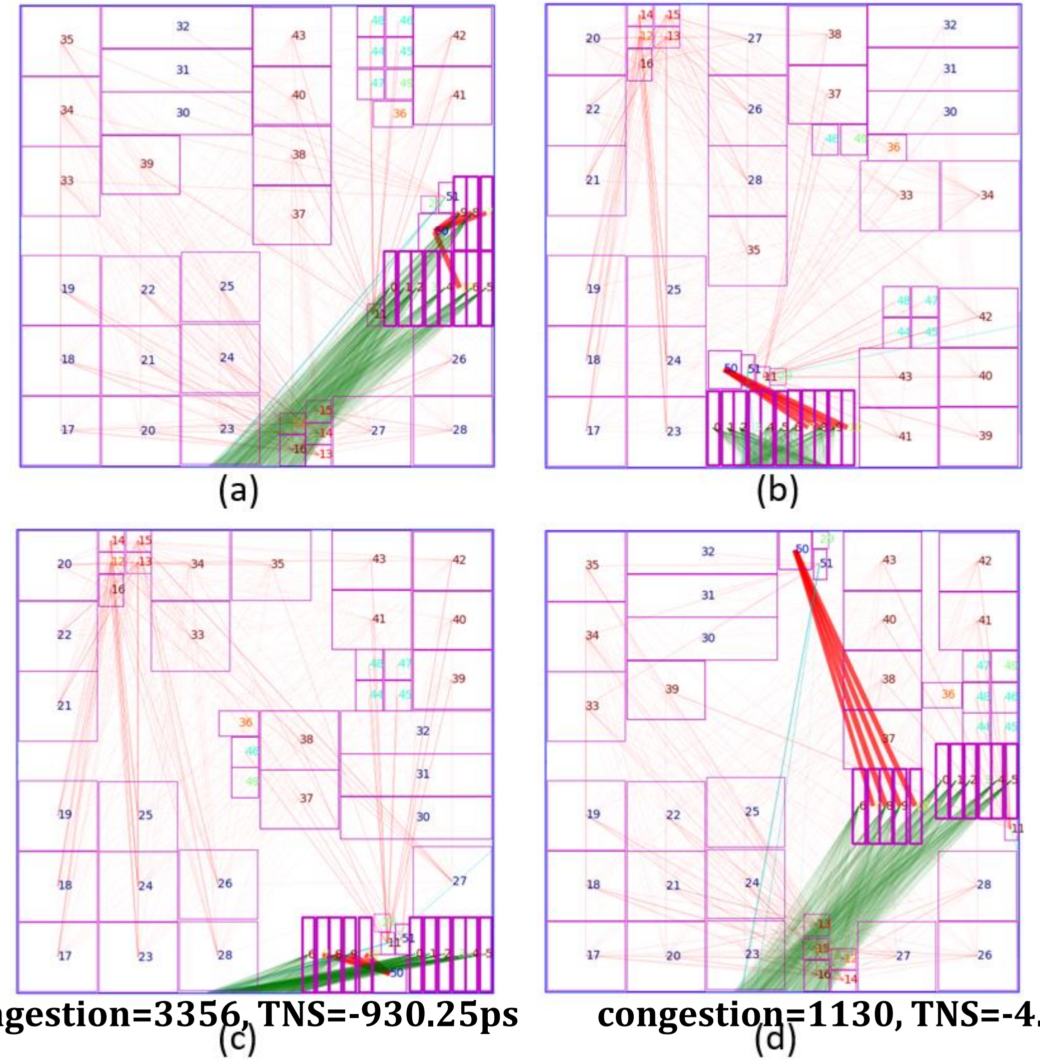
ACM ISBN 978-1-4503-9210-5/22/03…$15.00
https://doi.org/10.1145/3505170.3506722

Outline

1. Introduction
2. Our ML prediction Approach
3. Feature Engineering
    1. List of Features
    2. Feature ranking in 1-on-1, all-on-1 cross-design scenarios
    3. Feature visualization
    4. Cross-design prediction baselines
4. Discerning Ensemble
    1. Theory
    2. Applied to Macro Placement
5. Results
    1. on single design group (G3)
    2. on multiple design groups
    3. on 8 unseen leading-edge (7-16nm) designs
6. Conclusion

1. Motivation: Macro Placement has a Big Impact on Congestion and Timing

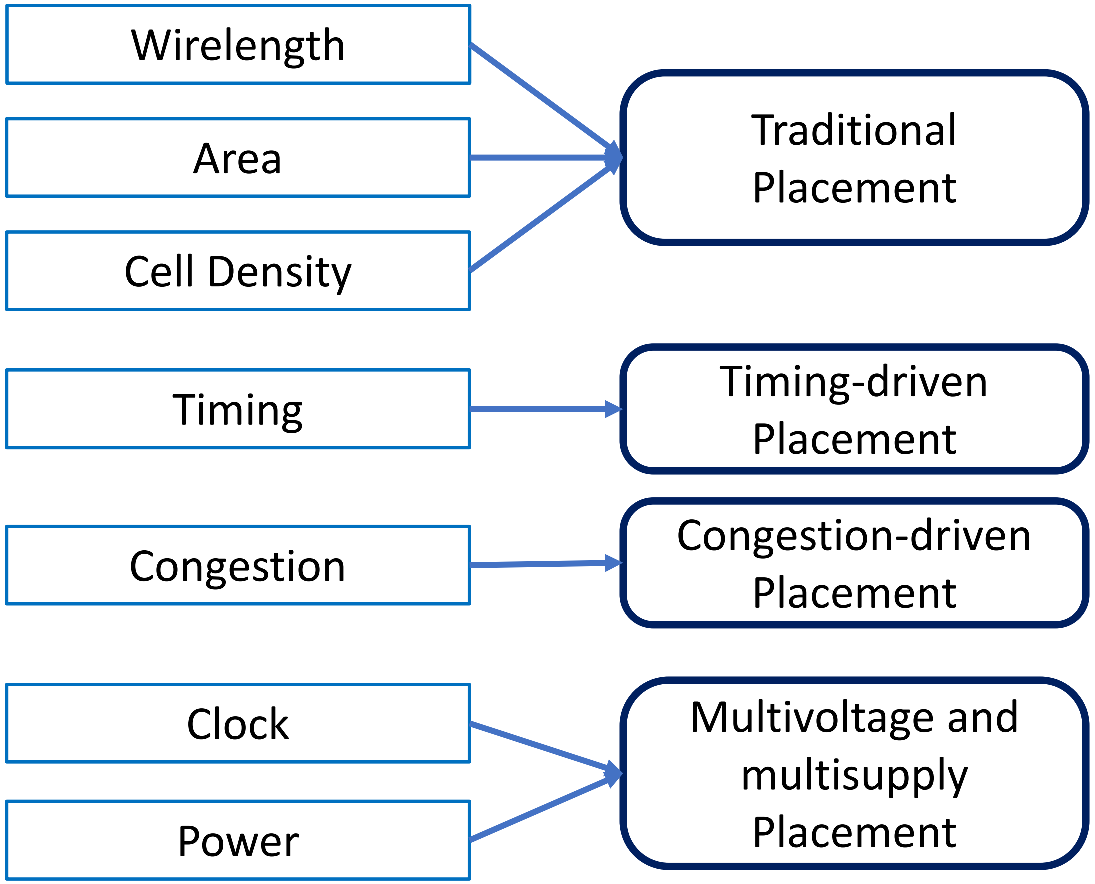
congestion=12355, TNS=-6.52ps congestion=1016, TNS=-10.72ps



congestion=3356, TNS=-930.25ps congestion=1130, TNS=-4.59ps

Figure 1: 4 macro placement candidates from 1 training design. The colored lines are connections among macros and IO ports.

1. Macro Placement in Commercial EDA Tool Can be Complicated



How much do we know the EDA tools' behavior?

- Many different tool options for a variety of reasons.
- Many different type of designs, technologies
- Customers' specific requirements
- P&R is an NP-hard problem

The steps after macro placement, such as place_opt, makes it even harder to predict the final QoR.

1. ML-Guided Macro Placement, A Recent Development in EDA

Category 1. Enhance the operations of traditional placement engines

- data path prediction, etc. [15,16]
- GNN-based embedding and clustering for cell clusters. [18]

Category 2. Look-ahead ML prediction for the traditional placers

- Routing-aware placement
  - Design rule checking (DRC) hot spot [20-28]
  - Congestion
    - global routing congestion map [27]
    - local routing congestion
- Timing, Cross talk [25]
- Scalar QoR metrics: wirelength, congestion, timing, power, etc. [8,9,27]

Category 3. ML, DL & RL Placers that can replace traditional placers

- RL + GCN [12,13,14]

[8] K. Jeong, A. B. Kahng, B. Lin, and K. Samadi. 2010. Accurate Machine-Learning-Based On-Chip Router Modeling. IEEE Embedded Systems Letters (ESL) 2, 3 (2010), 62–66.

[9] W.-T. J. Chan, Y. D., A. B. Kahng, S. Nath, and K. Samadi. 2016. BEOL Stack-Aware Routability Prediction from Placement Using Data Mining Techniques. In IEEE International Conference on Computer Design (ICCD). 41–48.

[12] A. Goldie and A. Mirhoseini. 2020. Placement Optimization with Deep Reinforcement Learning. In Proceedings of the 2020 International Symposium on Physical Design (ISPD '20), March 29-April 1, 2020, Taipei, Taiwan. ACM, New York, NY, USA, 5 pages. https://doi.org/10.1145/3372780.3378174

[13] A. Mirhoseini, A. Goldie, M. Yazgan, J. Jiang, E. M. Songhori, S. Wang, Y.-J. Lee, E. Johnson, O. Pathak, S. Bae, A. Nazi, J. Pak, A. Tong, K. Srinivasa, W. Hang, E. Tuncer, A. Babu, Quoc V. Le, J. Laudon, R. C. Ho, R. Carpenter, and J. Dean. 2020. Chip Placement with Deep Reinforcement Learning. CoRR abs/2004.10746 (2020). arXiv:2004.10746

[14] Z. Jiang, E. Songhori, S. Wang, A. Goldie, A. Mirhoseini, J. Jiang, Y.J. Lee, and D. Pan, 2021. Delving into Macro Placement with Reinforcement Learning. 1-3. 10.1109/MLCAD52597.2021.9531313.

[22] Z. Xie, Y-H Huang, G-Q Fang, H. Ren, S-Y Fang, Y. Chen, and J. Hu. 2018. RouteNet: Routability Prediction for Mixed-Size Designs Using Convolutional Neural Network. In IEEE/ACM International Conference on Computer-Aided Design (ICCAD). 80.

[23] Z. Xie, H. Ren, B. Khailany, Y. Sheng, S. Santosh, J. Hu, and Y. Chen. 2020. PowerNet: Transferable Dynamic IR Drop Estimation via Maximum Convolutional Neural Network. In IEEE/ACM Asia and South Pacific Design Automation Conference (ASPDAC). 13–18.

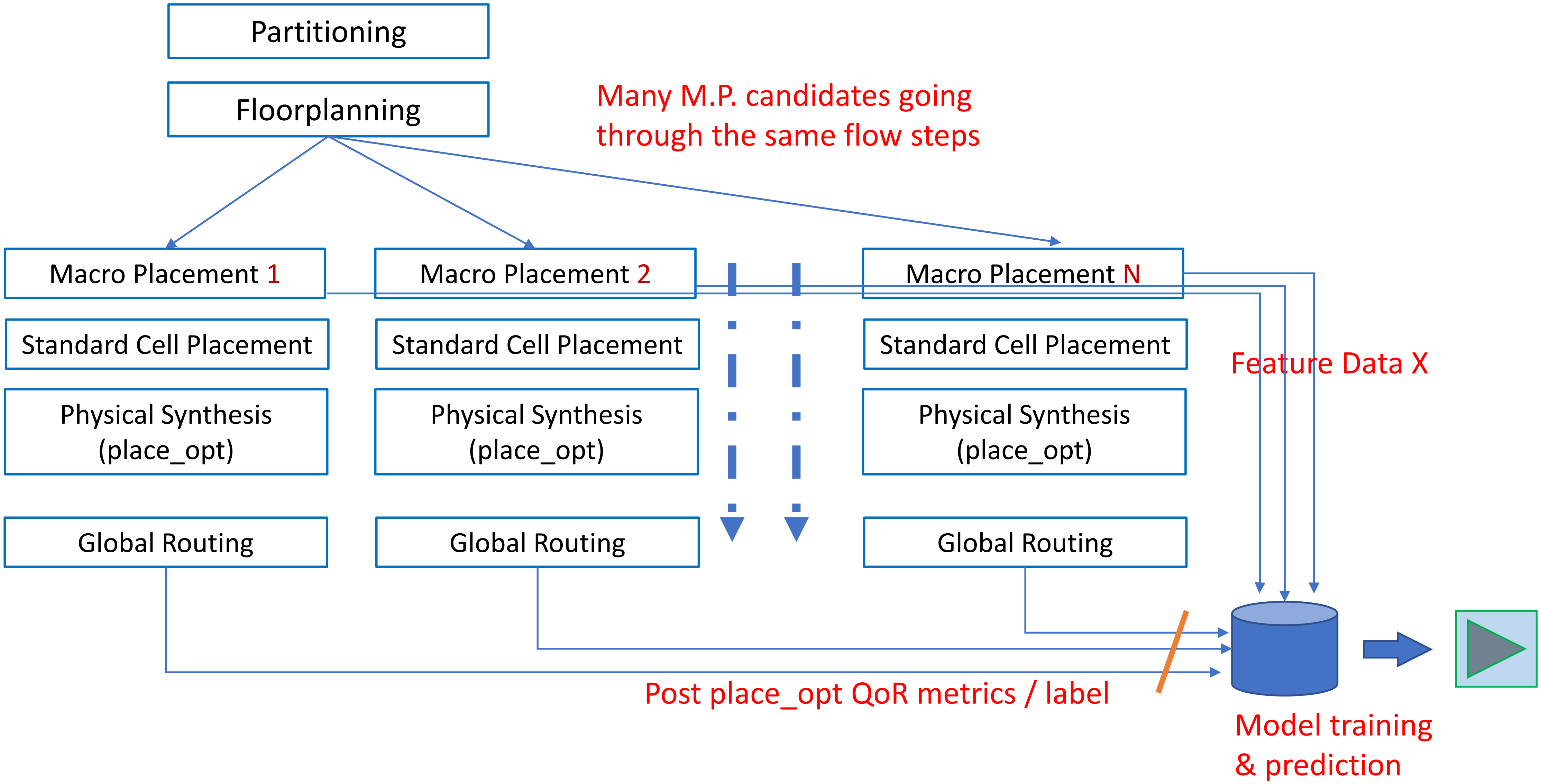
[24] R. Liang, H. Xiang, D. Pandey, L. N. Reddy, S. Ramji, G-J Nam, and J. Hu. 2020. DRC Hotspot Prediction at Sub-10nm Process Nodes Using Customized Convolutional Network. In ACM International Symposium on Physical Design (ISPD). 135–142.

[25] R. Liang, Z. Xie, J. Jung, V. Chauha, Y. Chen, J. Hu, H. Xiang, and G. J. Nam. 2020. Routing-Free Crosstalk Prediction. In 2020 IEEE/ACM International Conference on Computer Aided Design (ICCAD). 1–9.

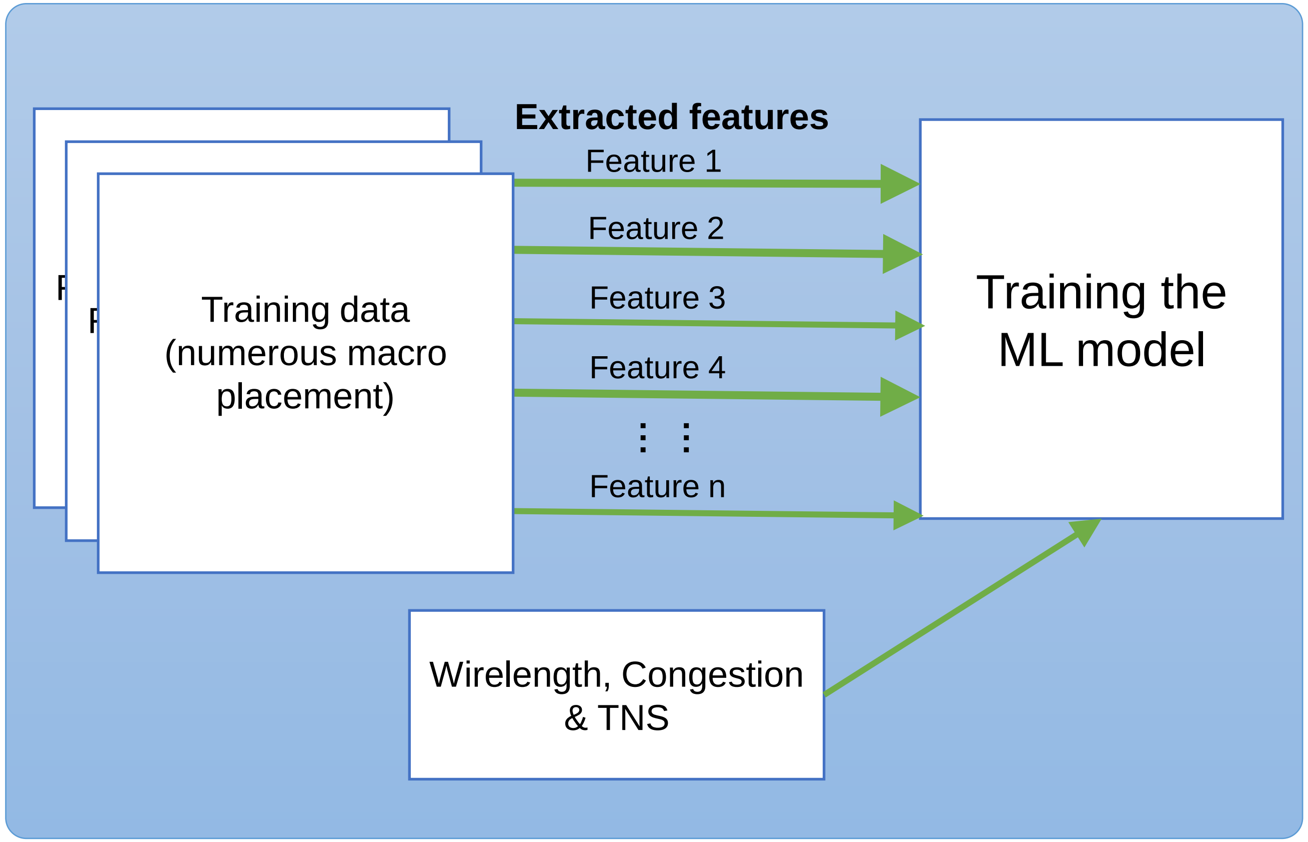
[27] J. Chen, J. Kuang, G. Zhao, D. J. -H. Huang and E. F. Y. Young, "PROS: A Plug-in for Routability Optimization applied in the State-of-the-art commercial EDA tool using deep learning," 2020 IEEE/ACM International Conference on Computer Aided Design (ICCAD), 2020, pp. 1-8.

[28] L. Li, Y. Cai and Q. Zhou, "An Efficient Approach for DRC Hotspot Prediction with Convolutional Neural Network," 2021 IEEE International Symposium on Circuits and Systems (ISCAS), 2021, pp. 1-5, doi: 10.1109/ISCAS51556.2021.9401274.

2. Our Congestion and Timing Aware ML Macro Placements Approach



3.1 Features of Macro Placement for Wirelength, Congestion & TNS Predictions



Extract 65 Features

1. Design statistics:

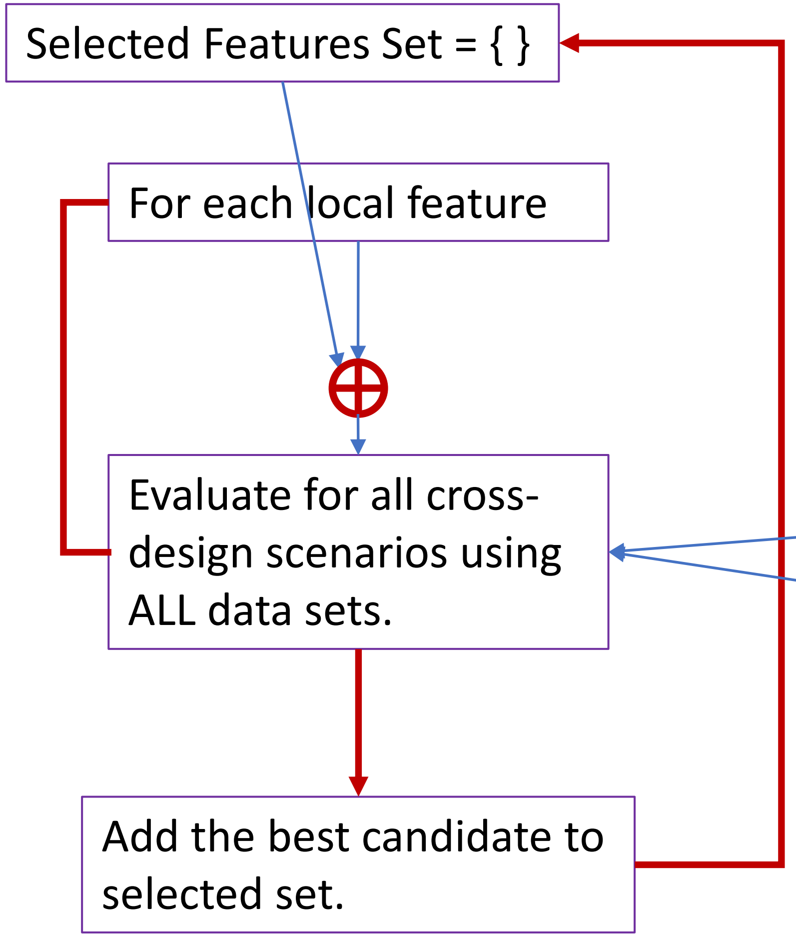
   # of cells, # of hard macros, # of nets, # of layers, #of ports, # of hard macro pins, # of blocks, technology node, etc.

2. Macro placement topology

   macro placement area, proportion of macro area/macro placement area, macro pin connectivity, features related to location of macros, features related to connections, etc.

3. Macro grouping, etc.

Feature Engineering Methodology



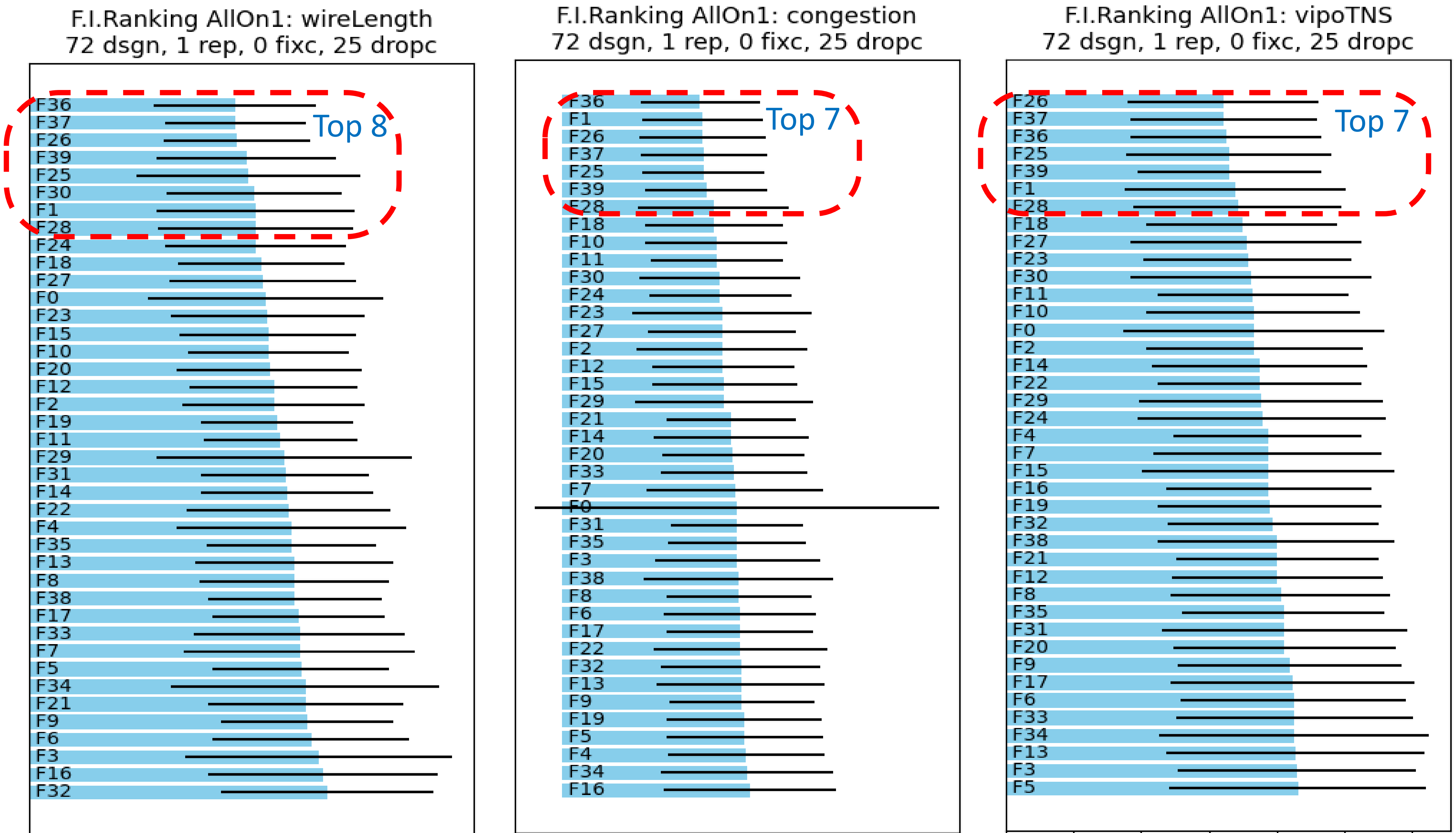
Cross-design (test design is excluded from training designs) prediction scenarios:

"1-on-1": A => B

"all-on-1": {not A} => A

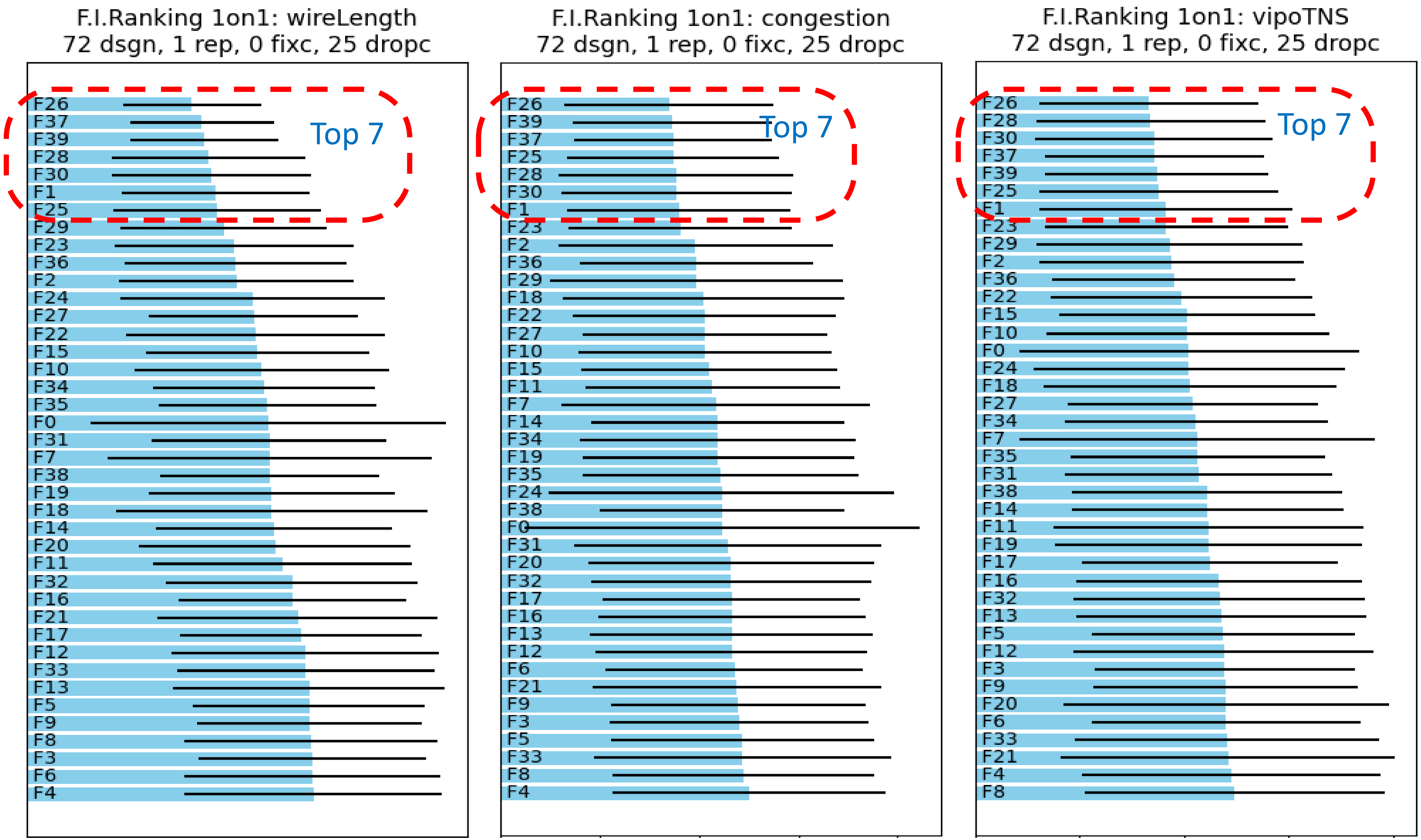
3.2 Feature Importance Ranking for all-on-1 Cross-Design Prediction

Important feature ranking are similar for 3 different targets:
F36, F26, F37, F25, F1, F39, F28 appear in top 7 or 8.

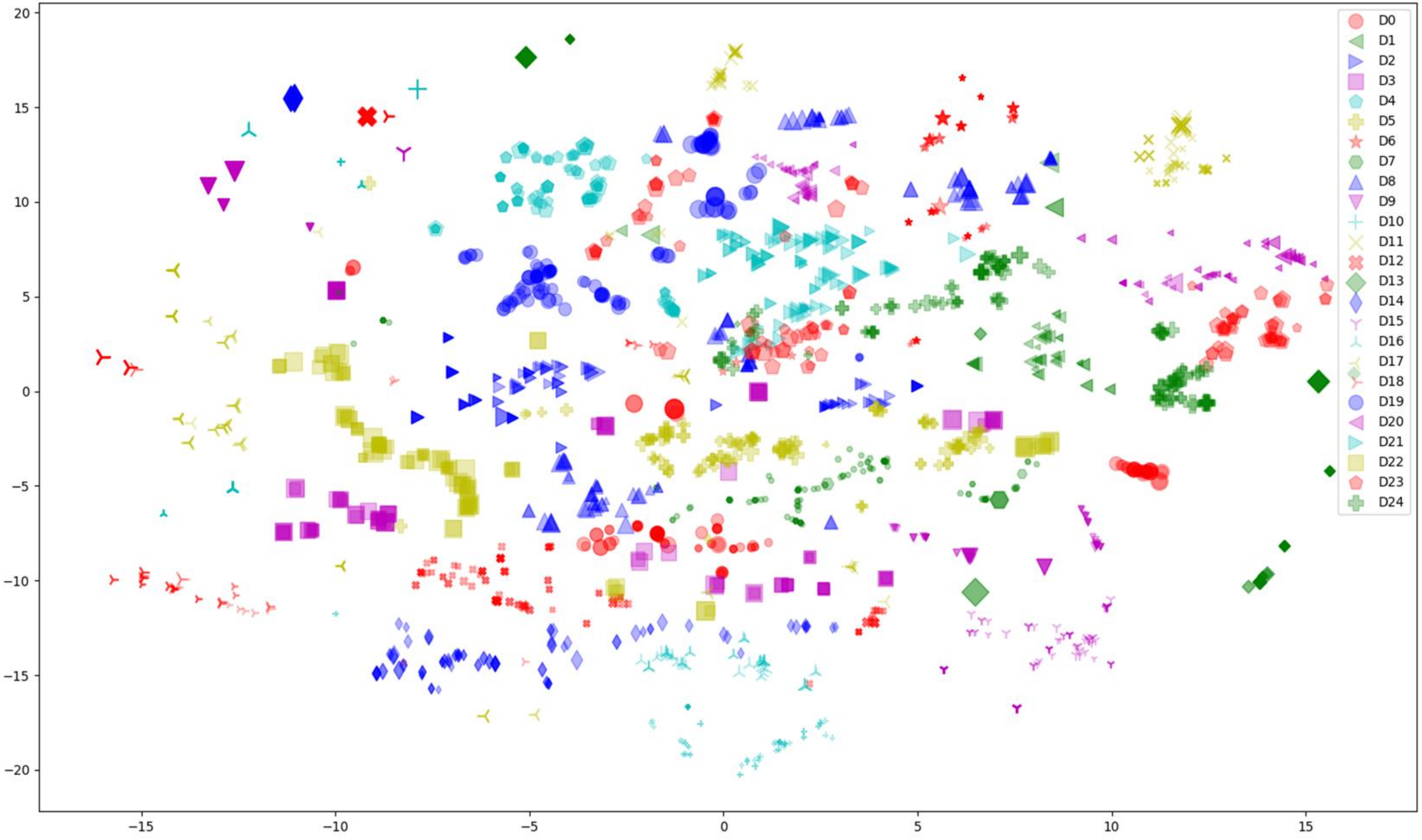


3.2 Feature Importance Ranking for 1-on-1 Cross-Design Prediction

Important feature ranking are similar for 3 different targets:
F26, F37, F39, F28, F30, F25, F1 appear in top 7.



3.3 Feature Data Visualization via t-SNE



25 designs from group 1, 2 and 3 {G1: D0-D7; G2: D8-18; G3: D19-D24} are shown here using tSNE. Data points were grouped by design. The marker size is proportional to the target "congestion" value. (Larger size corresponds to larger "congestion" value.

3.4 Cross-Design Baseline, 1-on-1 and all-on-1 Scenarios

3 groups of total 25 designs used as 1-on-1 cross-design prediction; MSE of “congestion”.

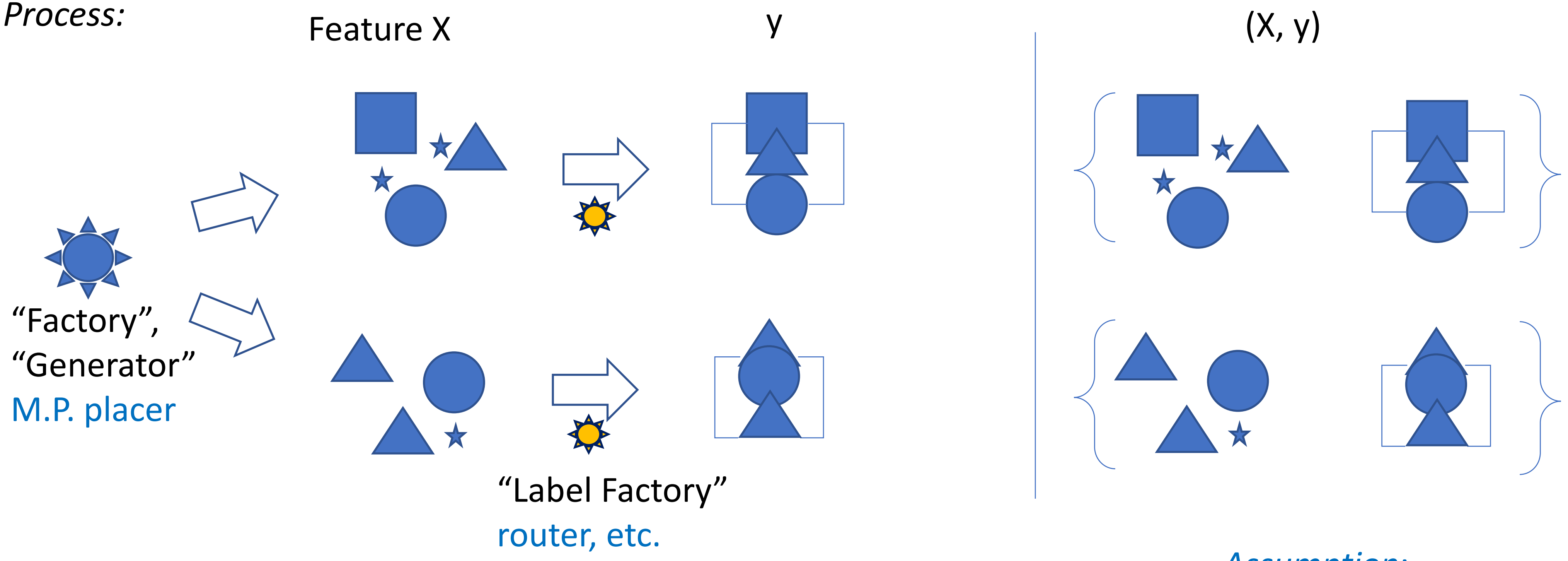
		G1.D1	G1.D2	G1.D3	G1.D4	G1.D5	G1.D6	G1.D7	G1.D8	G2.D1	G2.D2	G2.D3	G2.D4	G2.D5	G2.D6	G2.D7	G2.D8	G2.D9	G2.D10	G2.D11	G3.D1	G3.D2	G3.D3	G3.D4	G3.D5	G3.D6	Row-avg
Training	G1.D1		0.084	0.035	0.100	0.045	0.071	0.067	0.162	0.094	0.169	0.318	0.087	0.364	0.261	0.159	0.359	0.063	0.081	0.289	0.100	0.070	0.063	0.109	0.088	0.053	0.137
	G1.D2	0.066		0.036	0.140	0.041	0.068	0.088	0.077	0.068	0.066	0.130	0.099	0.139	0.139	0.084	0.202	0.042	0.104	0.152	0.100	0.062	0.081	0.147	0.116	0.101	0.098
	G1.D3	0.048	0.083		0.103	0.049	0.061	0.057	0.053	0.083	0.073	0.096	0.055	0.055	0.109	0.034	0.028	0.059	0.112	0.104	0.063	0.063	0.039	0.127	0.105	0.080	0.072
	G1.D4	0.084	0.143	0.046		0.063	0.097	0.134	0.194	0.068	0.198	0.237	0.096	0.119	0.079	0.090	0.070	0.080	0.061	0.066	0.042	0.259	0.043	0.086	0.088	0.062	0.104
	G1.D5	0.053	0.078	0.062	0.058		0.102	0.108	0.139	0.066	0.105	0.103	0.084	0.090	0.136	0.051	0.050	0.037	0.049	0.083	0.047	0.067	0.036	0.156	0.102	0.090	0.081
	G1.D6	0.063	0.073	0.090	0.057	0.074		0.146	0.053	0.073	0.062	0.171	0.108	0.073	0.146	0.054	0.068	0.061	0.089	0.079	0.079	0.063	0.056	0.142	0.128	0.062	0.086
	G1.D7	0.097	0.133	0.058	0.153	0.086	0.120		0.024	0.185	0.213	0.071	0.076	0.079	0.038	0.130	0.063	0.149	0.158	0.224	0.170	0.044	0.112	0.215	0.179	0.137	0.121
	G1.D8	0.146	0.132	0.079	0.259	0.112	0.143	0.049		0.262	0.230	0.074	0.077	0.049	0.065	0.132	0.079	0.184	0.285	0.268	0.213	0.035	0.160	0.255	0.231	0.193	0.155
	G2.D1	0.075	0.174	0.078	0.061	0.046	0.078	0.063	0.199		0.127	0.130	0.165	0.047	0.080	0.112	0.066	0.046	0.049	0.073	0.034	0.178	0.041	0.067	0.074	0.044	0.088
	G2.D2	0.149	0.132	0.055	0.123	0.089	0.096	0.263	0.129	0.100		0.310	0.265	0.217	0.050	0.253	0.219	0.096	0.085	0.117	0.169	0.271	0.256	0.121	0.192	0.162	0.163
	G2.D3	0.143	0.161	0.100	0.220	0.297	0.138	0.268	0.038	0.169	0.089		0.209	0.005	0.323	0.059	0.027	0.139	0.261	0.245	0.163	0.384	0.067	0.268	0.102	0.060	0.164
	G2.D4	0.099	0.103	0.054	0.203	0.064	0.076	0.042	0.030	0.197	0.175	0.015		0.008	0.057	0.037	0.050	0.145	0.252	0.226	0.149	0.031	0.133	0.218	0.158	0.158	0.112
	G2.D5	0.118	0.136	0.068	0.175	0.143	0.112	0.243	0.030	0.143	0.145	0.046	0.078		0.089	0.045	0.024	0.119	0.147	0.145	0.090	0.098	0.064	0.229	0.095	0.093	0.111
	G2.D6	0.065	0.092	0.035	0.156	0.057	0.080	0.038	0.046	0.166	0.067	0.106	0.116	0.052		0.100	0.045	0.113	0.188	0.195	0.111	0.070	0.099	0.169	0.151	0.120	0.101
	G2.D7	0.093	0.240	0.056	0.132	0.220	0.076	0.321	0.084	0.114	0.079	0.098	0.486	0.029	0.557		0.029	0.102	0.126	0.078	0.185	0.546	0.156	0.142	0.122	0.122	0.175
	G2.D8	0.069	0.267	0.052	0.111	0.221	0.058	0.409	0.097	0.124	0.067	0.062	0.368	0.020	0.261	0.011		0.081	0.117	0.057	0.148	0.423	0.190	0.139	0.158	0.145	0.152
	G2.D9	0.072	0.096	0.095	0.102	0.051	0.088	0.173	0.252	0.078	0.163	0.232	0.102	0.197	0.136	0.206	0.191		0.084	0.066	0.045	0.228	0.083	0.105	0.102	0.084	0.126
	G2.D10	0.244	0.133	0.120	0.049	0.087	0.059	0.071	0.209	0.084	0.190	0.310	0.121	0.164	0.084	0.221	0.179	0.199		0.163	0.068	0.201	0.057	0.091	0.066	0.053	0.134
	G2.D11	0.090	0.080	0.074	0.108	0.042	0.074	0.046	0.093	0.136	0.076	0.243	0.065	0.067	0.079	0.045	0.092	0.045	0.100		0.085	0.078	0.078	0.119	0.111	0.094	0.088
	G3.D1	0.075	0.056	0.051	0.055	0.075	0.060	0.174	0.115	0.060	0.146	0.220	0.112	0.228	0.128	0.147	0.117	0.049	0.085	0.096		0.082	0.051	0.096	0.067	0.050	0.100
	G3.D2	0.109	0.106	0.050	0.179	0.079	0.099	0.138	0.100	0.186	0.204	0.112	0.147	0.193	0.157	0.145	0.158	0.118	0.210	0.244	0.185		0.128	0.230	0.200	0.125	0.150
	G3.D3	0.169	0.074	0.019	0.084	0.035	0.049	0.266	0.071	0.112	0.090	0.189	0.059	0.307	0.255	0.050	0.164	0.062	0.105	0.111	0.079	0.051		0.123	0.094	0.062	0.112
	G3.D4	0.311	0.093	0.236	0.127	0.213	0.187	0.379	0.064	0.132	0.293	0.637	0.382	0.430	0.348	0.363	0.298	0.175	0.143	0.147	0.155	0.492	0.259		0.081	0.124	0.253
	G3.D5	0.297	0.072	0.240	0.152	0.239	0.269	0.438	0.063	0.168	0.333	0.573	0.454	0.378	0.328	0.363	0.347	0.162	0.098	0.216	0.158	0.405	0.153	0.125		0.101	0.256
	G3.D6	0.137	0.083	0.164	0.072	0.119	0.123	0.280	0.058	0.086	0.113	0.337	0.218	0.331	0.237	0.227	0.174	0.093	0.059	0.123	0.074	0.086	0.082	0.116	0.109		0.146
	Col-Avg	0.120	0.118	0.081	0.124	0.106	0.099	0.177	0.099	0.123	0.145	0.201	0.168	0.152	0.173	0.130	0.129	0.101	0.127	0.149	0.113	0.179	0.104	0.150	0.122	0.099	

All-on-1 cross-design prediction; MSE of “congestion”.

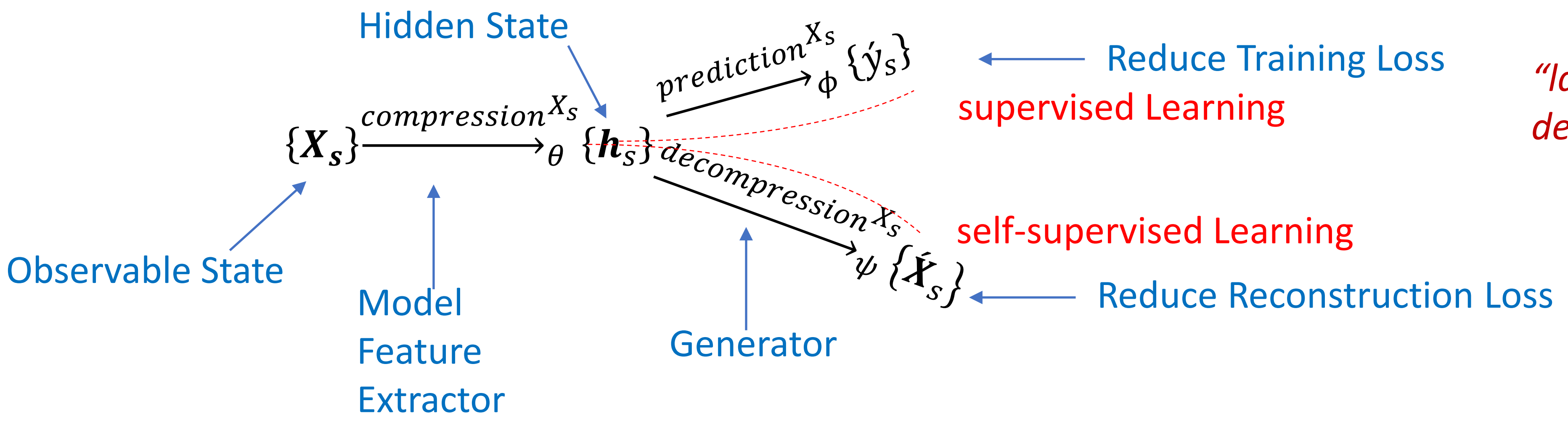
Target: congestion																									
	G1.D1	G1.D2	G1.D3	G1.D4	G1.D5	G1.D6	G1.D7	G1.D8	G2.D1	G2.D2	G2.D3	G2.D4	G2.D5	G2.D6	G2.D7	G2.D8	G2.D9	G2.D10	G2.D11	G3.D1	G3.D2	G3.D3	G3.D4	G3.D5	G3.D6
All Other Designs	0.082	0.130	0.035	0.106	0.079	0.075	0.049	0.024	0.208	0.076	0.094	0.069	0.006	0.049	0.064	0.062	0.954	0.101	0.171	0.086	0.053	0.059	0.138	0.192	0.074
Total Average	0.121																								

4.1 Discerning Ensemble Theory

Underpinning Process:



ML Abstraction:



Assumption:

“labels are primarily determined by feature X”

4.1 Training Loss, Reconstruction Loss & Projected Loss

Test Loss

Training Loss

$$\frac{L(y_t, \acute{y}_t)}{L(\boldsymbol{X}_t, \acute{\boldsymbol{X}}_t)} \approx \frac{L(y_s, \acute{y}_s)}{L(\boldsymbol{X}_s, \acute{\boldsymbol{X}}_s)}$$

Training Reconstruction Loss

Test Reconstruction Loss

$$L(y_t, \acute{y}_t) \approx max\left(\frac{L(\boldsymbol{X}_t, \acute{\boldsymbol{X}}_t)}{L(\boldsymbol{X}_s, \acute{\boldsymbol{X}}_s)}, 1\right) L(y_s, \acute{y}_s)$$

Projected Prediction Loss

Training Loss

$$\frac{Bias(y_t, \acute{y}_t)}{Bias(\boldsymbol{X}_t, \acute{\boldsymbol{X}}_t)} \approx \frac{Bias(y_s, \acute{y}_s)}{Bias(\boldsymbol{X}_s, \acute{\boldsymbol{X}}_s)}$$

$$L(y_t, \acute{y}_t) \approx \max\left(\frac{L(\boldsymbol{X}_t, \acute{\boldsymbol{X}}_t) - \sigma_X}{L(\boldsymbol{X}_s, \acute{\boldsymbol{X}}_s) - \sigma_X}, 1\right) L(y_s, \acute{y}_s) - \dot{\sigma}_y \max\left(\frac{L(\boldsymbol{X}_t, \acute{\boldsymbol{X}}_t) - L(\boldsymbol{X}_s, \acute{\boldsymbol{X}}_s)}{L(\boldsymbol{X}_s, \acute{\boldsymbol{X}}_s) - \sigma_X}, 0\right)$$

4.1 Discerning Ensemble is Different From Stacking

$$\acute{y} = \frac{1}{\sum_i w_i} \sum_i w_i \acute{y}_i(\boldsymbol{X})$$
$$w_i > 0$$

Stacking:

w_i

$\acute{y}_i(\theta)$

Minimize Cross validation error [30, 36]

Discerning:

Weighting is based on “projected loss” or “bias”, which dynamically changes with input data set.

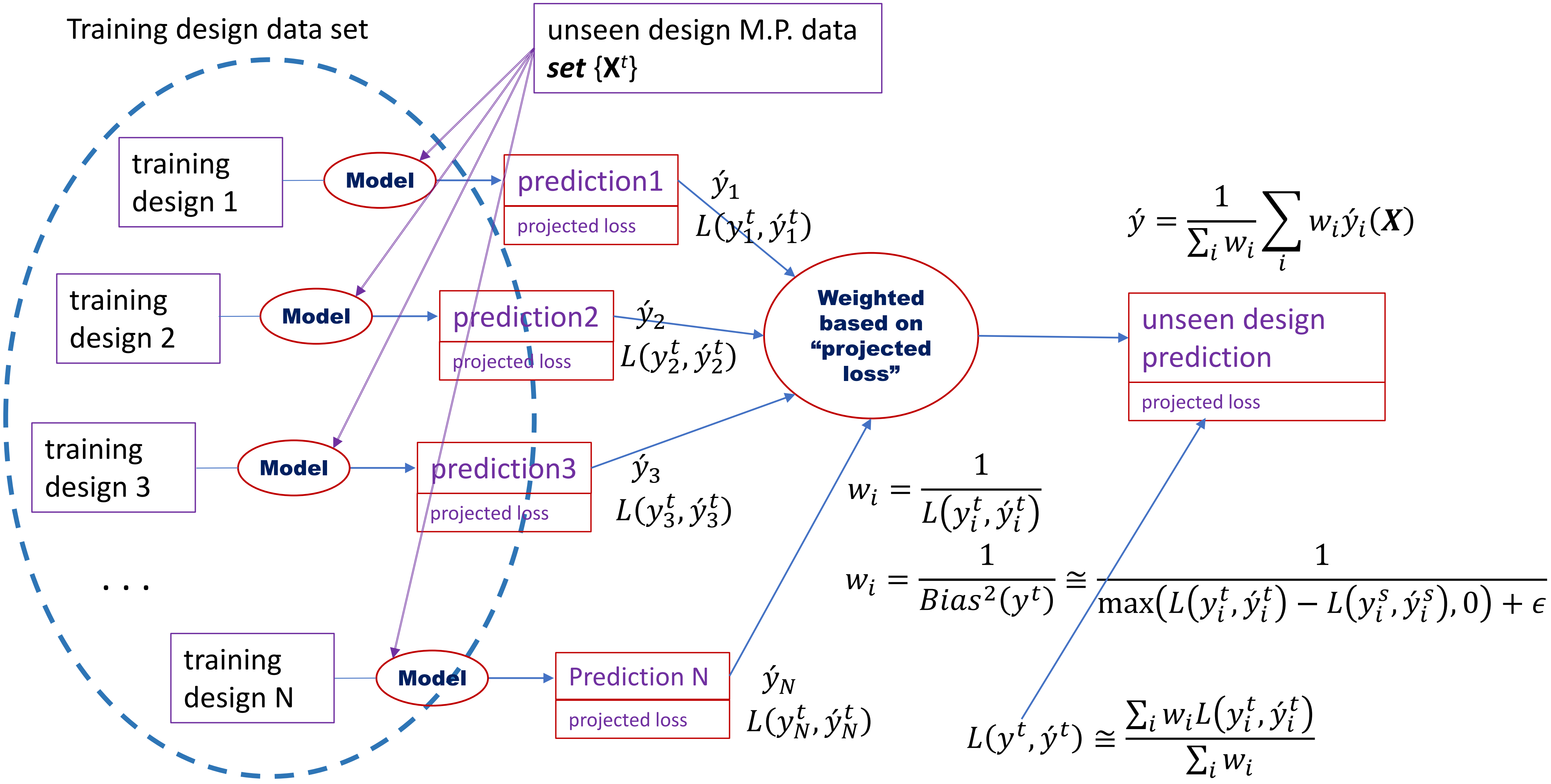
$$w_i = \frac{1}{L_i(y^t, \acute{y}^t)}$$

$$w_i = \frac{1}{Bias^2(y^t)} \cong \frac{1}{\max(L_i(y^t, \acute{y}^t) - L_i(y^s, \acute{y}^s), 0) + \epsilon}$$

Comment:
Interestingly, weighting is used as “attention” in GNN, and “gating” in RNN, the purpose of the weighting scalars is different.

[30] M. P. Perrone, L. N. Cooper, 1992. When networks disagree: Ensemble methods for hybrid neural networks: BROWN UNIV PROVIDENCE RI INST FOR BRAIN AND NEURAL SYSTEMS.
[36] M. Shahhosseini, G. Hu, H. Pham, “Optimizing Ensemble Weights and Hyperparameters of Machine Learning Models for Regression Problems”

4.2 Discerning Ensemble Applied to ML Training Data From Different Designs



5. ML Experiments

Table 1: Breakdown and description of 72 training and testing designs

Group	G1	G2	G3	G4	G5	G6	G7	G8
# of designs	8	11	6	2	5	12	12	16
Avg # of macros	29	125	526	200	230	25	52	23
Avg (macro area / core area)	0.30	0.46	0.56	0.50	0.34	0.28	0.17	0.40
Technology nodes	16nm	12nm	16nm	7nm	16nm	7nm	7nm	7nm
# of standard cell ranges	3M 6M	1.5M 2M	0.8M 1.3M	0.7M 1.3M	1.5M 2.2M	1.2M 2.1M	0.8M 1.5M	0.3M 1.0M

Study 1, we pick group #3 (G3) from the 8 design groups (see Table 1) as the test set

Study 2, Randomly pick several groups to form the "design universe", then inspect averaged leave-one-design-out prediction MSE.

Study 3: Apply discerning-ensemble-based ML solutions for 8 unseen leading-edge designs (7nm-16nm) and compare with user manual results.

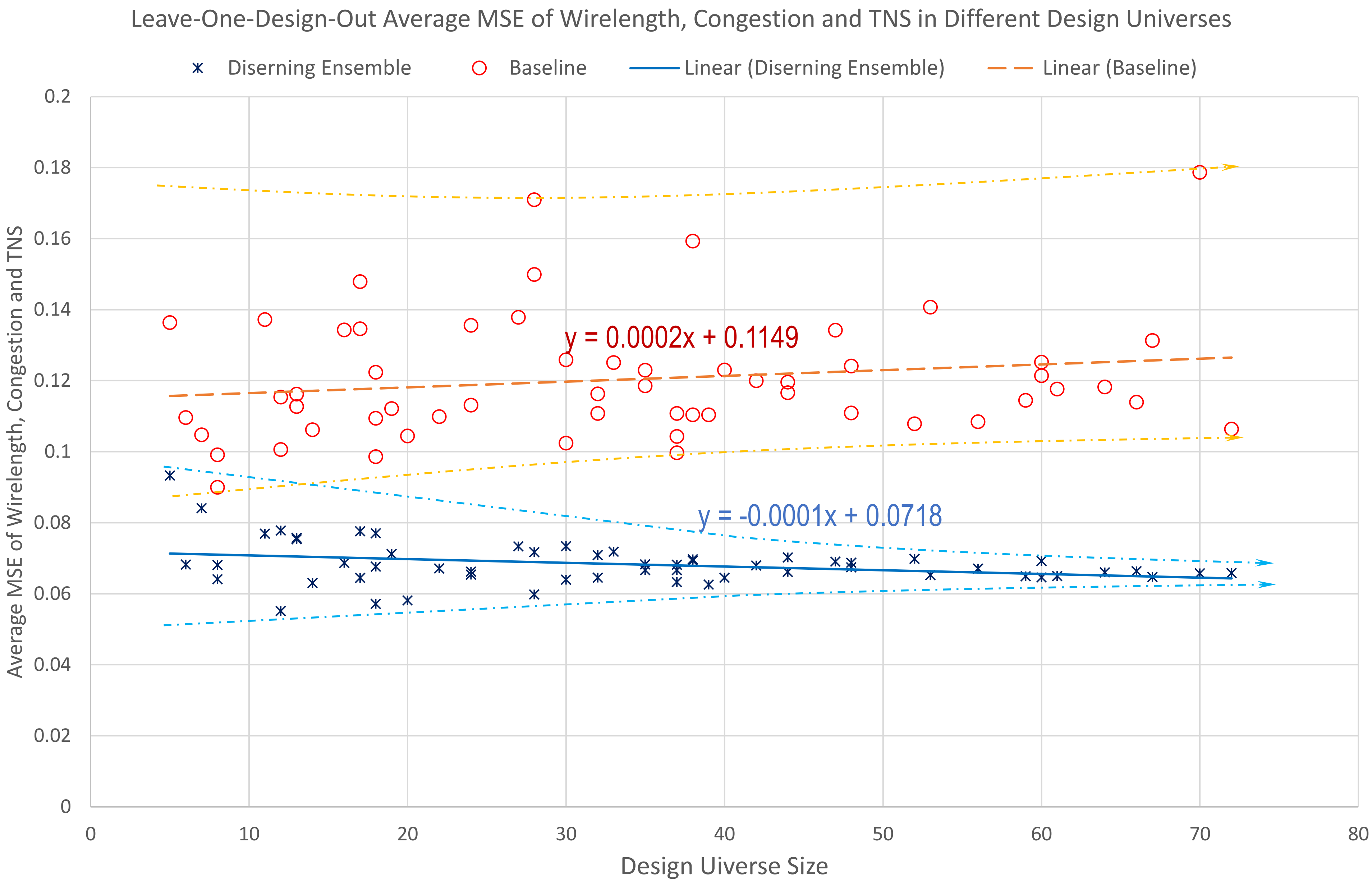
5.1 Study 1: Use Other Groups to Predict G3 designs

MSE loss of design group G3 predictions from other design groups.

MSE* Loss	wirelength	congestion	TNS	Mean MSE
Baseline	0.1637	0.1187	0.1131	0.1318
Discerning ensemble**	0.0617	0.0655	0.0495	0.0589

* MSE is computed on normalized ([0, 1]) target values. ** Weighting uses Eq. 13 and Eq.18, with $\sqrt{L(y^t)}$, $\sigma_X = 0.02$ and $\sigma_y = 0.03$.

5.2 Study2: Comparison of Discerning Ensemble Against Vanilla ML Model



5.3 Study 3: ML-Assisted (Discerning Ensemble) Compared with Manual Macro Placement with Post Place-Opt QoR for 8 Leading Edge Designs (7nm-16nm)

	WNS (ns)		TNS (ns)		Congestion GRC	
	ML	User	ML	User	ML	User
Design 1	-0.10	**-0.07**	**-0.62**	-4.23	**49 (0.01%)**	2136 (0.28%)
Design 2	**-0.02**	-0.05	-1.98	**-1.61**	**37035 (1.22%)**	NA*
Design 3	**-0.02**	-0.08	**-0.25**	-4.41	**446 (0.01%)**	2883 (0.09%)
Design 4	**-0.02**	-2.15	**-0.13**	-109.12	**18 (0.00%)**	96586 (2.91%)
Design 5	**-0.22**	-0.39	**-5.20**	-290.14	**503 (0.18%)**	595 (0.20%)
Design 6	-0.18	-0.18	**-2.11**	-20.13	**3644 (0.29%)**	4952 (0.31%)
Design 7	**-0.36**	-0.54	-434.78	**-396.77**	**2949 (0.03%)**	8611 (0.07%)
Design 8	-0.20	**-0.12**	**-390.96**	-3189.67	**4208 (0.09%)**	9853 (0.20%)

Table 3: Comparison of ML-assisted vs user manual placement results.

* Value is not provided from user due to failure in early stages.

6. Conclusion

Factors Contributed An ML-Prediction M.P. Solution:

1. Large number of features that can capture placement states that affect downstream outcome,

2. Feature engineering tailored for cross-design performance,

3. ML data management, such as the discerning ensemble framework

Other Possible Usages for Discerning Ensemble:

1. Special Purpose Learning Systems
   1. Each base learner can be separately trained without affecting others.
   2. Base learners independently developed and combined to form a dynamic learning system.

2. Debuggability for performance issues
   1. Warning for mismatch training data set.
   2. Trusted machine learning applications.