The uniform sparse FFT

with application to PDEs with random coefficients

Fabian Taubert

joint work with Lutz Kämmerer and Daniel Potts

11.02.2022

TECHNISCHE UNIVERSITÄT
CHEMNITZ

Outline

We consider the PDE problem

$$\begin{aligned} -\nabla_{\boldsymbol{x}} \cdot (a(\boldsymbol{x}, \boldsymbol{y}) \nabla_{\boldsymbol{x}} u(\boldsymbol{x}, \boldsymbol{y})) &= f(\boldsymbol{x}), & \boldsymbol{x} &\in D,\, \boldsymbol{y} \in D_{\boldsymbol{y}} \\ u(\boldsymbol{x}, \boldsymbol{y}) &= 0, & \forall\, \boldsymbol{x} &\in \partial D,\, \boldsymbol{y} \in D_{\boldsymbol{y}}. \end{aligned}$$

- spatial variable $\boldsymbol{x} \in D \subset \mathbb{R}^{d_{\boldsymbol{x}}}$, typically with $d_{\boldsymbol{x}} = 1, 2, 3$
- random variable $\boldsymbol{y} = (y_j)_{j=1}^{d} \in D_{\boldsymbol{y}}$, typically very high-dimensional or even infinite-dimensional

- affine random coefficient [Cohen, DeVore, Schwab '10], [Dick, Kuo, Le Gia, Schwab '16], [Bachmayr, Cohen, Dahmen '18],[Gantner, Herrmann, Schwab '18], [Nguyen, Nuyens '21], ...

$$a(\boldsymbol{x},\boldsymbol{y}) = a_0(\boldsymbol{x}) + \sum_{j=1}^{d} y_j\, \psi_j(\boldsymbol{x})$$

- periodic random coefficient

[Kaarnioja, Kuo, Sloan '20], [Kaarnioja, Kazashi, Kuo, Nobile, Sloan '20]

$$a(\boldsymbol{x},\boldsymbol{y}) = a_0(\boldsymbol{x}) + \sum_{j=1}^{d} \Theta_j(\boldsymbol{y})\, \psi_j(\boldsymbol{x}),$$

with Θ_j periodic, e.g., $\Theta_j(\boldsymbol{y}) := \frac{1}{\sqrt{6}} \sin(2\pi y_j)$

- lognormal random coefficient [Graham, Kuo, Nichols, Scheichl, Schwab, Sloan '13], [Cheng, Hou, Yan, Zhang '13], [Bachmayr, Cohen, DeVore, Migliorati '17], [Nguyen, Nuyens '21], ...

$$a(\boldsymbol{x},\boldsymbol{y}) = a_0(\boldsymbol{x}) + \exp(b(\boldsymbol{x},\boldsymbol{y})), \qquad b(\boldsymbol{x},\boldsymbol{y}) = b_0(\boldsymbol{x}) + \sum_{j=1}^{d} y_j\, \psi_j(\boldsymbol{x})$$

- Common aim: Approximation of the solution $u(\boldsymbol{x}, \boldsymbol{y})$ or some quantity of interest, e.g., the expectation value $\mathbb{E}[F(\boldsymbol{y})]$ of some functional $F(\boldsymbol{y}) \coloneqq F[u(\cdot, \boldsymbol{y})]$, using samples of the function.
- Main problem: high-dimensional approximation!

- Common aim: Approximation of the solution $u(\boldsymbol{x}, \boldsymbol{y})$ or some quantity of interest, e.g., the expectation value $\mathbb{E}[F(\boldsymbol{y})]$ of some functional $F(\boldsymbol{y}) \coloneqq F[u(\cdot, \boldsymbol{y})]$, using samples of the function.
- Main problem: high-dimensional approximation!
- Examples of other approaches:
  - Quasi-Monte Carlo methods

    [Kuo, Schwab, Sloan '15], [Dick, Le Gia, Schwab '16], [Nguyen, Nuyens '21], ...
  - collocation methods

    [Cheng, Hou, Yan, Zhang '13], [Ernst, Sprungk '14], [Zhang, Hu, Hou, Lin, Yan '14], ...
  - methods based on certain (tensorized) functions (e.g., Legendre polynomials) [Cohen, DeVore, Schwab '10], [Bachmayr, Cohen, Migliorati '17], ...

- Common aim: Approximation of the solution $u(\boldsymbol{x}, \boldsymbol{y})$ or some quantity of interest, e.g., the expectation value $\mathbb{E}[F(\boldsymbol{y})]$ of some functional $F(\boldsymbol{y}) \coloneqq F[u(\cdot, \boldsymbol{y})]$, using samples of the function.
- Main problem: high-dimensional approximation!
- Examples of other approaches:
  - Quasi-Monte Carlo methods
    [Kuo, Schwab, Sloan '15], [Dick, Le Gia, Schwab '16], [Nguyen, Nuyens '21], ...
  - collocation methods
    [Cheng, Hou, Yan, Zhang '13], [Ernst, Sprungk '14], [Zhang, Hu, Hou, Lin, Yan '14], ...
  - methods based on certain (tensorized) functions (e.g., Legendre polynomials) [Cohen, DeVore, Schwab '10], [Bachmayr, Cohen, Migliorati '17], ...
- These approaches are often heavily influenced by the choice (or computation) of some weights, functions or kernels in advance!

- We aim for an approximation of $u(\boldsymbol{x}_0, \cdot)$ for fixed $\boldsymbol{x}_0 \in D$ using the sparse FFT (sFFT) based on rank-1 lattice sampling.

  [Potts, Volkmer '16], [Kämmerer, Krahmer, Volkmer '20], [Kämmerer, Potts, Volkmer '21]

- We aim for an approximation of $u(\boldsymbol{x}_0, \cdot)$ for fixed $\boldsymbol{x}_0 \in D$ using the sparse FFT (sFFT) based on rank-1 lattice sampling.

  [Potts, Volkmer '16], [Kämmerer, Krahmer, Volkmer '20], [Kämmerer, Potts, Volkmer '21]

- The goal of the sFFT is the computation of the index set $\mathrm{I} = \operatorname{supp} \hat{p} \subset \mathbb{Z}^d$ in the given search space $\Gamma \supset \mathrm{I}$ and the coefficients $\hat{p}_{\boldsymbol{k}}, \boldsymbol{k} \in \mathrm{I}$, of the multivariate trigonometric polynomial

$$p(\boldsymbol{y}) = \sum_{\boldsymbol{k} \in \mathrm{I}} \hat{p}_{\boldsymbol{k}} \mathrm{e}^{2\pi \mathrm{i} \boldsymbol{k} \cdot \boldsymbol{y}}$$

  from sampling values.

- We aim for an approximation of $u(\boldsymbol{x}_0, \cdot)$ for fixed $\boldsymbol{x}_0 \in D$ using the sparse FFT (sFFT) based on rank-1 lattice sampling.

  [Potts, Volkmer '16], [Kämmerer, Krahmer, Volkmer '20], [Kämmerer, Potts, Volkmer '21]

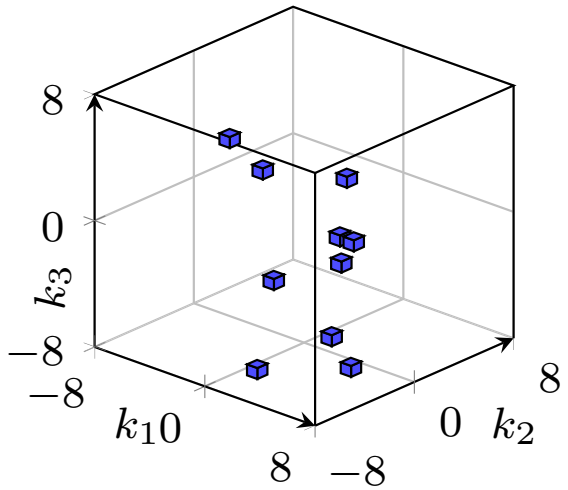
- The goal of the sFFT is the computation of the index set $\mathrm{I} = \operatorname{supp} \hat{p} \subset \mathbb{Z}^d$ in the given search space $\Gamma \supset \mathrm{I}$ and the coefficients $\hat{p}_{\boldsymbol{k}}, \boldsymbol{k} \in \mathrm{I}$, of the multivariate trigonometric polynomial

$$p(\boldsymbol{y}) = \sum_{\boldsymbol{k} \in \mathrm{I}} \hat{p}_{\boldsymbol{k}} \mathrm{e}^{2\pi \mathrm{i} \boldsymbol{k} \cdot \boldsymbol{y}}$$

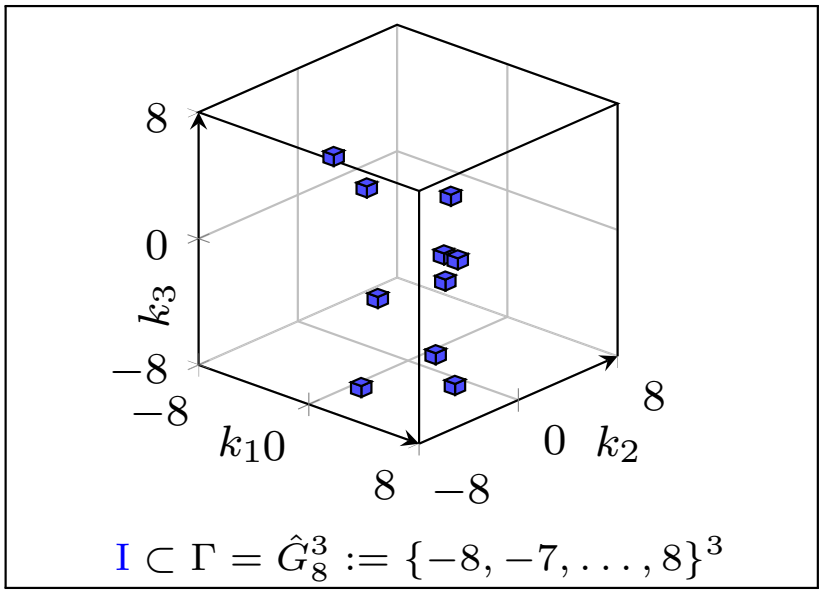
  from sampling values.

- Function approximation of $u(\boldsymbol{x}_0, \cdot)$ can be realized using the sFFT by assuming it to be a trigonometric polynomial p with some noise η, i.e.,

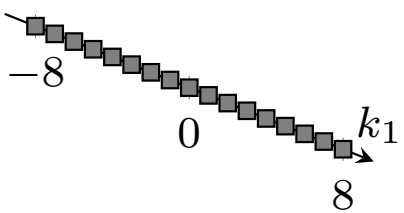
$$u(\boldsymbol{x}_0, \boldsymbol{y}) = p(\boldsymbol{y}) + \eta(\boldsymbol{y}).$$



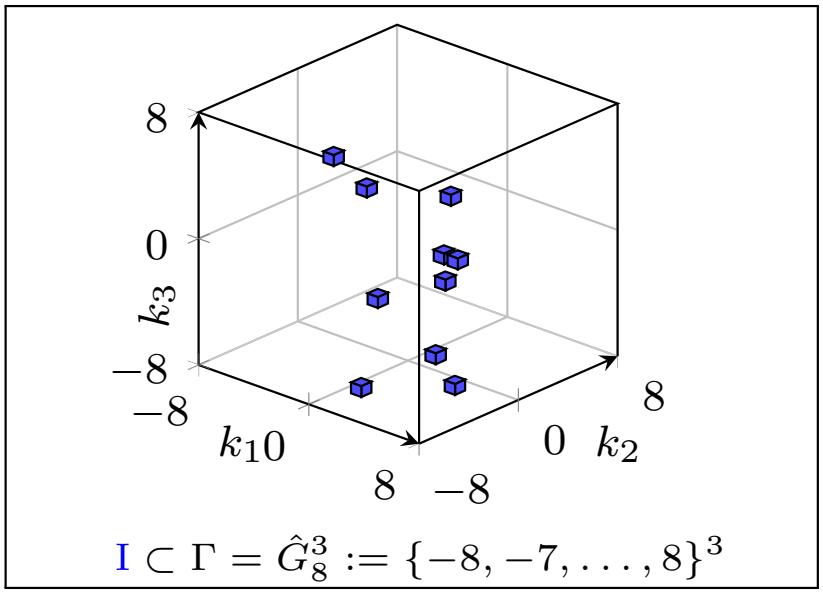
$I \subset \Gamma = \hat{G}_8^3 := \{-8, -7, \ldots, 8\}^3$



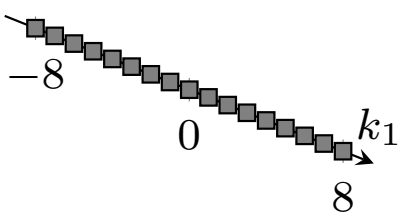
$I \subset \Gamma = \hat{G}_8^3 := \{-8, -7, \ldots, 8\}^3$



frequency candidates

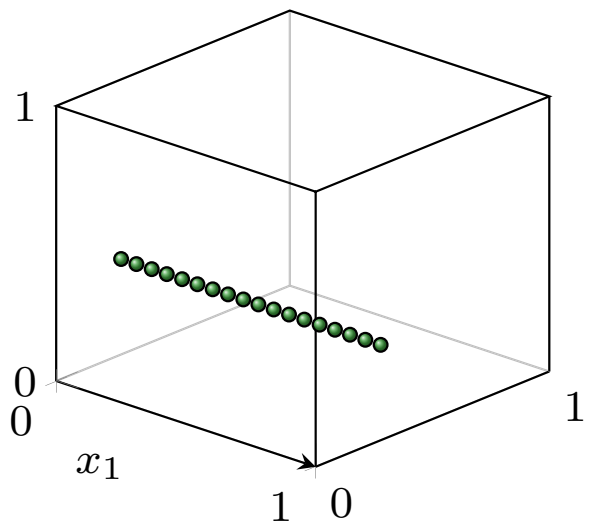


$I \subset \Gamma = \hat{G}_8^3 := \{-8, -7, \ldots, 8\}^3$

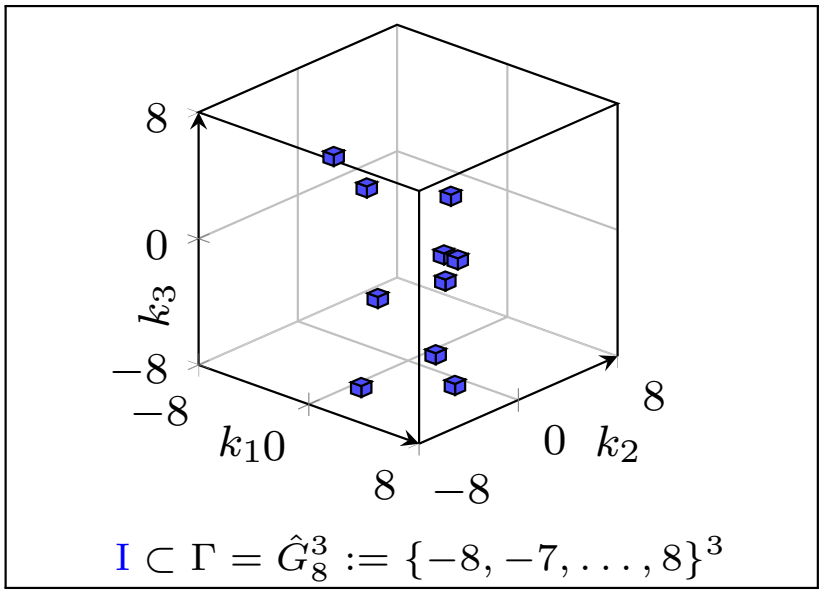


frequency candidates

construct
sampling set



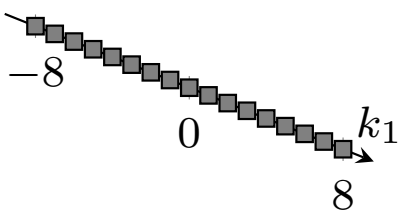
sampling nodes $\left\{\left(\frac{\ell}{17}, x_2', x_3'\right)\right\}_{\ell=0}^{16}$



$I \subset \Gamma = \hat{G}_8^3 := \{-8, -7, \ldots, 8\}^3$

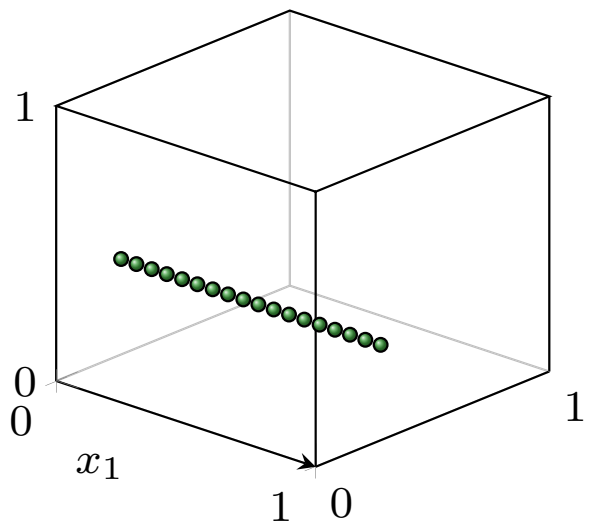
$$\hat{p}_{k_1} := \frac{1}{17} \sum_{\ell=0}^{16} p\left(\begin{pmatrix} \ell/17 \\ x_2' \\ x_3' \end{pmatrix}\right) \mathrm{e}^{-2\pi \mathrm{i} \frac{\ell k_1}{17}}$$

$$k_1 = -8, \ldots, 8$$

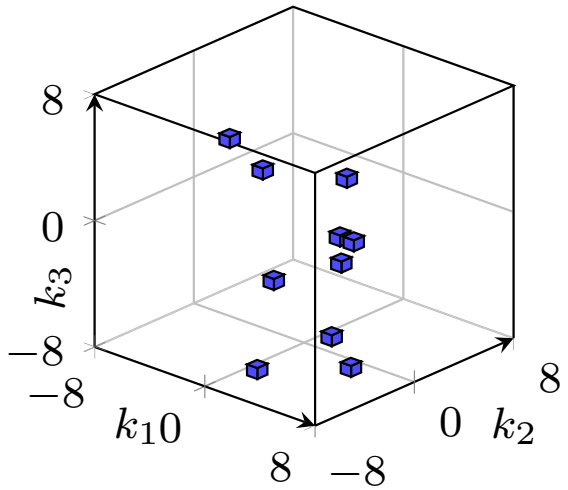


frequency candidates

1-dim. FFT



sampling nodes $\{(\frac{\ell}{17}, x_2', x_3')\}_{\ell=0}^{16}$

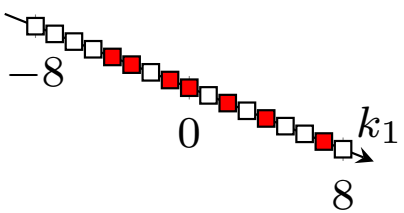


$\mathrm{I} \subset \Gamma = \hat{G}_8^3 := \{-8, -7, \ldots, 8\}^3$

$$\hat{p}_{k_1} := \frac{1}{17} \sum_{\ell=0}^{16} p\left(\begin{pmatrix} \ell/17 \\ x_2' \\ x_3' \end{pmatrix}\right) \mathrm{e}^{-2\pi \mathrm{i} \frac{\ell k_1}{17}}$$

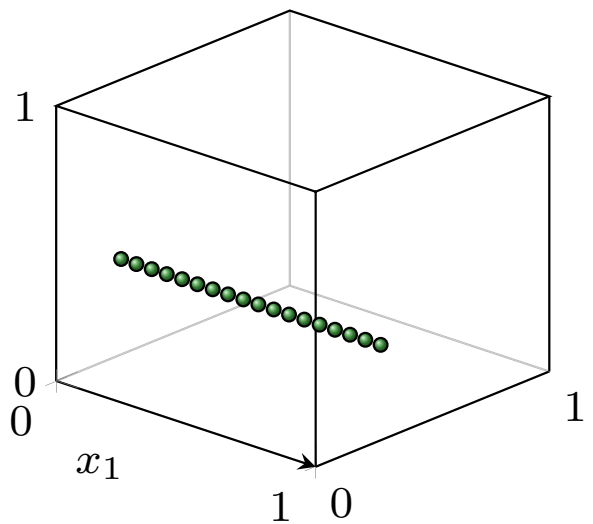
$$= \sum_{\substack{(h_2,h_3)\in\{-8,\ldots,8\}^2 \\ (k_1,h_2,h_3)^\top \in \operatorname{supp} \hat{p}}} \hat{p}\begin{pmatrix} k_1 \\ h_2 \\ h_3 \end{pmatrix} \mathrm{e}^{2\pi \mathrm{i}(h_2 x_2' + h_3 x_3')},$$

$k_1 = -8, \ldots, 8$

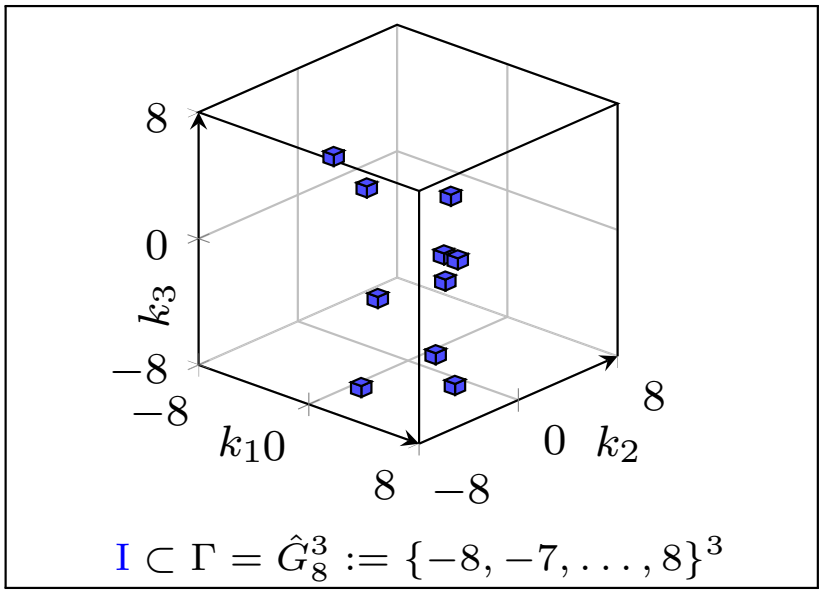


detected frequencies $\mathrm{I}^{(1)}$

1-dim. FFT



sampling nodes $\left\{\left(\frac{\ell}{17}, x_2', x_3'\right)\right\}_{\ell=0}^{16}$

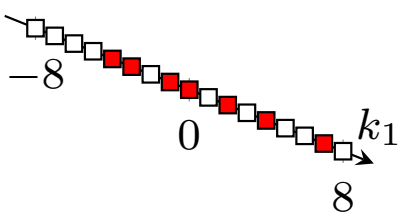


$\mathrm{I} \subset \Gamma = \hat{G}_8^3 := \{-8, -7, \ldots, 8\}^3$

$$\hat{p}_{k_1} := \frac{1}{17} \sum_{\ell=0}^{16} p\left(\begin{pmatrix} \ell/17 \\ x_2' \\ x_3' \end{pmatrix}\right) \mathrm{e}^{-2\pi \mathrm{i} \frac{\ell k_1}{17}}$$

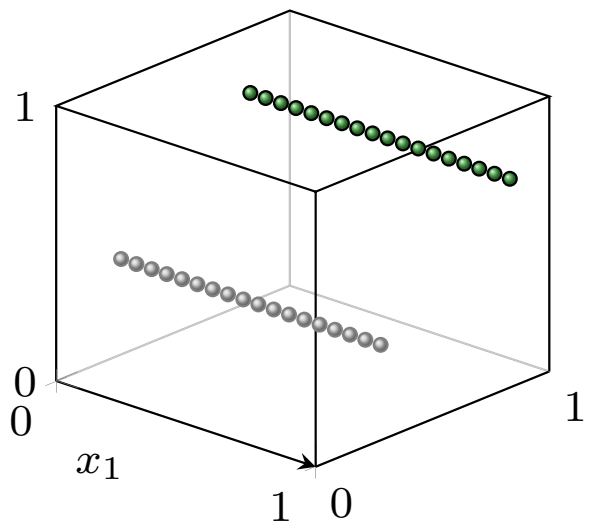
$$= \sum_{\substack{(h_2,h_3)\in\{-8,\ldots,8\}^2 \\ (k_1,h_2,h_3)^\top \in \operatorname{supp} \hat{p}}} \hat{p}\begin{pmatrix} k_1 \\ h_2 \\ h_3 \end{pmatrix} \mathrm{e}^{2\pi \mathrm{i}(h_2 x_2' + h_3 x_3')},$$

$k_1 = -8, \ldots, 8$



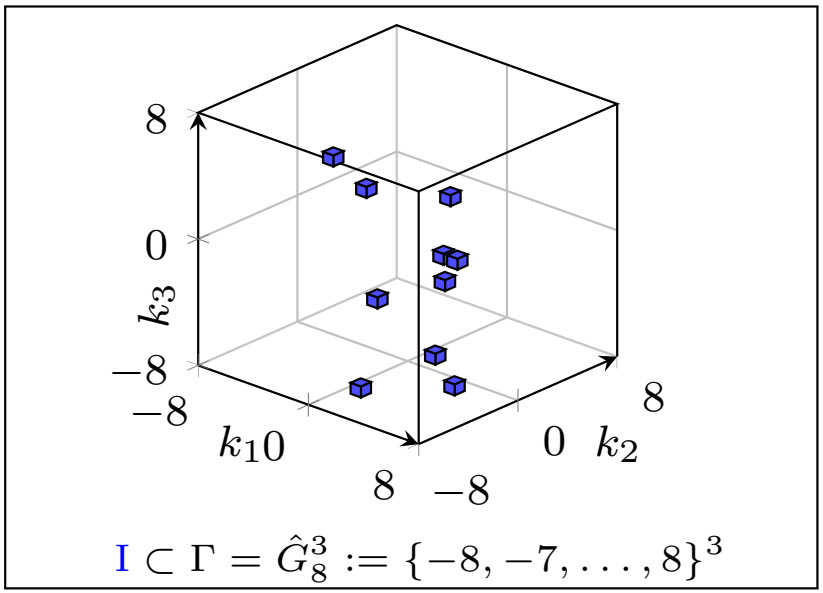
detected frequencies $\mathrm{I}^{(1)}$

1-dim. FFT

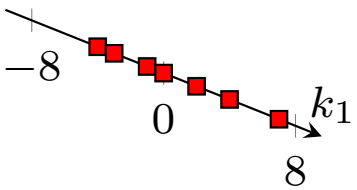


sampling nodes $\{(\frac{\ell}{17}, x_2', x_3')\}_{\ell=0}^{16}$

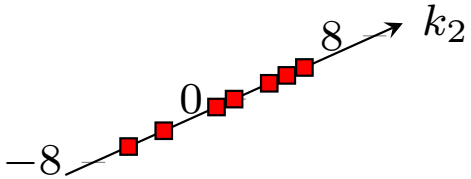
\+ repeat (r detection iterations)



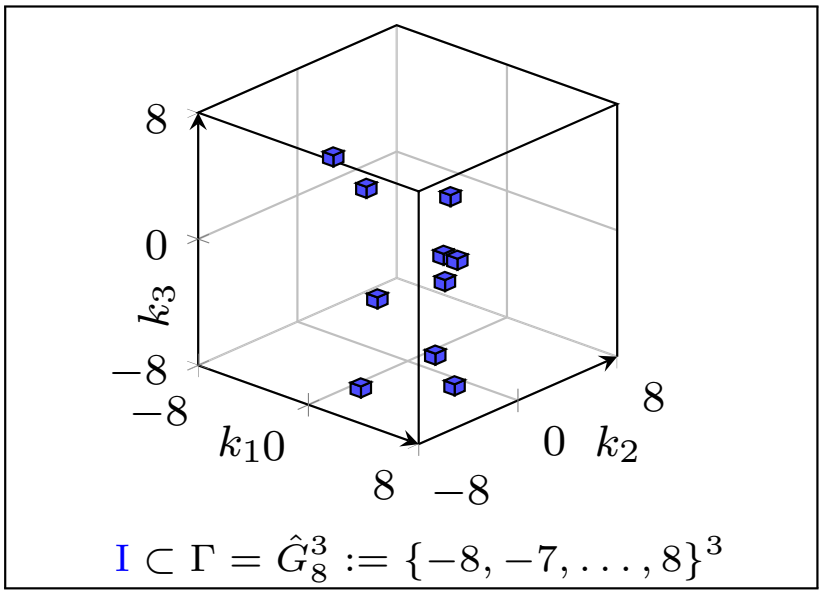
$\mathrm{I} \subset \Gamma = \hat{G}_8^3 := \{-8, -7, \ldots, 8\}^3$



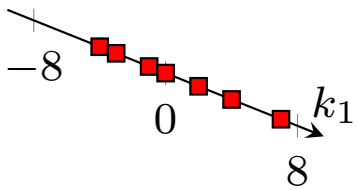
detected frequencies
$\mathrm{I}^{(1)}$



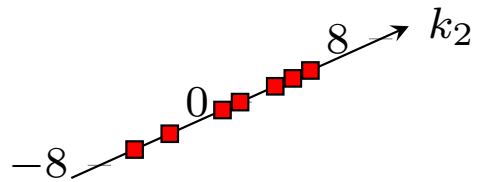
detected frequencies
$\mathrm{I}^{(2)}$



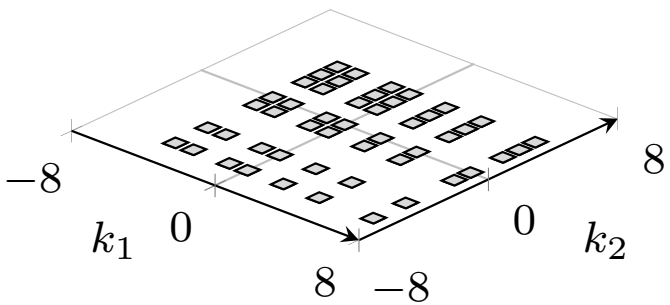
$\mathrm{I} \subset \Gamma = \hat{G}_8^3 := \{-8, -7, \ldots, 8\}^3$



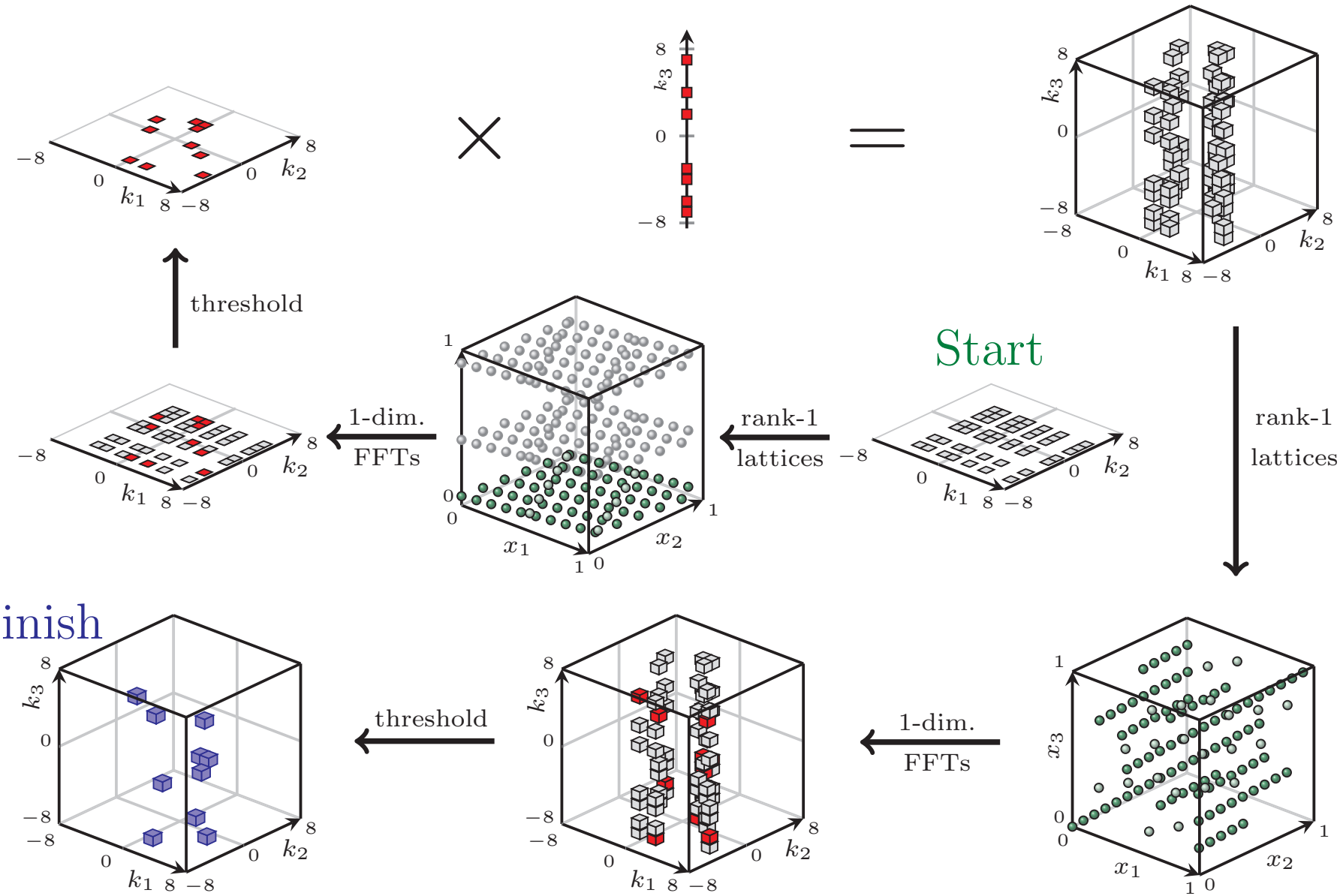
detected frequencies
$\mathrm{I}^{(1)}$



detected frequencies
$\mathrm{I}^{(2)}$



frequency candidates $\mathrm{I}^{(1)} \times \mathrm{I}^{(2)}$


Start
rank-1 lattices
1-dim. FFTs
threshold
1-dim. FFTs
threshold
Finish

- Back to our original problem:
  Approximating $u_{\boldsymbol{x}_0}(\boldsymbol{y}) \coloneqq u(\boldsymbol{x}_0, \boldsymbol{y})$ for a given $\boldsymbol{x}_0 \in \mathcal{D}$.
- Up to now:

$$u_{\boldsymbol{x}_0}(\boldsymbol{y}) \approx u_{\boldsymbol{x}_0}^{\mathtt{sFFT}}(\boldsymbol{y}) \coloneqq \sum_{\boldsymbol{k} \in \mathrm{I}_{\boldsymbol{x}_0}} c_{\boldsymbol{k}}^{\mathtt{sFFT}}(u_{\boldsymbol{x}_0})\, \mathrm{e}^{2\pi \mathrm{i} \boldsymbol{k} \cdot \boldsymbol{y}}$$

- Back to our original problem:
  Approximating $u_{\boldsymbol{x}_0}(\boldsymbol{y}) \coloneqq u(\boldsymbol{x}_0, \boldsymbol{y})$ for a given $\boldsymbol{x}_0 \in \mathcal{D}$.
- Up to now:

$$u_{\boldsymbol{x}_0}(\boldsymbol{y}) \approx u_{\boldsymbol{x}_0}^{\mathtt{sFFT}}(\boldsymbol{y}) \coloneqq \sum_{\boldsymbol{k} \in \mathrm{I}_{\boldsymbol{x}_0}} c_{\boldsymbol{k}}^{\mathtt{sFFT}}(u_{\boldsymbol{x}_0})\, \mathrm{e}^{2\pi \mathrm{i} \boldsymbol{k} \cdot \boldsymbol{y}}$$

- In practice: Consider $\{\boldsymbol{x}_g \in D,\, g = 1, ..., G\}$ instead of $\{\boldsymbol{x}_0 \in D\}$.
- Need G calls of the sFFT to compute $\mathrm{I}_{\boldsymbol{x}_g}$ and $c_{\boldsymbol{k}}^{\mathtt{sFFT}}(u_{\boldsymbol{x}_g})$, $g = 1, \ldots, G$.

- Back to our original problem:
  Approximating $u_{\boldsymbol{x}_0}(\boldsymbol{y}) \coloneqq u(\boldsymbol{x}_0, \boldsymbol{y})$ for a given $\boldsymbol{x}_0 \in \mathcal{D}$.
- Up to now:

$$u_{\boldsymbol{x}_0}(\boldsymbol{y}) \approx u_{\boldsymbol{x}_0}^{\mathtt{sFFT}}(\boldsymbol{y}) \coloneqq \sum_{\boldsymbol{k} \in \mathrm{I}_{\boldsymbol{x}_0}} c_{\boldsymbol{k}}^{\mathtt{sFFT}}(u_{\boldsymbol{x}_0})\, \mathrm{e}^{2\pi \mathrm{i} \boldsymbol{k} \cdot \boldsymbol{y}}$$

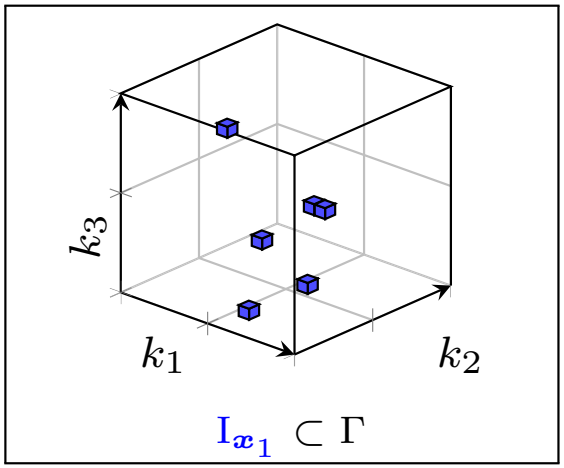
- In practice: Consider $\{\boldsymbol{x}_g \in D,\, g = 1, ..., G\}$ instead of $\{\boldsymbol{x}_0 \in D\}$.
- Need G calls of the sFFT to compute $\mathrm{I}_{\boldsymbol{x}_g}$ and $c_{\boldsymbol{k}}^{\mathtt{sFFT}}(u_{\boldsymbol{x}_g})$, $g = 1, \ldots, G$.
- Problem: Sampling is very expensive!
  One sample $u_{\boldsymbol{x}_g}(\boldsymbol{y}) \mathrel{\widehat{=}}$ one call of the PDE solver for fixed $\boldsymbol{y}$
- Sampling nodes $\boldsymbol{y}$ differ for each $\boldsymbol{x}_g \longrightarrow$ we can't reuse the PDE solutions.

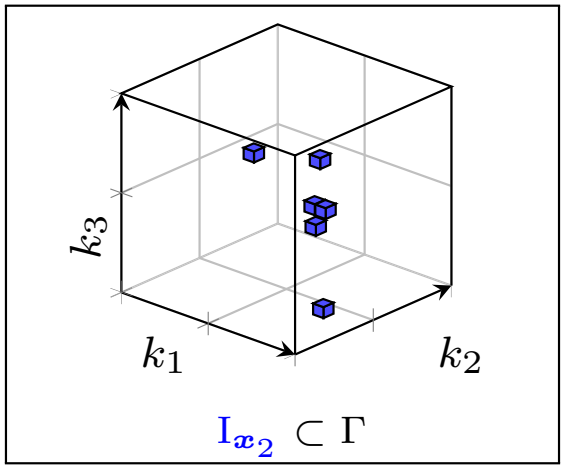
- Back to our original problem:
  Approximating $u_{\boldsymbol{x}_0}(\boldsymbol{y}) \coloneqq u(\boldsymbol{x}_0, \boldsymbol{y})$ for a given $\boldsymbol{x}_0 \in \mathcal{D}$.
- Up to now:

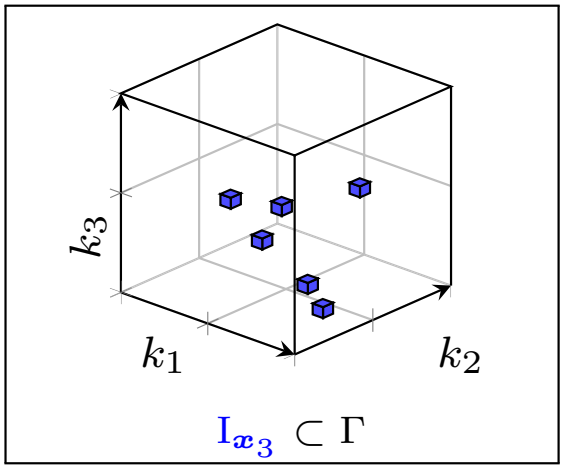
$$u_{\boldsymbol{x}_0}(\boldsymbol{y}) \approx u_{\boldsymbol{x}_0}^{\mathtt{sFFT}}(\boldsymbol{y}) \coloneqq \sum_{\boldsymbol{k} \in \mathrm{I}_{\boldsymbol{x}_0}} c_{\boldsymbol{k}}^{\mathtt{sFFT}}(u_{\boldsymbol{x}_0})\, \mathrm{e}^{2\pi \mathrm{i} \boldsymbol{k} \cdot \boldsymbol{y}}$$

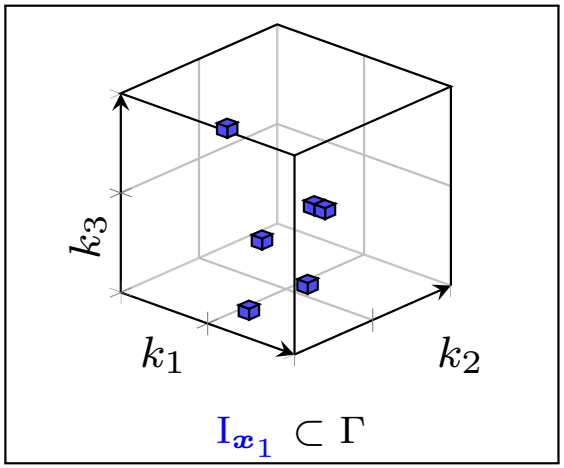
- In practice: Consider $\{\boldsymbol{x}_g \in D,\, g = 1, ..., G\}$ instead of $\{\boldsymbol{x}_0 \in D\}$.
- Need G calls of the sFFT to compute $\mathrm{I}_{\boldsymbol{x}_g}$ and $c_{\boldsymbol{k}}^{\mathtt{sFFT}}(u_{\boldsymbol{x}_g})$, $g = 1, \ldots, G$.
- Problem: Sampling is very expensive!
  One sample $u_{\boldsymbol{x}_g}(\boldsymbol{y}) \mathrel{\widehat{=}}$ one call of the PDE solver for fixed $\boldsymbol{y}$
- Sampling nodes $\boldsymbol{y}$ differ for each $\boldsymbol{x}_g \longrightarrow$ we can't reuse the PDE solutions.

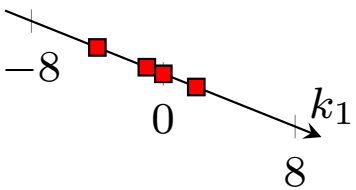
- Solution: the *uniform* sFFT!


k_3
k_1
k_2
$\mathrm{I}_{\boldsymbol{x}_1} \subset \Gamma$

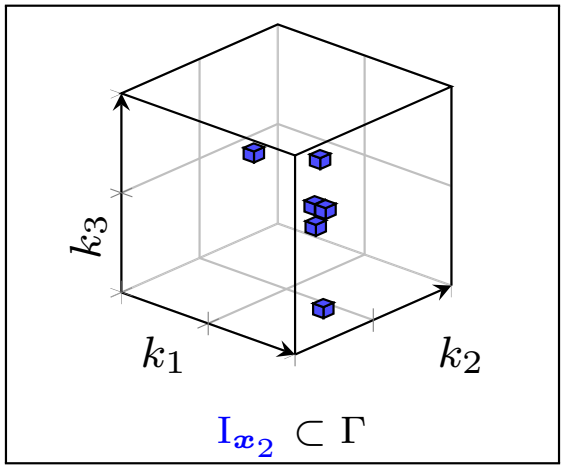

k_3
k_1
k_2
$\mathrm{I}_{\boldsymbol{x}_2} \subset \Gamma$

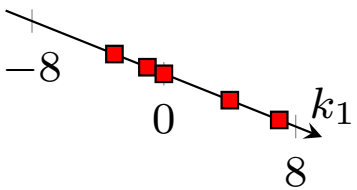

k_3
k_1
k_2
$\mathrm{I}_{\boldsymbol{x}_3} \subset \Gamma$



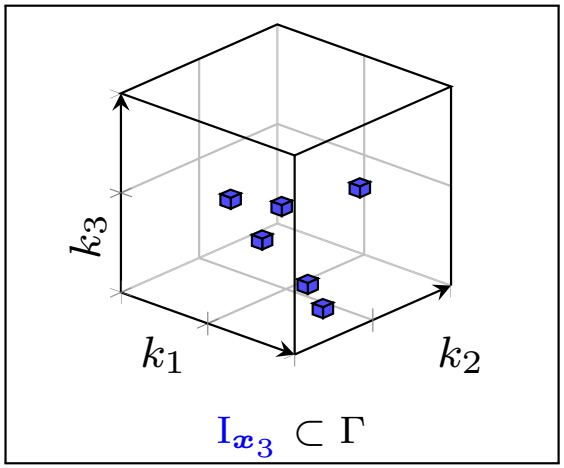


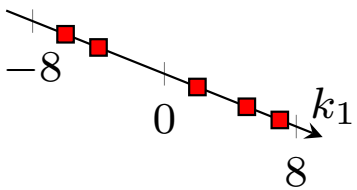
detected frequencies
$\mathrm{I}^{(1)}_{\boldsymbol{x}_1}$



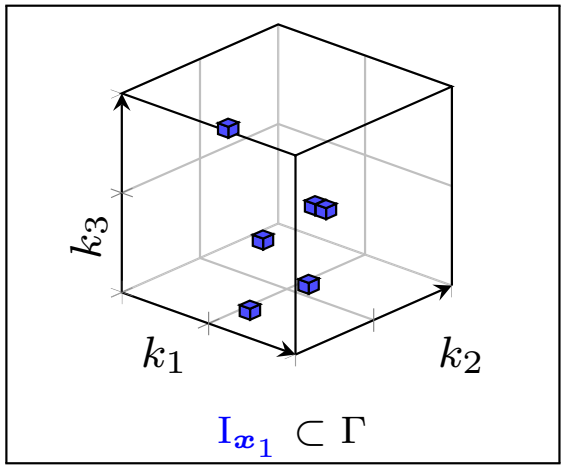


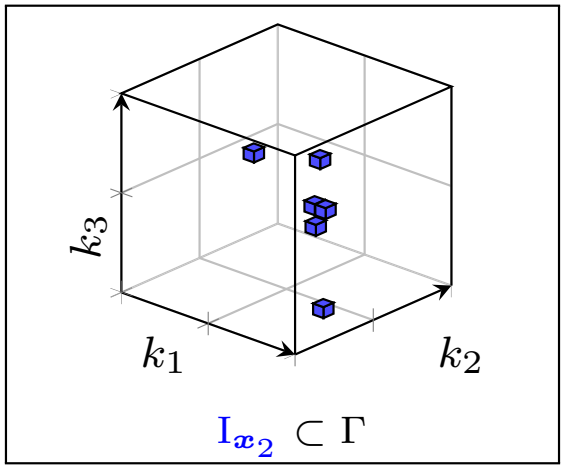
detected frequencies
$\mathrm{I}^{(1)}_{\boldsymbol{x}_2}$

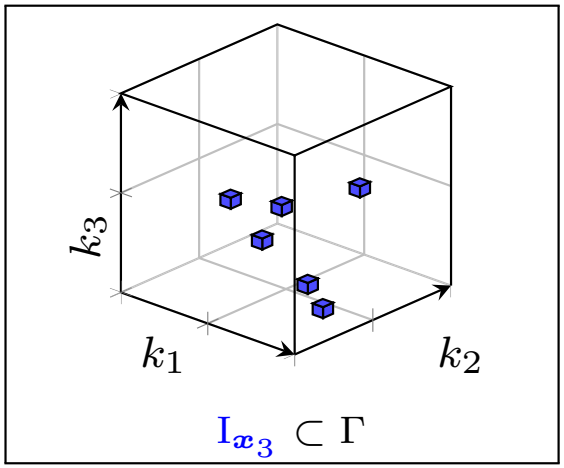


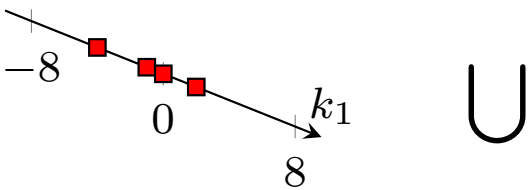


detected frequencies
$\mathrm{I}^{(1)}_{\boldsymbol{x}_3}$



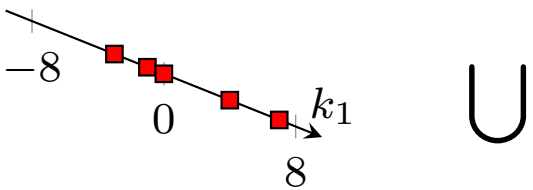






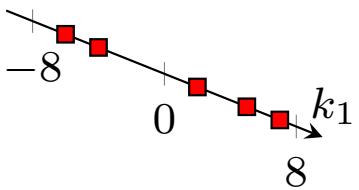
detected frequencies
$\mathrm{I}_{\boldsymbol{x}_1}^{(1)}$

$\cup$

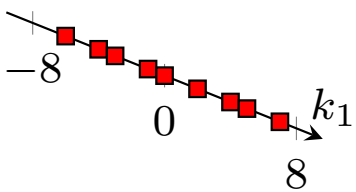


detected frequencies
$\mathrm{I}_{\boldsymbol{x}_2}^{(1)}$

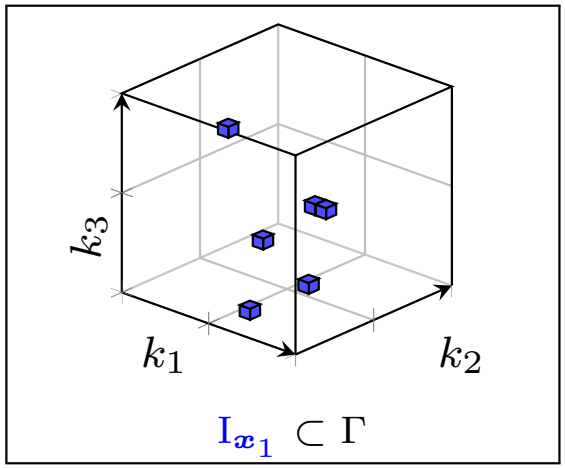
$\cup$



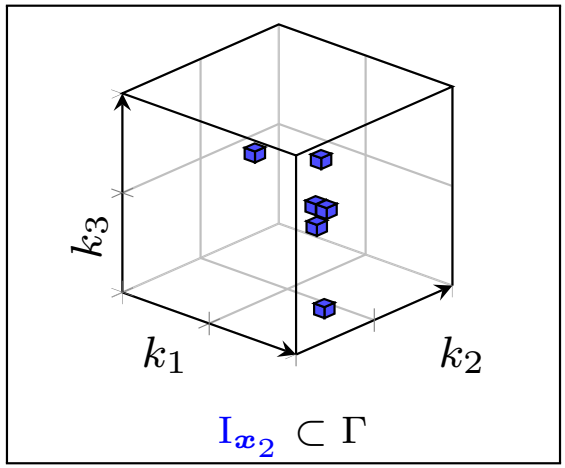
detected frequencies
$\mathrm{I}_{\boldsymbol{x}_3}^{(1)}$



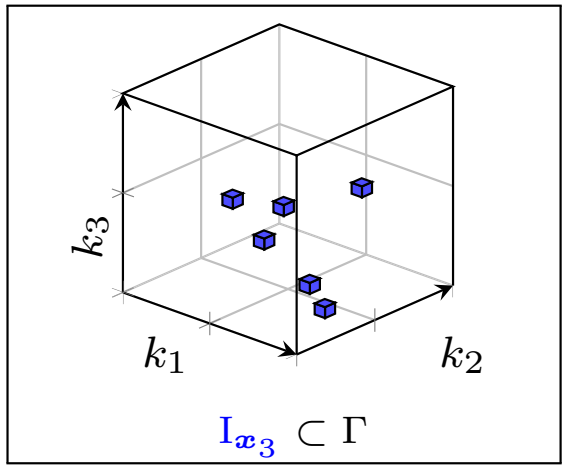
detected frequencies
$\mathrm{I}^{(1)} := \mathrm{I}_{\boldsymbol{x}_1}^{(1)} \cup \mathrm{I}_{\boldsymbol{x}_2}^{(1)} \cup \mathrm{I}_{\boldsymbol{x}_3}^{(1)}$



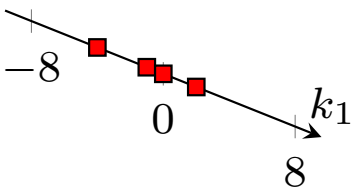
$\mathrm{I}_{\boldsymbol{x}_1} \subset \Gamma$



$\mathrm{I}_{\boldsymbol{x}_2} \subset \Gamma$

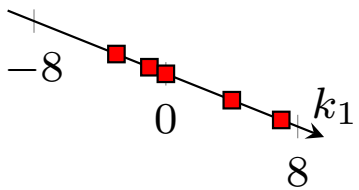


$\mathrm{I}_{\boldsymbol{x}_3} \subset \Gamma$



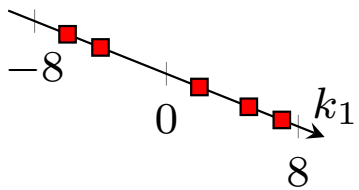
detected frequencies
$\mathrm{I}^{(1)}_{\boldsymbol{x}_1}$

$\cup$

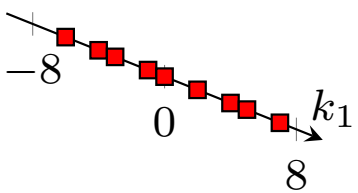


detected frequencies
$\mathrm{I}^{(1)}_{\boldsymbol{x}_2}$

$\cup$

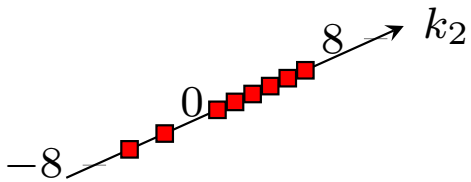


detected frequencies
$\mathrm{I}^{(1)}_{\boldsymbol{x}_3}$



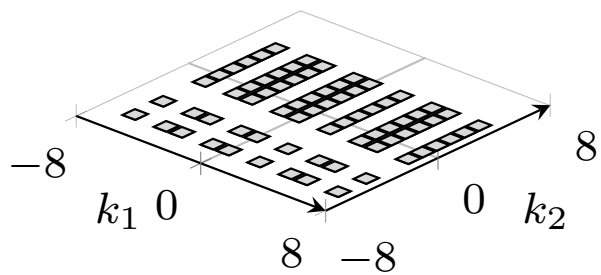
detected frequencies
$\mathrm{I}^{(1)} := \mathrm{I}^{(1)}_{\boldsymbol{x}_1} \cup \mathrm{I}^{(1)}_{\boldsymbol{x}_2} \cup \mathrm{I}^{(1)}_{\boldsymbol{x}_3}$

$\times$

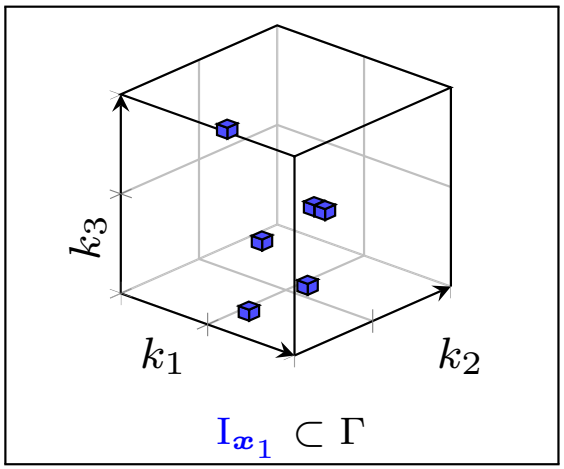


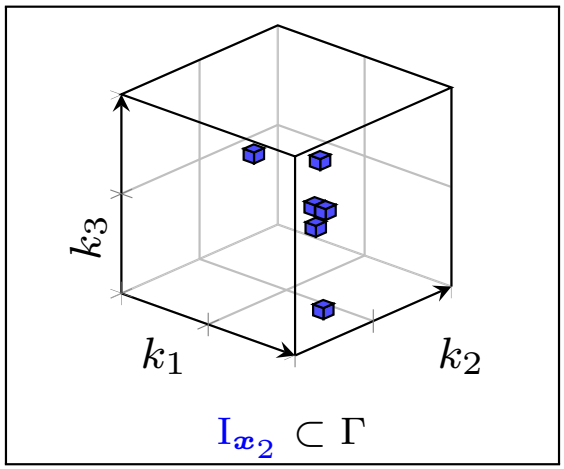
detected frequencies
$\mathrm{I}^{(2)} := \mathrm{I}^{(2)}_{\boldsymbol{x}_1} \cup \mathrm{I}^{(2)}_{\boldsymbol{x}_2} \cup \mathrm{I}^{(2)}_{\boldsymbol{x}_3}$

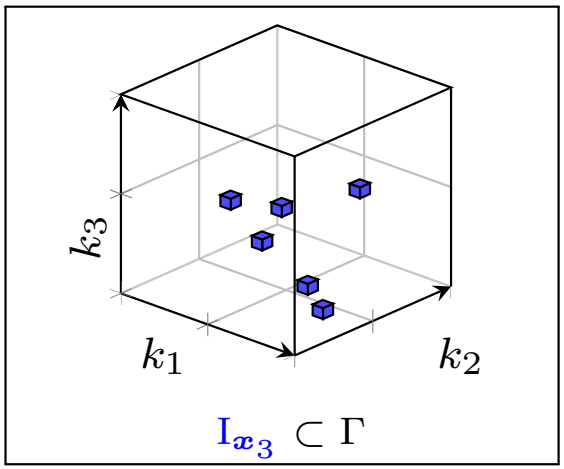
$=$

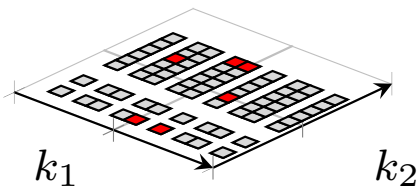


frequency candidates $\mathrm{I}^{(1)} \times \mathrm{I}^{(2)}$

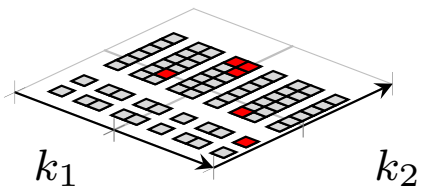




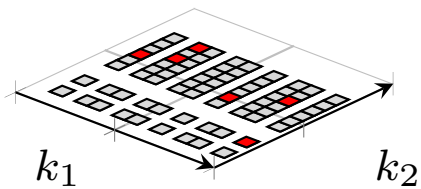




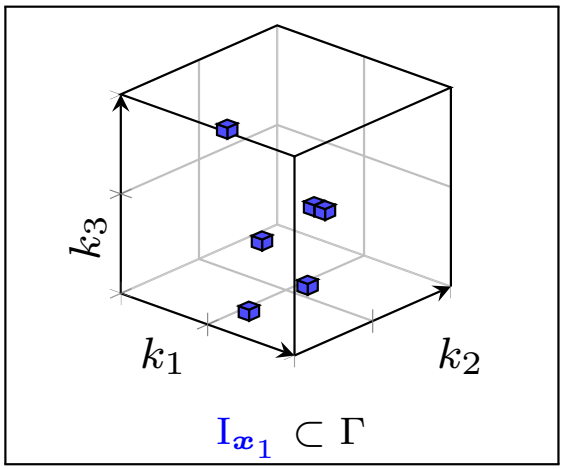
detected frequencies
$\mathrm{I}_{\boldsymbol{x}_1}^{(1,2)}$



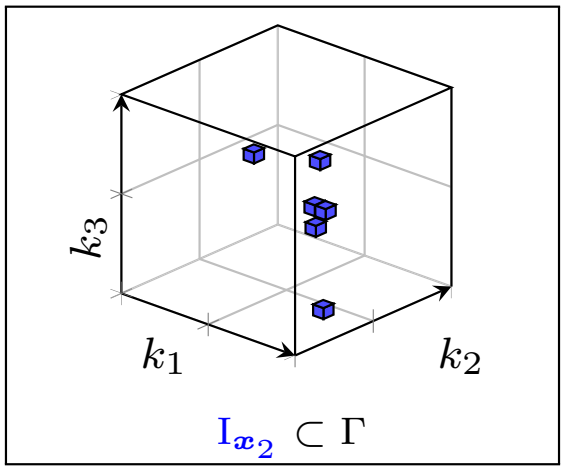
detected frequencies
$\mathrm{I}_{\boldsymbol{x}_2}^{(1,2)}$



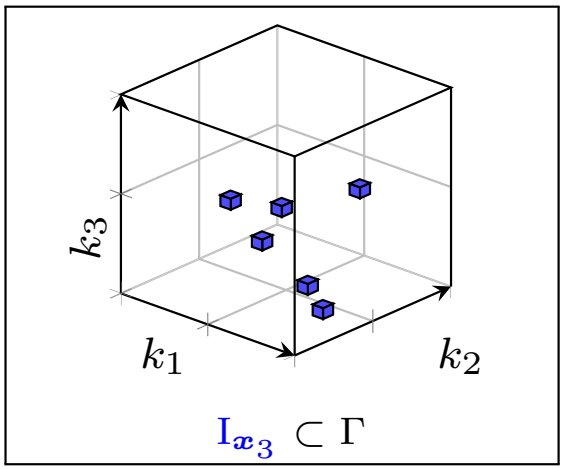
detected frequencies
$\mathrm{I}_{\boldsymbol{x}_3}^{(1,2)}$



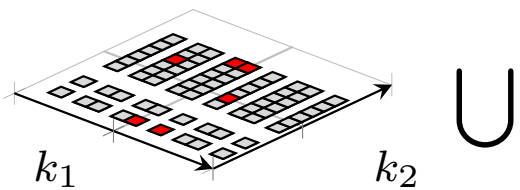
$\mathrm{I}_{\boldsymbol{x}_1} \subset \Gamma$



$\mathrm{I}_{\boldsymbol{x}_2} \subset \Gamma$

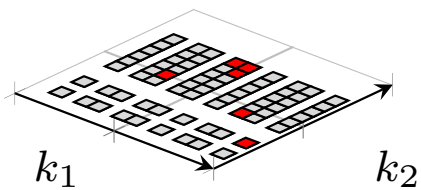


$\mathrm{I}_{\boldsymbol{x}_3} \subset \Gamma$



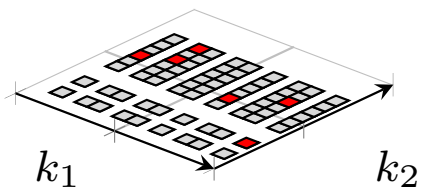
$\bigcup$

detected frequencies
$\mathrm{I}_{\boldsymbol{x}_1}^{(1,2)}$

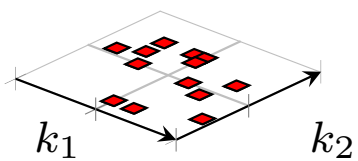


$\bigcup$

detected frequencies
$\mathrm{I}_{\boldsymbol{x}_2}^{(1,2)}$

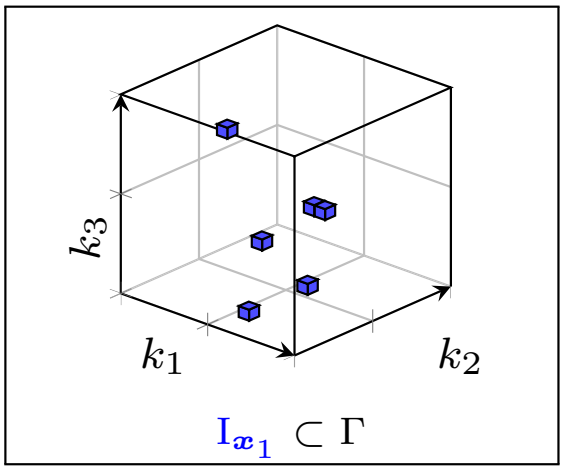


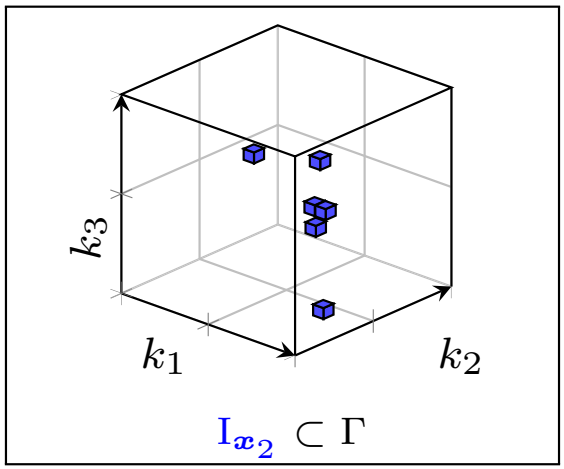
detected frequencies
$\mathrm{I}_{\boldsymbol{x}_3}^{(1,2)}$

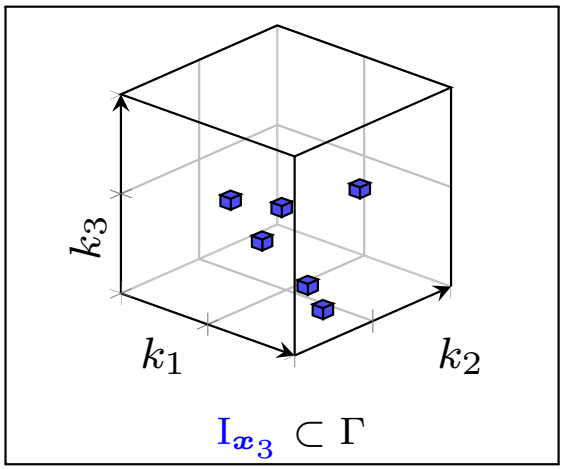


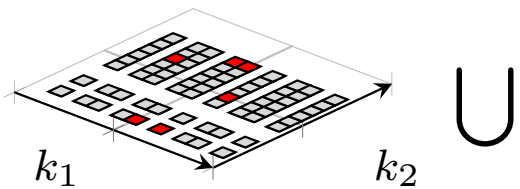
detected frequencies

$$\mathrm{I}^{(1,2)} := \mathrm{I}_{\boldsymbol{x}_1}^{(1,2)} \cup \mathrm{I}_{\boldsymbol{x}_2}^{(1,2)} \cup \mathrm{I}_{\boldsymbol{x}_3}^{(1,2)}$$



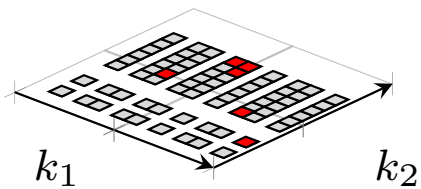






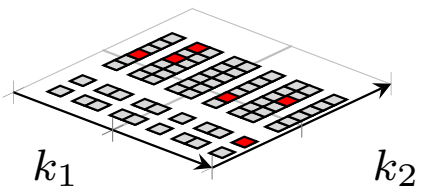
$\bigcup$

detected frequencies
$\mathrm{I}_{\boldsymbol{x}_1}^{(1,2)}$

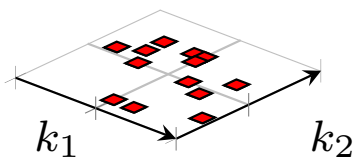


$\bigcup$

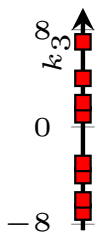
detected frequencies
$\mathrm{I}_{\boldsymbol{x}_2}^{(1,2)}$



detected frequencies
$\mathrm{I}_{\boldsymbol{x}_3}^{(1,2)}$



$\times$

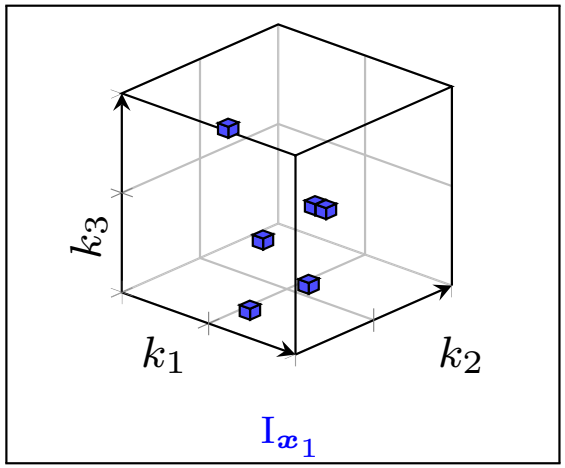


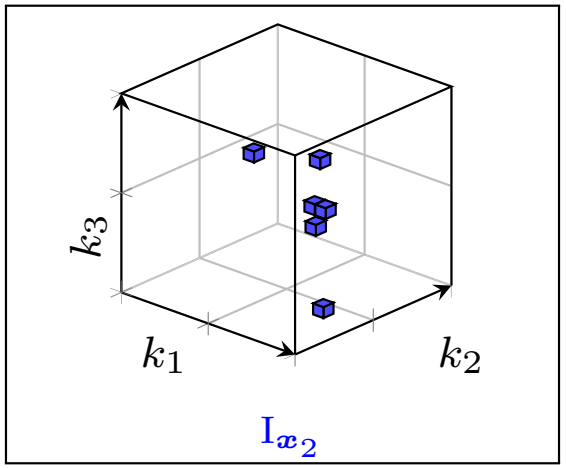
$=$

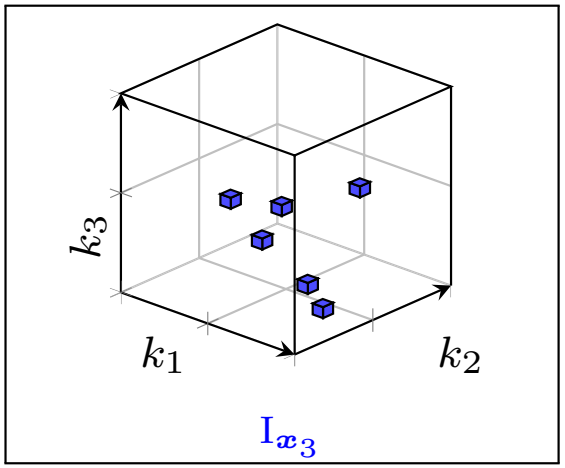
$\cdots$

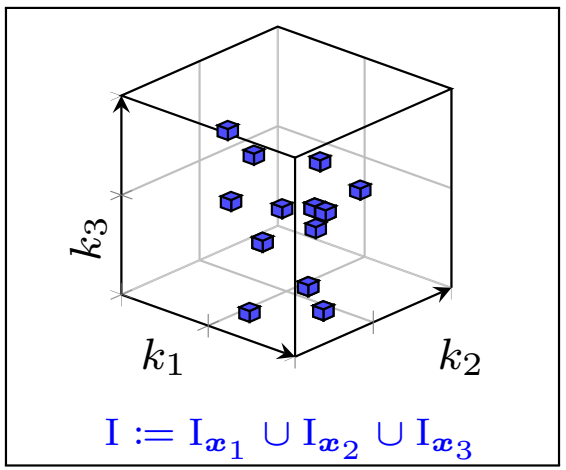
detected frequencies
$\mathrm{I}^{(1,2)} := \mathrm{I}_{\boldsymbol{x}_1}^{(1,2)} \cup \mathrm{I}_{\boldsymbol{x}_2}^{(1,2)} \cup \mathrm{I}_{\boldsymbol{x}_3}^{(1,2)}$

detected frequencies
$\mathrm{I}^{(3)} := \mathrm{I}_{\boldsymbol{x}_1}^{(3)} \cup \mathrm{I}_{\boldsymbol{x}_2}^{(3)} \cup \mathrm{I}_{\boldsymbol{x}_3}^{(3)}$









- Result:

$$u_{\boldsymbol{x}_g}(\boldsymbol{y}) \approx u_{\boldsymbol{x}_g}^{\mathtt{usFFT}}(\boldsymbol{y}) := \sum_{\boldsymbol{k}\in\mathrm{I}} c_{\boldsymbol{k}}^{\mathtt{usFFT}}\left(u_{\boldsymbol{x}_g}\right) \mathrm{e}^{2\pi\mathrm{i}\boldsymbol{k}\cdot\boldsymbol{y}} \qquad g = 1, \ldots, G$$

- Output:
  - index set $\mathrm{I} \in \Gamma$ with $\mathrm{I}_{\boldsymbol{x}_g} \subset \mathrm{I}$ for all $g = 1, ..., G$
  - approximations $c_{\boldsymbol{k}}^{\mathtt{usFFT}}(u_{\boldsymbol{x}_g}), \boldsymbol{k} \in \mathrm{I}$, for all $g = 1, ..., G$

- Output:
  - index set $\mathrm{I} \in \Gamma$ with $\mathrm{I}_{\boldsymbol{x}_g} \subset \mathrm{I}$ for all $g = 1, ..., G$
  - approximations $c_{\boldsymbol{k}}^{\texttt{usFFT}}(u_{\boldsymbol{x}_g}), \boldsymbol{k} \in \mathrm{I}$, for all $g = 1, ..., G$

- samples $\hat{=}$ amount of PDE solutions needed:

$$\mathcal{O}\left(d\,s\,\max(s, N_\Gamma)\,\log^2\frac{d\,s\,G\,N_\Gamma}{\delta} + \max(sG, N_\Gamma)\,\log\frac{d\,s\,G}{\delta}\right)$$

- Output:
  - index set $\mathrm{I} \in \Gamma$ with $\mathrm{I}_{\boldsymbol{x}_g} \subset \mathrm{I}$ for all $g = 1, ..., G$
  - approximations $c_{\boldsymbol{k}}^{\texttt{usFFT}}(u_{\boldsymbol{x}_g}), \boldsymbol{k} \in \mathrm{I}$, for all $g = 1, ..., G$

- samples $\hat{=}$ amount of PDE solutions needed:

$$\mathcal{O}\left(d\, s\, \max(s, N_\Gamma)\, \log^2 \frac{d\, s\, G\, N_\Gamma}{\delta} + \max(sG, N_\Gamma)\, \log \frac{d\, s\, G}{\delta}\right)$$

- computational complexity of the usFFT:

$$\mathcal{O}\left(d^2\, s^2\, G^2\, N_\Gamma\, \log^3 \frac{d\, s\, G\, N_\Gamma}{\delta}\right) \quad \text{with high probability } 1 - \delta$$

$$\mathcal{O}\left(d^2\, s^3\, G^2\, N_\Gamma\, \log^3 \frac{d\, s\, G\, N_\Gamma}{\delta}\right) \quad \text{worst case}$$

- Output:
  - index set $\mathrm{I} \in \Gamma$ with $\mathrm{I}_{\boldsymbol{x}_g} \subset \mathrm{I}$ for all $g = 1, ..., G$
  - approximations $c_{\boldsymbol{k}}^{\texttt{usFFT}}(u_{\boldsymbol{x}_g}), \boldsymbol{k} \in \mathrm{I}$, for all $g = 1, ..., G$

- samples $\hat{=}$ amount of PDE solutions needed:

$$\mathcal{O}\left(d\, s\, \max(s, N_\Gamma)\, \log^2 \frac{d\, s\, G\, N_\Gamma}{\delta} + \max(sG, N_\Gamma)\, \log \frac{d\, s\, G}{\delta}\right)$$

- computational complexity of the usFFT:

$$\mathcal{O}\left(d^2\, s^2\, G^2\, N_\Gamma\, \log^3 \frac{d\, s\, G\, N_\Gamma}{\delta}\right) \quad \text{with high probability } 1 - \delta$$

$$\mathcal{O}\left(d^2\, s^3\, G^2\, N_\Gamma\, \log^3 \frac{d\, s\, G\, N_\Gamma}{\delta}\right) \quad \text{worst case}$$

- Output:
  - index set $\mathrm{I} \in \Gamma$ with $\mathrm{I}_{\boldsymbol{x}_g} \subset \mathrm{I}$ for all $g = 1, ..., G$
  - approximations $c_{\boldsymbol{k}}^{\texttt{usFFT}}(u_{\boldsymbol{x}_g}), \boldsymbol{k} \in \mathrm{I}$, for all $g = 1, ..., G$

- samples $\hat{=}$ amount of PDE solutions needed: $G \lesssim ds$

$$\mathcal{O}\left(d\,s\,\max(s, N_\Gamma)\,\log^2\frac{d\,s\,G\,N_\Gamma}{\delta} + \max(sG, N_\Gamma)\,\log\frac{d\,s\,G}{\delta}\right)$$

- computational complexity of the usFFT:

$$\mathcal{O}\left(d^2\,s^2\,G^2\,N_\Gamma\,\log^3\frac{d\,s\,G\,N_\Gamma}{\delta}\right) \quad \text{with high probability } 1-\delta$$

$$\mathcal{O}\left(d^2\,s^3\,G^2\,N_\Gamma\,\log^3\frac{d\,s\,G\,N_\Gamma}{\delta}\right) \quad \text{worst case}$$

Consider the problem

$$\begin{aligned} -\nabla_{\boldsymbol{x}} \cdot (a(\boldsymbol{x},\boldsymbol{y})\nabla_{\boldsymbol{x}} u(\boldsymbol{x},\boldsymbol{y})) &= 1 && \boldsymbol{x} \in D,\, \boldsymbol{y} \in D_{\boldsymbol{y}} \\ u(\boldsymbol{x},\boldsymbol{y}) &= 0 && \forall \boldsymbol{x} \in \partial D,\, \boldsymbol{y} \in D_{\boldsymbol{y}} \end{aligned}$$

with $D = (0,1)^2$, $D_{\boldsymbol{y}} = [-1,1]^{20}$, $\boldsymbol{y} \sim \mathcal{U}\left([-1,1]^{20}\right)$ and the random coefficient

$$a(\boldsymbol{x},\boldsymbol{y}) = 1 + \sum_{j=1}^{20} y_j \psi_j(\boldsymbol{x})$$

with

$$\psi_j(\boldsymbol{x}) := \frac{0.9}{\zeta(2)} j^{-2} \cos(2\pi m_1(j) x_1) \cos(2\pi m_2(j) x_2), \qquad \boldsymbol{x} \in D,\, j \geq 1.$$

j	1	2	3	4	5	6	7	8	9	10	11	12	13	14	...
$m_1(j)$	0	1	0	1	2	0	1	2	3	0	1	2	3	4	...
$m_2(j)$	1	0	2	1	0	3	2	1	0	4	3	2	1	0	...

Example taken from
M. Eigel, C. J. Gittelson, C. Schwab and E. Zander.
Adaptive stochastic Galerkin FEM.
Comput. Methods Appl. Mech. Engrg., 270: 247–269, 2014.

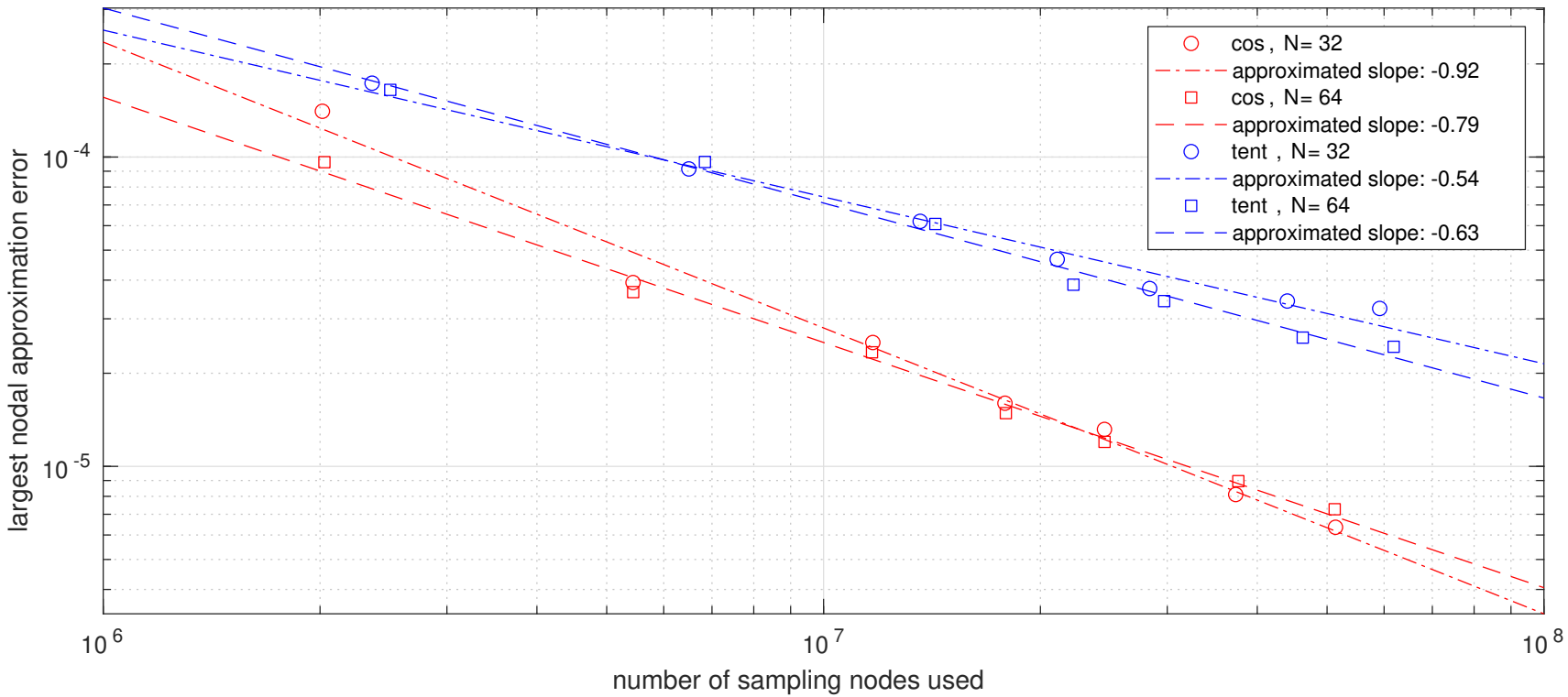


Figure: Largest error err_2^η w.r.t. the nodes $\boldsymbol{x}_g$ for different parameter settings, i.e., $s = 100, 250, 500, 750, 1000, 1500, 2000$, for the affine example.

$$\mathrm{err}_2^\eta(\boldsymbol{x}_g) := \sqrt{\frac{1}{n_{\text{test}}} \sum_{j=1}^{n_{\text{test}}} \left| \check{u}\left(\boldsymbol{x}_g, \boldsymbol{y}^{(j)}\right) - u^{\texttt{usFFT}}\left(\boldsymbol{x}_g, \boldsymbol{y}^{(j)}\right) \right|^2}$$

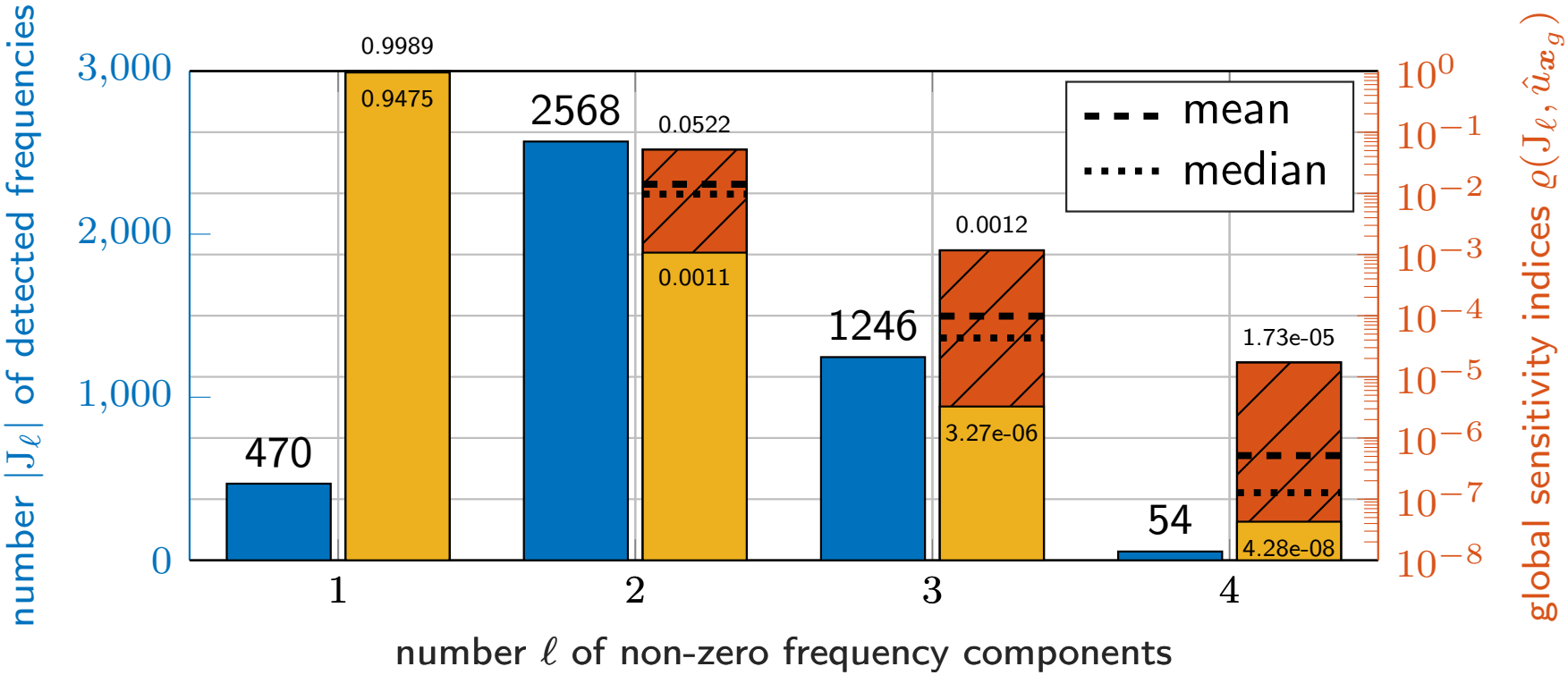


Figure: Analysis of the approximation for the affine example with $s = 2000,\ N = 32$.

$$\varrho(\mathrm{J}, \tilde{u}^{\mathtt{usFFT}}_{\boldsymbol{x}_g}) := \frac{\sigma^2(\tilde{u}^{\mathtt{usFFT}}_{\boldsymbol{x}_g,\mathrm{J}})}{\sigma^2(\tilde{u}^{\mathtt{usFFT}}_{\boldsymbol{x}_g})} = \frac{\sum_{\boldsymbol{k}\in\mathrm{J}\setminus\{\mathbf{0}\}} |c^{\mathtt{usFFT}}_{\boldsymbol{k}}(\tilde{u}_{\boldsymbol{x}_g})|^2}{\sum_{\boldsymbol{k}\in\mathrm{I}\setminus\{\mathbf{0}\}} |c^{\mathtt{usFFT}}_{\boldsymbol{k}}(\tilde{u}_{\boldsymbol{x}_g})|^2} \in [0,1],$$

Consider the problem

$$\begin{aligned} -\nabla_{\boldsymbol{x}} \cdot (a(\boldsymbol{x},\boldsymbol{y})\nabla_{\boldsymbol{x}} u(\boldsymbol{x},\boldsymbol{y})) &= f(\boldsymbol{x}) && \boldsymbol{x} \in D,\, \boldsymbol{y} \in D_{\boldsymbol{y}} \\ u(\boldsymbol{x},\boldsymbol{y}) &= 0 && \forall \boldsymbol{x} \in \partial D,\, \boldsymbol{y} \in D_{\boldsymbol{y}} \end{aligned}$$

with $f(\boldsymbol{x}) = \sin(1.3\pi x_1 + 3.4\pi x_2)\cos(4.3\pi x_1 - 3.1\pi x_2)$, $D = (0,1)^2$, $D_{\boldsymbol{y}} = \mathbb{R}^{10}$, $\boldsymbol{y} \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{I})$ and the random coefficient

$$\log a(\boldsymbol{x},\boldsymbol{y}) = \sum_{j=1}^{10} \frac{y_j}{j} \sin(2\pi j x_1)\cos(2\pi(11-j)x_2).$$

Example taken and modified from
M. Cheng, T. Y. Hou, M. Yan and Z. Zhang.
A data-driven stochastic method for elliptic PDEs with random coefficients.
SIAM/ASA J. Uncertain. Quantif., 1: 452-493, 2013.

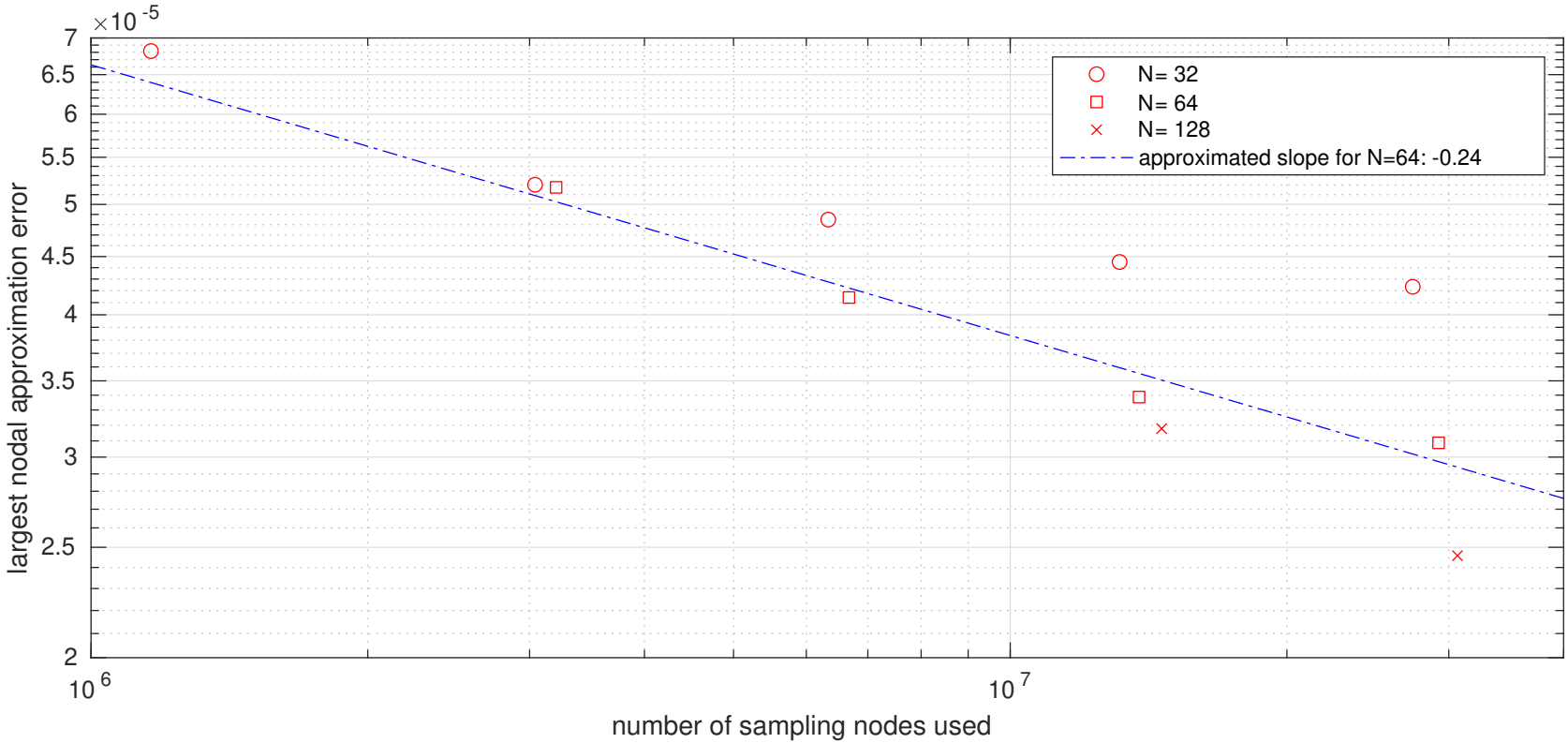


Figure: Largest error err_2^η w.r.t. the nodes $\boldsymbol{x}_g$ for different parameter settings, i.e., $s = 100, 250, 500, 1000, 2000$, for the lognormal example.

$$\mathrm{err}_2^\eta(\boldsymbol{x}_g) := \sqrt{\frac{1}{n_{\text{test}}} \sum_{j=1}^{n_{\text{test}}} \left| \breve{u}\left(\boldsymbol{x}_g, \boldsymbol{y}^{(j)}\right) - u^{\texttt{usFFT}}\left(\boldsymbol{x}_g, \boldsymbol{y}^{(j)}\right) \right|^2}$$

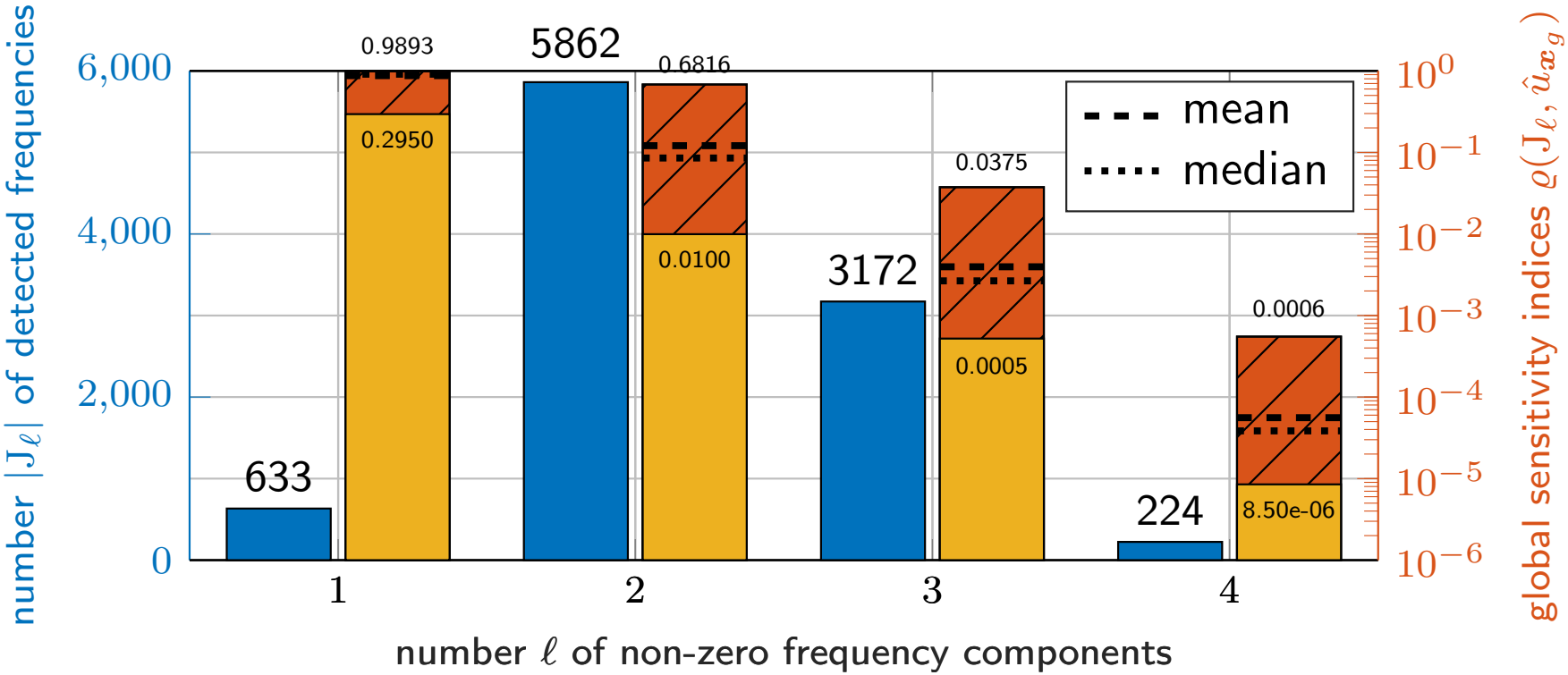


Figure: Analysis of the approximation for the lognormal example with $s = 2000,\ N = 32$.

$$\varrho(\mathrm{J}, \tilde{u}_{\boldsymbol{x}_g}^{\mathtt{usFFT}}) := \frac{\sigma^2(\tilde{u}_{\boldsymbol{x}_g,\mathrm{J}}^{\mathtt{usFFT}})}{\sigma^2(\tilde{u}_{\boldsymbol{x}_g}^{\mathtt{usFFT}})} = \frac{\sum_{\boldsymbol{k}\in\mathrm{J}\setminus\{\mathbf{0}\}} |c_{\boldsymbol{k}}^{\mathtt{usFFT}}(\tilde{u}_{\boldsymbol{x}_g})|^2}{\sum_{\boldsymbol{k}\in\mathrm{I}\setminus\{\mathbf{0}\}} |c_{\boldsymbol{k}}^{\mathtt{usFFT}}(\tilde{u}_{\boldsymbol{x}_g})|^2} \in [0,1],$$

- Main advantages of the usFFT:
  - fully adaptive, no critical a priori choice needed
  - sample efficient (in terms of sampling locations)
  - approximation gives insight on the influence and interactions of the y_j
  - adapts easily to other domains, boundary conditions, ...
  - non-intrusive and parallelizable

- Main advantages of the usFFT:
  - fully adaptive, no critical a priori choice needed
  - sample efficient (in terms of sampling locations)
  - approximation gives insight on the influence and interactions of the y_j
  - adapts easily to other domains, boundary conditions, ...
  - non-intrusive and parallelizable

- Lutz Kämmerer, Daniel Potts, Fabian Taubert
  The uniform sparse FFT with application to PDEs with random coefficients
  ArXiv e-prints, 2021. arXiv:2109.04131 [math.NA]

- Thank you for your attention!