TOSHIBA
Leading Innovation

Optical Proximity Correction with Hierarchical Bayes Model

Tetsuaki Matsunawa[1], Bei Yu[2] and David Z. Pan[2]

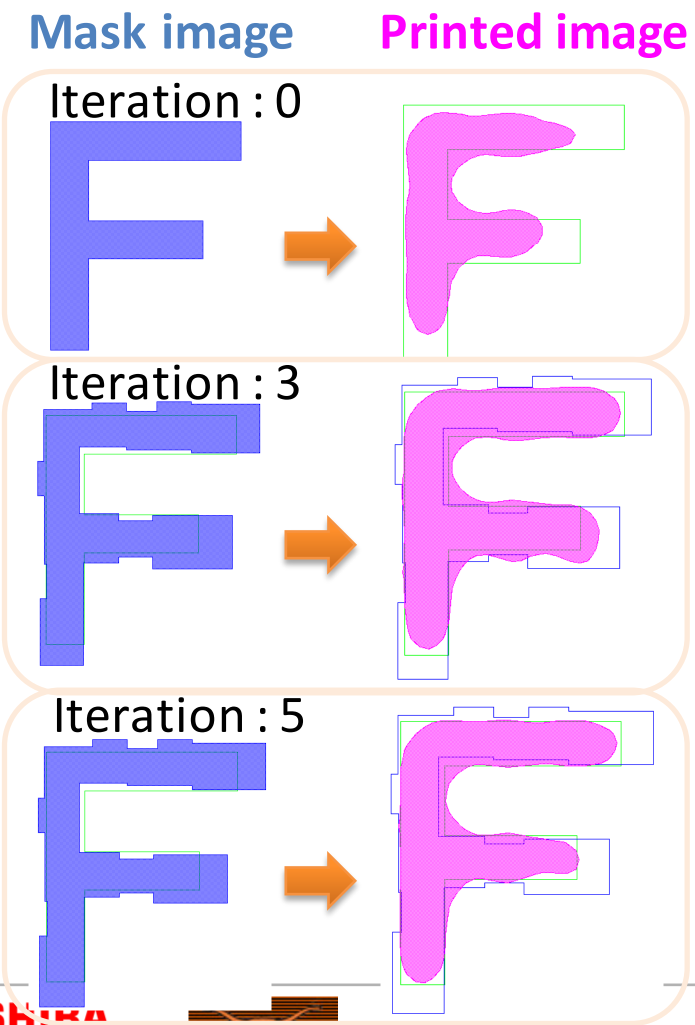
[1]Toshiba Corporation
[2]The University of Texas at Austin

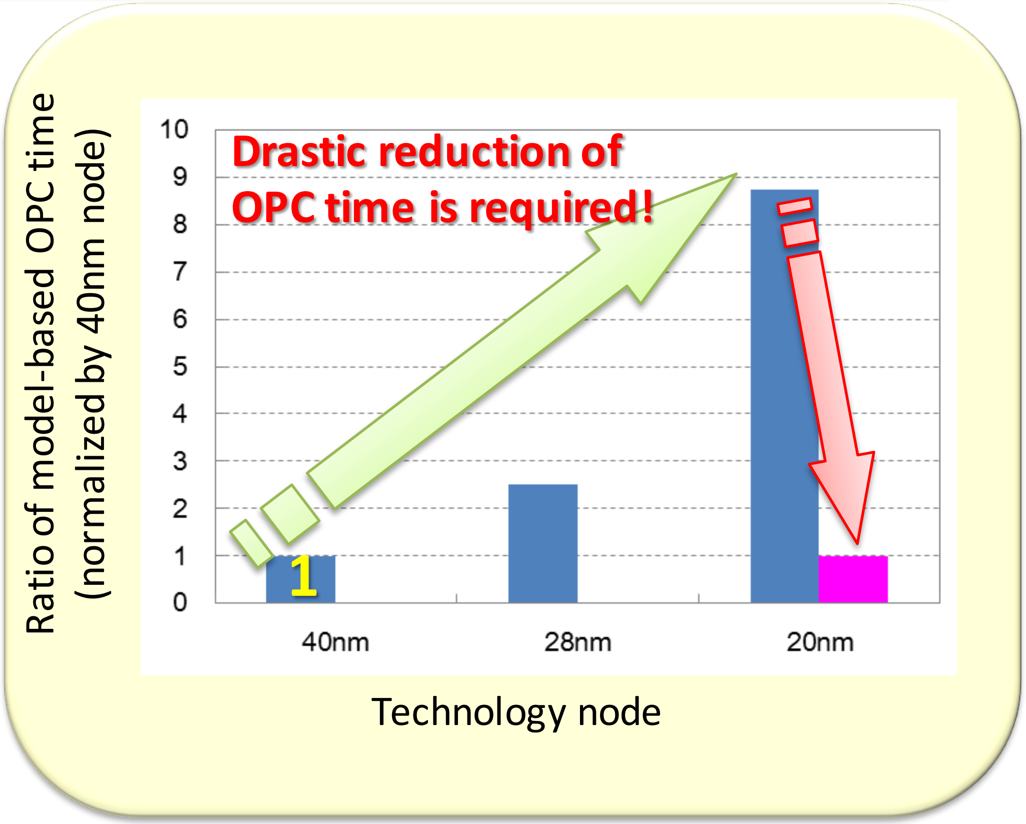
Outline

- Background
- OPC with Linear Regression
- Hierarchical Bayes Model (HBM)
- Markov Chain Monte Carlo (MCMC)
- Experimental results
- Conclusion

Optical Proximity Correction

- **Issue: Conventional OPC consumes very long time**
- **Goal: High accurate correction in short runtime**

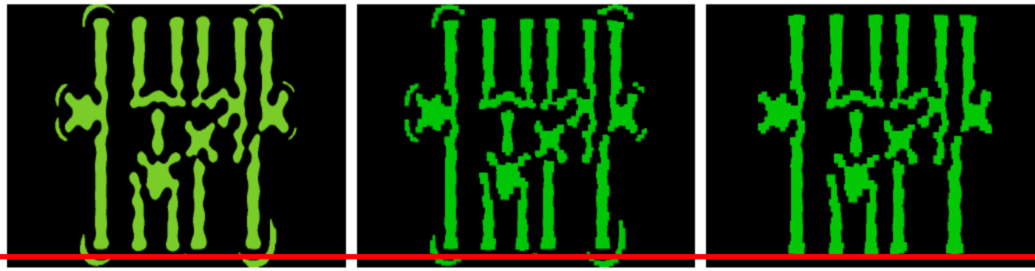




Model-based OPC is the most widely used technique

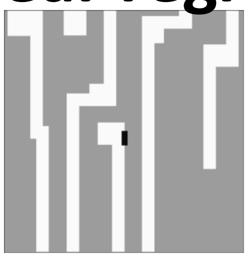
Related works

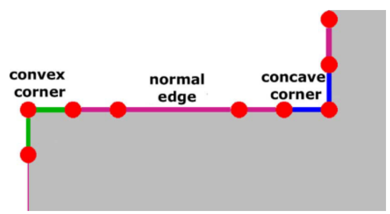
- **Inverse lithography**



D. S. Abrams, et al., SPIE, 2006

- **Linear regression**

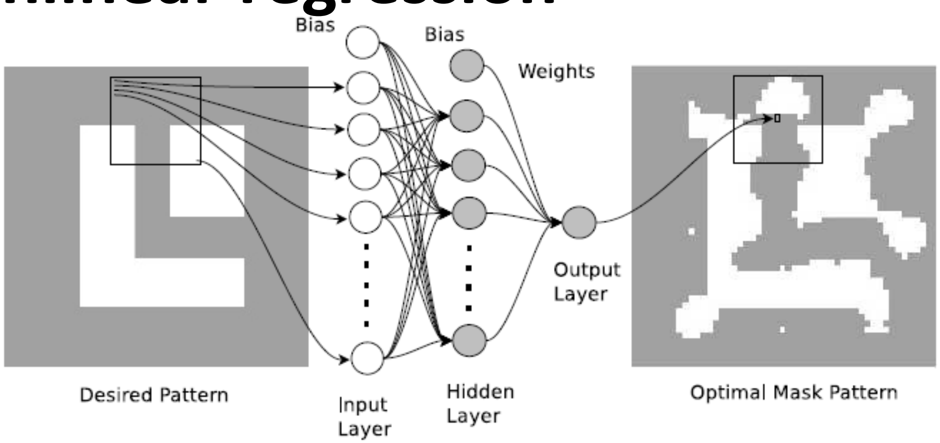




$$y = \boldsymbol{\beta}\mathbf{x} + \epsilon$$

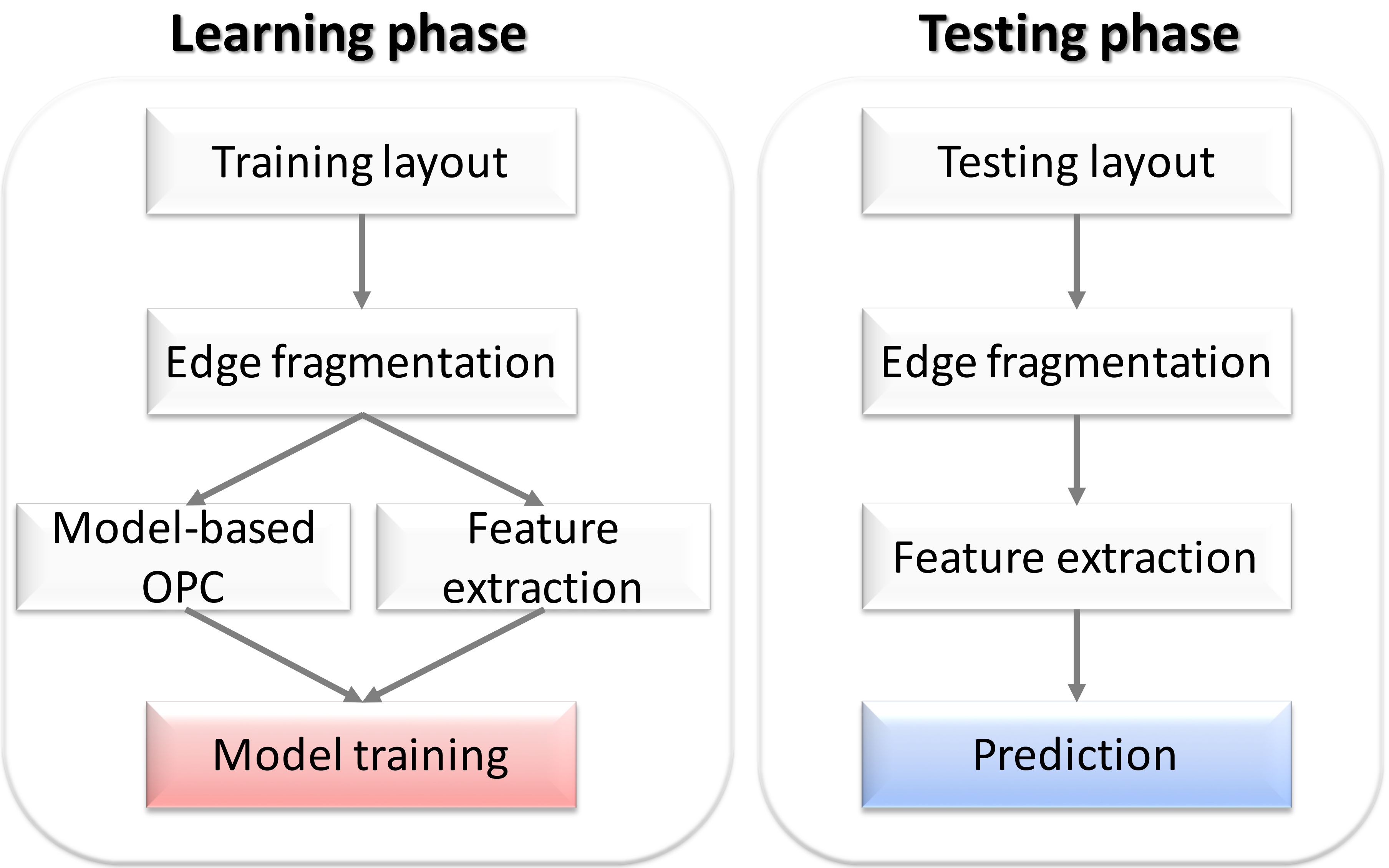
Allan Gu, et al., IEEE, 2008

- **Nonlinear regression**



Rui Luo, Journal of Optics, 2013

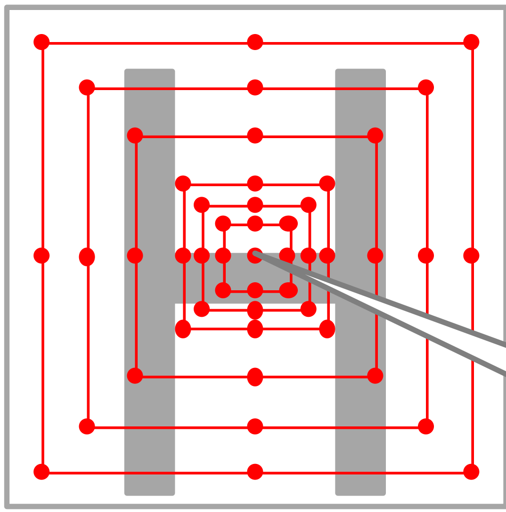
Regression-based OPC



OPC with Linear Regression

Layout feature

Concentric square sampling (CSS)



$$\mathbf{x} = \left(0, 1, 0, 0, 1, \ldots, v_j\right)^{\mathrm{T}}$$

V_j : pixel value of j-th dimension

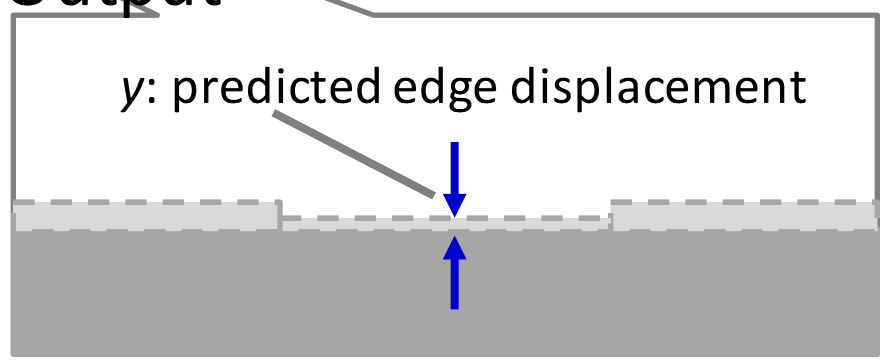
Linear model

$$y(\mathbf{x}, \mathbf{w}) = w_0 + \sum_{j=1}^{D} w_j x_j$$

Least squares solution

$$\mathbf{w} = \left(\mathbf{X}^{\mathrm{T}}\mathbf{X}\right)^{-1}\mathbf{X}^{\mathrm{T}}\mathbf{y}$$

Output



Pros and Cons

- Fast runtime
- Reasonable prediction accuracy
- Easy to understand

- Over-fitting, especially even with small data
- Required a large amount of training data
- Cannot model complex phenomena

OPC with Hierarchical Bayes Model (HBM)

- Fast runtime
- Reasonable prediction accuracy
- Easy to understand

- Prevent over-fitting
- Model complicated phenomena with small amount of data

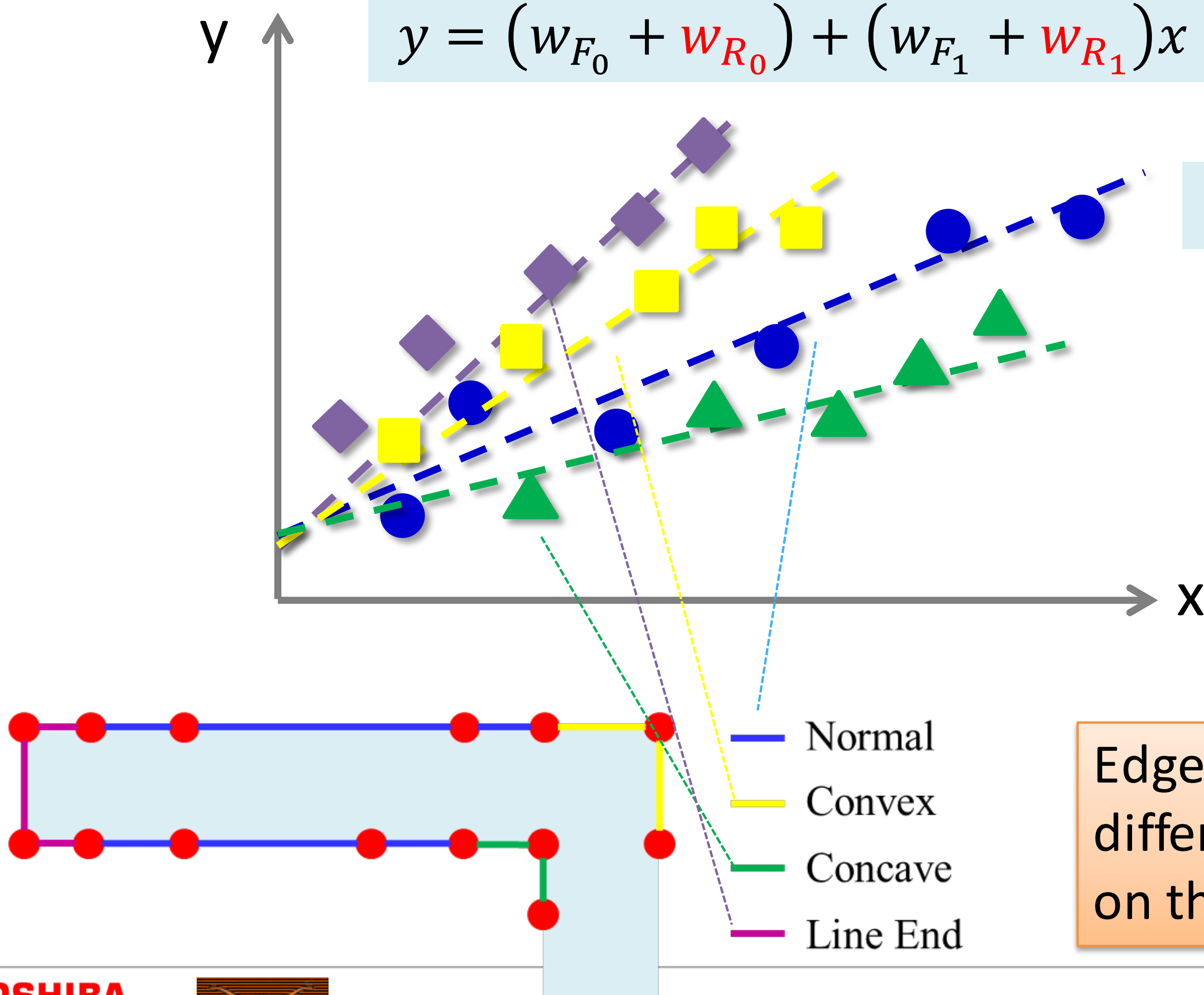
Outline

- Background
- OPC with Linear Regression
- **Hierarchical Bayes Model (HBM)**
- Markov Chain Monte Carlo (MCMC)
- Experimental results
- Conclusion

Regression example

$$y = (w_{F_0} + w_{R_0}) + (w_{F_1} + w_{R_1})x$$

$$y = w_0 + w_1 x$$



Edge displacement differs depending on the edge type

GLM vs GLMM

Generalized Linear Model

Generalized Linear Mixed Model

#dimensions

$$y(\mathbf{x}, \mathbf{w}) = \sum_{j=0}^{D} w_j x_j$$

$$y(\mathbf{x}, \mathbf{w}_F, \mathbf{w}_R) = \sum_{j=0}^{D} \left(w_{F_j} + w_{R_j} \right) x_j$$

Parameter

Feature vector

Fixed effects

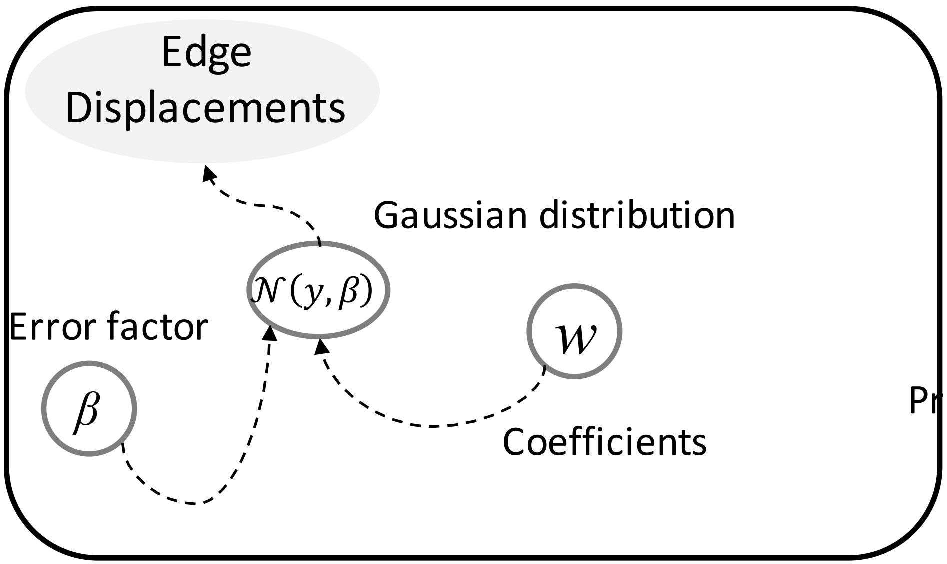
Random effects

Solution: Maximum likelihood estimation

Concept of Hierarchical Bayes Model (HBM)

Generalized Linear Model

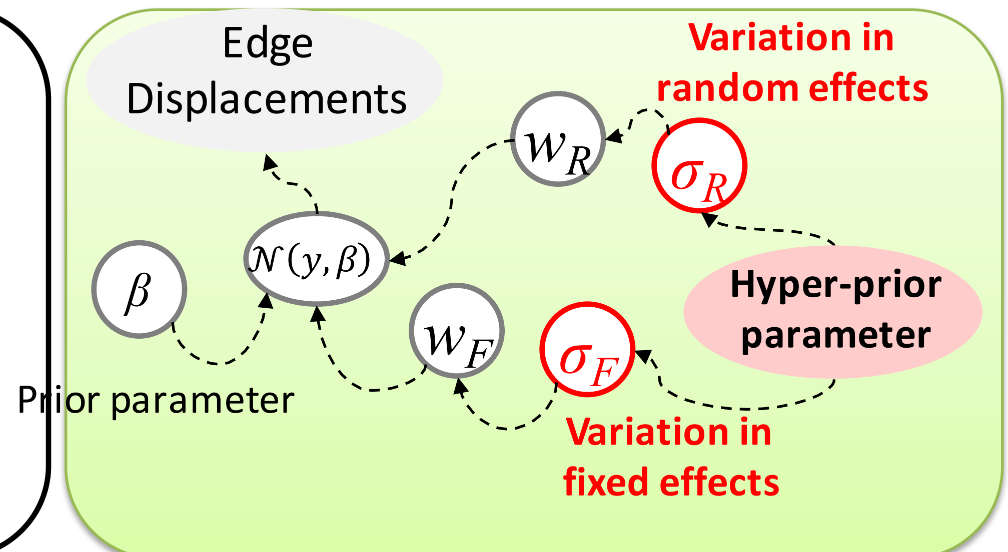
$$y(\mathbf{x}, \mathbf{w}) = \sum_{j=1}^{D} w_j x_j$$



Solution: Maximum likelihood estimation

Introduction of hidden variables

$$y(\mathbf{x}, \mathbf{w}_F, \mathbf{w}_R) = \sum_{j=1}^{D} \left(w_{F_j} + w_{R_j} \right) x_j$$



Solution: MCMC

Hierarchical Bayes Model (HBM)

Model

$$p(\mathbf{y}|\mathbf{x},\boldsymbol{\theta}) = \mathcal{N}\left(y(\mathbf{x},\boldsymbol{\theta}),\sigma_y\right)$$

Prior distributions

$$w_{f_j} \sim \mathcal{N}\left(0,\sigma_f\right)$$

$$w_{r_j} \sim \mathcal{N}\left(0,\sigma_r\right)$$

$$\sigma_y \sim \mathcal{U}(0,10^4)$$

Hyper-prior distributions

$$\sigma_f \sim \mathcal{U}(0,10^4)$$

$$\sigma_r \sim \mathcal{U}(0,10^4)$$

Non-informative hyper-prior

Posterior distribution

$$p(\boldsymbol{\theta}|\mathbf{y}) \propto \prod_{i=1}^{N}\prod_{j=0}^{D} p\left(y_i|\theta_j\right)p\left(w_{f_j}|\sigma_f\right)p\left(w_{r_j}|\sigma_r\right)p\left(\sigma_y\right)p\left(\sigma_f\right)p\left(\sigma_r\right)$$

Hierarchical Bayes

Bayes' theorem

$$p(\boldsymbol{\theta}|\mathbf{y}) \propto p(y|\theta)\ p(\theta)$$

Posterior Likelihood **Prior**

Posterior distribution for OPC model

$$p(\boldsymbol{\theta}|\mathbf{y}) \propto \prod_{i=1}^{N}\prod_{j=0}^{D} p(y_i|\theta_j)\, p\left(w_{f_j}|\sigma_f\right) p\left(w_{r_j}|\sigma_r\right) p(\sigma_y)\, p(\sigma_f)\, p(\sigma_r)$$

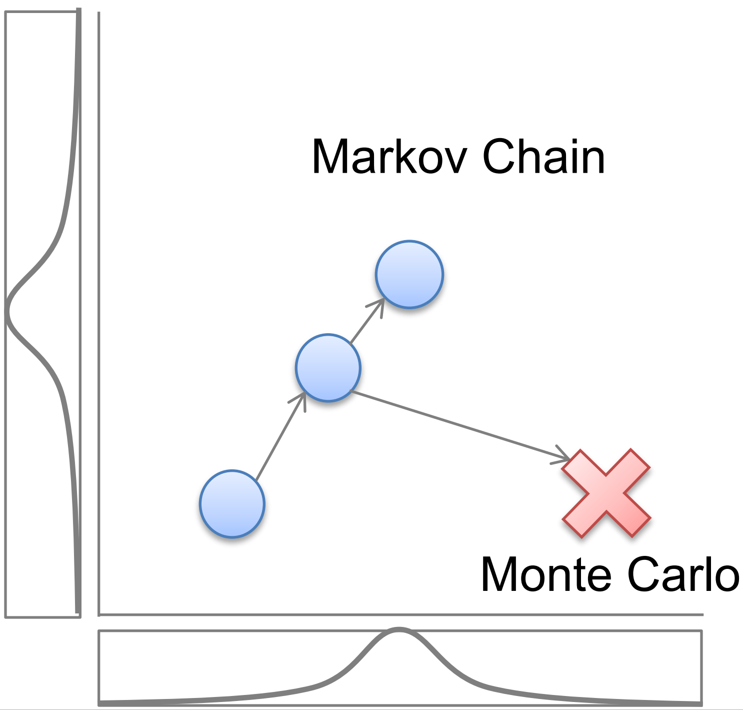
Likelihood **Priors** **Hyper-priors**

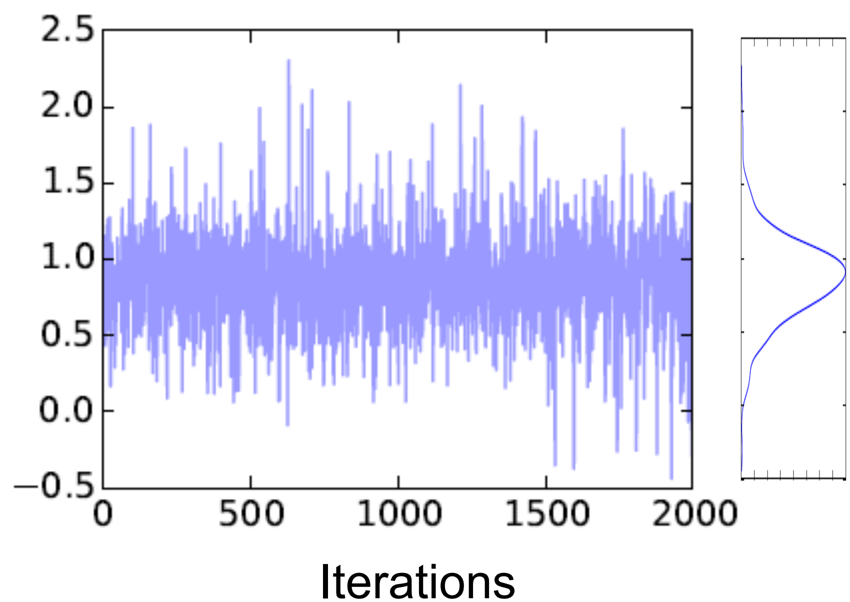
Outline

- Background
- OPC with Linear Regression
- Hierarchical Bayes Model (HBM)
- **Markov Chain Monte Carlo (MCMC)**
- Experimental results
- Conclusion

Markov Chain Monte Carlo (MCMC)

- **Markov Chain**
  - Next state is defined based on the previous state
- **Monte Carlo**
  - Stochastic algorithm



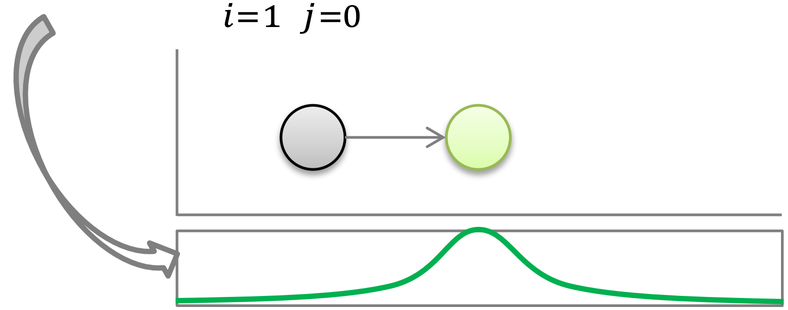


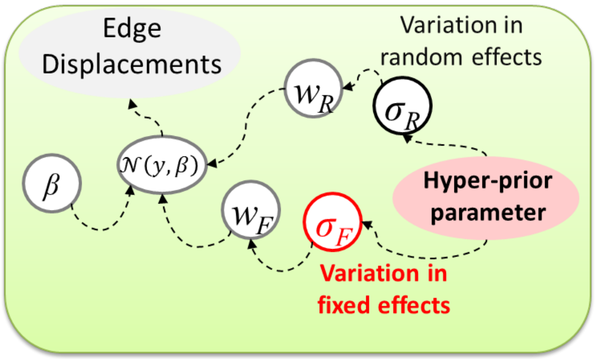
Markov Chain Monte Carlo (MCMC)

- **All parameters are fixed except sampling parameter**

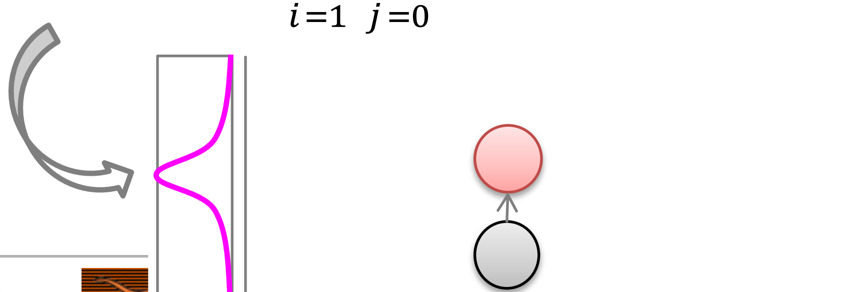
$$p(\boldsymbol{\theta}|\mathbf{y}) \propto \prod_{i=1}^{N}\prod_{j=0}^{D} p(y_i|\theta_j)\,p\left(w_{F_j}|\sigma_F\right)p\left(w_{R_j}|\sigma_R\right)p(\sigma_y)\,p(\sigma_F)\,p(\sigma_R)$$

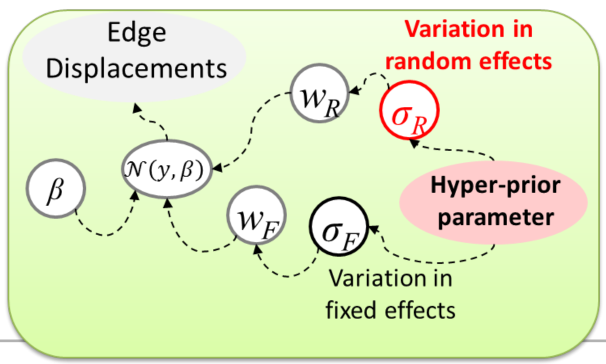
$$p(\sigma_F|\cdots) \propto \prod_{i=1}^{N}\prod_{j=0}^{D} p(y_i|\theta_j)\,p\left(w_{F_j}|\sigma_F\right)p(\sigma_F)$$





$$p(\sigma_R|\cdots) \propto \prod_{i=1}^{N}\prod_{j=0}^{D} p(y_i|\theta_j)\,p\left(w_{R_j}|\sigma_R\right)p(\sigma_R)$$





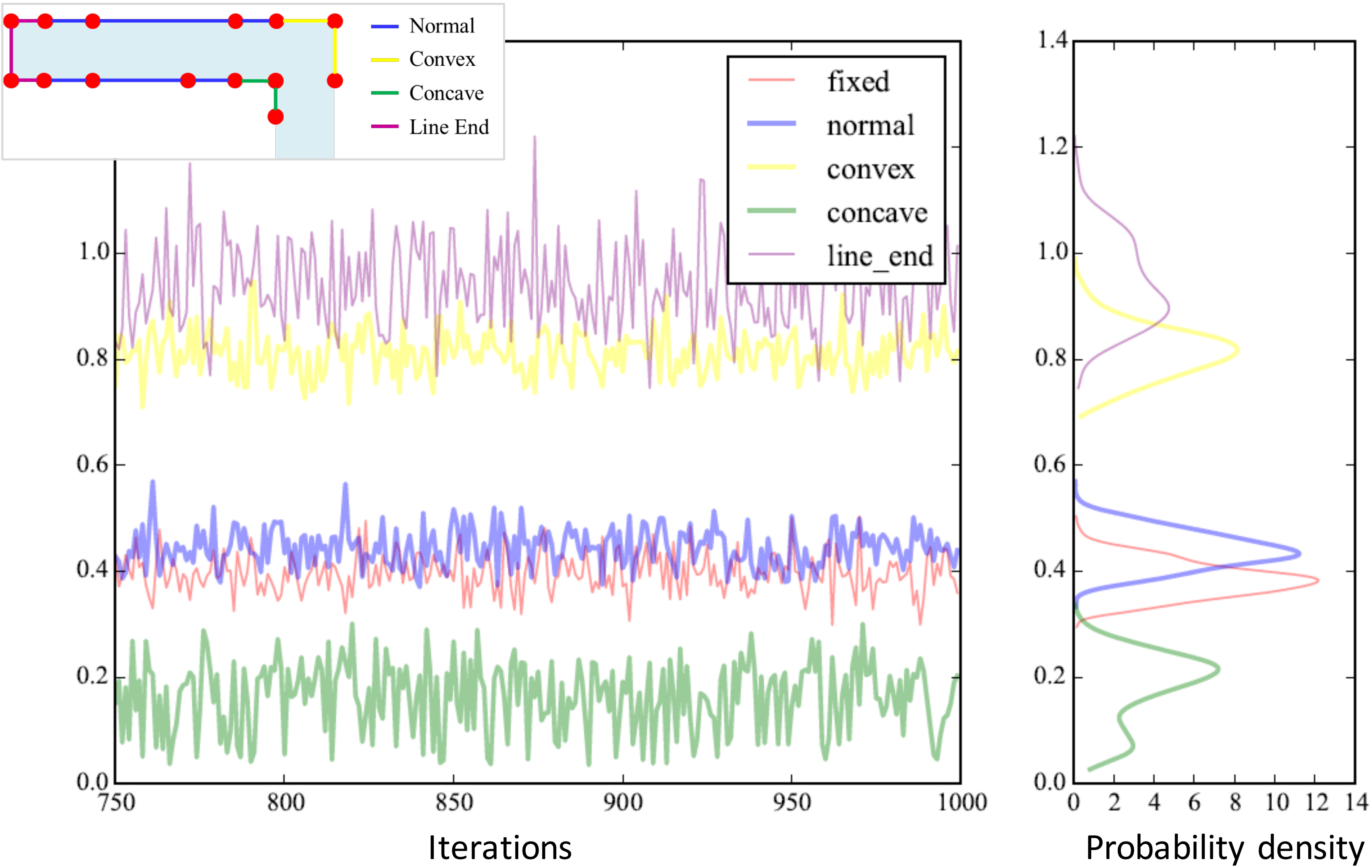
Outline

- Background
- OPC with Linear Regression
- Hierarchical Bayes Model (HBM)
- Markov Chain Monte Carlo (MCMC)
- **Experimental results**
- Conclusion

Experiments

- **Model training**
  - Motif : Layout A (training layout)
- **Comparison with linear/nonlinear regression**
  - Motif : Layout A (training), Layout B (testing)
  - Algorithms: linear regression (LR), support vector regression (SVR)
- **Comparison with model-based OPC**
  - Motif : Layout B (testing)
  - Optical model : 32nm node (λ=193nm, NA=1.35)
  - Model-based OPC : Calibre nmOPC

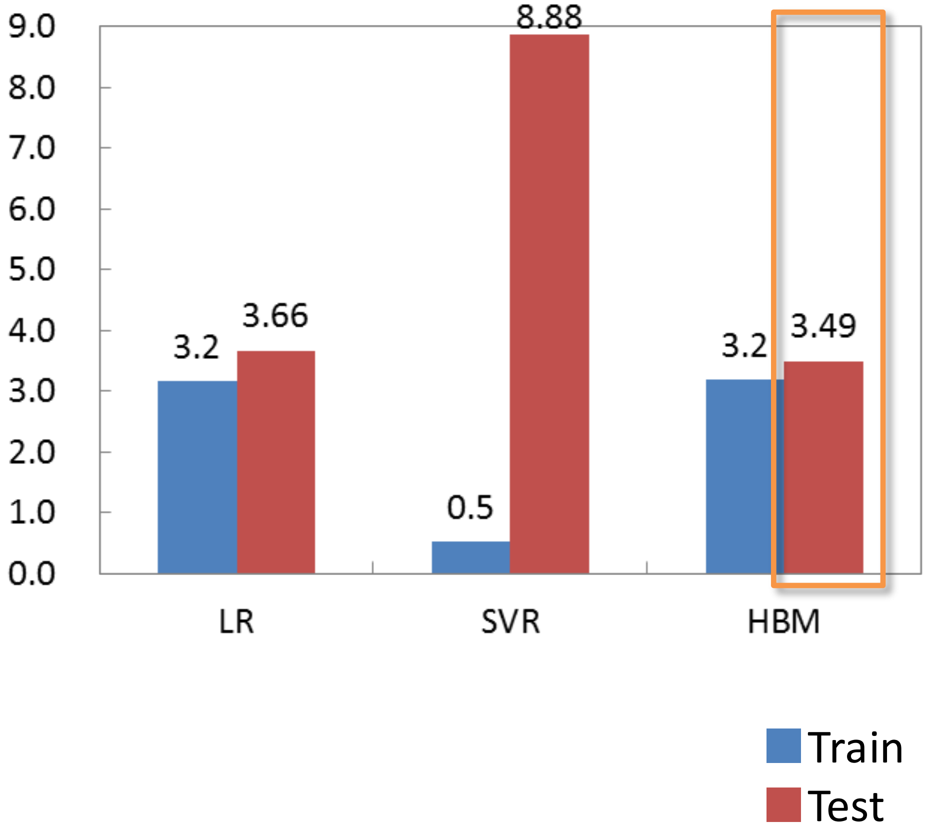
Sampling results of hidden parameters



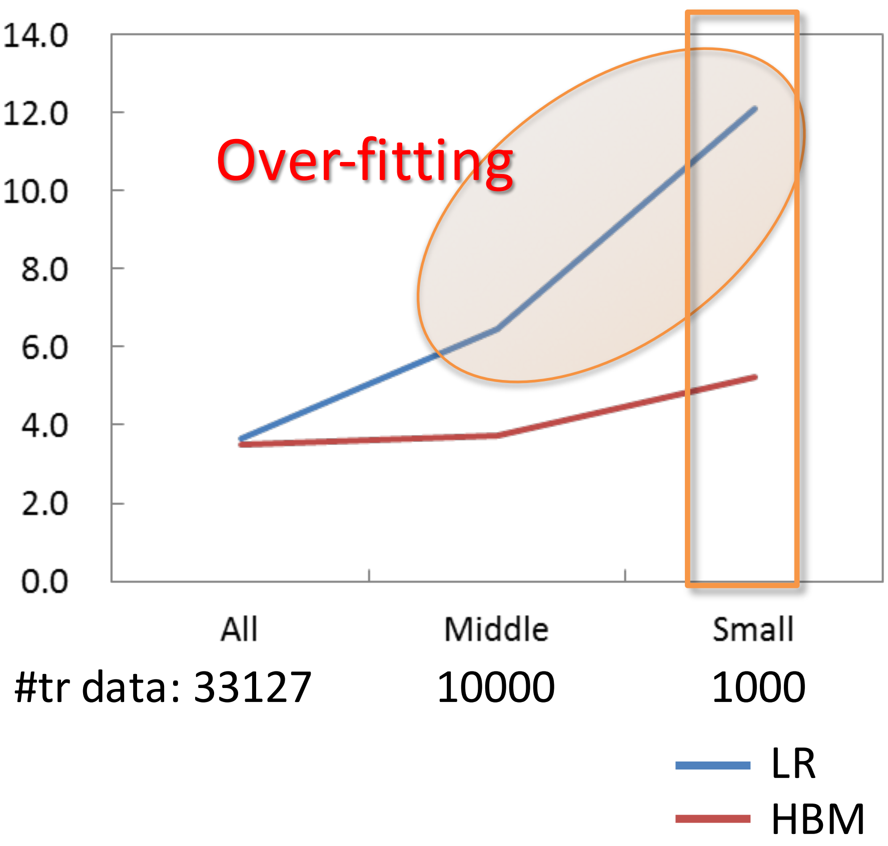
Clear difference among edge types

Comparison with linear/nonlinear regression

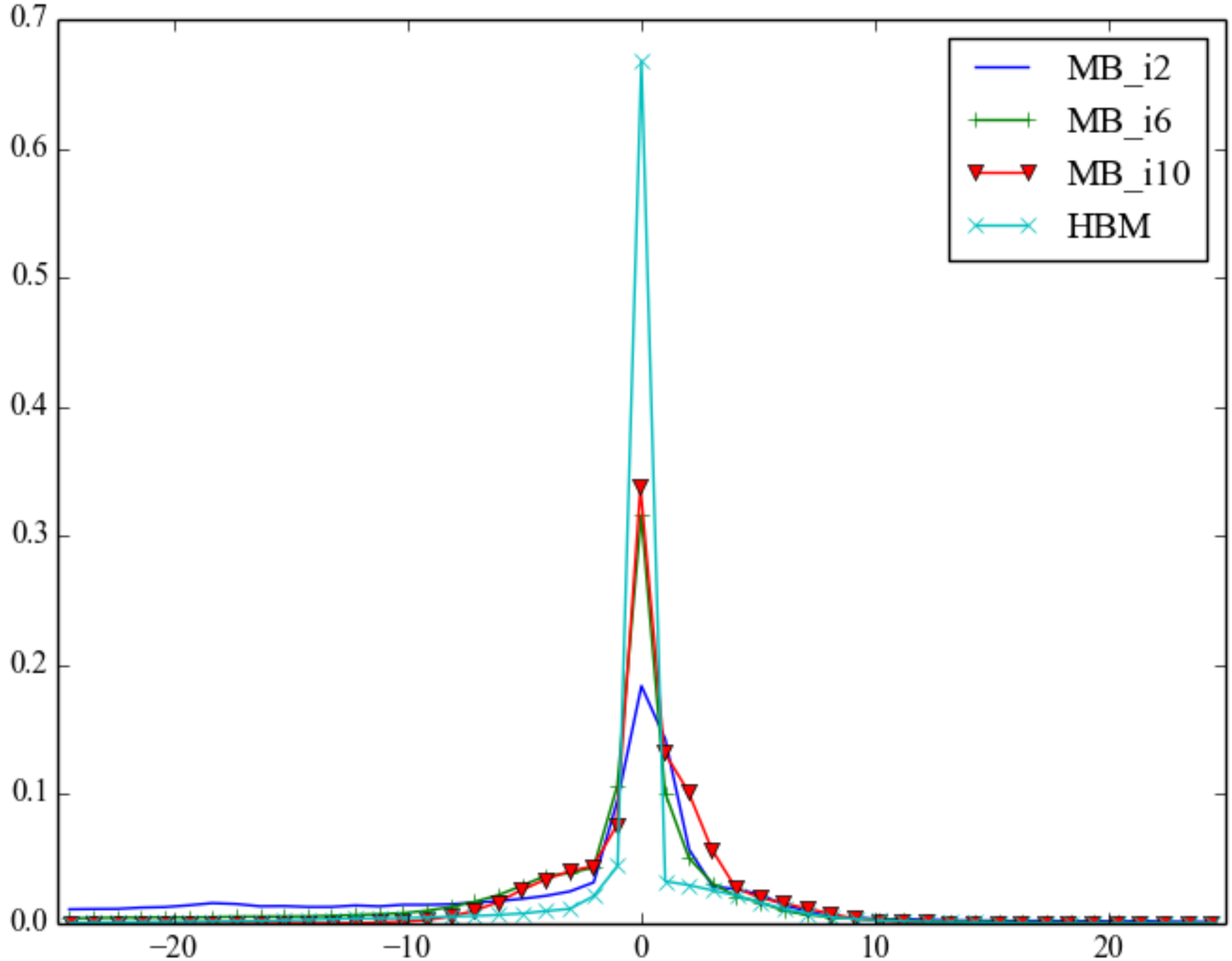
RMSE (nm)
(Root mean square error)



Prediction error
(RMSE @ Test layout)

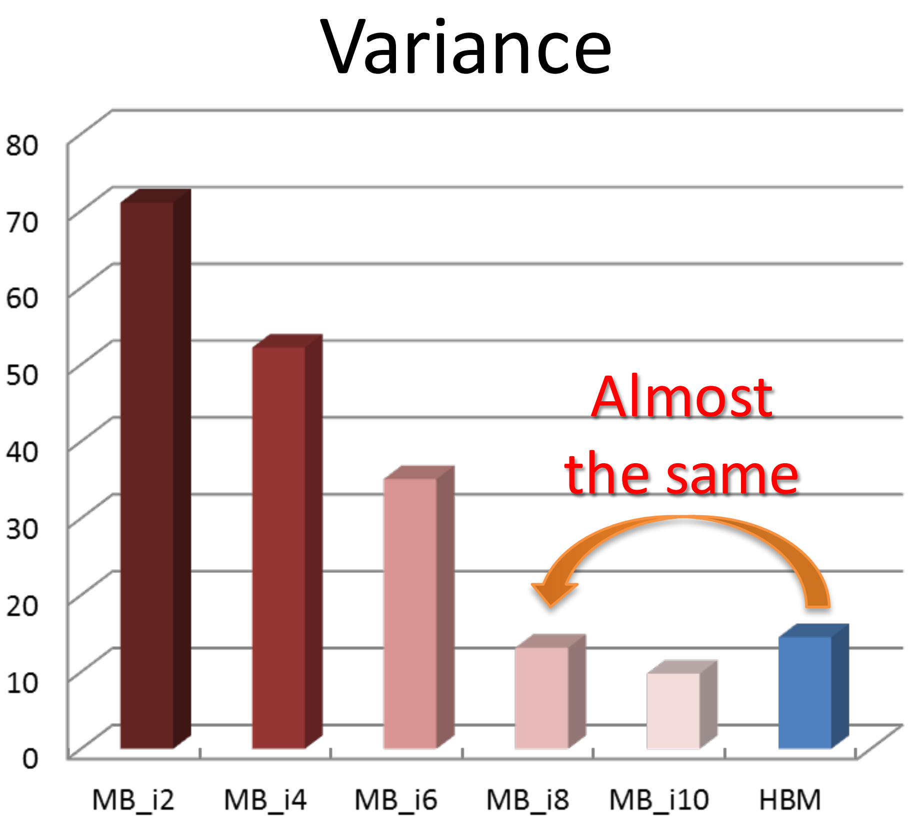


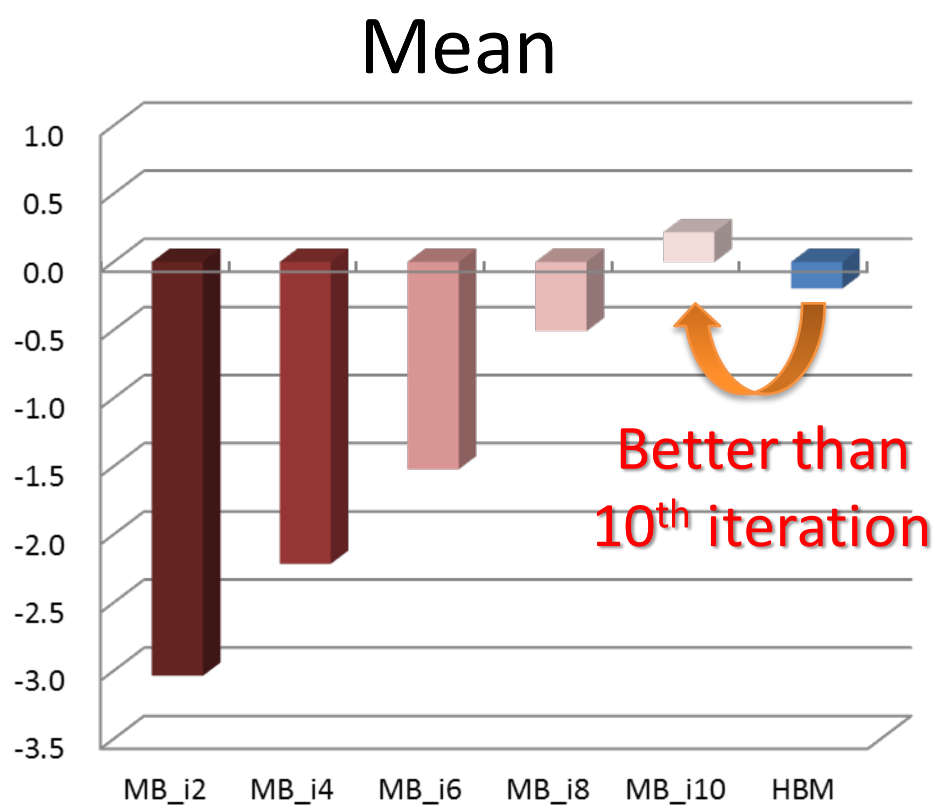
Comparison with model-based OPC (1/2)



Edge Placement Error (nm)

Comparison with model-based OPC (2/2)





The number of iterations of model-based OPC can be reduced 2.

Conclusion

- **Toshiba and UTDA developed a new Regression-based OPC using hierarchical Bayes model (HBM).**

- **The experimental results showed:**
  - High accurate prediction model can be achieved with small amount of data while preventing over-fitting.
  - The number of iterations in model-based OPC can be reduced 2.