Exploration, sampling and free energy surface reconstruction: using Gaussian processes to improve adaptive potential of mean force methods

Letif Mones, Thomas Stecher, Noam Bernstein, Gábor Csányi



Engineering Laboratory

UNIVERSITY OF CAMBRIDGE

Outline

- Interpolation with Gaussian processes
- Application to free energy reconstruction from MD data
- Exploration and biasing:
  - Metadynamics, Adaptive Biasing Force, Gaussian process
  - Numerical experiments

Function fitting with kernels

Fit a function $f(x)$ based on observations $\mathbf{y} \equiv \{y_i\}$ at $\{x_i\}$

$$f(x) = \sum_{i=1}^{N} \alpha_i k(x_i, x) \qquad \text{e.g. } k(x, x') = \sigma_w^2 e^{-|x-x'|^2/2\sigma^2}$$

Function fitting with kernels

Fit a function $f(x)$ based on observations $\mathbf{y} \equiv \{y_i\}$ at $\{x_i\}$

$$f(x) = \sum_{i=1}^{N} \alpha_i k(x_i, x)$$

e.g. $k(x, x') = \sigma_w^2 e^{-|x-x'|^2/2\sigma^2}$

$$y_j = \sum_{i=1}^{N} \alpha_i k(x_i, x_j)$$

Function fitting with kernels

Fit a function $f(x)$ based on observations $\mathbf{y} \equiv \{y_i\}$ at $\{x_i\}$

$$f(x) = \sum_{i=1}^{N} \alpha_i k(x_i, x)$$

e.g. $k(x, x') = \sigma_w^2 e^{-|x-x'|^2/2\sigma^2}$

$$y_j = \sum_{i=1}^{N} \alpha_i \left(k(x_i, x_j) + \sigma_\nu^2 \delta_{ij}\right)$$

regularised fit:
arbitrary $\sigma, \sigma_w, \sigma_\nu$

Function fitting with kernels

Fit a function $f(x)$ based on observations $\mathbf{y} \equiv \{y_i\}$ at $\{x_i\}$

$$f(x) = \sum_{i=1}^{N} \alpha_i k(x_i, x)$$

e.g. $k(x, x') = \sigma_w^2 e^{-|x-x'|^2/2\sigma^2}$

$$y_j = \sum_{i=1}^{N} \alpha_i \left(k(x_i, x_j) + \sigma_\nu^2 \delta_{ij}\right)$$

regularised fit:
arbitrary σ,σ_w,σ_ν

$$\mathbf{y} = (\mathbf{K} + \sigma_\nu^2 \mathbf{I})\boldsymbol{\alpha}$$

$$\boldsymbol{\alpha} = \mathbf{C}^{-1}\mathbf{y}$$

$$[\mathbf{K}]_{ij} \equiv k(x_i, x_j)$$

$$\mathbf{C} \equiv \mathbf{K} + \sigma_\nu^2 \mathbf{I}$$

$$\mathbf{k} \equiv k(x_i, x)$$

Function fitting with kernels

Fit a function $f(x)$ based on observations $\mathbf{y} \equiv \{y_i\}$ at $\{x_i\}$

$$f(x) = \sum_{i=1}^{N} \alpha_i k(x_i, x)$$

e.g. $k(x, x') = \sigma_w^2 e^{-|x-x'|^2/2\sigma^2}$

$$y_j = \sum_{i=1}^{N} \alpha_i \left(k(x_i, x_j) + \sigma_\nu^2 \delta_{ij}\right)$$

regularised fit:
arbitrary σ,σ_w,σ_ν

$$\mathbf{y} = (\mathbf{K} + \sigma_\nu^2 \mathbf{I})\boldsymbol{\alpha}$$

$$\boldsymbol{\alpha} = \mathbf{C}^{-1}\mathbf{y}$$

$$\boxed{f(x) = \mathbf{k}^T \mathbf{C}^{-1} \mathbf{y}}$$

$$[\mathbf{K}]_{ij} \equiv k(x_i, x_j)$$

$$\mathbf{C} \equiv \mathbf{K} + \sigma_\nu^2 \mathbf{I}$$

$$\mathbf{k} \equiv k(x_i, x)$$

Function fitting with kernels

Fit a function $f(x)$ based on observations $\mathbf{y} \equiv \{y_i\}$ at $\{x_i\}$

$$f(x) = \sum_{i=1}^{N} \alpha_i k(x_i, x)$$

e.g. $k(x, x') = \sigma_w^2 e^{-|x-x'|^2/2\sigma^2}$

$$y_j = \sum_{i=1}^{N} \alpha_i \left(k(x_i, x_j) + \sigma_\nu^2 \delta_{ij} \right)$$

regularised fit:
arbitrary $\sigma, \sigma_w, \sigma_\nu$

$$\mathbf{y} = (\mathbf{K} + \sigma_\nu^2 \mathbf{I})\boldsymbol{\alpha}$$

$$\boldsymbol{\alpha} = \mathbf{C}^{-1}\mathbf{y}$$

$$[\mathbf{K}]_{ij} \equiv k(x_i, x_j)$$

$$\mathbf{C} \equiv \mathbf{K} + \sigma_\nu^2 \mathbf{I}$$

$$\mathbf{k} \equiv k(x_i, x)$$

$$f(x) = \mathbf{k}^T \mathbf{C}^{-1} \mathbf{y}$$

Bayesian function fitting: Gaussian Process Regression

Fit a function $f(x)$ based on observations $\mathbf{y} \equiv \{y_i\}$ at $\{x_i\}$

$$f(x) = \sum_{h=1}^{H} w_h \phi_h(x)$$

basis functions $\phi_h(x)$

$$f_i = \sum_h R_{ih} w_h$$

$$R_{ih} \equiv \phi_h(x_i)$$

$$f(x) = \mathbf{k}^T \mathbf{C}^{-1} \mathbf{y}$$

Bayesian function fitting: Gaussian Process Regression

Fit a function $f(x)$ based on observations $\mathbf{y} \equiv \{y_i\}$ at $\{x_i\}$

$$f(x) = \sum_{h=1}^{H} w_h \phi_h(x)$$

basis functions $\phi_h(x)$

$$f_i = \sum_h R_{ih} w_h$$

$$R_{ih} \equiv \phi_h(x_i)$$

Gaussian prior: $P(\mathbf{w}) = \text{Normal}(0, \sigma_w^2 \mathbf{I})$

Expected function Var

$$P(\mathbf{f}) = \text{Normal}(0, \sigma_w^2 \mathbf{R}\mathbf{R}^T)$$

$$f(x) = \mathbf{k}^T \mathbf{C}^{-1} \mathbf{y}$$

Bayesian function fitting: Gaussian Process Regression

Fit a function $f(x)$ based on observations $\mathbf{y} \equiv \{y_i\}$ at $\{x_i\}$

$$f(x) = \sum_{h=1}^{H} w_h \phi_h(x)$$

basis functions $\phi_h(x)$

$$f_i = \sum_h R_{ih} w_h$$

$$R_{ih} \equiv \phi_h(x_i)$$

Gaussian prior: $P(\mathbf{w}) = \text{Normal}(0, \sigma_w^2 \mathbf{I})$

$$P(\mathbf{f}) = \text{Normal}(0, \sigma_w^2 \mathbf{R}\mathbf{R}^T)$$

Expected function Var

Gaussian likelihood: $\mathbf{y} = \mathbf{f} + \varepsilon$ $\quad \varepsilon \sim \text{Normal}(0, \sigma_\nu^2 \mathbf{I})$

Var of data

$$f(x) = \mathbf{k}^T \mathbf{C}^{-1} \mathbf{y}$$

Bayesian function fitting: Gaussian Process Regression

Fit a function $f(x)$ based on observations $\mathbf{y} \equiv \{y_i\}$ at $\{x_i\}$

$$f(x) = \sum_{h=1}^{H} w_h \phi_h(x)$$

basis functions $\phi_h(x)$

$$f_i = \sum_h R_{ih} w_h$$

$$R_{ih} \equiv \phi_h(x_i)$$

Gaussian prior: $P(\mathbf{w}) = \text{Normal}(0, \sigma_w^2 \mathbf{I})$

$$P(\mathbf{f}) = \text{Normal}(0, \sigma_w^2 \mathbf{R}\mathbf{R}^T)$$

Expected function Var

Gaussian likelihood: $\mathbf{y} = \mathbf{f} + \varepsilon \qquad \varepsilon \sim \text{Normal}(0, \sigma_\nu^2 \mathbf{I})$

Var of data

Joint of observations also Gaussian:

$$P(\mathbf{y}) = \text{Normal}(0, \sigma_w^2 \mathbf{R}\mathbf{R}^T + \sigma_\nu^2 \mathbf{I})$$

$$f(x) = \mathbf{k}^T \mathbf{C}^{-1} \mathbf{y}$$

Bayesian function fitting: Gaussian Process Regression

Fit a function $f(x)$ based on observations $\mathbf{y} \equiv \{y_i\}$ at $\{x_i\}$

$$f(x) = \sum_{h=1}^{H} w_h \phi_h(x)$$

basis functions $\phi_h(x)$

$$f_i = \sum_h R_{ih} w_h$$

$$R_{ih} \equiv \phi_h(x_i)$$

Gaussian prior: $P(\mathbf{w}) = \text{Normal}(0, \sigma_w^2 \mathbf{I})$

Expected function Var

$$P(\mathbf{f}) = \text{Normal}(0, \sigma_w^2 \mathbf{R}\mathbf{R}^T)$$

Var of data

Gaussian likelihood: $\mathbf{y} = \mathbf{f} + \varepsilon \qquad \varepsilon \sim \text{Normal}(0, \sigma_\nu^2 \mathbf{I})$

Joint of observations also Gaussian:

$$P(\mathbf{y}) = \text{Normal}(0, \sigma_w^2 \mathbf{R}\mathbf{R}^T + \sigma_\nu^2 \mathbf{I})$$

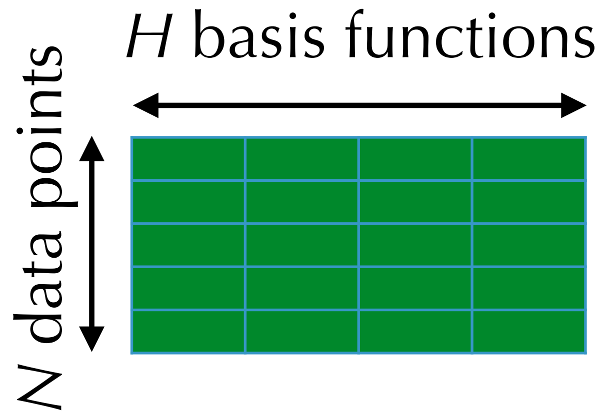
Predict the next observation using the conditional

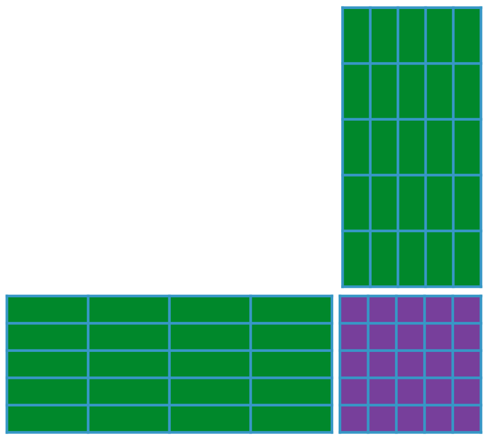
$$P(y_{N+1} | \mathbf{y}_N) = P(y_{N+1}, \mathbf{y}_N) / P(\mathbf{y}_N)$$

$$f(x) = \mathbf{k}^T \mathbf{C}^{-1} \mathbf{y}$$

Limit of ∞ basis functions

$$R_{ih} \equiv \phi_h(x_i)$$





$\mathbf{RR}^{\mathrm{T}}$

Use Gaussian basis functions, take $H \to \infty$, $\mathbf{RR}^{\mathrm{T}}$ is finite

Prior is a Gaussian Process with

$$\mathrm{Cov}[f(x_i), f(x_j)] \propto \exp\left[-(x_i - x_j)^2/\sigma^2\right]$$

Similarly for the observations:

$$\mathrm{Cov}[y(x_i), y(x_j)] \propto \exp\left[-(x_i - x_j)^2/\sigma^2\right] + \sigma_\nu^2 \delta_{ij}$$

$$f(x) = \mathbf{k}^T \mathbf{C}^{-1} \mathbf{y}$$

Gaussian Process connection to linear fit

$$f(x) = \mathbf{k}^T \mathbf{C}^{-1} \mathbf{y}$$

Gaussian Process connection to linear fit

$$P(y_{N+1}|\mathbf{y}_N) = P(y_{N+1}, \mathbf{y}_N)/P(\mathbf{y}_N) \propto P(y_{N+1}, \mathbf{y}_N)$$

$$f(x) = \mathbf{k}^T \mathbf{C}^{-1} \mathbf{y}$$

Gaussian Process connection to linear fit

$$P(y_{N+1}|\mathbf{y}_N) = P(y_{N+1}, \mathbf{y}_N)/P(\mathbf{y}_N) \propto P(y_{N+1}, \mathbf{y}_N)$$

$$P(\mathbf{y}_N) \propto \text{Normal}(0, \mathbf{C}_N) \propto \exp(-\mathbf{y}_N^T \mathbf{C}_N^{-1} \mathbf{y}_N)$$

$$f(x) = \mathbf{k}^T \mathbf{C}^{-1} \mathbf{y}$$

Gaussian Process connection to linear fit

$$P(y_{N+1}|\mathbf{y}_N) = P(y_{N+1}, \mathbf{y}_N)/P(\mathbf{y}_N) \propto P(y_{N+1}, \mathbf{y}_N)$$

$$P(\mathbf{y}_N) \propto \mathrm{Normal}(0, \mathbf{C}_N) \propto \exp(-\mathbf{y}_N^T \mathbf{C}_N^{-1} \mathbf{y}_N)$$

$$P(y_{N+1}, \mathbf{y}_N) \propto \mathrm{Normal}(0, \mathbf{C}^{N+1}) \propto \exp(-[\mathbf{y}_N^T \, y_{N+1}] \mathbf{C}_{N+1}^{-1} [\mathbf{y}_N \, y_{N+1}])$$

$$f(x) = \mathbf{k}^T \mathbf{C}^{-1} \mathbf{y}$$

Gaussian Process connection to linear fit

$$P(y_{N+1}|\mathbf{y}_N) = P(y_{N+1}, \mathbf{y}_N)/P(\mathbf{y}_N) \propto P(y_{N+1}, \mathbf{y}_N)$$

$$P(\mathbf{y}_N) \propto \text{Normal}(0, \mathbf{C}_N) \propto \exp(-\mathbf{y}_N^T \mathbf{C}_N^{-1} \mathbf{y}_N)$$

$$P(y_{N+1}, \mathbf{y}_N) \propto \text{Normal}(0, \mathbf{C}^{N+1}) \propto \exp(-[\mathbf{y}_N^T \, y_{N+1}] \mathbf{C}_{N+1}^{-1} [\mathbf{y}_N \, y_{N+1}])$$

$$\mathbf{C}_{N+1} \equiv \begin{bmatrix} \begin{bmatrix} \mathbf{C}_N \end{bmatrix} & \begin{bmatrix} \mathbf{k} \end{bmatrix} \\ \begin{bmatrix} \mathbf{k}^T \end{bmatrix} & [\kappa] \end{bmatrix} \qquad \mathbf{C}_{N+1}^{-1} \equiv \begin{bmatrix} \begin{bmatrix} \mathbf{M} \end{bmatrix} & \begin{bmatrix} \mathbf{m} \end{bmatrix} \\ \begin{bmatrix} \mathbf{m}^T \end{bmatrix} & [\mu] \end{bmatrix}$$

$$\begin{aligned} \mu &= \left(\kappa - \mathbf{k}^T \mathbf{C}_N^{-1} \mathbf{k}\right)^{-1} \\ \mathbf{m} &= -\mu \, \mathbf{C}_N^{-1} \mathbf{k} \\ \mathbf{M} &= \mathbf{C}_N^{-1} + \frac{1}{\mu} \mathbf{m}\mathbf{m}^T \end{aligned}$$

$$[\mathbf{k}]_i \equiv C(x_i, x_{N+1})$$

$$\kappa \equiv C(x_{N+1}, x_{N+1})$$

$$f(x) = \mathbf{k}^T \mathbf{C}^{-1} \mathbf{y}$$

Gaussian Process connection to linear fit

$$P(y_{N+1}|\mathbf{y}_N) = P(y_{N+1}, \mathbf{y}_N)/P(\mathbf{y}_N) \propto P(y_{N+1}, \mathbf{y}_N)$$

$$P(\mathbf{y}_N) \propto \text{Normal}(0, \mathbf{C}_N) \propto \exp(-\mathbf{y}_N^T \mathbf{C}_N^{-1} \mathbf{y}_N)$$

$$P(y_{N+1}, \mathbf{y}_N) \propto \text{Normal}(0, \mathbf{C}^{N+1}) \propto \exp(-[\mathbf{y}_N^T \, y_{N+1}] \mathbf{C}_{N+1}^{-1} [\mathbf{y}_N \, y_{N+1}])$$

$$\mathbf{C}_{N+1} \equiv \begin{bmatrix} \begin{bmatrix} \mathbf{C}_N \end{bmatrix} & \begin{bmatrix} \mathbf{k} \end{bmatrix} \\ \begin{bmatrix} \mathbf{k}^T \end{bmatrix} & [\kappa] \end{bmatrix} \qquad \mathbf{C}_{N+1}^{-1} \equiv \begin{bmatrix} \begin{bmatrix} \mathbf{M} \end{bmatrix} & \begin{bmatrix} \mathbf{m} \end{bmatrix} \\ \begin{bmatrix} \mathbf{m}^T \end{bmatrix} & [\mu] \end{bmatrix}$$

$$\begin{aligned} \mu &= \left(\kappa - \mathbf{k}^T \mathbf{C}_N^{-1} \mathbf{k}\right)^{-1} \\ \mathbf{m} &= -\mu \, \mathbf{C}_N^{-1} \mathbf{k} \\ \mathbf{M} &= \mathbf{C}_N^{-1} + \frac{1}{\mu} \mathbf{m}\mathbf{m}^T \end{aligned}$$

$$[\mathbf{k}]_i \equiv C(x_i, x_{N+1})$$
$$\kappa \equiv C(x_{N+1}, x_{N+1})$$

$$\begin{aligned} [\mathbf{y}_N^T \, y_{N+1}] \mathbf{C}_{N+1}^{-1} [\mathbf{y}_N \, y_{N+1}] &= \mathbf{y}_N^T \mathbf{M} \mathbf{y} + 2\mathbf{y}^T \mathbf{m} y_{N+1} + \mu y_{N+1}^2 \\ &= \mu (y_{N+1} + \mathbf{y}^T \mathbf{m}/\mu)^2 + \ldots \end{aligned}$$

$$f(x) = \mathbf{k}^T \mathbf{C}^{-1} \mathbf{y}$$

Gaussian Process connection to linear fit

$$P(y_{N+1}|\mathbf{y}_N) = P(y_{N+1}, \mathbf{y}_N)/P(\mathbf{y}_N) \propto P(y_{N+1}, \mathbf{y}_N)$$

$$P(\mathbf{y}_N) \propto \text{Normal}(0, \mathbf{C}_N) \propto \exp(-\mathbf{y}_N^T \mathbf{C}_N^{-1} \mathbf{y}_N)$$

$$P(y_{N+1}, \mathbf{y}_N) \propto \text{Normal}(0, \mathbf{C}^{N+1}) \propto \exp(-[\mathbf{y}_N^T \, y_{N+1}]\mathbf{C}_{N+1}^{-1}[\mathbf{y}_N \, y_{N+1}])$$

$$\mathbf{C}_{N+1} \equiv \begin{bmatrix} \begin{bmatrix} \mathbf{C}_N \end{bmatrix} & \begin{bmatrix} \mathbf{k} \end{bmatrix} \\ [\; \mathbf{k}^T \;] & [\kappa] \end{bmatrix} \quad \mathbf{C}_{N+1}^{-1} \equiv \begin{bmatrix} \begin{bmatrix} \mathbf{M} \end{bmatrix} & \begin{bmatrix} \mathbf{m} \end{bmatrix} \\ [\; \mathbf{m}^T \;] & [\mu] \end{bmatrix}$$

$$[\mathbf{k}]_i \equiv C(x_i, x_{N+1})$$

$$\kappa \equiv C(x_{N+1}, x_{N+1})$$

$$\begin{aligned} \mu &= \left(\kappa - \mathbf{k}^T \mathbf{C}_N^{-1} \mathbf{k}\right)^{-1} \\ \mathbf{m} &= -\mu \, \mathbf{C}_N^{-1} \mathbf{k} \\ \mathbf{M} &= \mathbf{C}_N^{-1} + \frac{1}{\mu} \mathbf{m}\mathbf{m}^T \end{aligned}$$

$$\begin{aligned} [\mathbf{y}_N^T \, y_{N+1}]\mathbf{C}_{N+1}^{-1}[\mathbf{y}_N \, y_{N+1}] &= \mathbf{y}_N^T \mathbf{M}\mathbf{y} + 2\mathbf{y}^T \mathbf{m} y_{N+1} + \mu y_{N+1}^2 \\ &= \mu(y_{N+1} + \mathbf{y}^T \mathbf{m}/\mu)^2 + \ldots \end{aligned}$$

$$P(y_{N+1}|\mathbf{y}_N) \propto \exp(-(y_{N+1} - \bar{y})^2 / 2\hat{\sigma}^2)$$

$$\bar{y} = \mathbf{k}^T \mathbf{C}_N^{-1} \mathbf{y}_N$$

$$\hat{\sigma} = \kappa - \mathbf{k}^T \mathbf{C}_N^{-1} \mathbf{k}$$

$$f(x) = \mathbf{k}^T \mathbf{C}^{-1} \mathbf{y}$$

Gaussian Process connection to linear fit

$$P(y_{N+1}|\mathbf{y}_N) = P(y_{N+1}, \mathbf{y}_N)/P(\mathbf{y}_N) \propto P(y_{N+1}, \mathbf{y}_N)$$

$$P(\mathbf{y}_N) \propto \mathrm{Normal}(0, \mathbf{C}_N) \propto \exp(-\mathbf{y}_N^T \mathbf{C}_N^{-1} \mathbf{y}_N)$$

$$P(y_{N+1}, \mathbf{y}_N) \propto \mathrm{Normal}(0, \mathbf{C}^{N+1}) \propto \exp(-[\mathbf{y}_N^T\, y_{N+1}]\mathbf{C}_{N+1}^{-1}[\mathbf{y}_N\, y_{N+1}])$$

$$\mathbf{C}_{N+1} \equiv \begin{bmatrix} \begin{bmatrix} \mathbf{C}_N \end{bmatrix} & \begin{bmatrix} \mathbf{k} \end{bmatrix} \\ \begin{bmatrix} \mathbf{k}^T \end{bmatrix} & [\kappa] \end{bmatrix} \qquad \mathbf{C}_{N+1}^{-1} \equiv \begin{bmatrix} \begin{bmatrix} \mathbf{M} \end{bmatrix} & \begin{bmatrix} \mathbf{m} \end{bmatrix} \\ \begin{bmatrix} \mathbf{m}^T \end{bmatrix} & [\mu] \end{bmatrix}$$

$$\begin{aligned} \mu &= \left(\kappa - \mathbf{k}^T \mathbf{C}_N^{-1} \mathbf{k}\right)^{-1} \\ \mathbf{m} &= -\mu\, \mathbf{C}_N^{-1} \mathbf{k} \\ \mathbf{M} &= \mathbf{C}_N^{-1} + \frac{1}{\mu} \mathbf{m}\mathbf{m}^T \end{aligned}$$

$$[\mathbf{k}]_i \equiv C(x_i, x_{N+1})$$
$$\kappa \equiv C(x_{N+1}, x_{N+1})$$

$$\begin{aligned} [\mathbf{y}_N^T\, y_{N+1}]\mathbf{C}_{N+1}^{-1}[\mathbf{y}_N\, y_{N+1}] &= \mathbf{y}_N^T \mathbf{M} \mathbf{y} + 2\mathbf{y}^T \mathbf{m} y_{N+1} + \mu y_{N+1}^2 \\ &= \mu(y_{N+1} + \mathbf{y}^T \mathbf{m}/\mu)^2 + \ldots \end{aligned}$$

$$P(y_{N+1}|\mathbf{y}_N) \propto \exp(-(y_{N+1} - \bar{y})^2/2\hat{\sigma}^2)$$

$$\bar{y} = \mathbf{k}^T \mathbf{C}_N^{-1} \mathbf{y}_N$$

$$\hat{\sigma} = \kappa - \mathbf{k}^T \mathbf{C}_N^{-1} \mathbf{k}$$

$$\arg\max_{y_{N+1}} P(y_{N+1}|\mathbf{y}_N) = \bar{y} = \mathbf{k}^T \mathbf{C}_N^{-1} \mathbf{y}_N$$

$$f(x) = \mathbf{k}^T \mathbf{C}^{-1} \mathbf{y}$$

Gaussian Process Regression summary

- Covariance:

$$K(x_i, x_j) = \exp(-(x_i - x_j)^2/2\sigma^2)$$

Prior assumption

$$f(x) = \arg\max_f P(f|\text{data}) = \sum_i \alpha_i K(x, x_{(i)})$$

Maximum of posterior

$$\alpha = \mathbf{C}^{-1}\mathbf{y} \equiv (\sigma_w^2 \mathbf{K} + \sigma_\nu^2 \mathbf{I})^{-1}\mathbf{y}$$

$$C_{ii'} = \sigma_w^2 K(x_i, x_{i'}) + \sigma_\nu^2 \delta_{ii'}$$

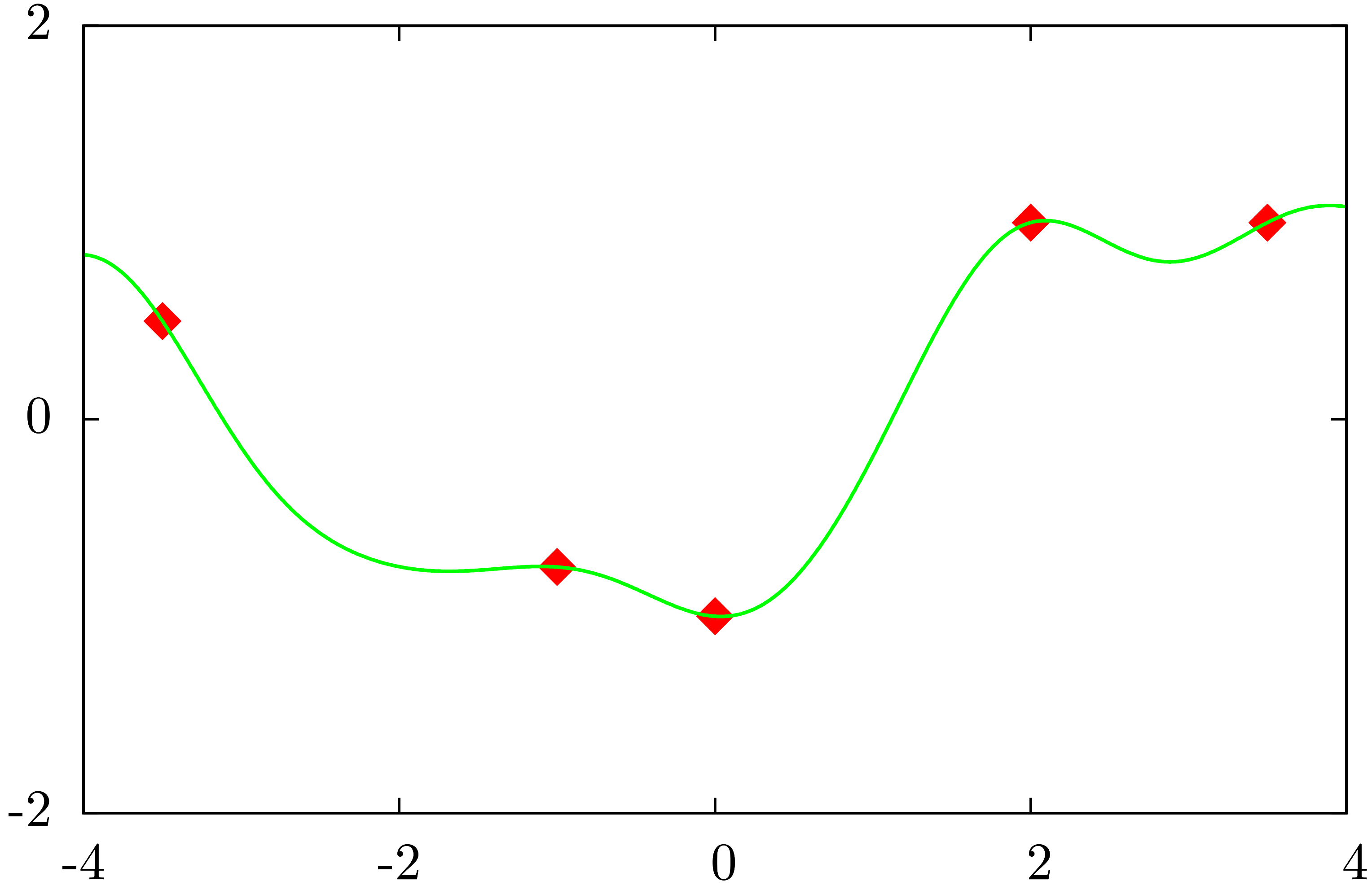
- Meaningful hyper-parameters:
  - σ: smoothness (x-scale) of f
  - σ_w: y-scale of f
  - σ_ν: variance (noise) of input data
- This is *not an optimised fit*, but a closed form estimate!

Mean of posterior distribution

$$\bar{y}(x) = \mathbf{k}^T(x)(\mathbf{K} + \sigma_\nu \mathbf{I})^{-1}\mathbf{y}$$

$$\mathrm{cov}(y(x)) = k(x,x) - \mathbf{k}^T(x)(\mathbf{K} + \sigma_\nu \mathbf{I})^{-1}\mathbf{k}(x)$$

$$[\mathbf{k}(x)]_i = k(x, x_i) \quad K_{ij} = k(x_i, x_j) = e^{-|x_i - x_j'|^2/\sigma}$$

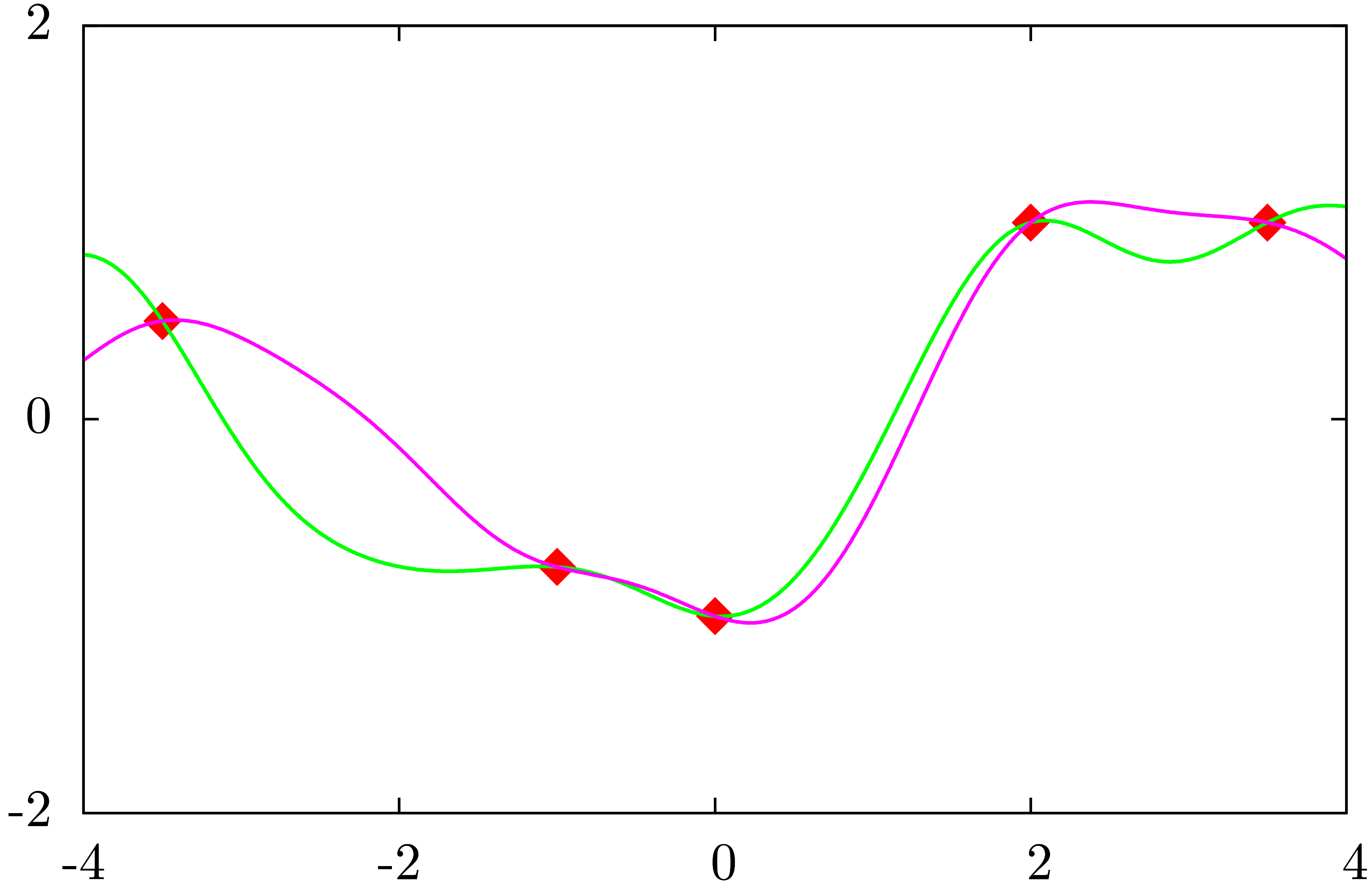


Mean of posterior distribution

$$\bar{y}(x) = \mathbf{k}^T(x)(\mathbf{K} + \sigma_\nu \mathbf{I})^{-1}\mathbf{y}$$

$$\mathrm{cov}(y(x)) = k(x,x) - \mathbf{k}^T(x)(\mathbf{K} + \sigma_\nu \mathbf{I})^{-1}\mathbf{k}(x)$$

$$[\mathbf{k}(x)]_i = k(x, x_i) \quad K_{ij} = k(x_i, x_j) = e^{-|x_i - x_j'|^2/\sigma}$$

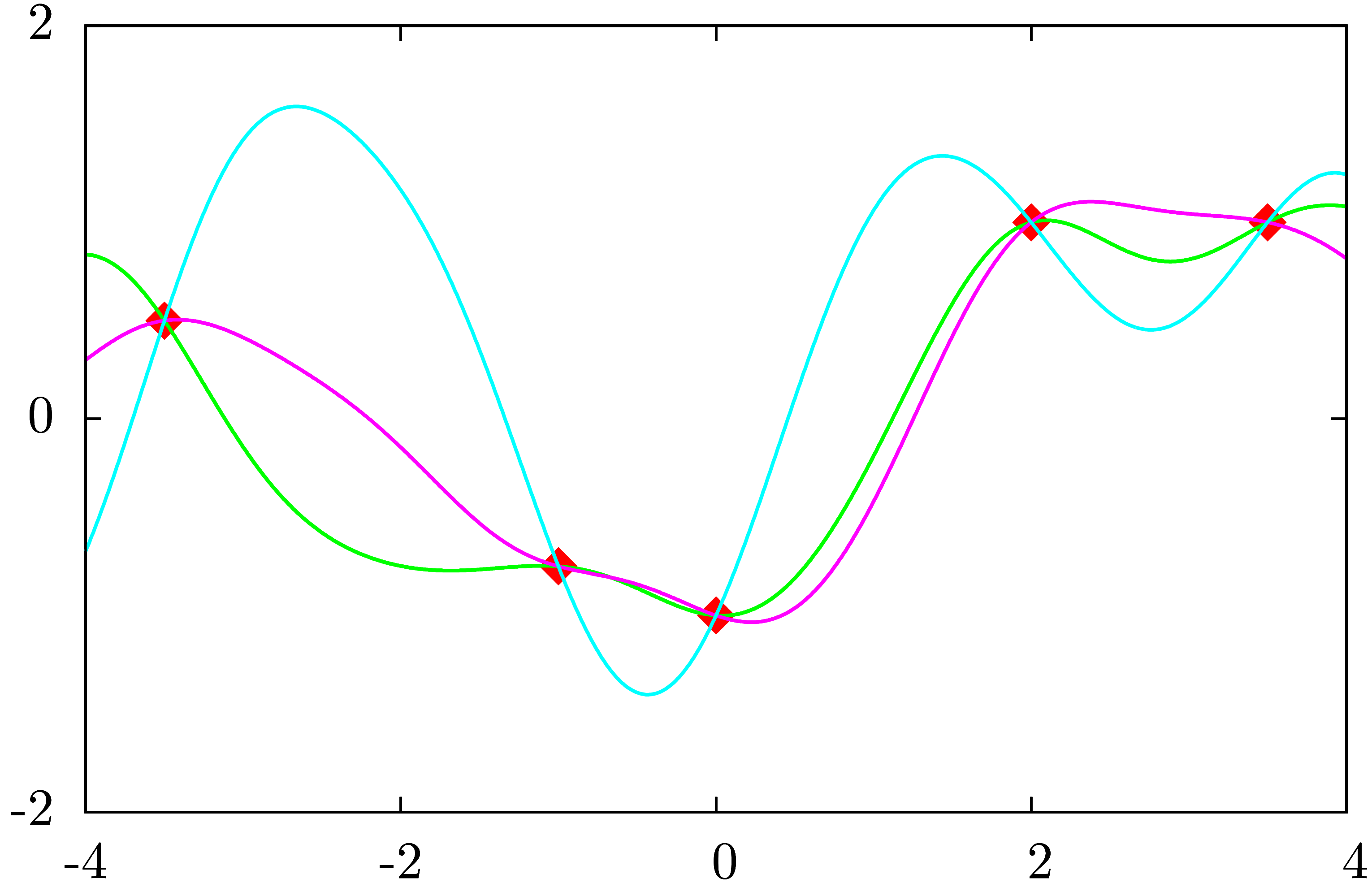


Mean of posterior distribution

$$\bar{y}(x) = \mathbf{k}^T(x)(\mathbf{K} + \sigma_\nu \mathbf{I})^{-1}\mathbf{y}$$

$$\mathrm{cov}(y(x)) = k(x,x) - \mathbf{k}^T(x)(\mathbf{K} + \sigma_\nu \mathbf{I})^{-1}\mathbf{k}(x)$$

$$[\mathbf{k}(x)]_i = k(x, x_i) \quad K_{ij} = k(x_i, x_j) = e^{-|x_i - x_j'|^2/\sigma}$$

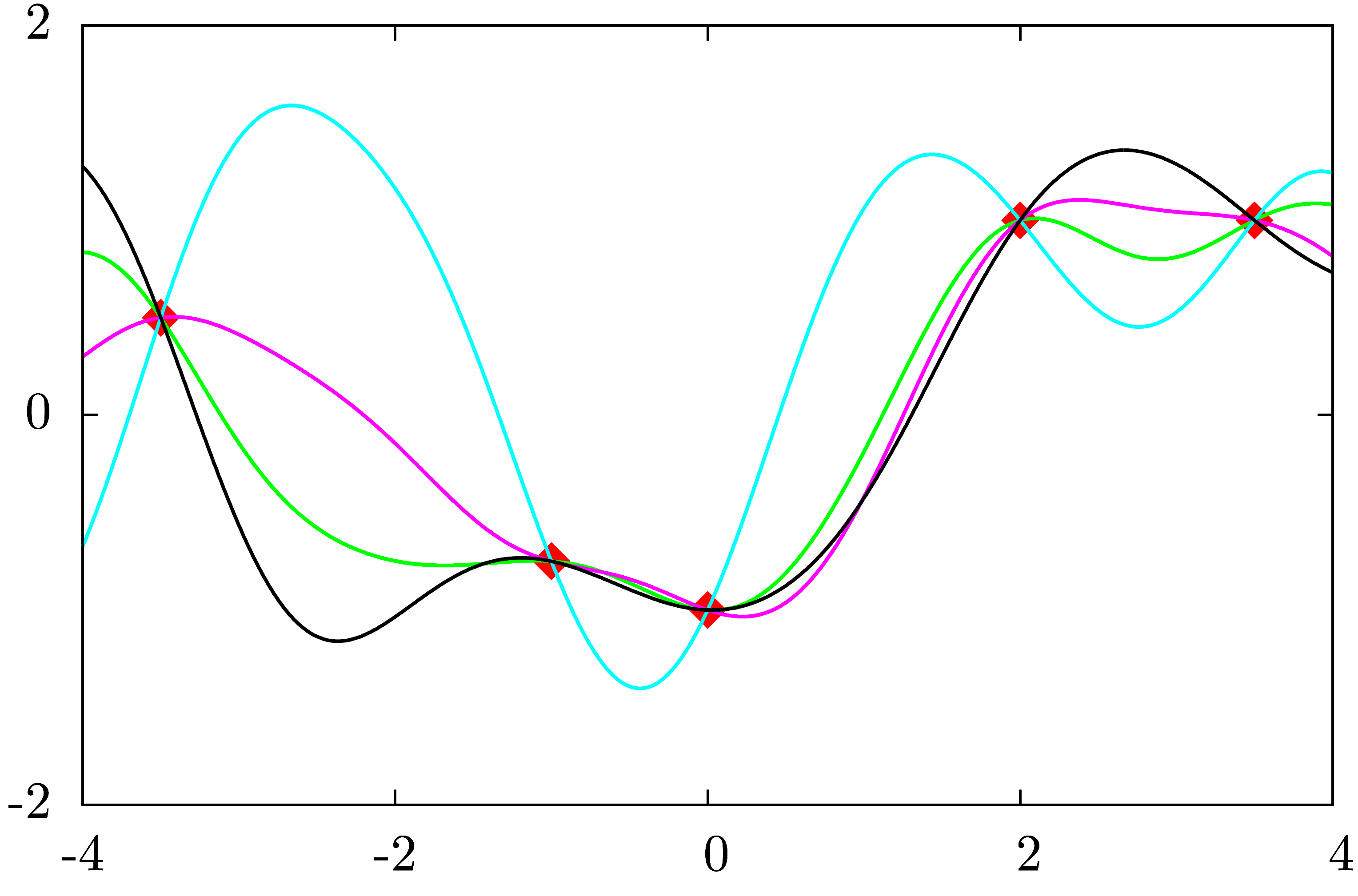


Mean of posterior distribution

$$\bar{y}(x) = \mathbf{k}^T(x)(\mathbf{K} + \sigma_\nu \mathbf{I})^{-1}\mathbf{y}$$

$$\mathrm{cov}(y(x)) = k(x,x) - \mathbf{k}^T(x)(\mathbf{K} + \sigma_\nu \mathbf{I})^{-1}\mathbf{k}(x)$$

$$[\mathbf{k}(x)]_i = k(x, x_i) \quad K_{ij} = k(x_i, x_j) = e^{-|x_i - x'_j|^2/\sigma}$$

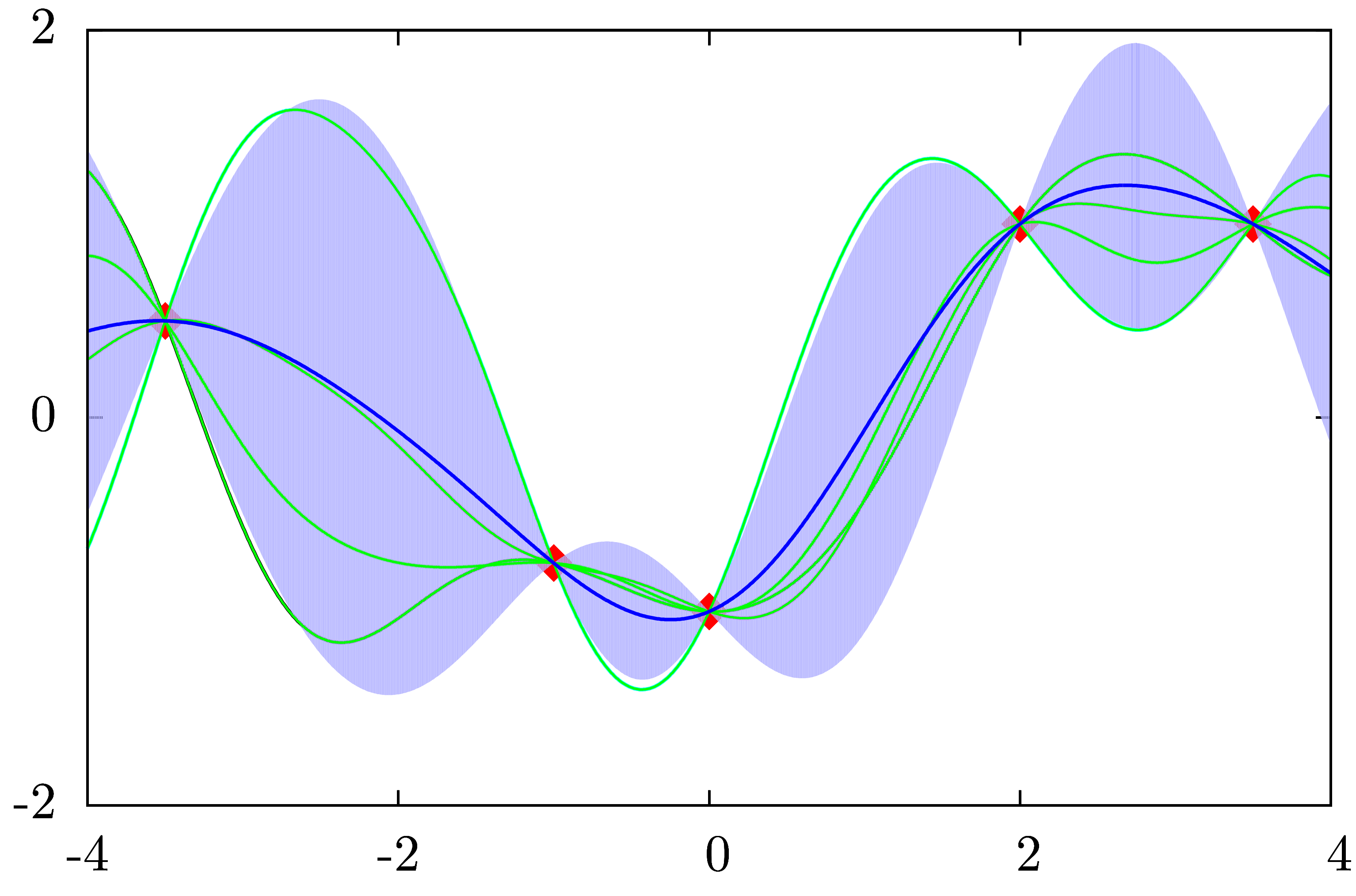


Mean of posterior distribution

$$\bar{y}(x) = \mathbf{k}^T(x)(\mathbf{K} + \sigma_\nu \mathbf{I})^{-1}\mathbf{y}$$

$$\mathrm{cov}(y(x)) = k(x,x) - \mathbf{k}^T(x)(\mathbf{K} + \sigma_\nu \mathbf{I})^{-1}\mathbf{k}(x)$$

$$[\mathbf{k}(x)]_i = k(x, x_i) \quad K_{ij} = k(x_i, x_j) = e^{-|x_i - x_j'|^2/\sigma}$$



Generalised input data

- Often direct measurements of $\mathbf{y}_i$ not available
- Consider any linear operator $L(\mathbf{y})$
  Compute $P(y_{N+1}|L(\mathbf{y}))$

  L can be built using Σ , $\partial/\partial x$ etc

- Basis functions adapt to input data, e.g. $\partial k(x, x')/\partial x$ for derivative data
- Can deal with gradient data, sums, etc.

History of free energy reconstruction

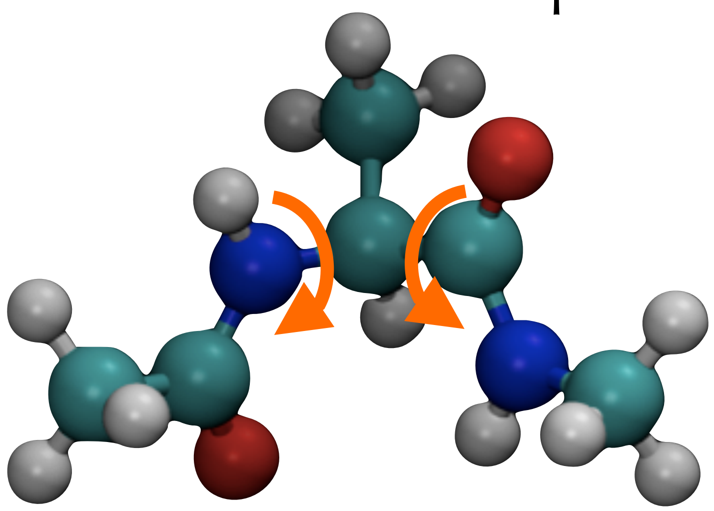
- 1989: Weighted histogram analysis (WHAM): direct estimation of P from biased histogram with overlaps
- 2005: Umbrella Integration: obtain gradient of log P by computing mean of biased distribution
- 2008: Least-squares fit of Gaussians to log P using gradients in multiple dimensions
- 2009: Gaussian mixture model of P from histogram
- 2011: WHAM = maximum likelihood estimate of unknown shifts
- 2013: Spline fit of log P using MLE

Free energy: marginalisation

Boltzmann / Gibbs measure at $T\,(=1/k\beta)$ on configuration space $\{q\}$

$$P(q) = \frac{1}{Z} e^{-\beta V(q)} \qquad Z = \int dq\, e^{-\beta V(q)}$$

We are only interested in some low-dimensional aspect of the system:



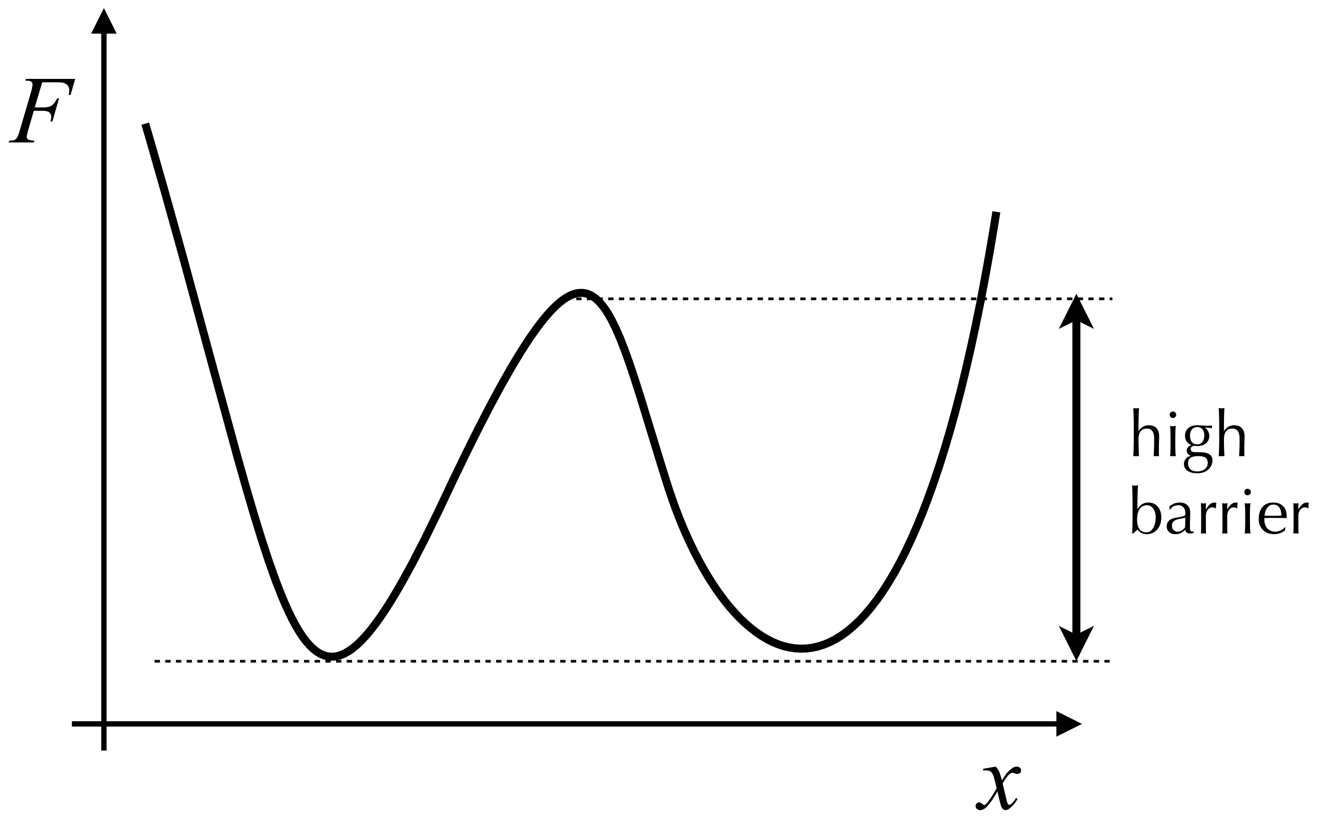
Collective variables $x = s(q)$ e.g. torsion angles of polymer chain

$$P(x) = \frac{1}{Z} \int dq\, e^{-\beta V(q)} \delta(s(q) - x) = e^{-\beta F(x)}$$

Free energy: $$F(x) = -\frac{1}{\beta} \ln P(x)$$

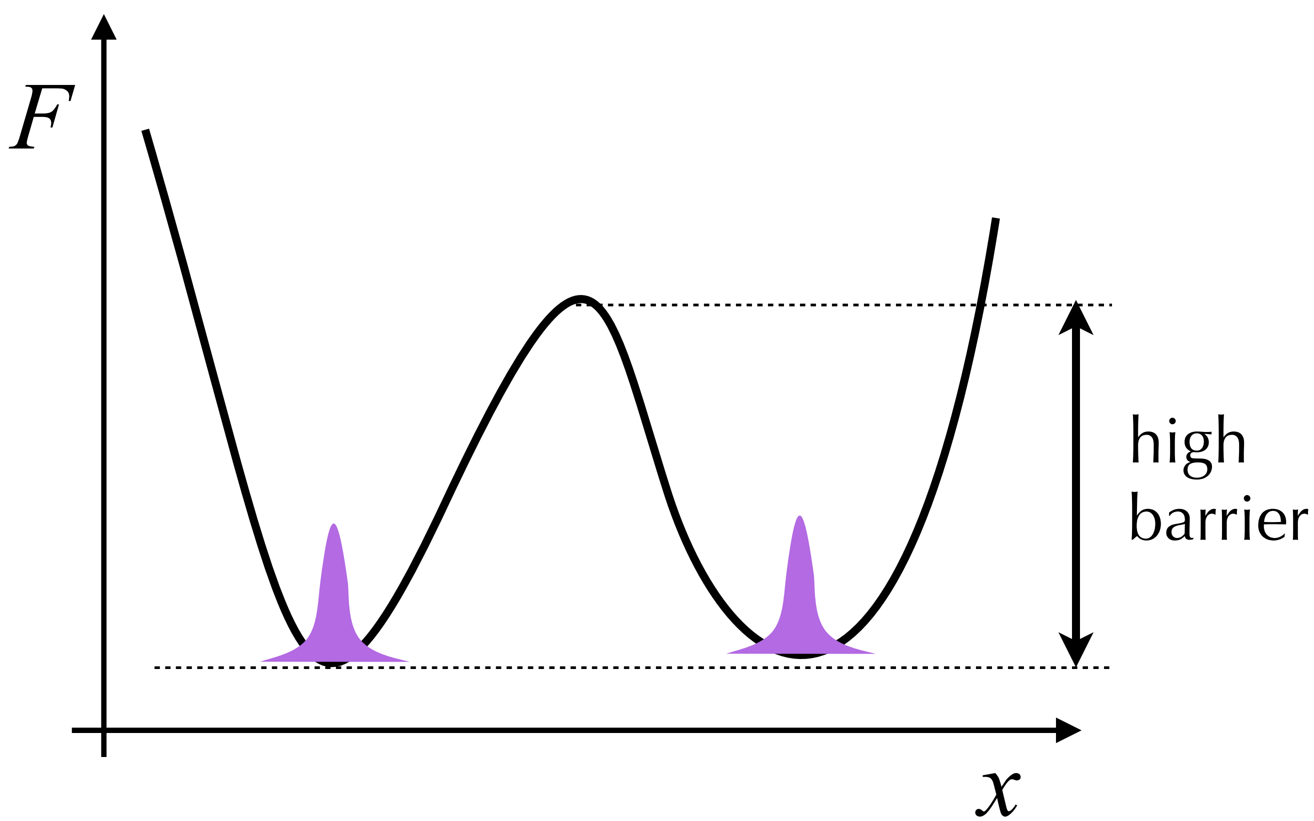
Two problems

- Where (in x) is interesting ? *Exploration*
- Bias the simulation – *Reconstruction*
  - Umbrella sampling: collect histograms of biased simulations
  - Each bias introduces an unknown normalisation constant (shift in *F*)



Two problems

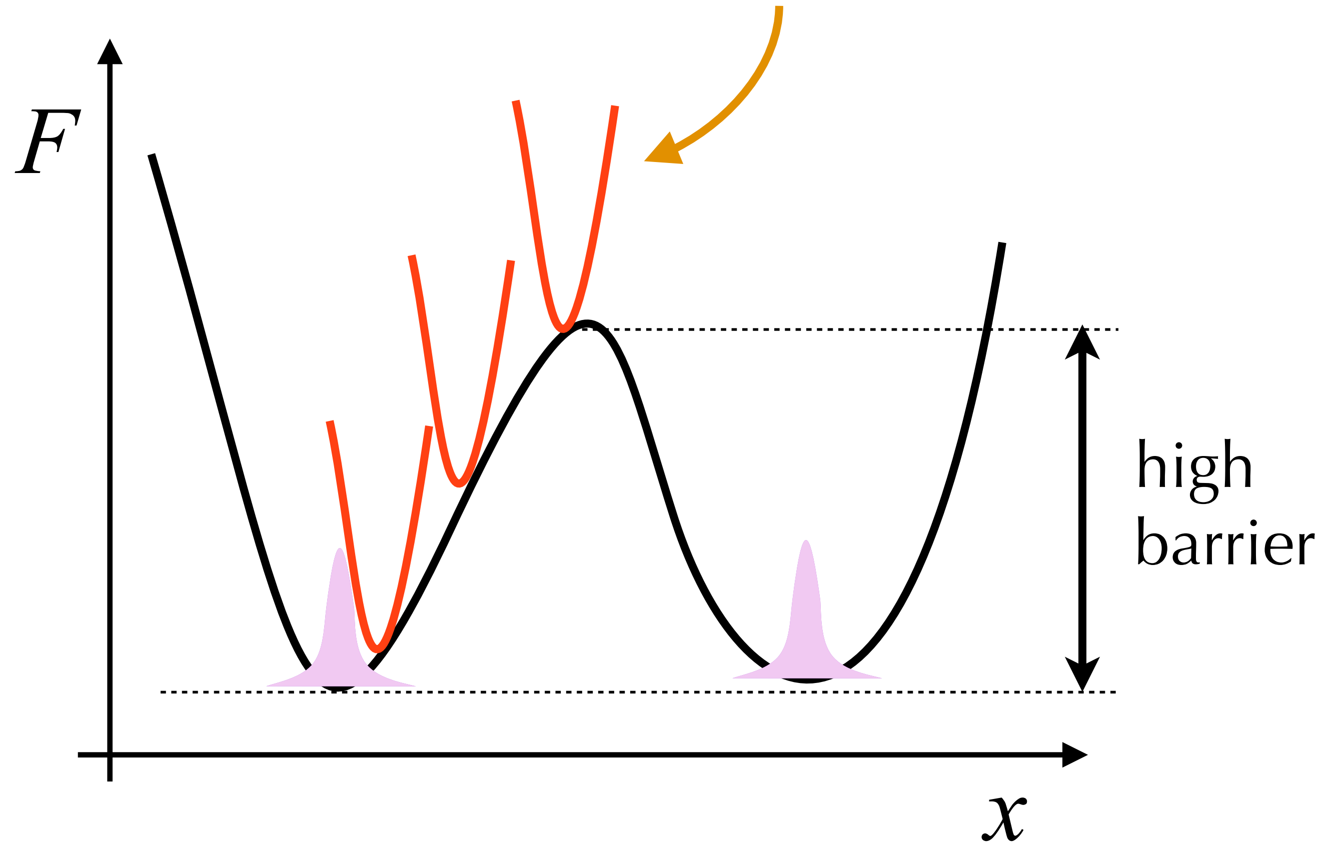
- Where (in x) is interesting ? *Exploration*
- Bias the simulation – *Reconstruction*
  - Umbrella sampling: collect histograms of biased simulations
  - Each bias introduces an unknown normalisation constant (shift in F)



Two problems

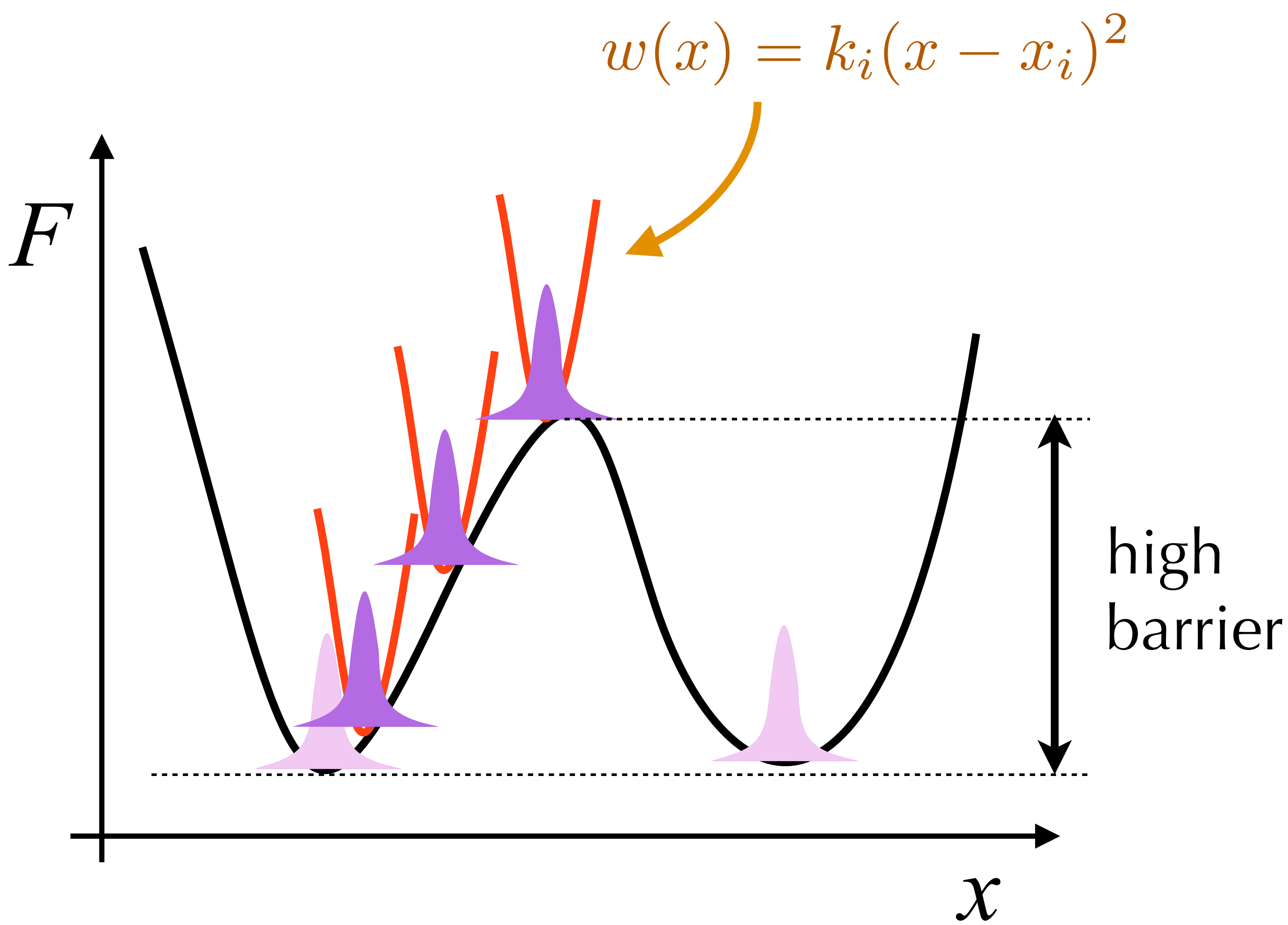
- Where (in x) is interesting ? *Exploration*
- Bias the simulation – *Reconstruction*
  - Umbrella sampling: collect histograms of biased simulations
  - Each bias introduces an unknown normalisation constant (shift in F)

$$w(x) = k_i(x - x_i)^2$$



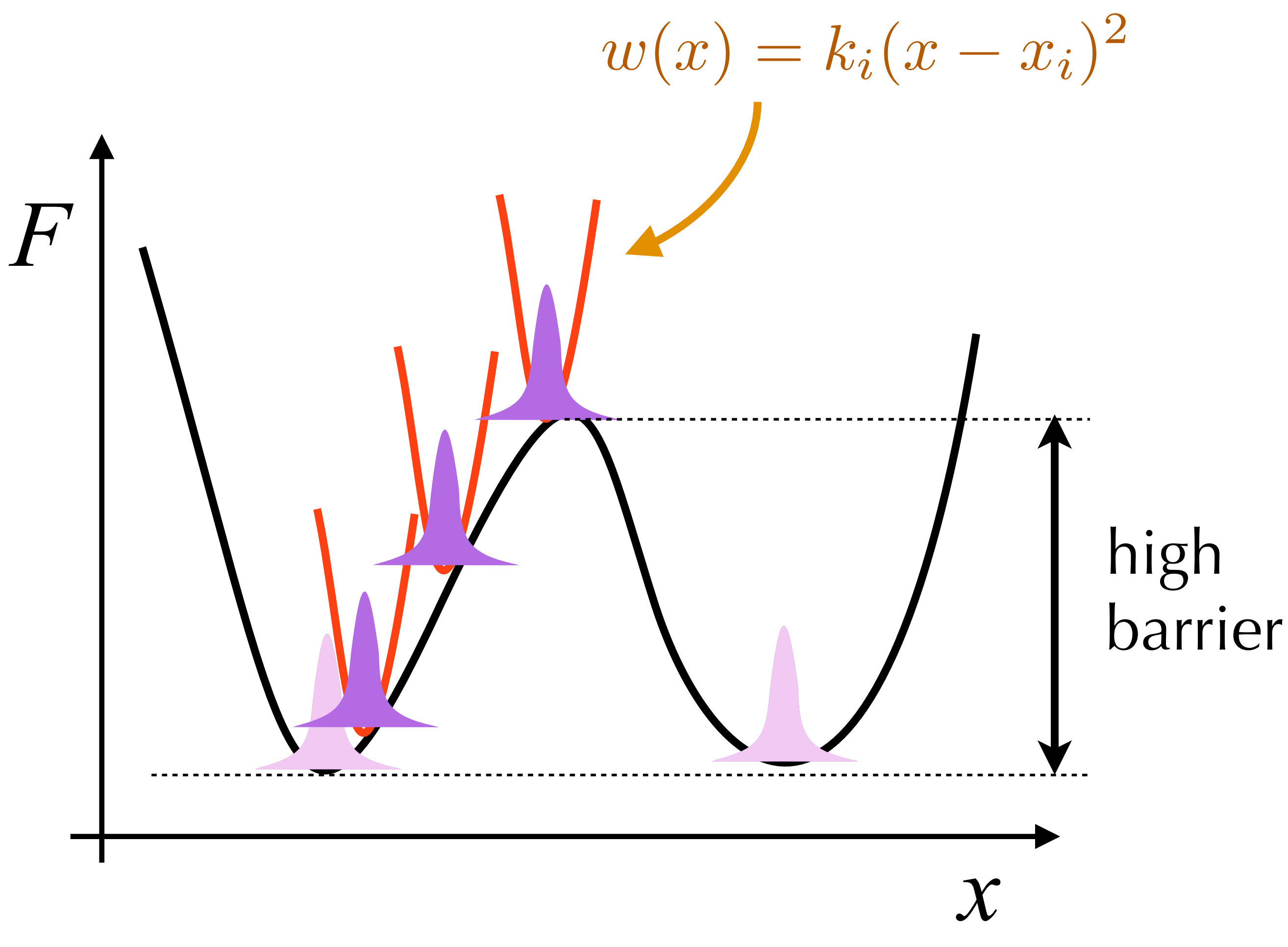
Two problems

- Where (in x) is interesting ? *Exploration*
- Bias the simulation – *Reconstruction*
    - Umbrella sampling: collect histograms of biased simulations
    - Each bias introduces an unknown normalisation constant (shift in F)



Two problems

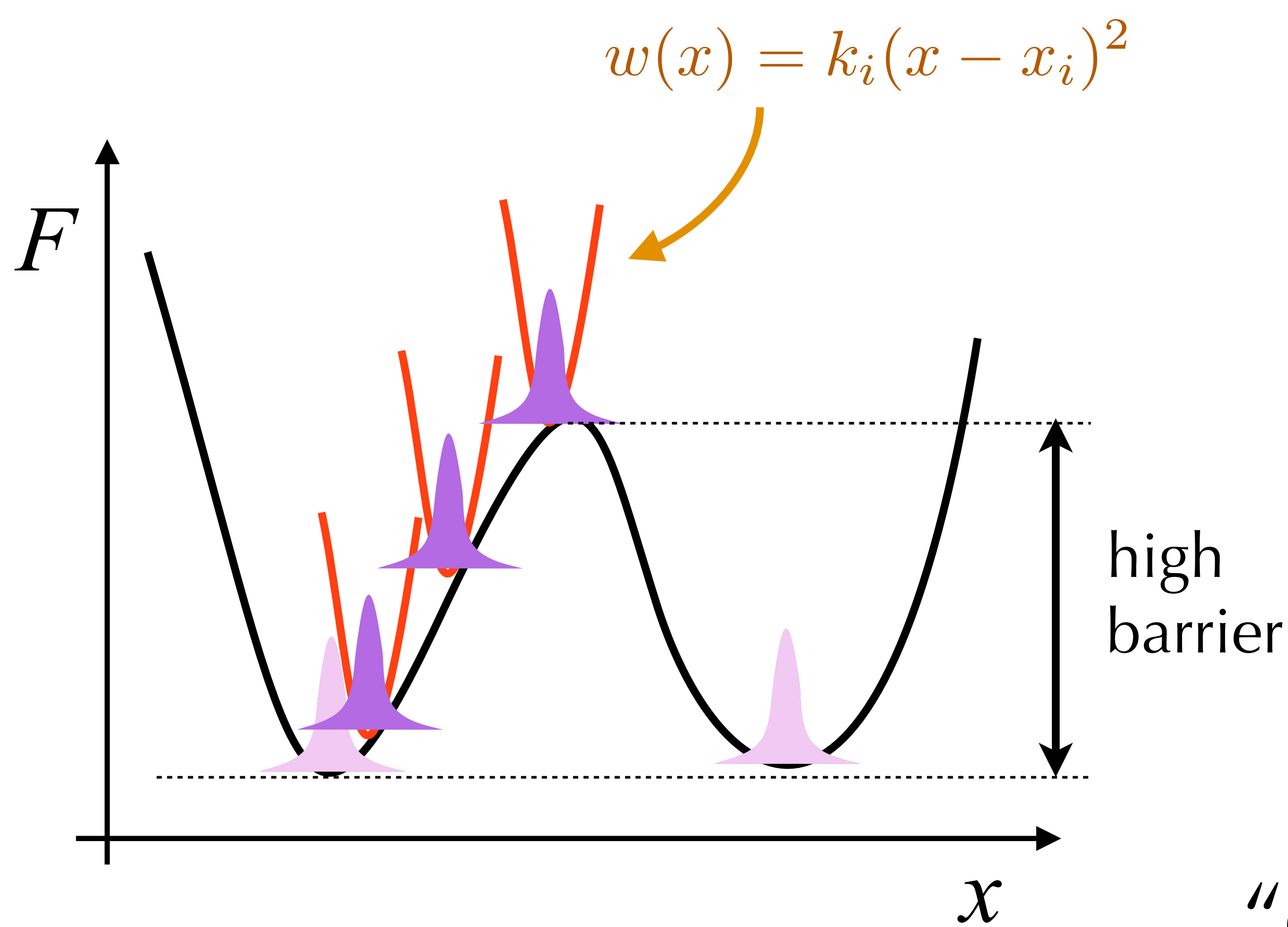
- Where (in x) is interesting ? *Exploration*
- Bias the simulation – *Reconstruction*
  - Umbrella sampling: collect histograms of biased simulations
  - Each bias introduces an unknown normalisation constant (shift in F)



$$F(x) = F^b(x) - w(x) + C$$

Two problems

- Where (in x) is interesting ? *Exploration*
- Bias the simulation – *Reconstruction*
  - Umbrella sampling: collect histograms of biased simulations
  - Each bias introduces an unknown normalisation constant (shift in F)



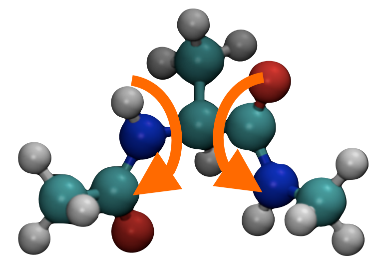
$$F(x) = F^b(x) - w(x) + C$$

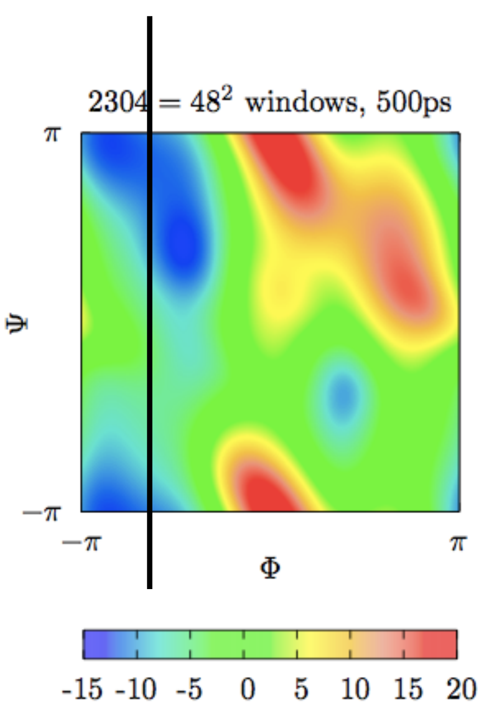
Problem: C varies from place to place

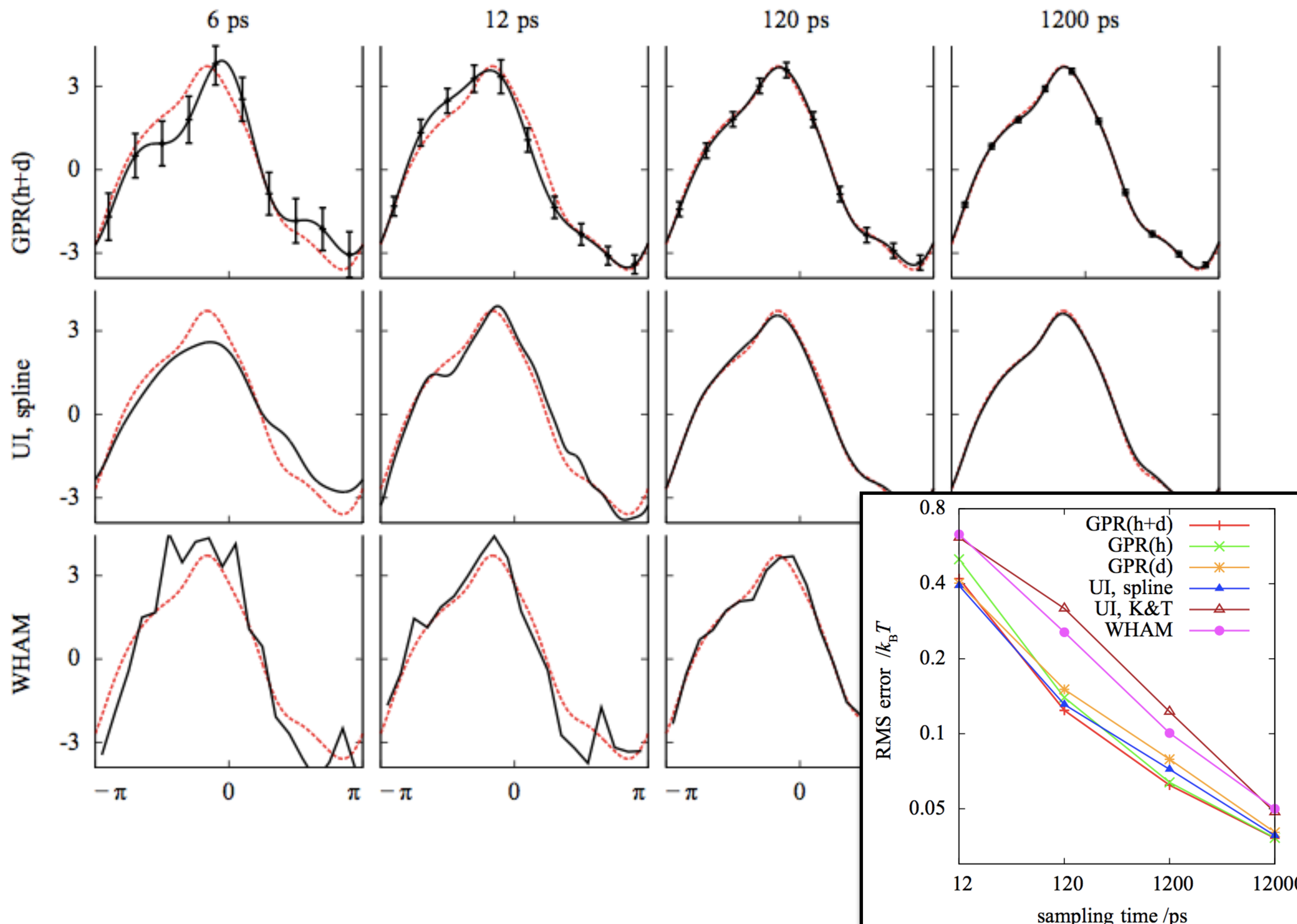
$$\left.\frac{\partial F(x)}{\partial x}\right|_{\bar{x}_i} \approx k_i(\bar{x}_i - x_i)$$

"deviation of mean position"

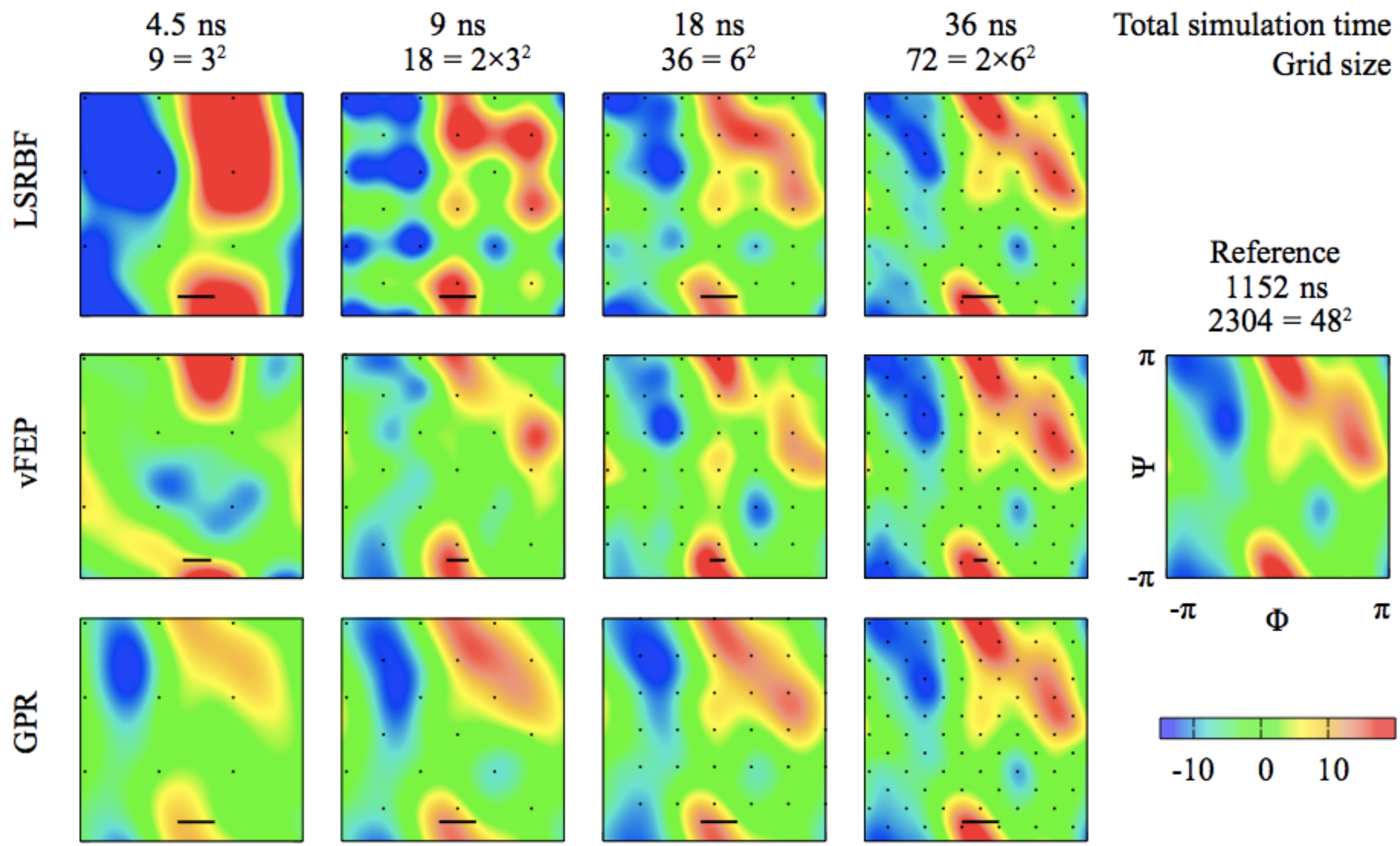
GPR 1D example



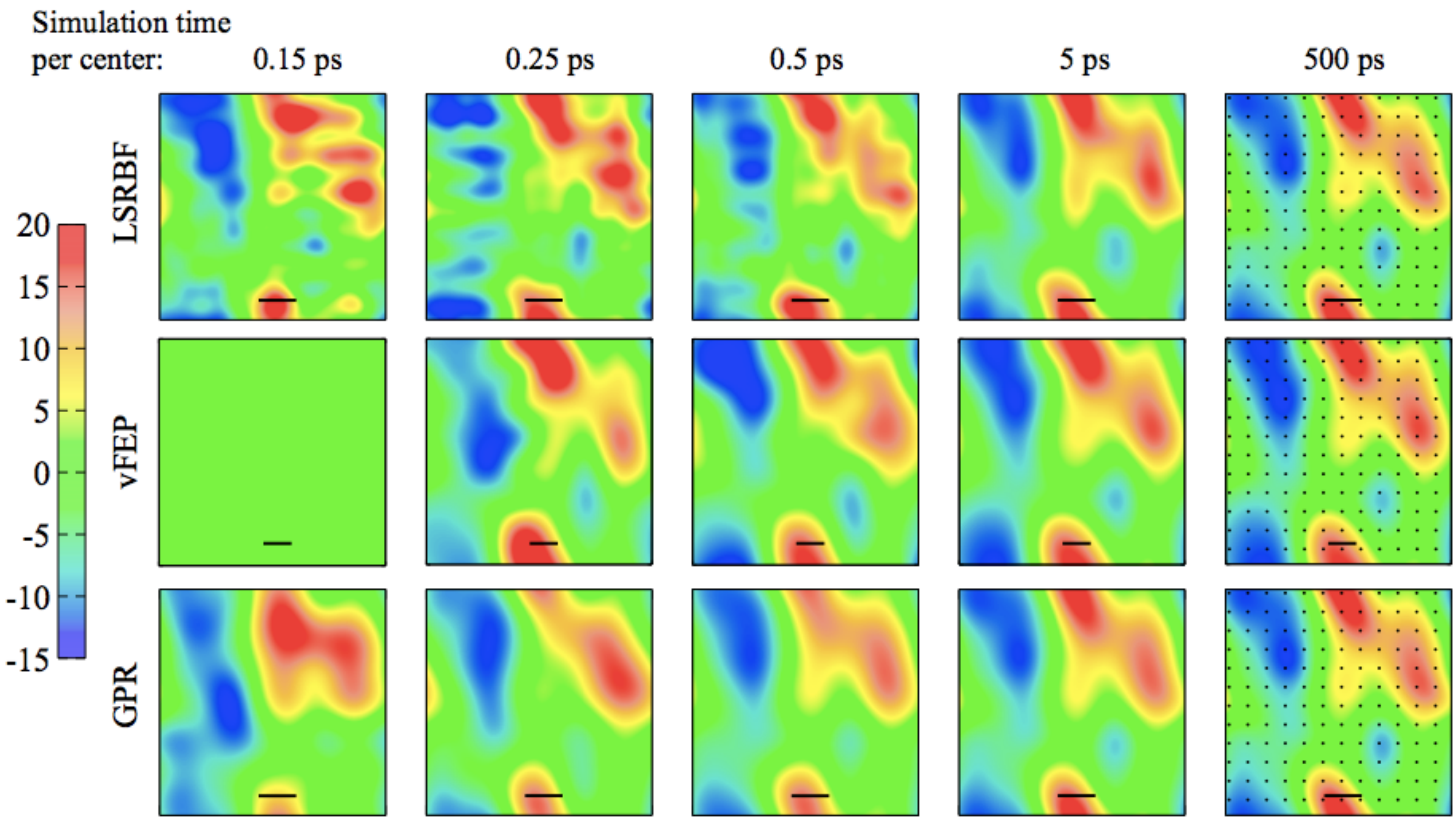




2D: effect of grid size

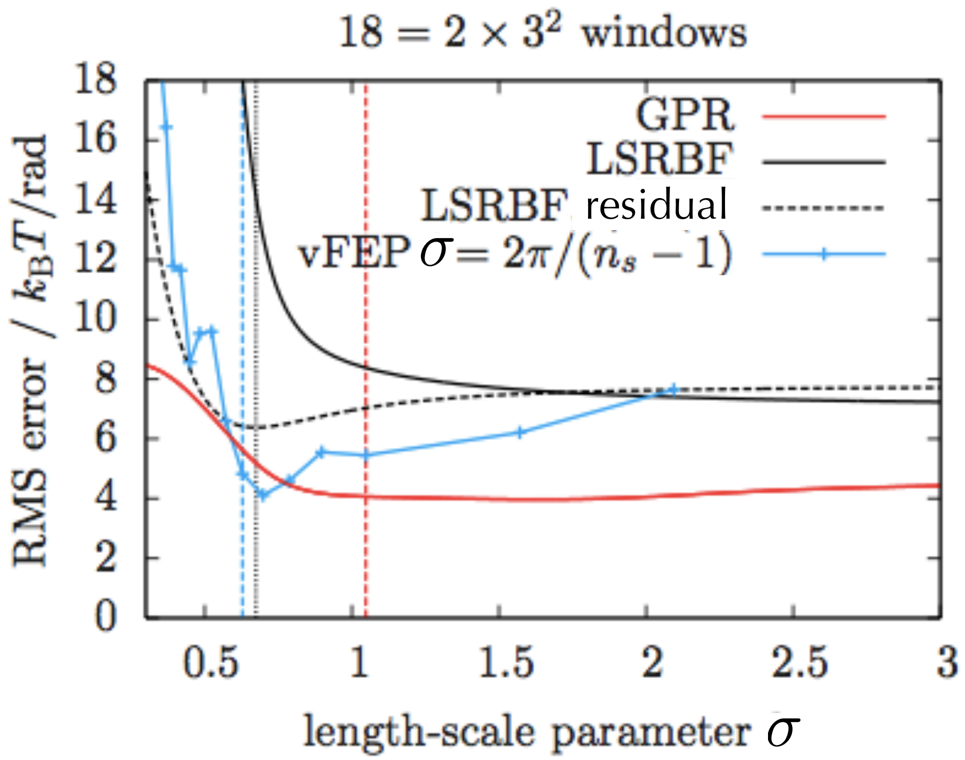


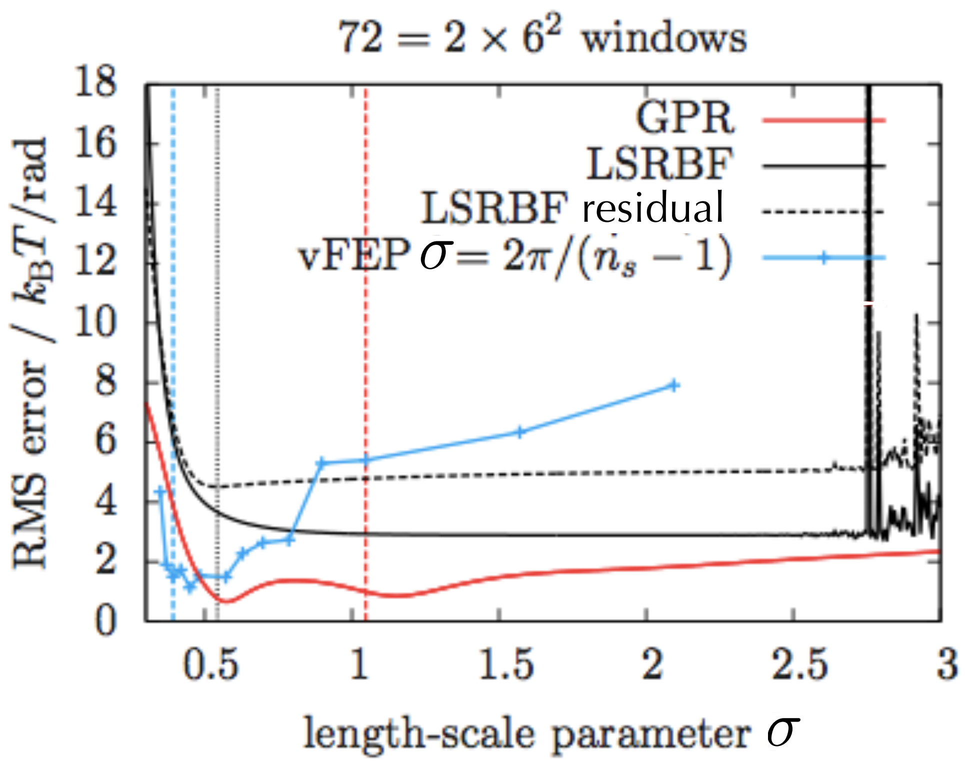
2D: effect of trajectory length (noise)



GPR vs LSRBF

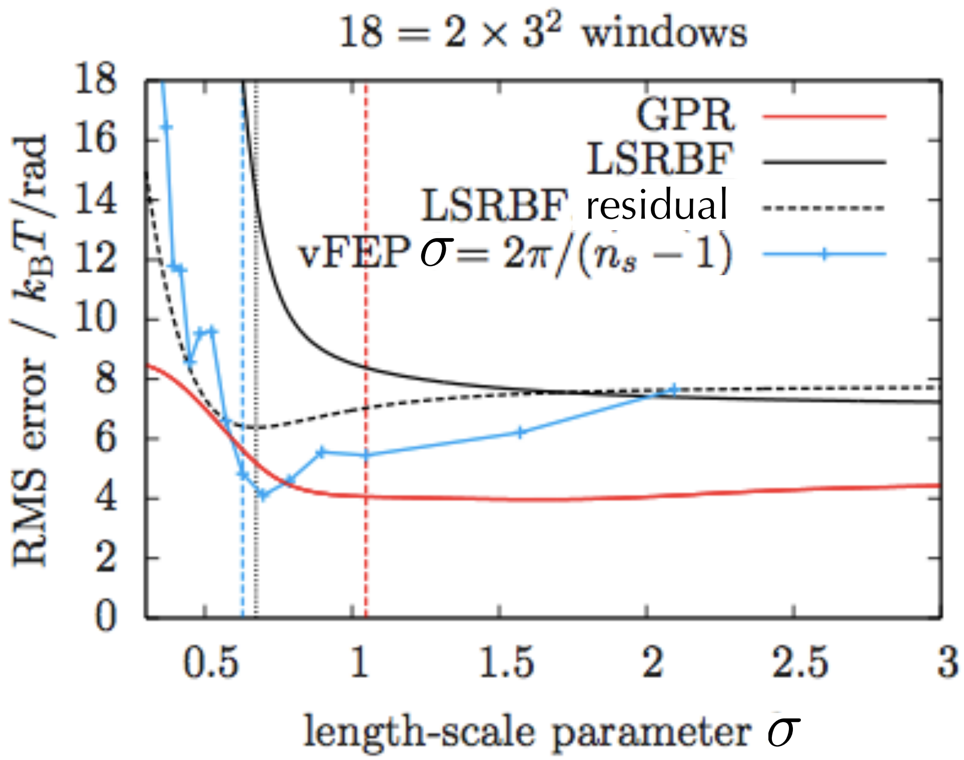
Should we optimise the length scale?

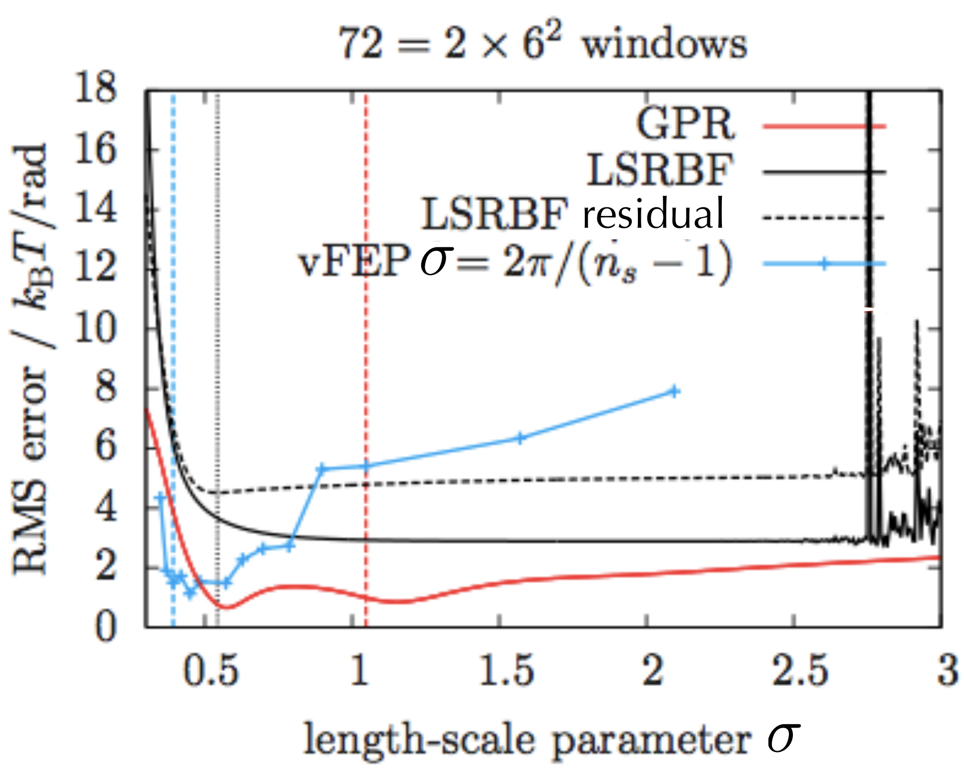




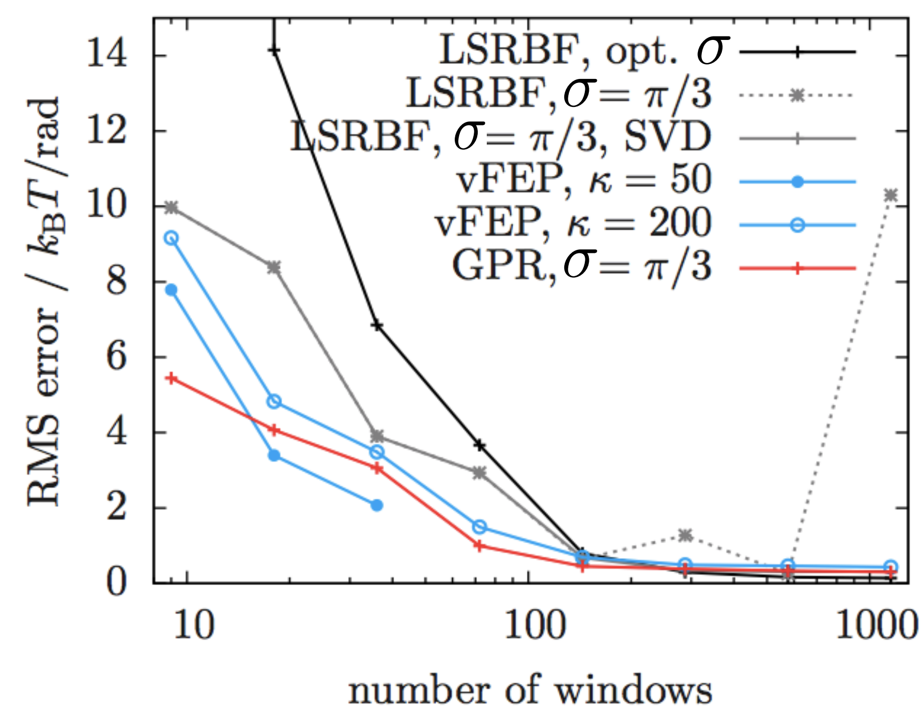
GPR vs LSRBF

Should we optimise the length scale?



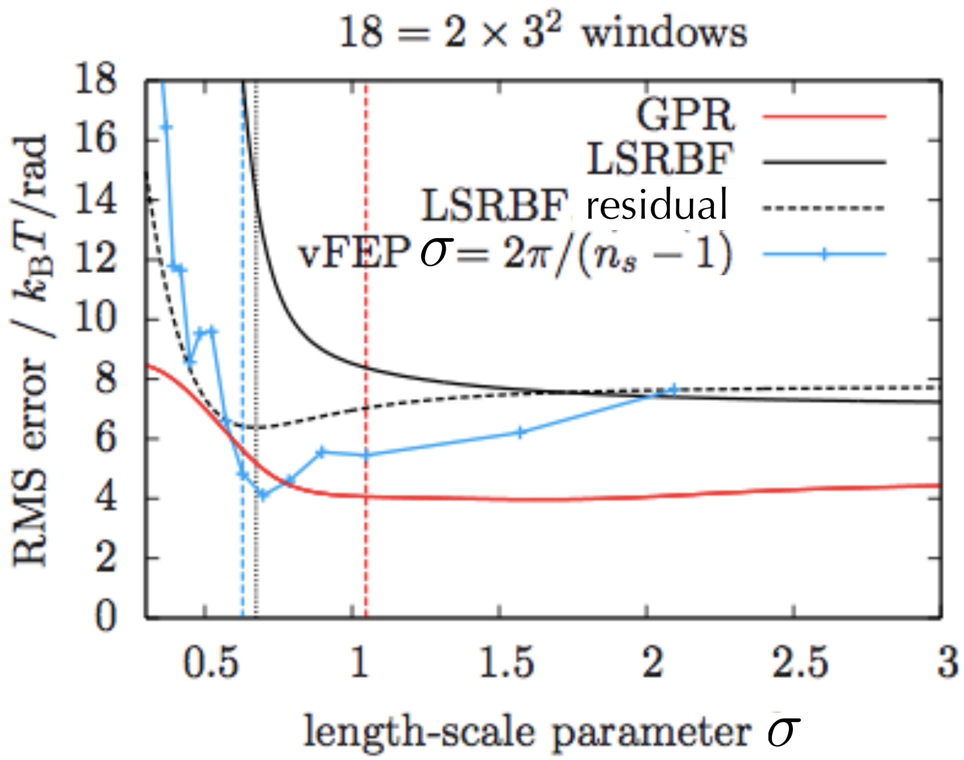


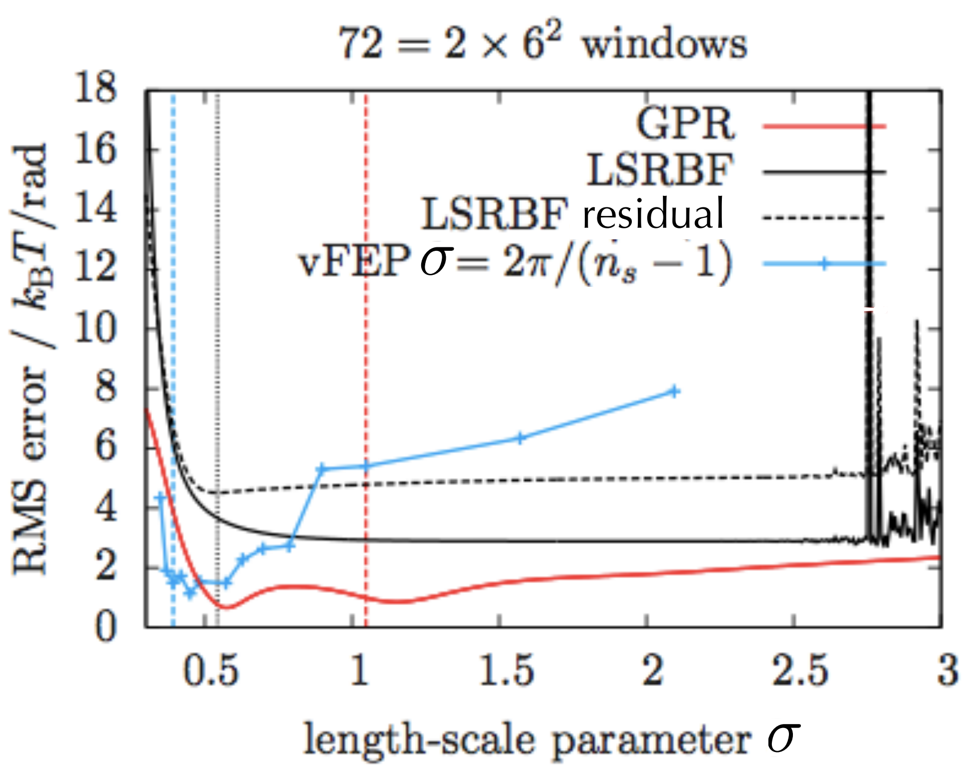
SVD for LSRBF ?



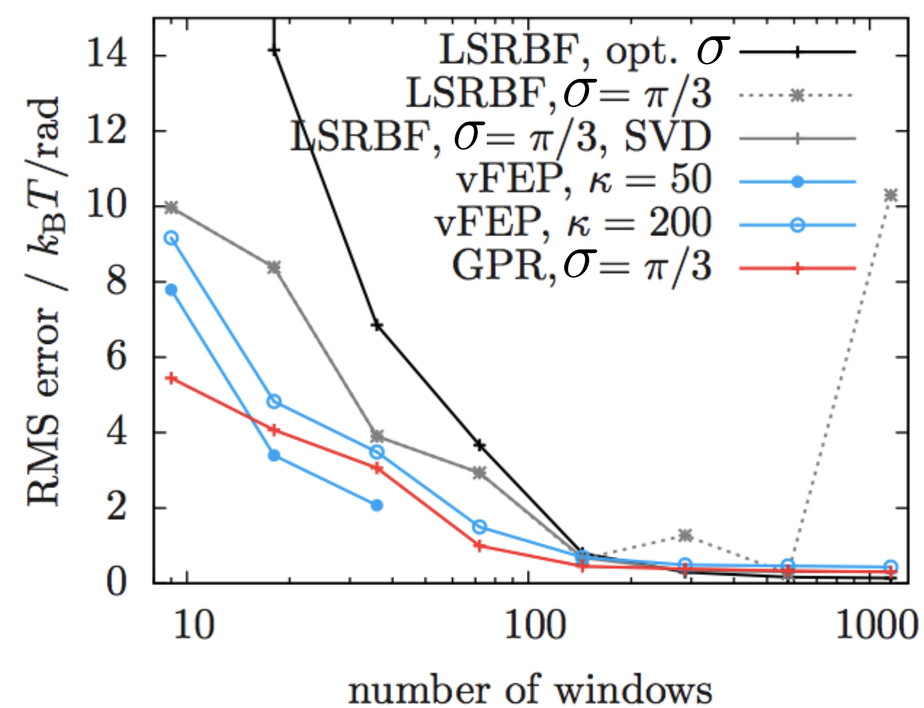
GPR vs LSRBF

Should we optimise the length scale?

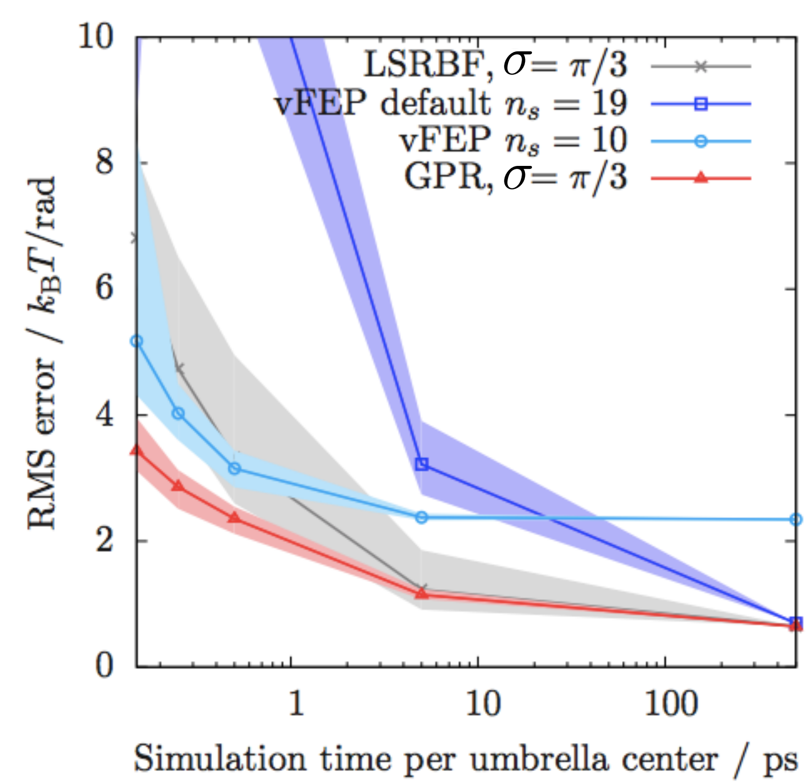




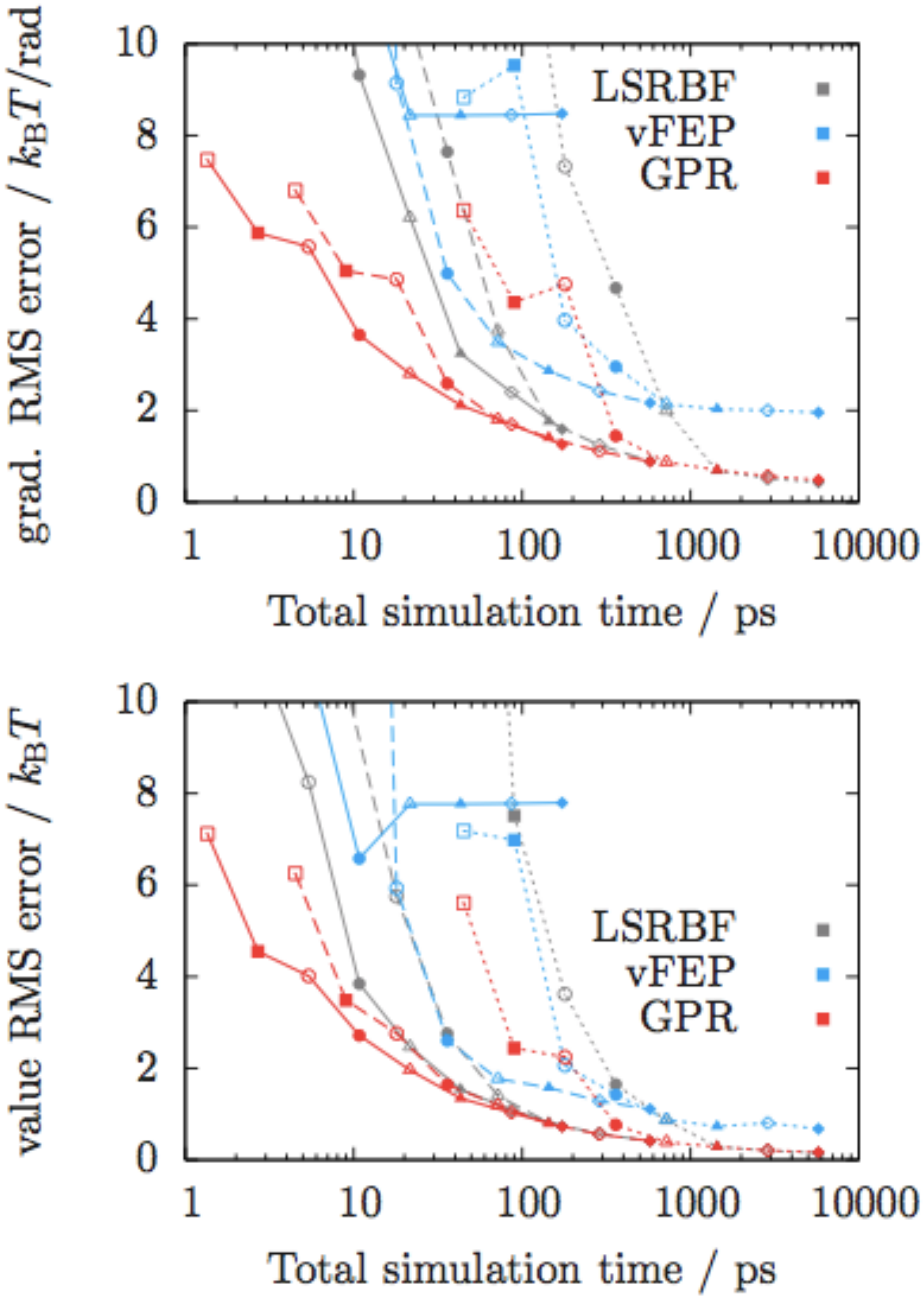
SVD for LSRBF ?



Noise tolerance

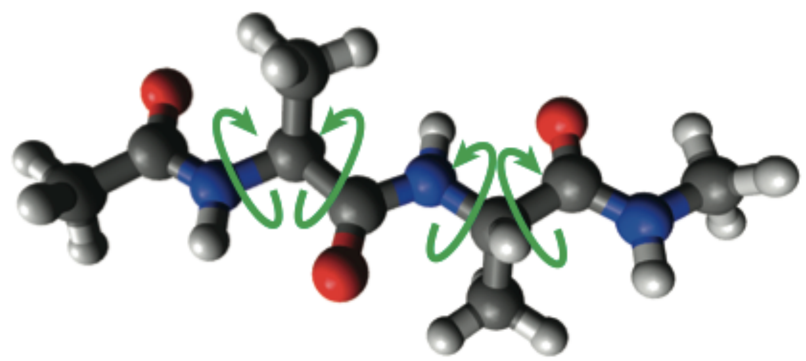


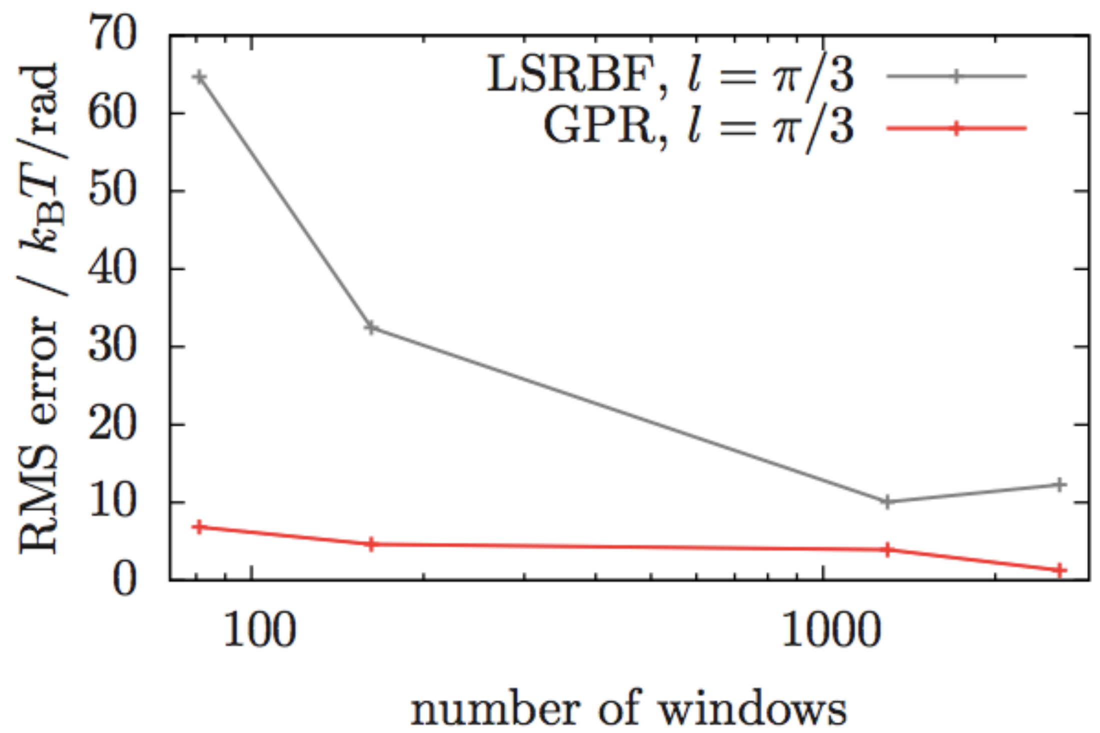
Error vs total effort

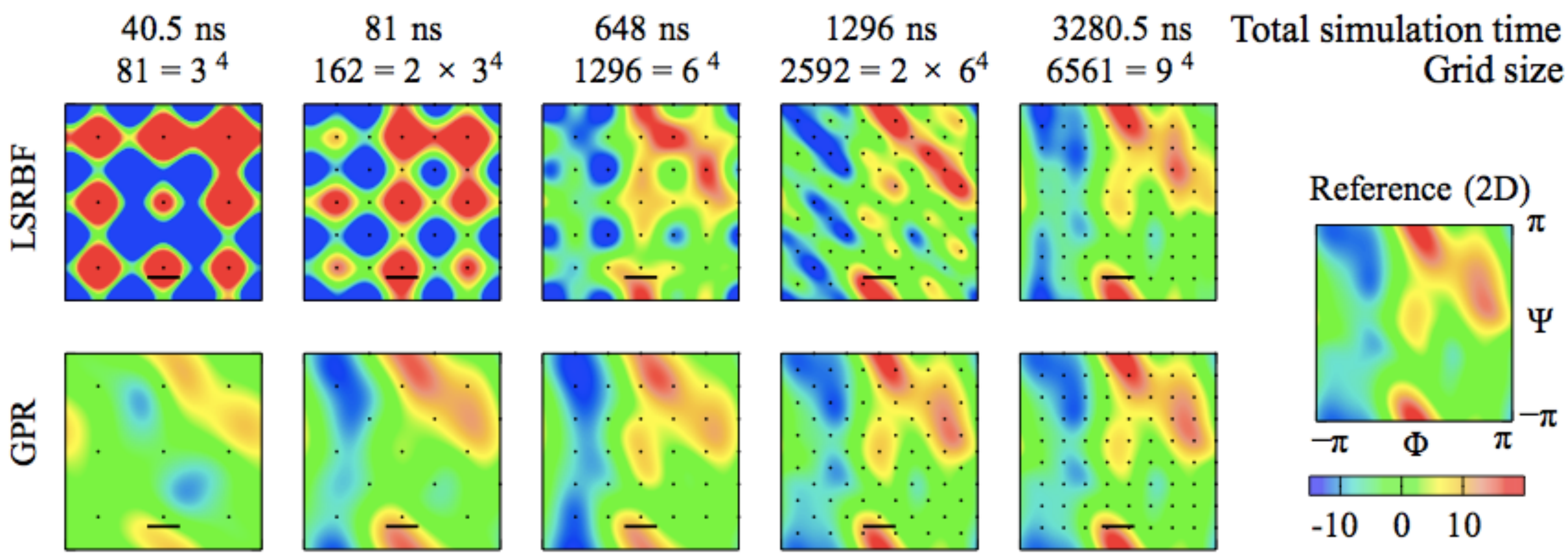


GPR: denser grid with noisier data always better

4D: tripeptide







Exploration

Biasing	Measurement	Reconstruction
Umbrella sampling	Histogram of positions	Gaussian process ✓
Metadynamics	"local mean force"	⋯
Adaptive Biasing Force	…	⋯
…	…	…

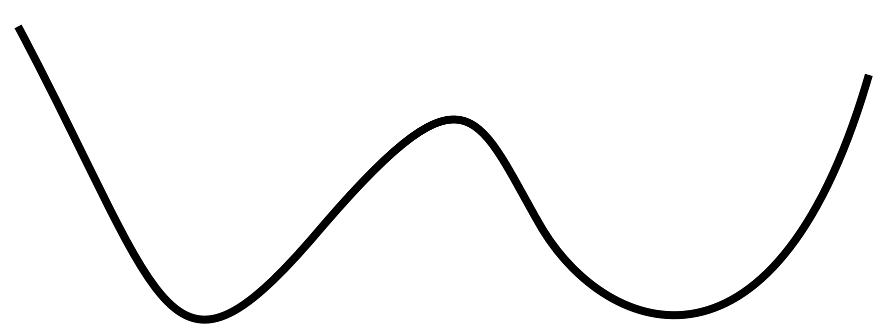
Exploration

Biasing →	Measurement →	Reconstruction
Umbrella sampling	Histogram of positions	Gaussian process ✓
Metadynamics	"local mean force"	…
Adaptive Biasing Force	…	…
…	…	…

add partial reconstruction of F to dynamics

Exploration

Biasing	→	Measurement	→	Reconstruction
Umbrella sampling		Histogram of positions		Gaussian process ✓
Metadynamics		"local mean force"		…
Adaptive Biasing Force		…		…
…		…		…

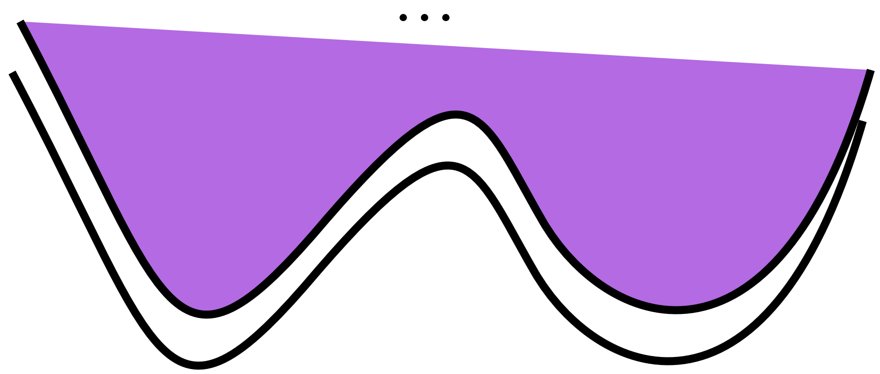


add partial reconstruction of F to dynamics

Exploration

Biasing → Measurement → Reconstruction

Biasing	Measurement	Reconstruction
Umbrella sampling	Histogram of positions	Gaussian process ✓
Metadynamics	"local mean force"	…
Adaptive Biasing Force	…	…
…	…	…



add partial reconstruction of F to dynamics

Measuring the gradient: local mean force

- Direct estimator for ∇F when x is an atomic coordinate

$$\nabla F(x') = \langle \nabla V(q) \rangle_{x=x'}$$

- More complicated for general collective variable, called "local mean force"

- Which is better?

-log(histogram of positions) vs $\int$ gradient estimator

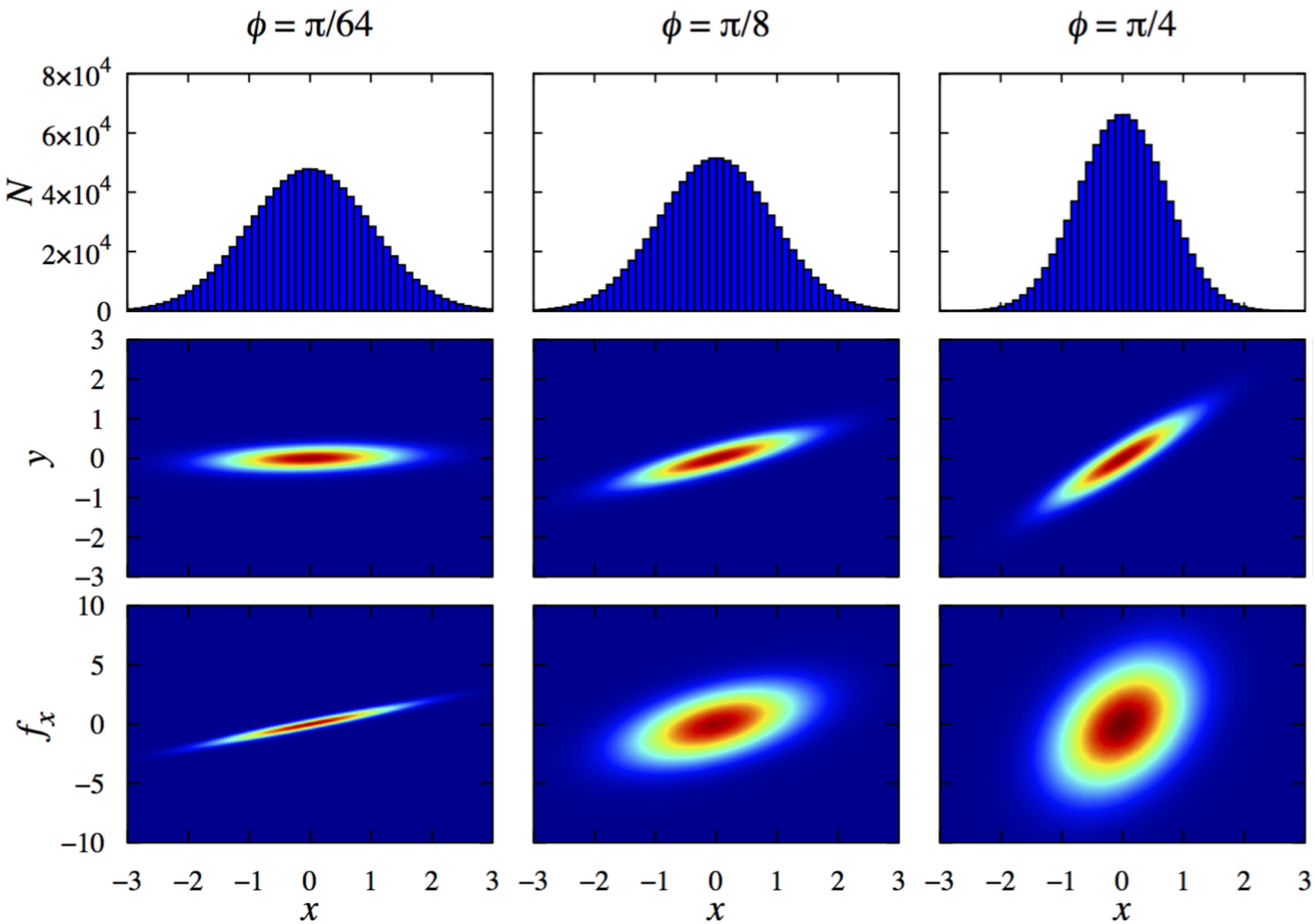
Simple test: quadratic potential

$$V_{\boldsymbol{\sigma}^2,\phi}(x,y) = \frac{1}{2}\begin{pmatrix} x & y \end{pmatrix} \mathbf{C}^{-1}_{\boldsymbol{\sigma}^2,\phi} \begin{pmatrix} x \\ y \end{pmatrix}$$

$$\mathbf{C}_{\boldsymbol{\sigma}^2} = \begin{pmatrix} \sigma_x^2 & 0 \\ 0 & \sigma_y^2 \end{pmatrix} \quad \mathbf{R}_\phi = \begin{pmatrix} \cos\phi & -\sin\phi \\ \sin\phi & \cos\phi \end{pmatrix}$$

$$\mathbf{C}_{\boldsymbol{\sigma}^2,\phi} = \mathbf{R}_\phi \mathbf{C}_{\boldsymbol{\sigma}^2} \mathbf{R}_\phi^{-1} = \begin{pmatrix} \sigma_x^2 c^2 + \sigma_y^2 s^2 & \sigma_x^2 cs - \sigma_y^2 cs \\ \sigma_x^2 cs - \sigma_y^2 cs & \sigma_x^2 s^2 + \sigma_y^2 c^2 \end{pmatrix}$$

$$\sigma_x/\sigma_y = 5$$

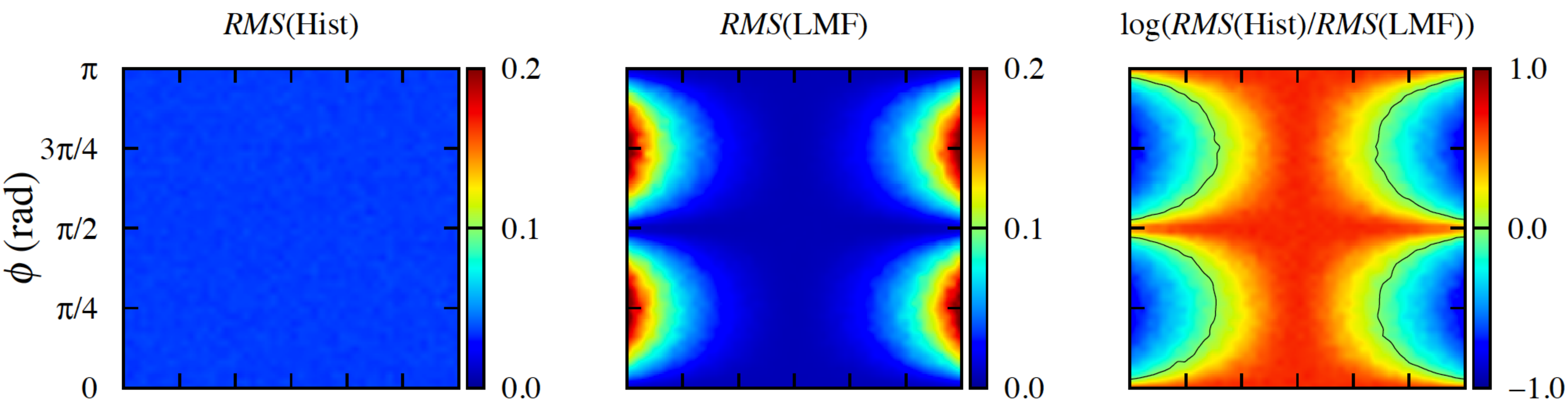


$P(x)$

$P(x,y)$

$P(\nabla V)$

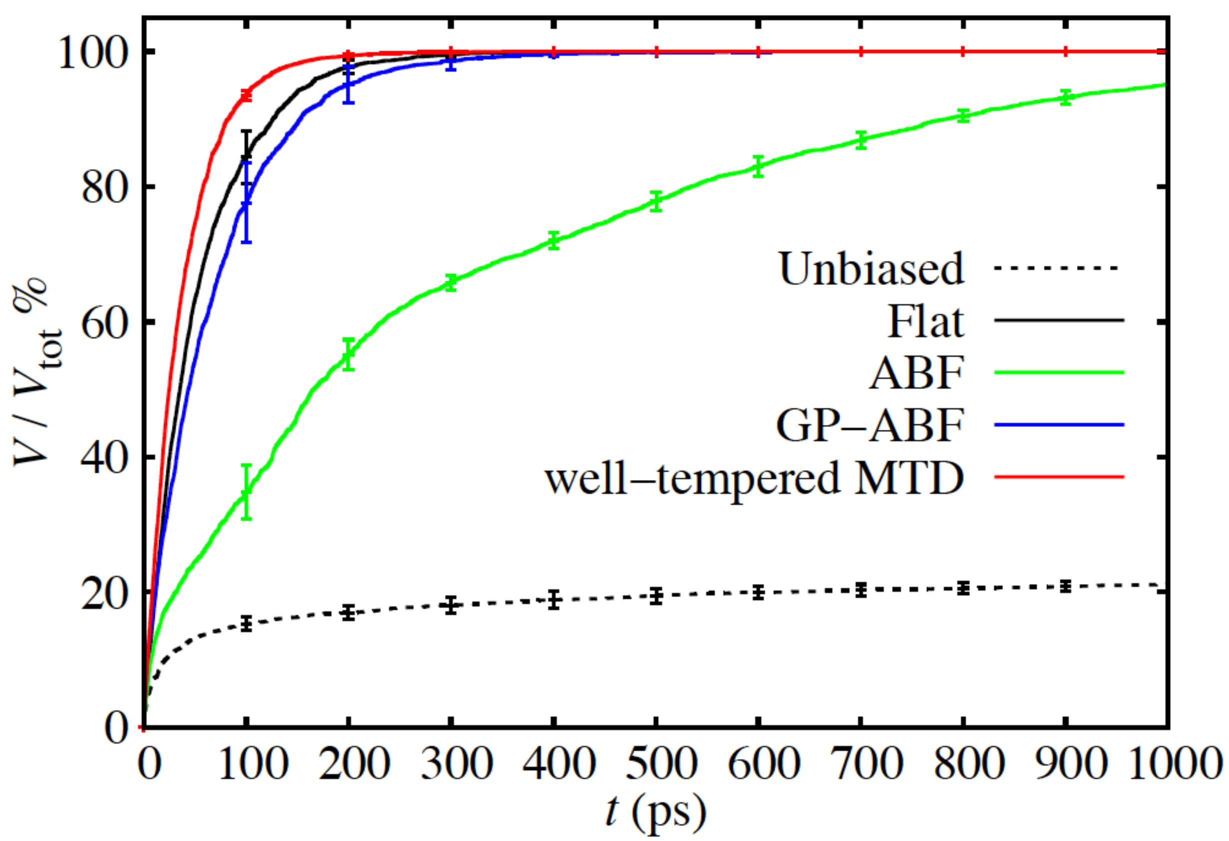
Simple test: quadratic potential



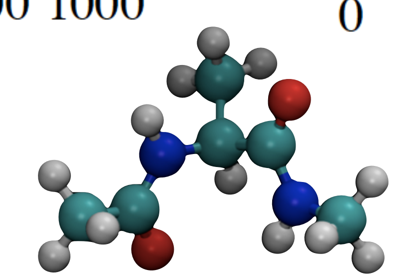
(No GPR, purely a test of "local mean force" vs position histogram)

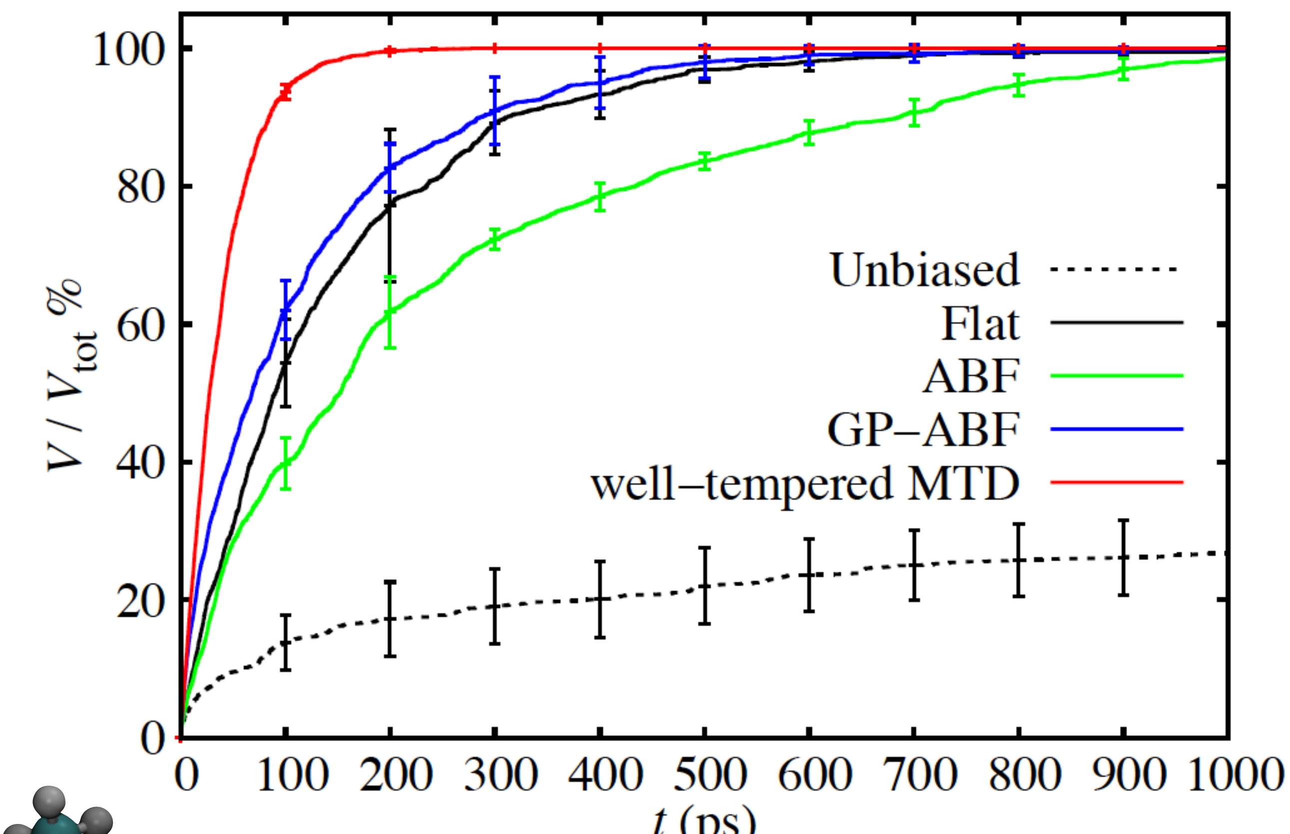
Adaptive biasing: add partial reconstruction to potential

- Metadynamics (MTD): add Gaussian hills centred on trajectory to potential (i.e. kernel-density estimate of F)
- Adaptive Biasing Force (ABF): add -ve of ∇V to forces
- GP-ABF: use Gaussian processed local mean forces (LMF), **use all data, no grid**



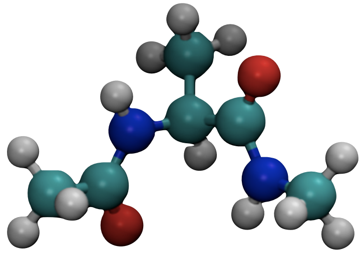
Gas phase

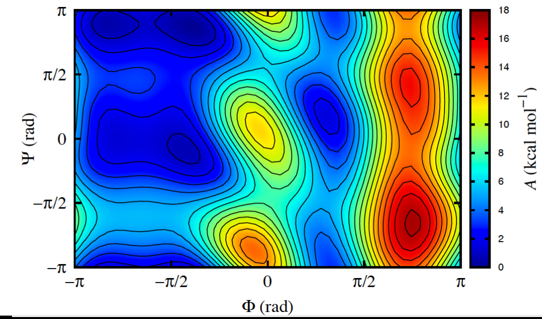


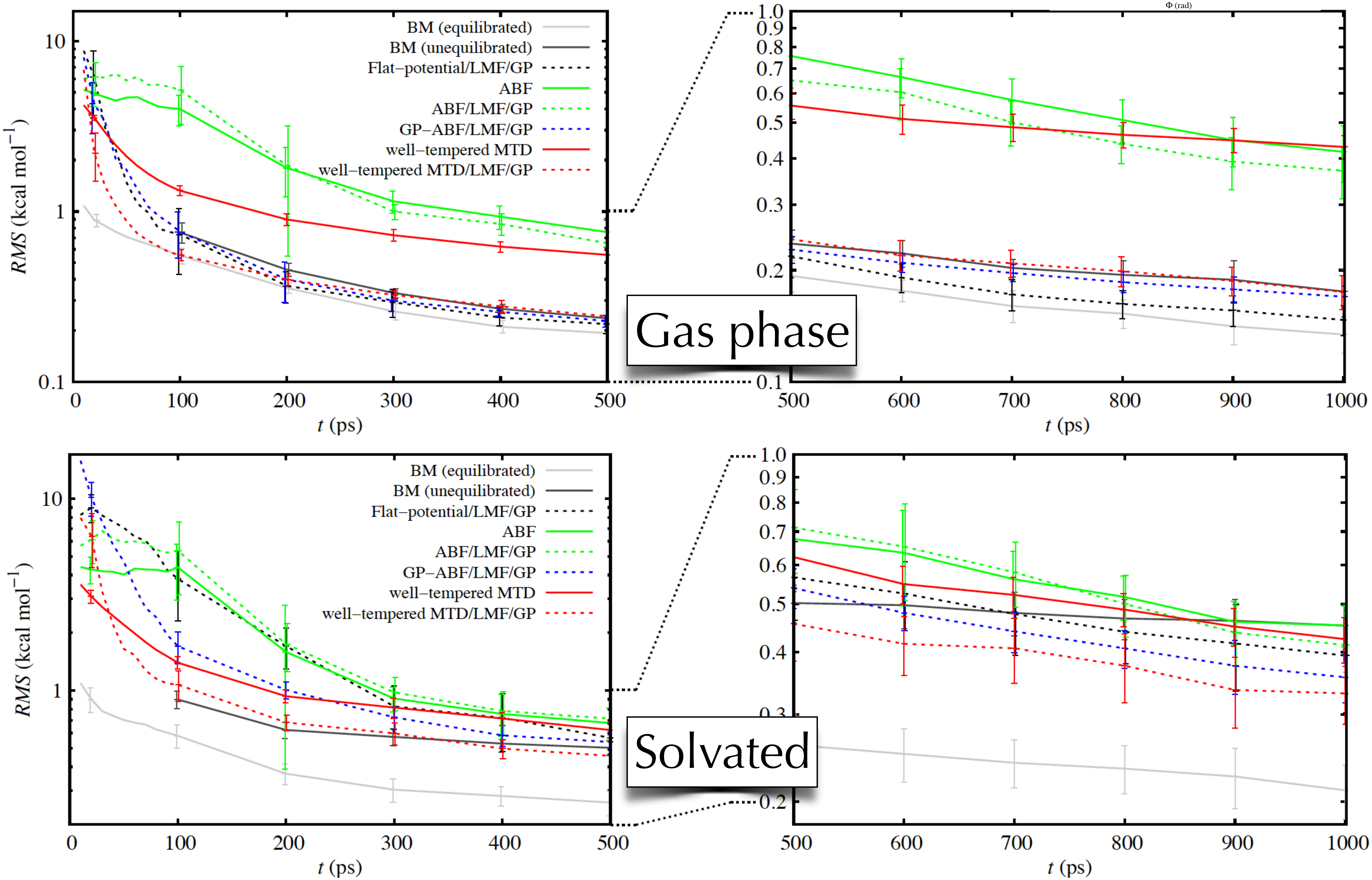


Solvated

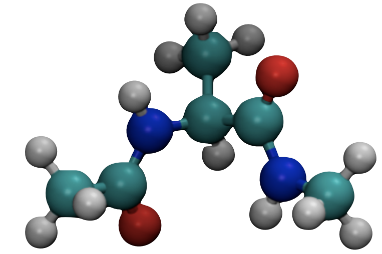
Putting it all together: 2D

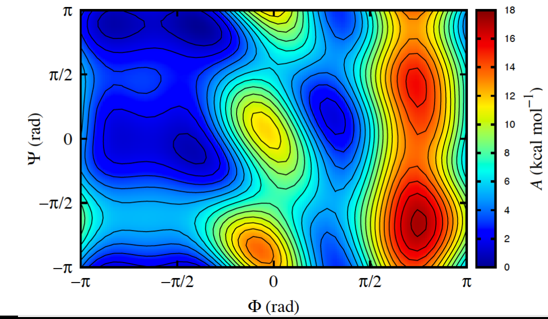


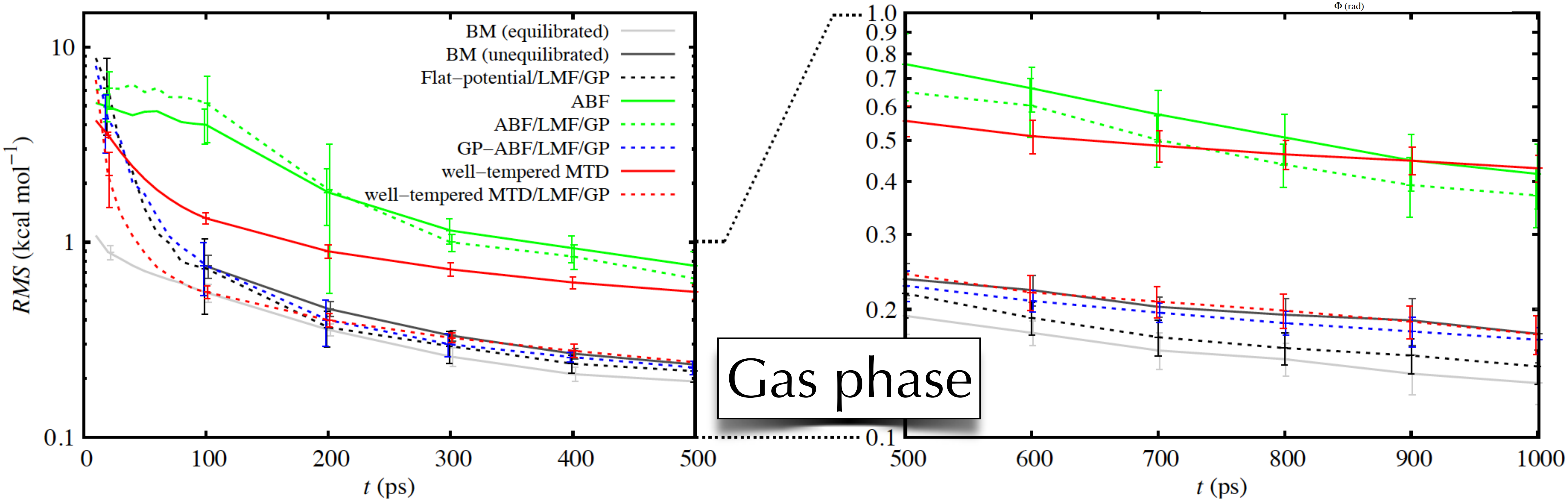


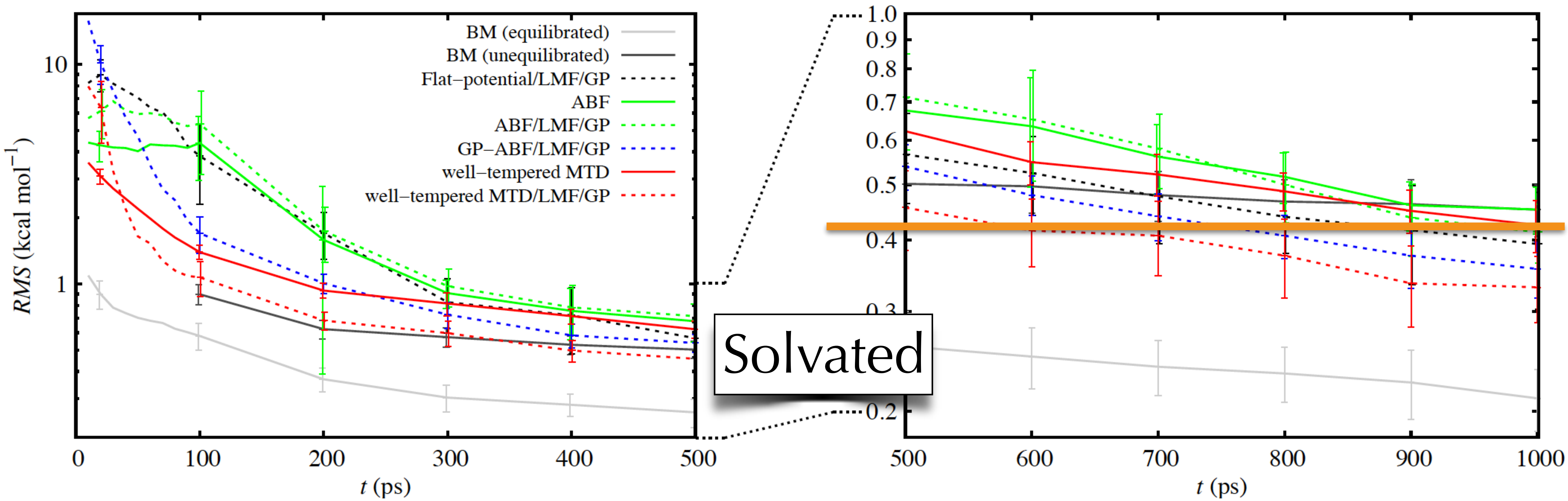


Putting it all together: 2D

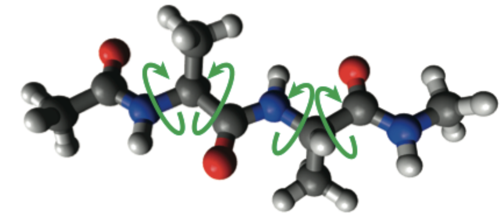


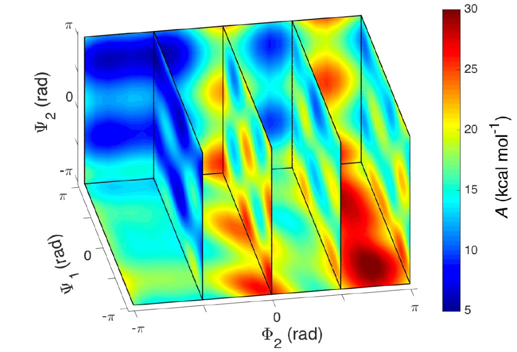


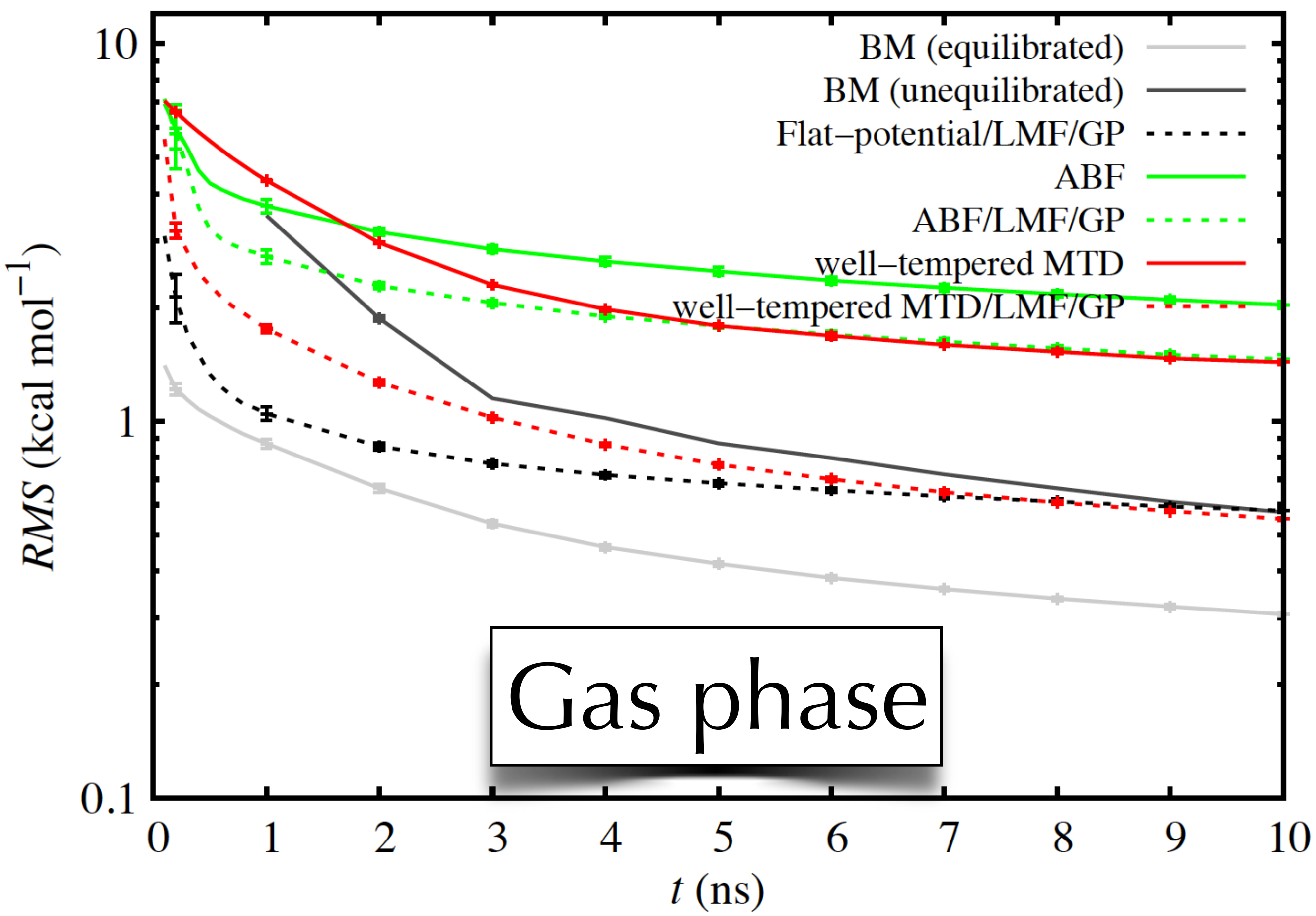


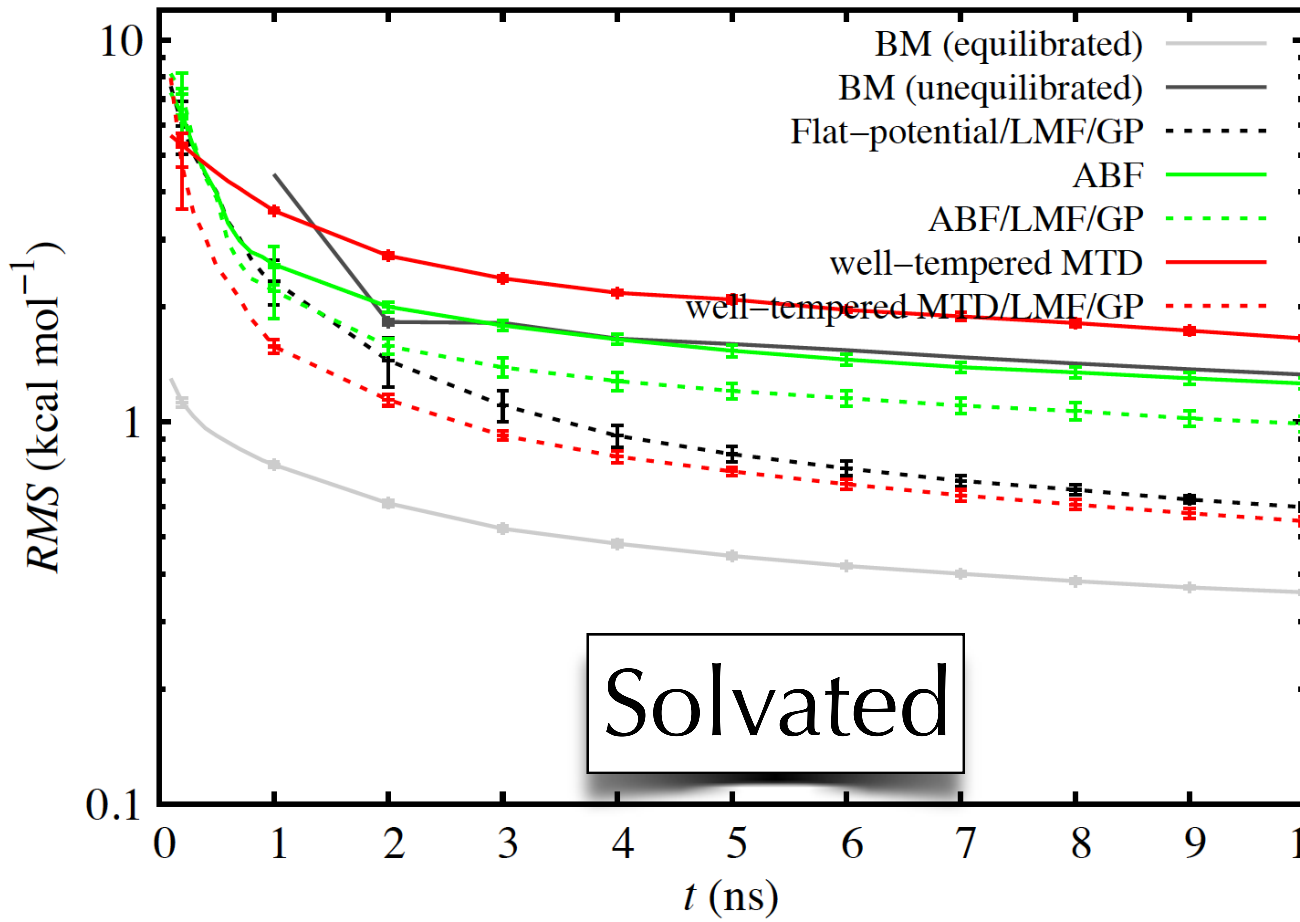


4D: Alanine tripeptide

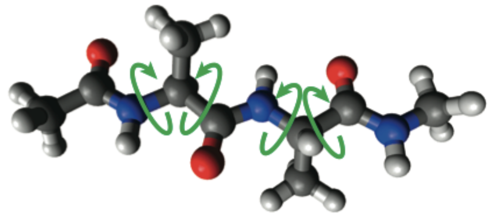


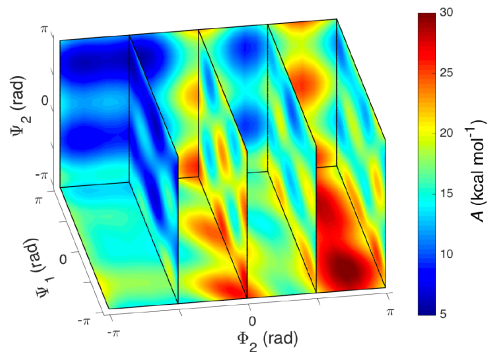


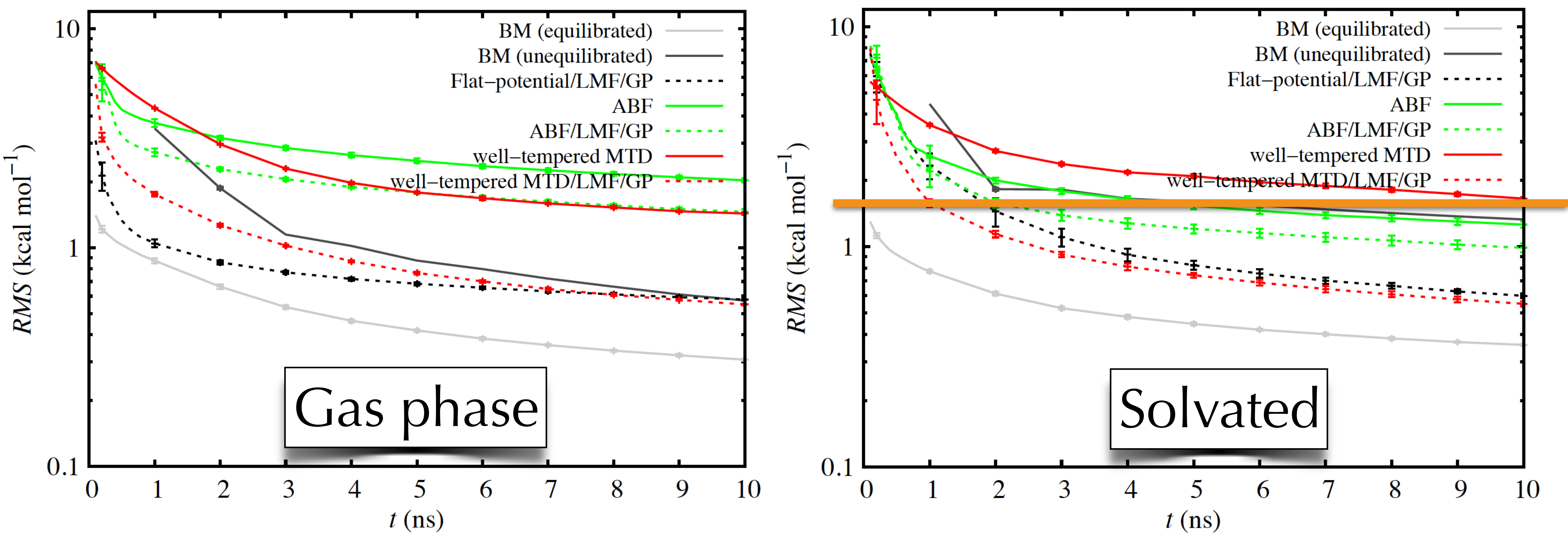




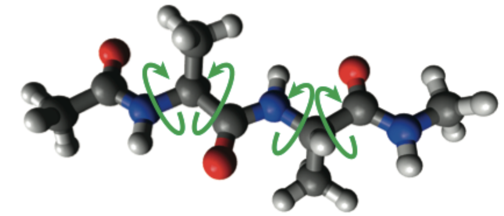
4D: Alanine tripeptide

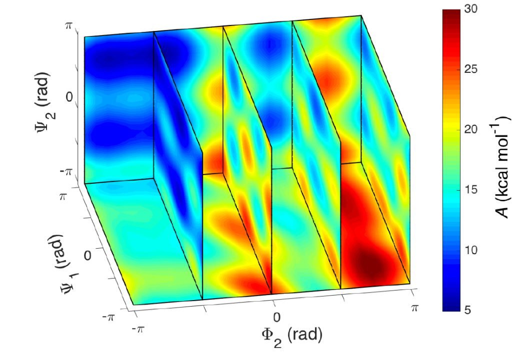


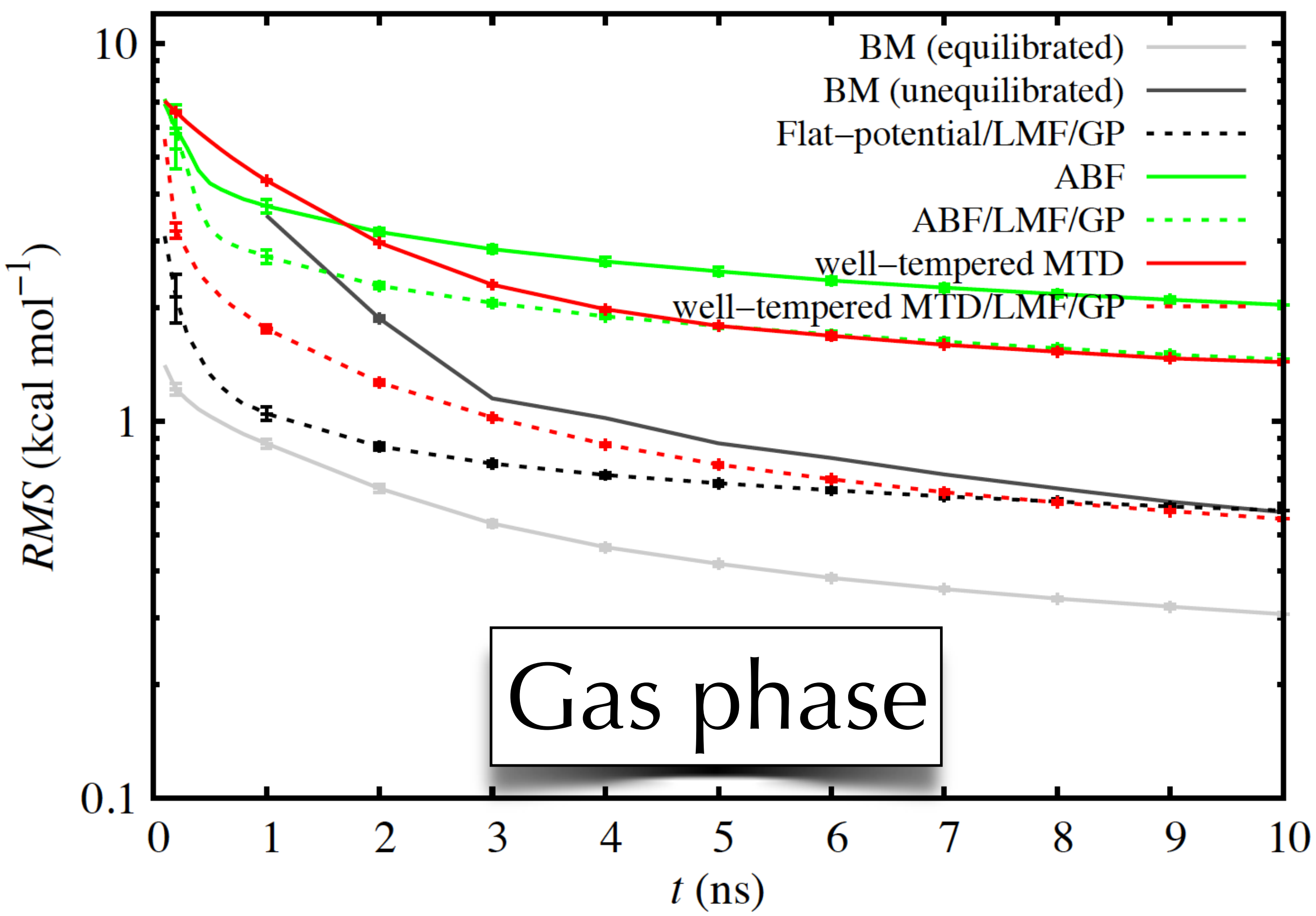


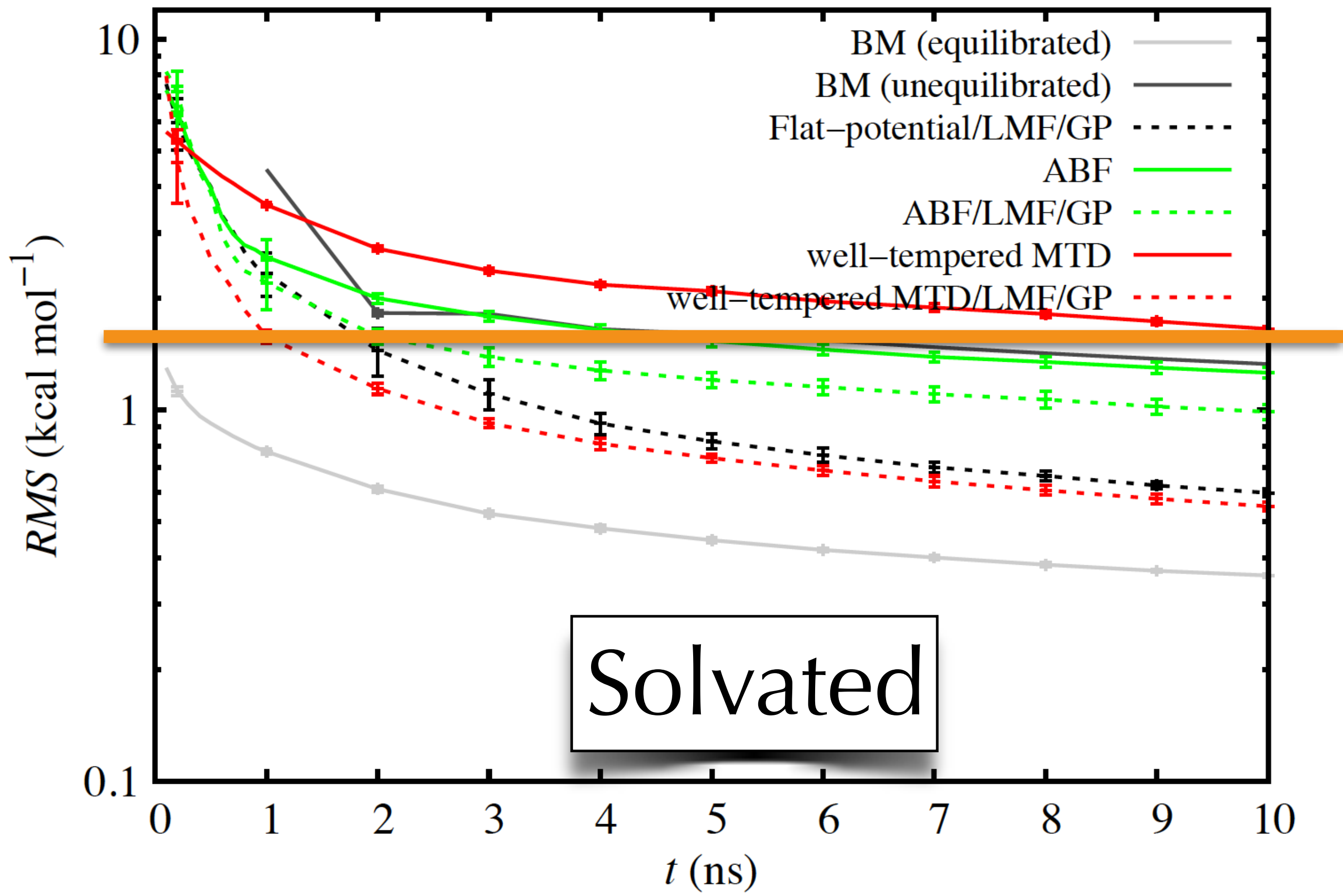


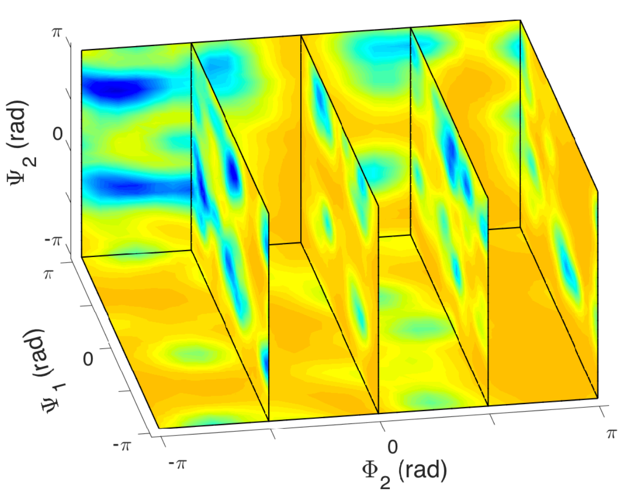
4D: Alanine tripeptide



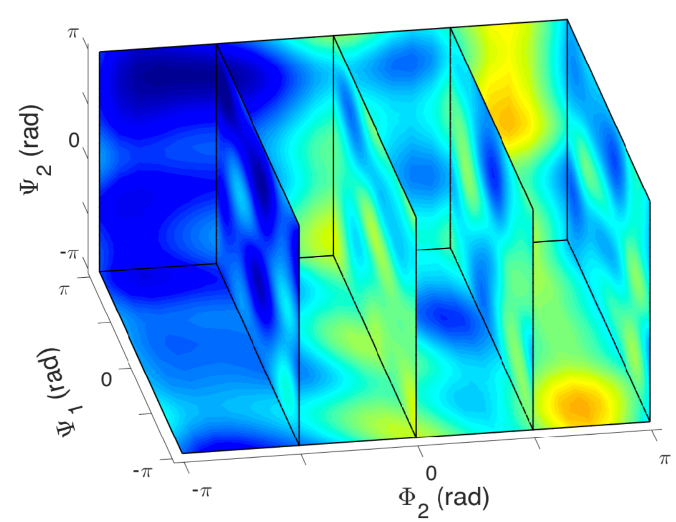




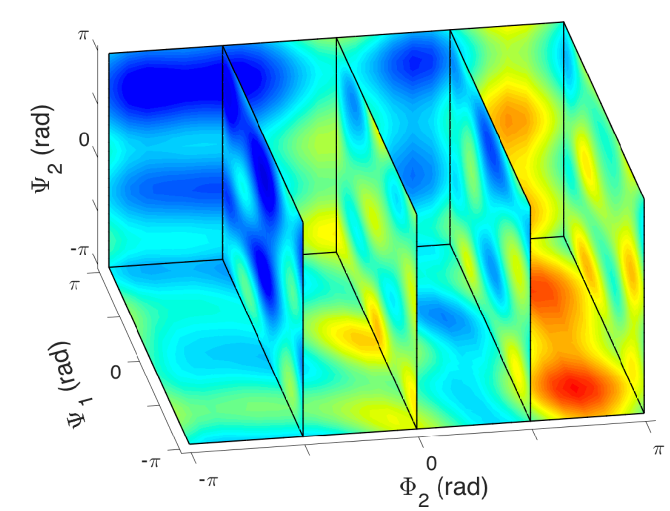




(a) MTD, t = 1 ns



(g) ABF, t = 1 ns



(d) MTD/LMF/GP, t = 1 ns