Low-rank Kernel Matrix Approximation using Skeletonized Interpolation with Endo- or Exo-Vertices

Zixi Xu, Léopold Cambier, Eric Darve (Stanford University)
Pierre L'Eplattenier, François-Henry Rouet, Yun Huang,
and Cleve Ashcraft (Livermore Software Technology Corporation)

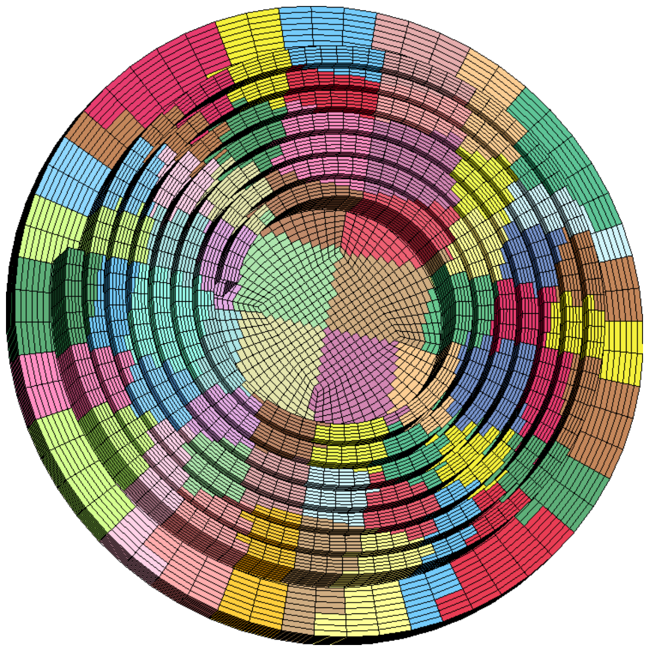
Integral equations & linear systems

$$a(x)u(x) + \int_X \boldsymbol{\mathcal{K}}(x,y)u(y)dy = f(x)$$

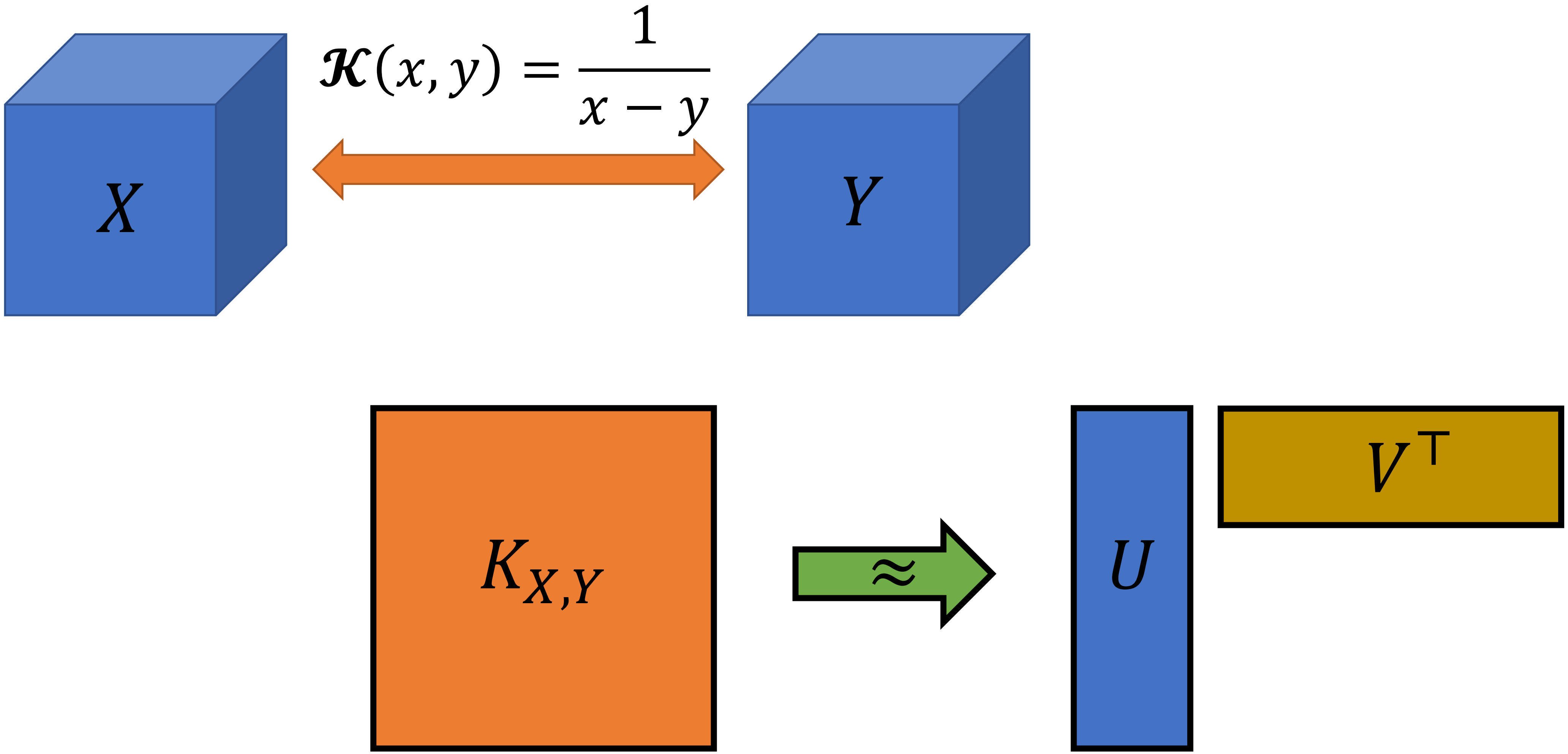
$$(aI)u + K_{n,n}u = f$$

$$K_{n,n} = \begin{bmatrix} K_{X_1,X_1} & \dots & K_{X_1,X_n} \\ \dots & \dots & \dots \\ K_{X_n,X_1} & \dots & K_{X_n,X_n} \end{bmatrix}$$

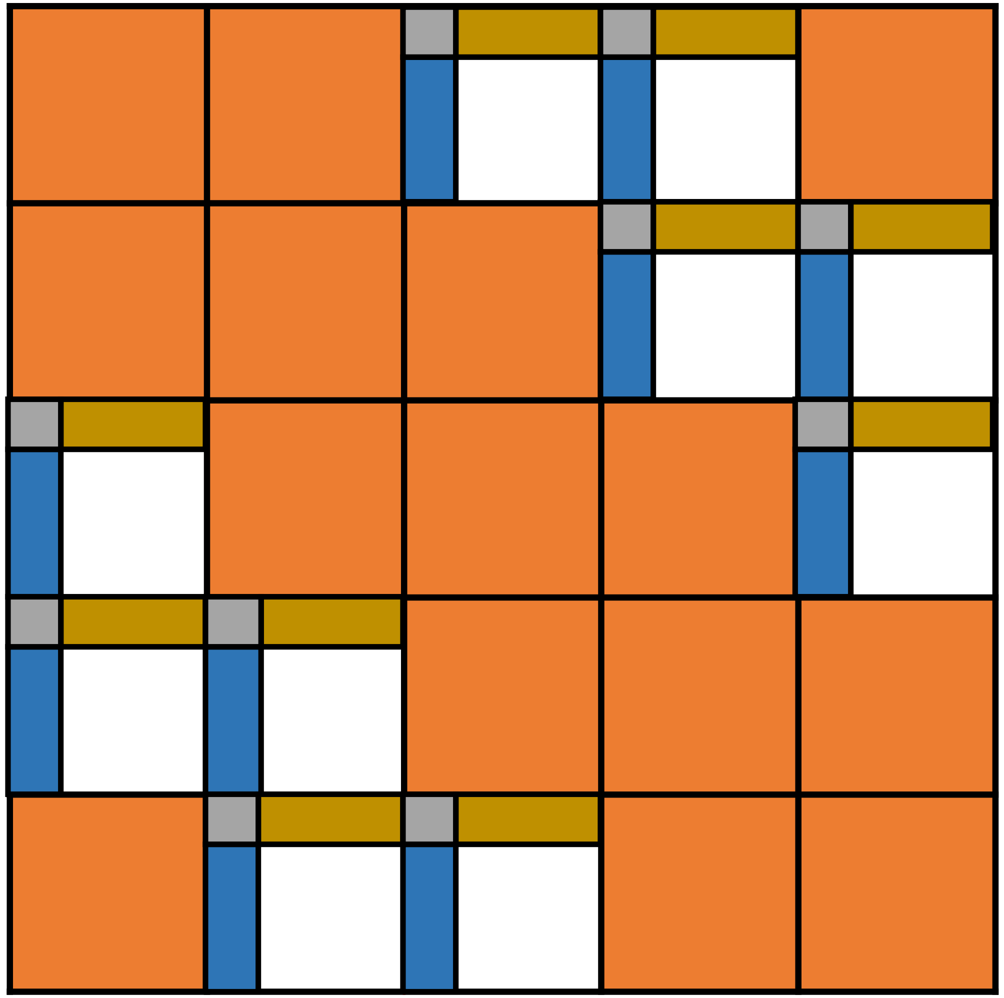
$$K_{ij} = \boldsymbol{\mathcal{K}}(x_i, y_j)$$



Low-rank off-diagonal blocks



Low-rank off-diagonal blocks



How to build the low-rank approximation

Rank-revealing LU factorization

$$P_{\hat{X}} K_{XY} P_{\hat{Y}} = P_{\hat{X}} \begin{bmatrix} K_{x_1 y_1} & \cdots & K_{x_1 y_n} \\ \vdots & \ddots & \vdots \\ K_{x_n y_1} & \cdots & K_{x_n y_n} \end{bmatrix} P_{\hat{Y}}$$

$$\approx L_{1:n,1:r} U_{1:r,1:n} = K_{X\hat{Y}} K_{\hat{X}\hat{Y}}^{-1} K_{\hat{X}Y}$$

How to build the low-rank approximation

Rank-revealing LU factorization with extended set

$$P_{\hat{X}} \begin{bmatrix} K_{X^\circ Y^\circ} & K_{X^\circ Y} \\ K_{X Y^\circ} & K_{XY} \end{bmatrix} P_{\hat{Y}} \approx K_{X\hat{Y}} K_{\hat{X}\hat{Y}}^{-1} K_{\hat{X}Y}$$

Skeletonized Interpolation

How to pick $\hat{X}$ and $\hat{Y}$?

<u>Algorithm</u>

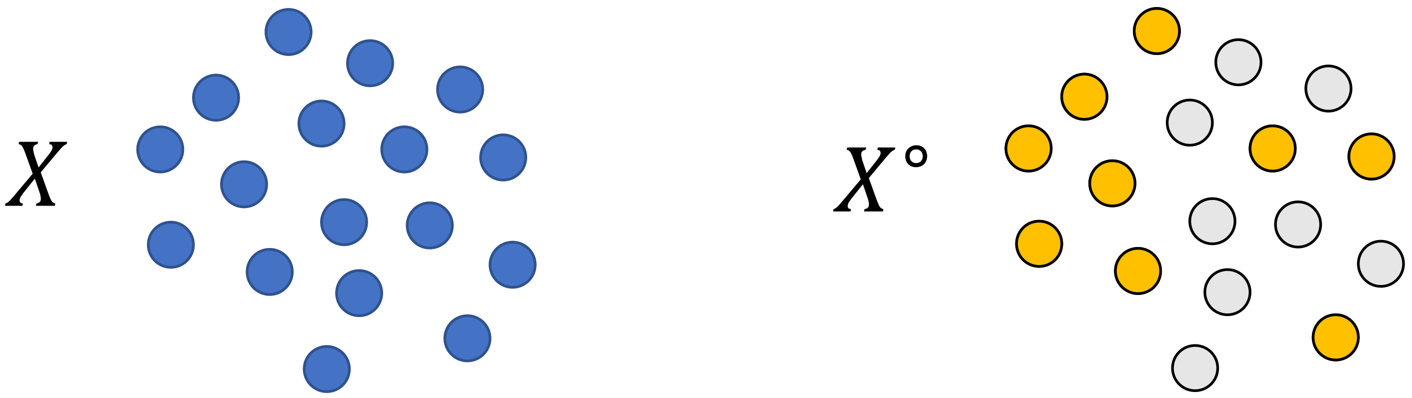
1. Generate candidates X° and Y°
2. Build $K^\circ = K_{X^\circ,Y^\circ}$
3. Select $\hat{X} \subset X^\circ, \hat{Y} \subset Y^\circ$ by performing RRQR over K° and $K^{\circ\top}$ up to tolerance ε
4. Return

$$K_{X,Y} \approx K_{X,\hat{Y}} K_{\hat{X},\hat{Y}}^{-1} K_{\hat{X},Y}$$

Different ways to define X° and Y°

Endo-vertices: subsets of X and Y

$$X^\circ = \text{random}(X)$$

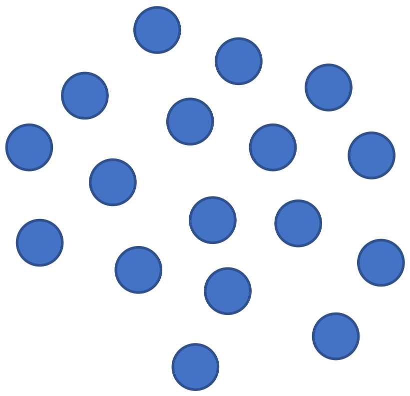


Different ways to define X° and Y°

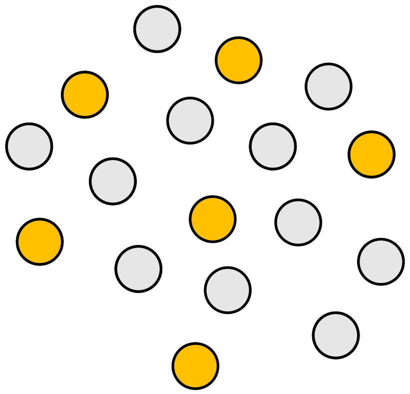
Endo-vertices: subsets of X and Y

$$X^\circ = \mathrm{MDV}(X)$$

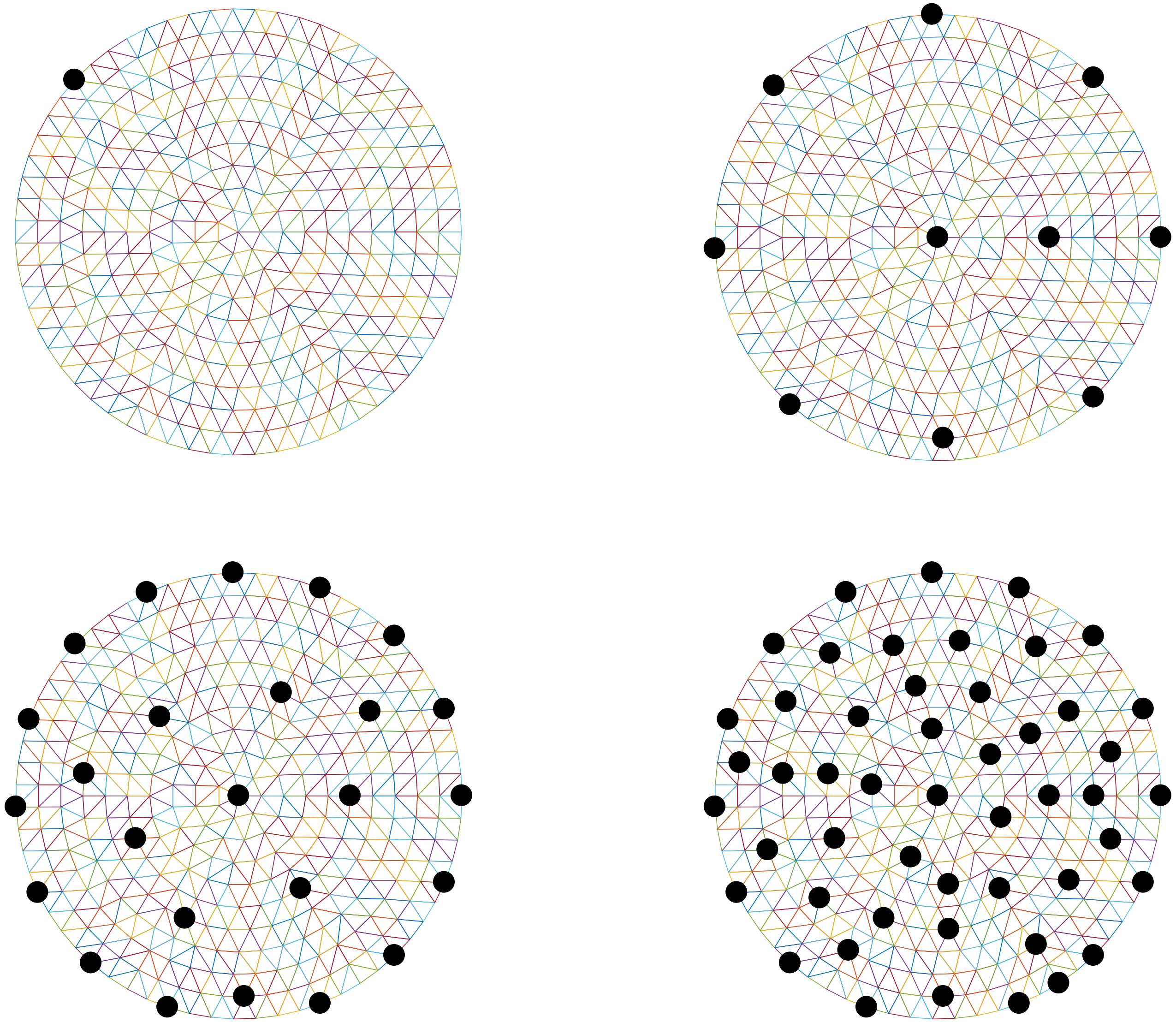
X



X°



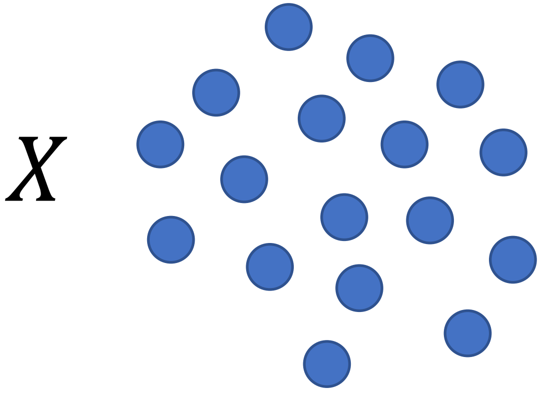
MDV

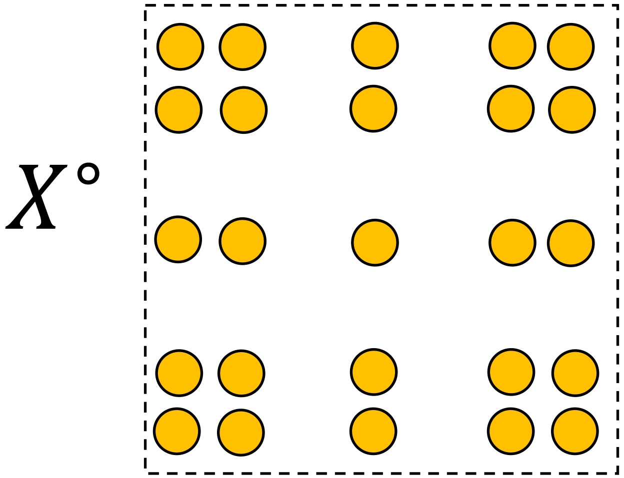


Different ways to define $X°$ and $Y°$

Exo-vertices: outside of X and Y

$$X° = \text{chebyshev}(X)$$

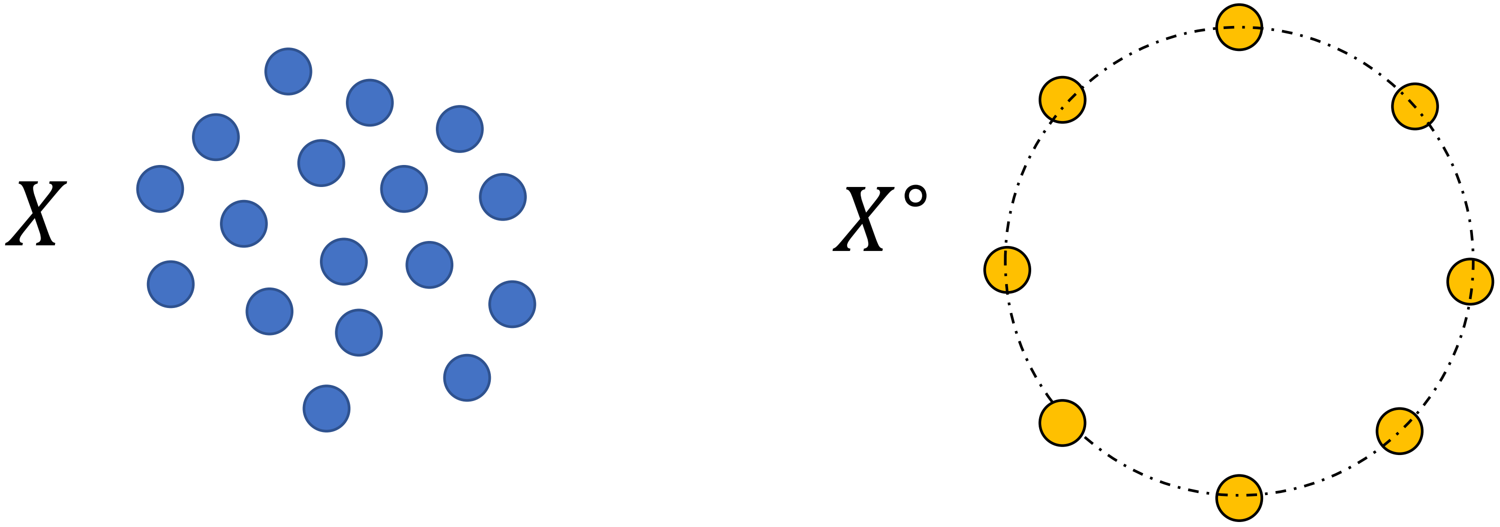




Different ways to define X° and Y°

Exo-vertices: outside of X and Y

$$X^\circ = \text{enclosing\_surface}(X)$$



Which one is best?

$$K^\circ = K_{X^\circ, Y^\circ}$$

$$\boldsymbol{r_0 \times r_0}$$

$$K_{X,Y} \approx K_{X,\hat{Y}} K_{\hat{X},\hat{Y}}^{-1} K_{\hat{X},Y}$$

$$\boldsymbol{r_1 \times r_1}$$

We want

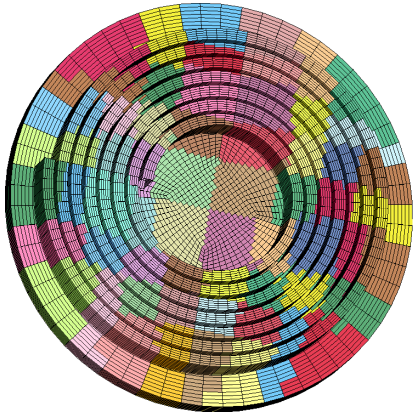
$\boldsymbol{r_0}$ as small as possible

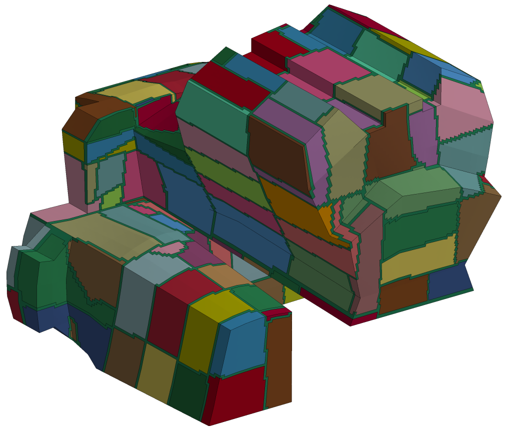
$\boldsymbol{r_1}$ close to optimal SVD-rank

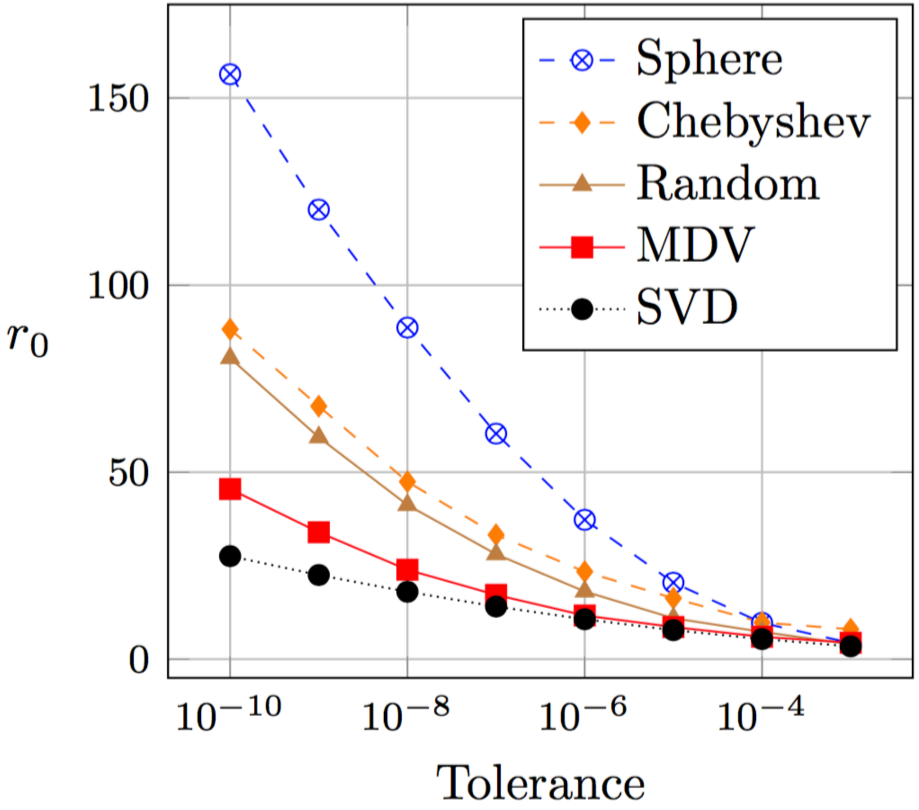
such that

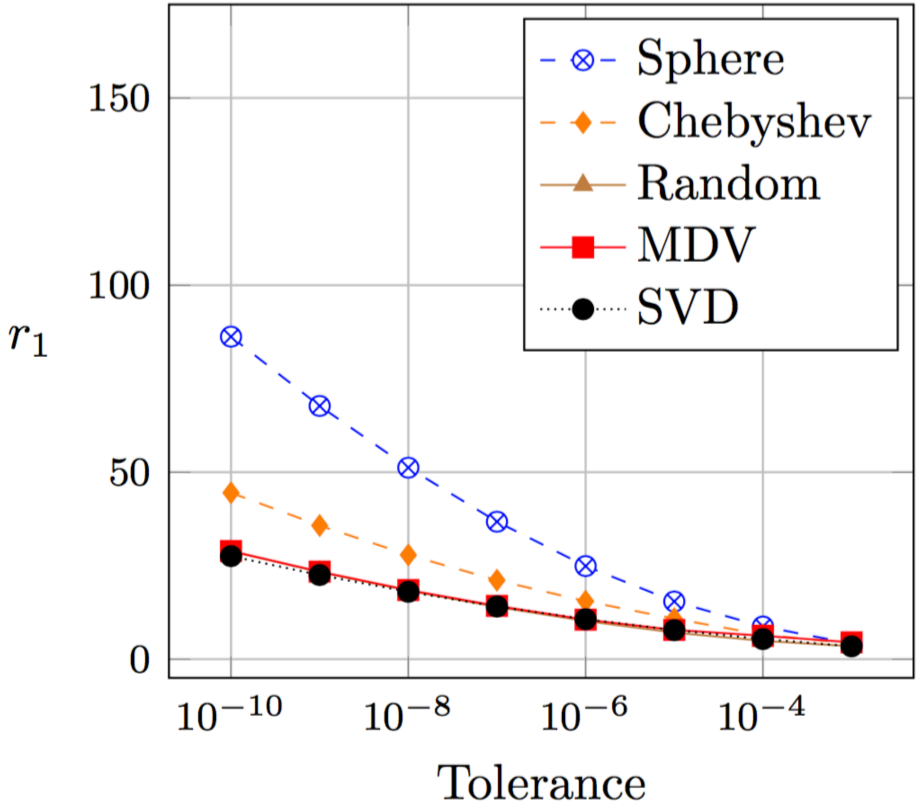
$$||K_{X,Y} - K_{X,\hat{Y}} K_{\hat{X},\hat{Y}}^{-1} K_{\hat{X},Y}|| \approx \epsilon$$

Overall, MDV is best: near-field

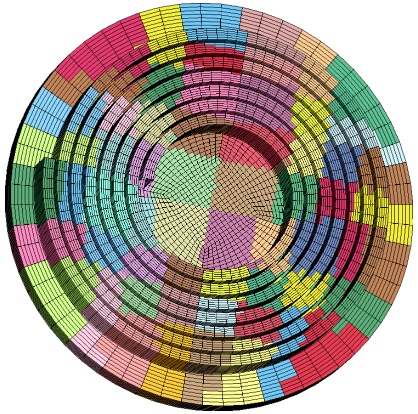


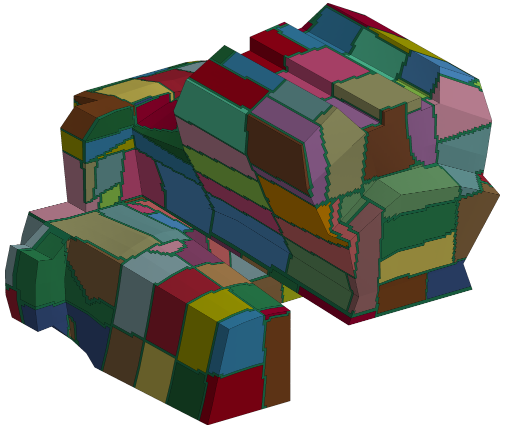


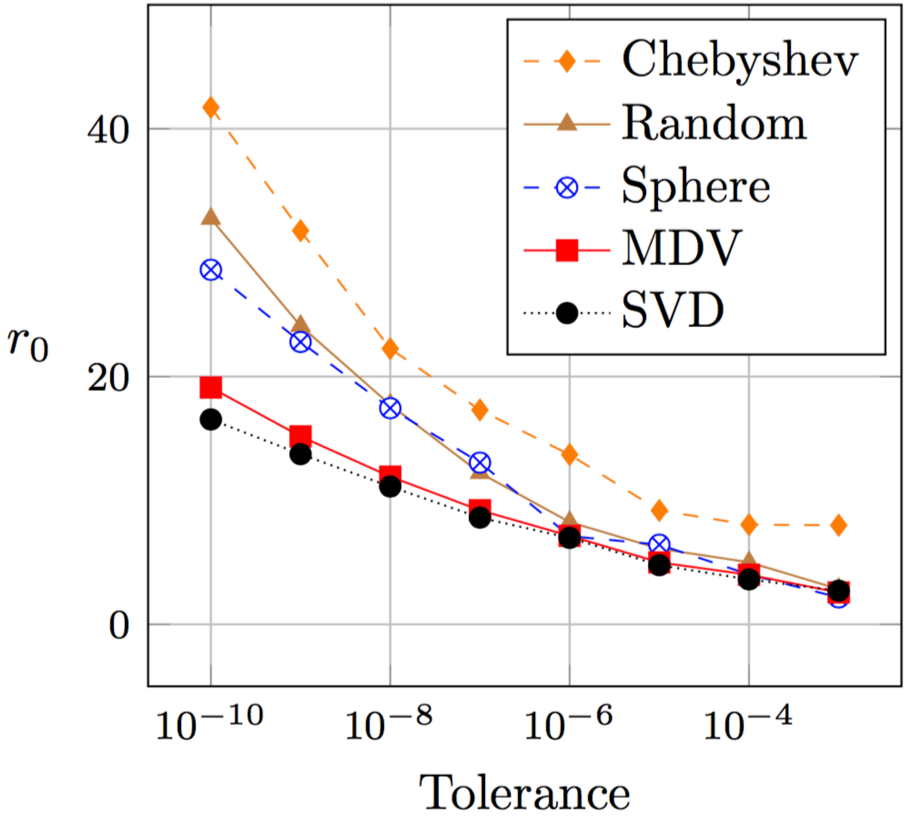


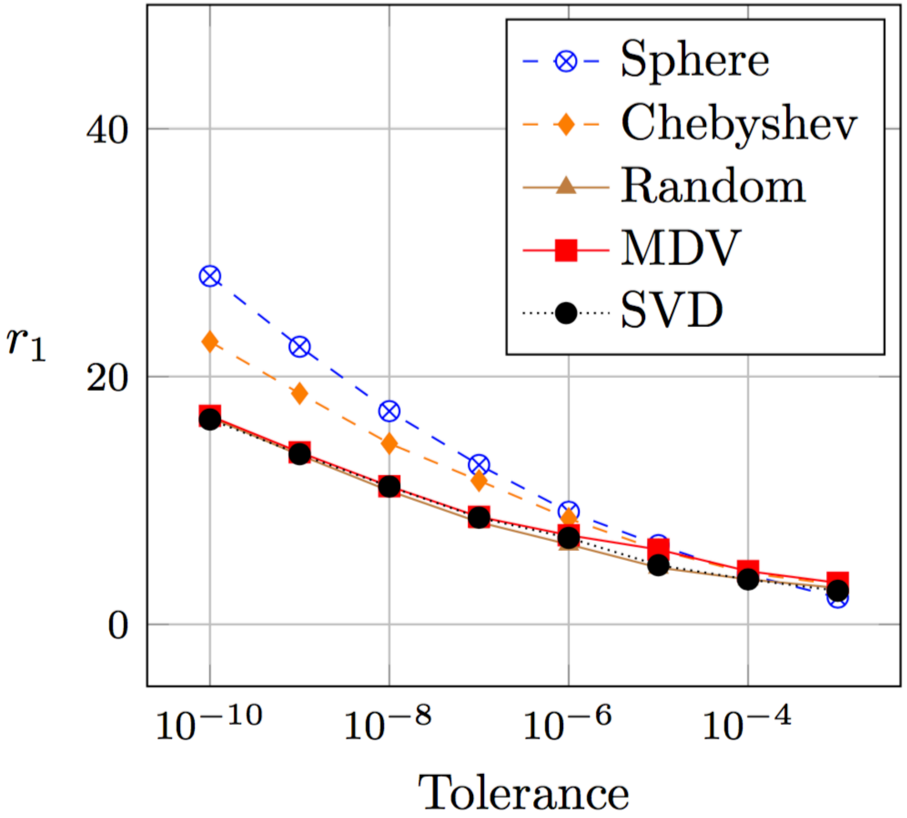


Overall, MDV is best: far-field

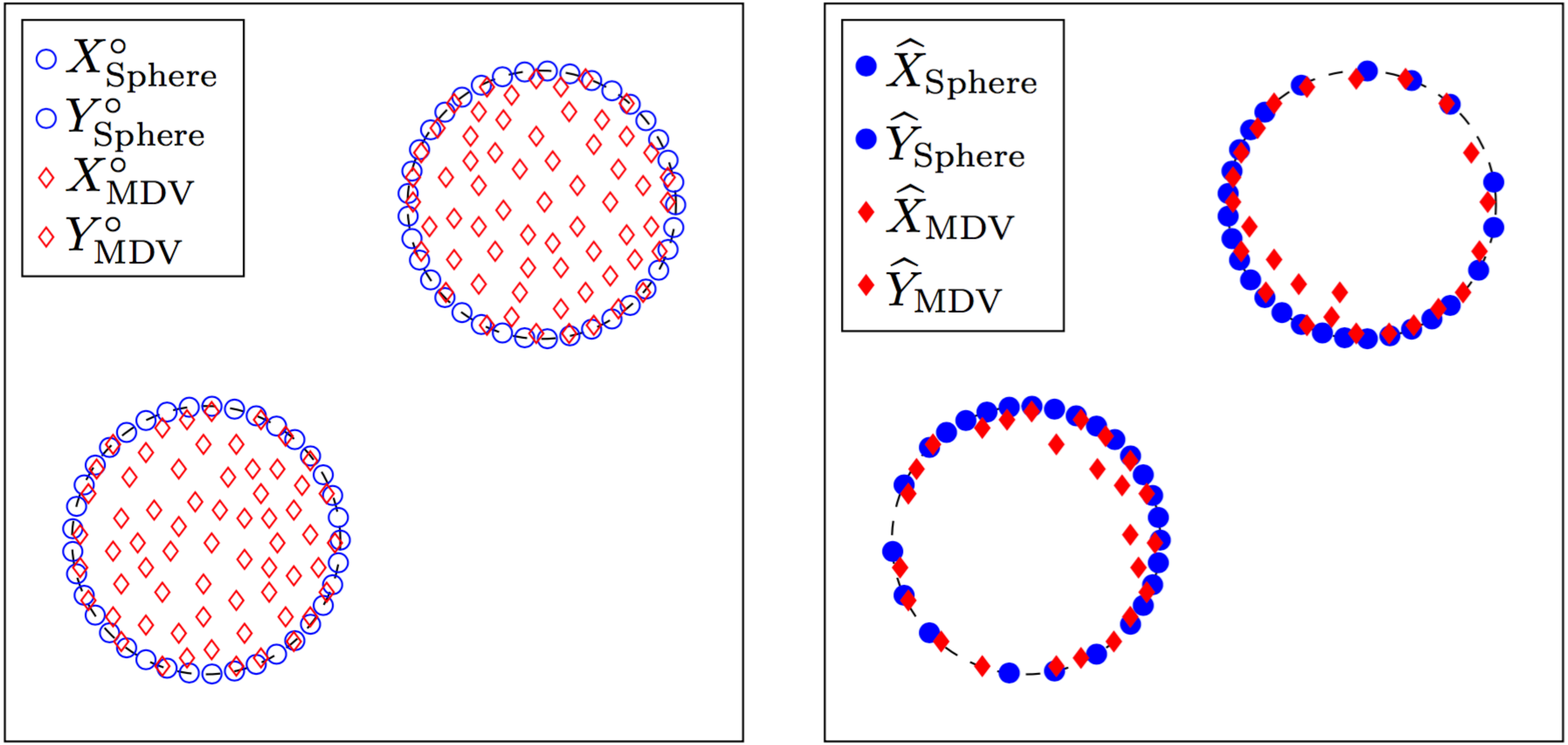




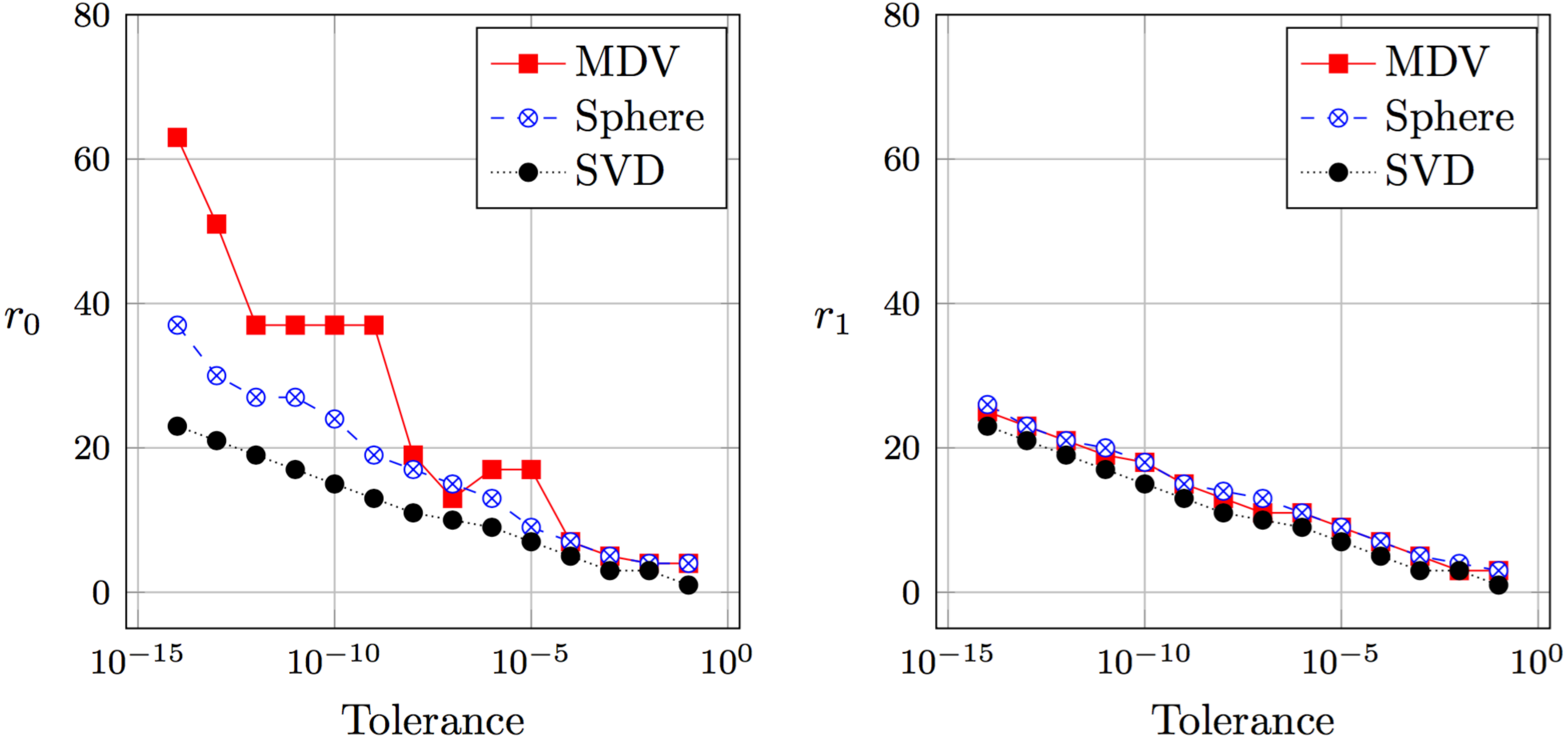




But not always: enclosing surface best



But not always: enclosing surface best



Conclusion

- Smooth kernel functions have low-rank kernel matrices for well-separated clusters
- Pre-selecting vertices using MDV leads to cheap & near-optimal factorization in most cases
- However, not always the best algorithm for all problems