Bayesian Analysis of Nuclear Dynamics

ROSE

Theory and implementation

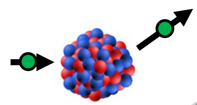
ROSEs are red,
violets are blue
the Reduced Order Scattering Emulator
is just the tool for you

Daniel Odell
Pablo Giuliani

BAND Camp May 2023

Outline

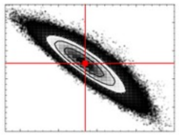
Main ideas

Nuclear Scattering 

Reduced Basis Method 

Main results

Accuracy and time 

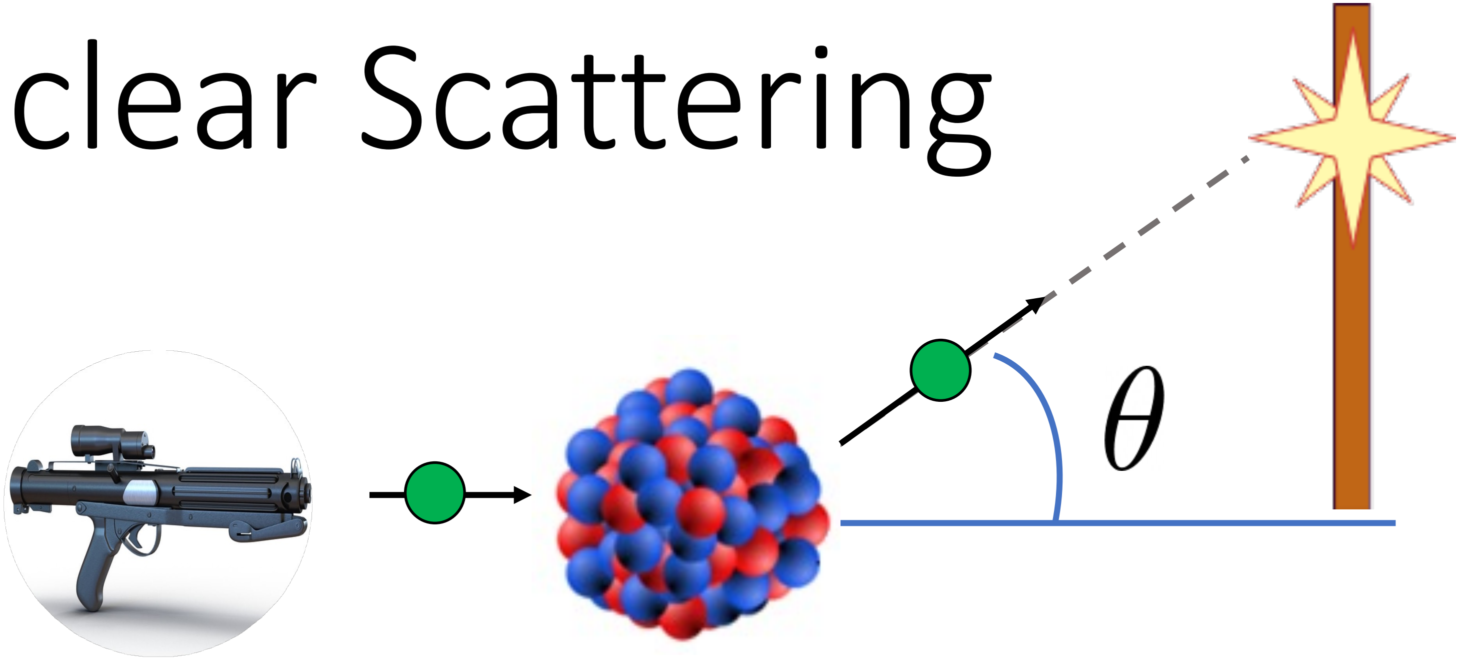
Going Bayesian with Surmise 

Tutorials

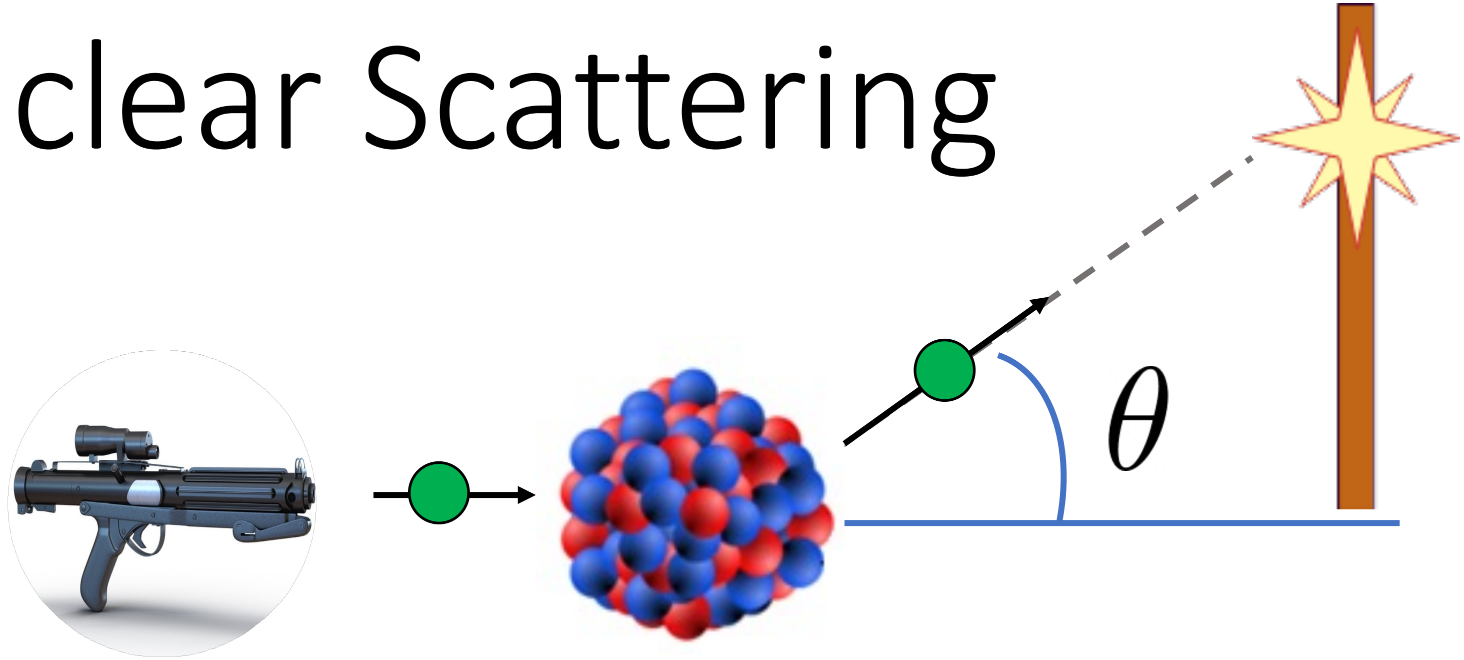
1) Emulating without ROSE

2) Emulating with ROSE

Nuclear Scattering



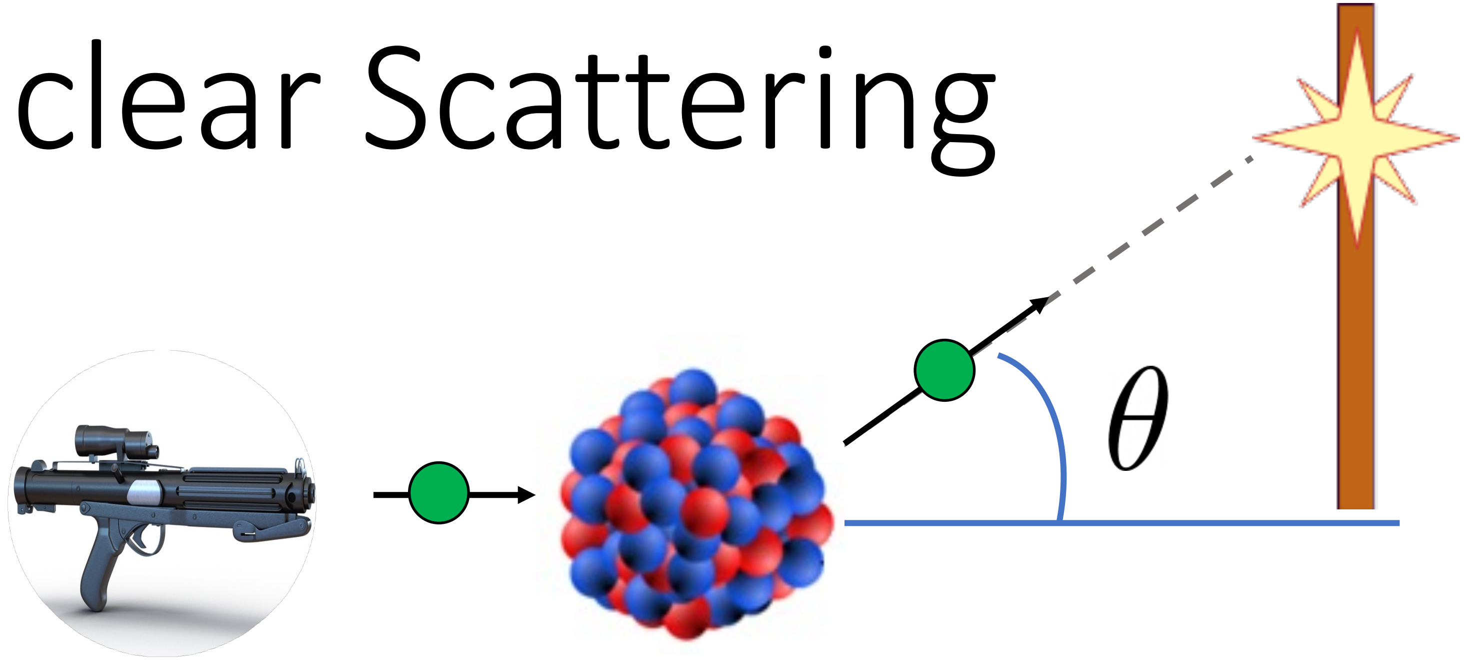
Nuclear Scattering



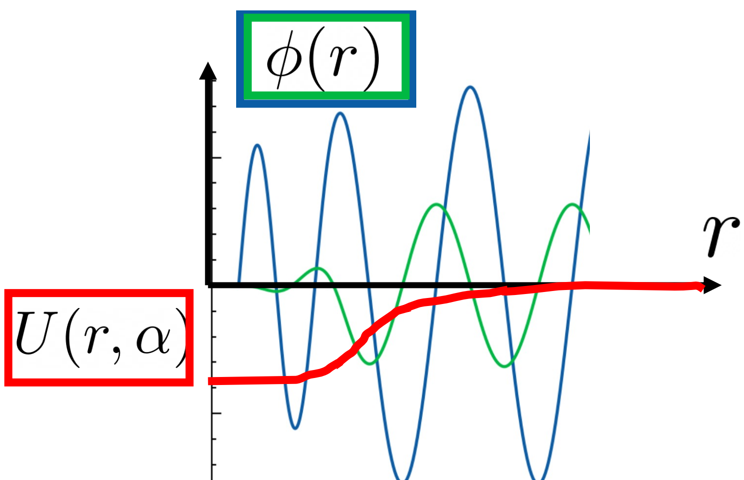
$$F_\alpha(\phi) = \left(-\frac{d^2}{dr^2} + \frac{\ell(\ell+1)}{r^2} + U(r,\alpha) - p^2\right)\phi(r) = 0$$

Parameters

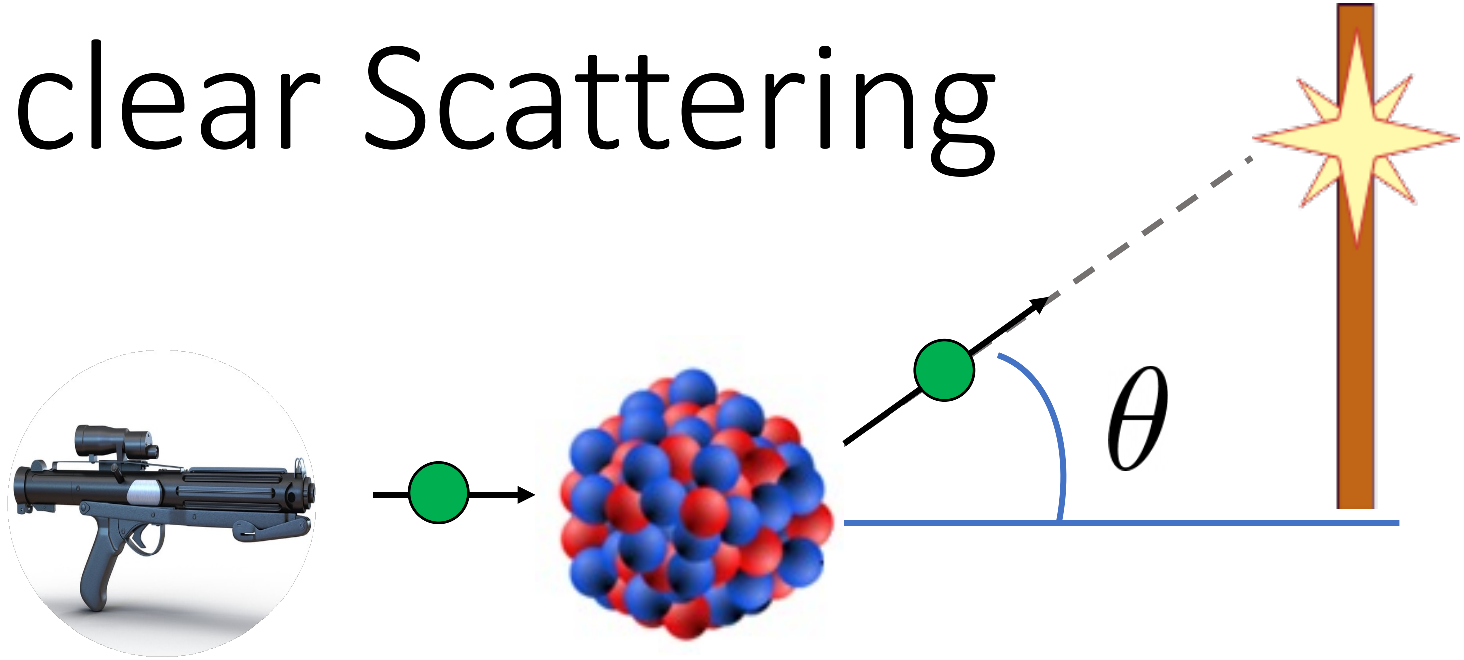
Nuclear Scattering



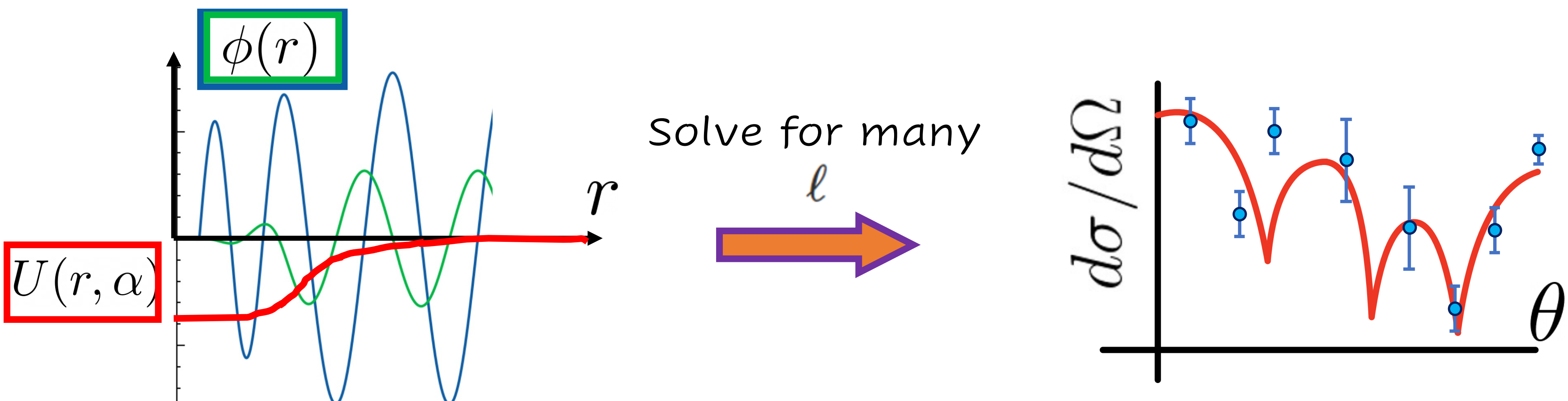
$$F_\alpha(\phi) = \left(-\frac{d^2}{dr^2} + \frac{\ell(\ell+1)}{r^2} + U(r,\alpha) - p^2\right)\phi(r) = 0$$



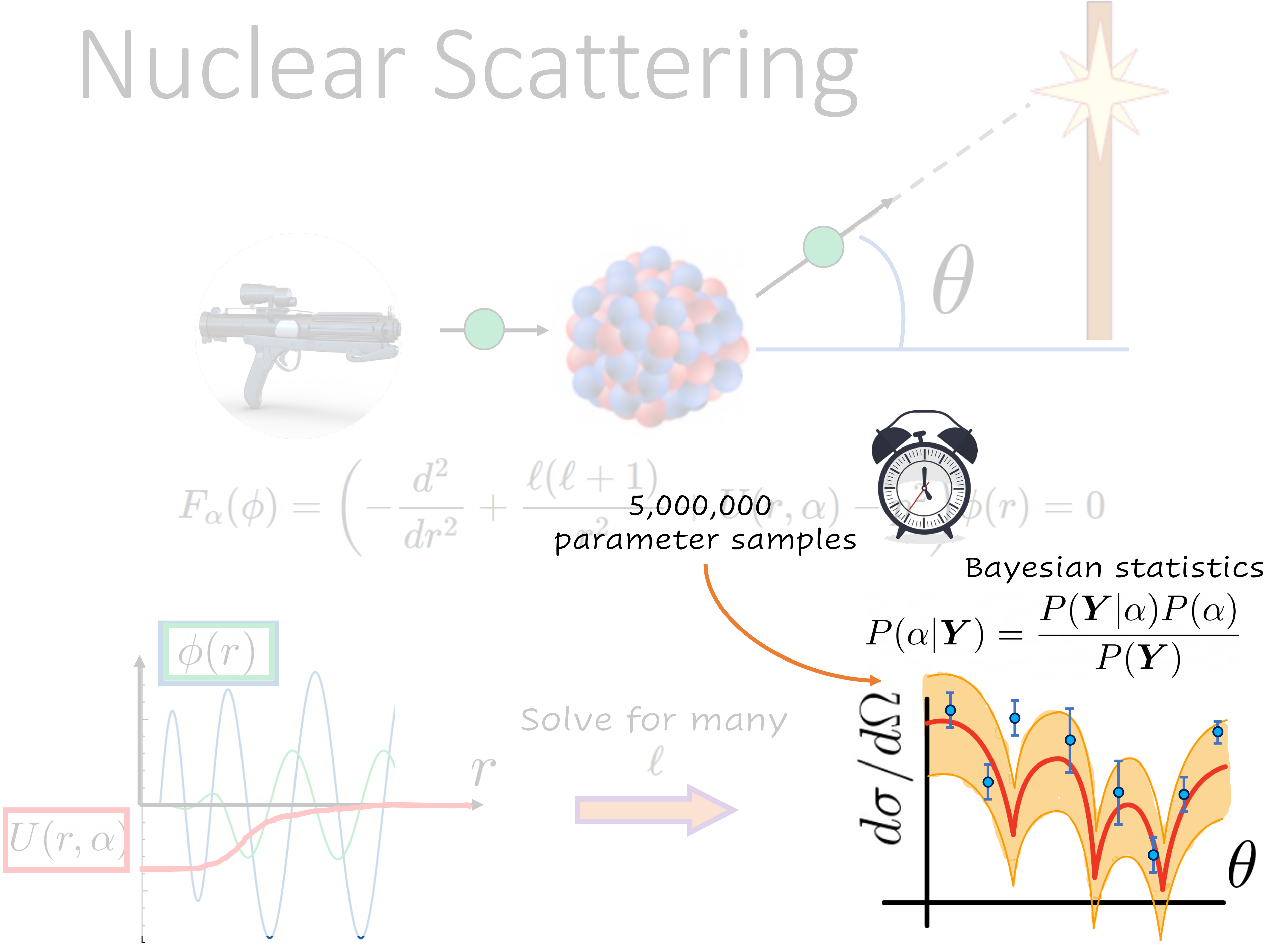
Nuclear Scattering



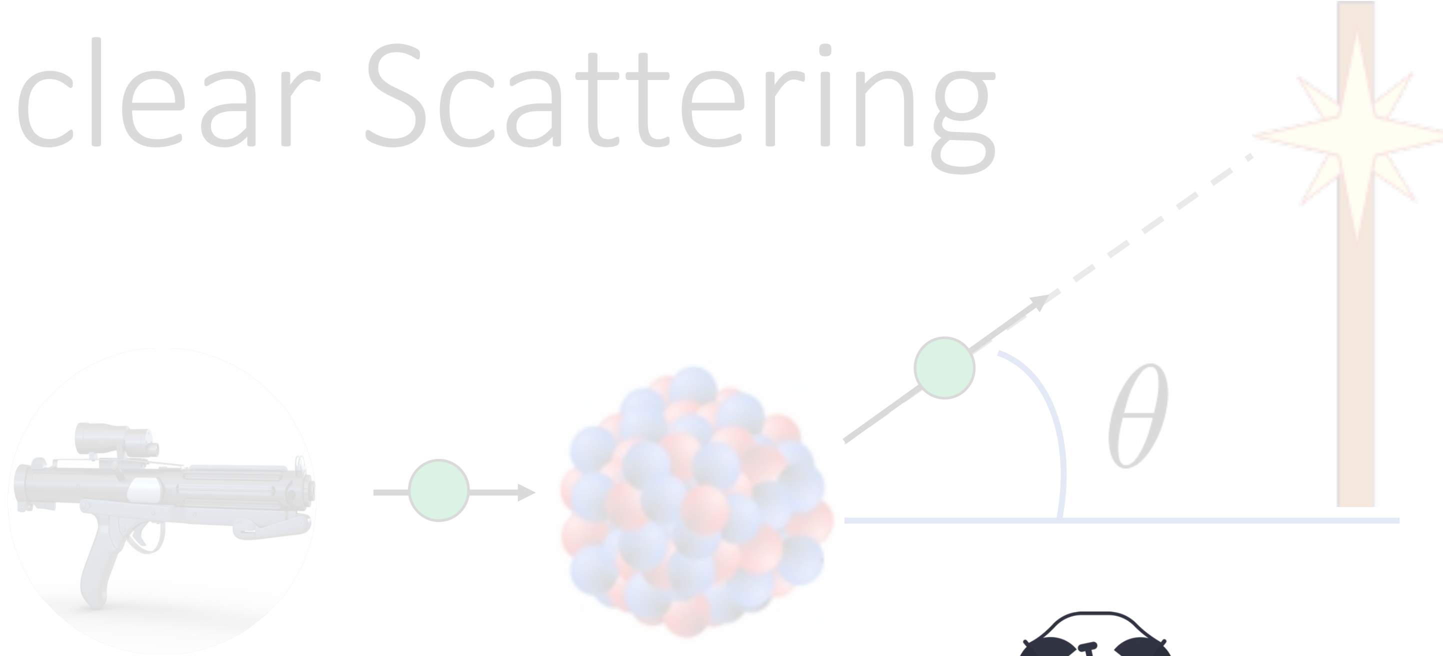
$$F_\alpha(\phi) = \left(-\frac{d^2}{dr^2} + \frac{\ell(\ell+1)}{r^2} + U(r,\alpha) - p^2\right)\phi(r) = 0$$



Nuclear Scattering



Nuclear Scattering



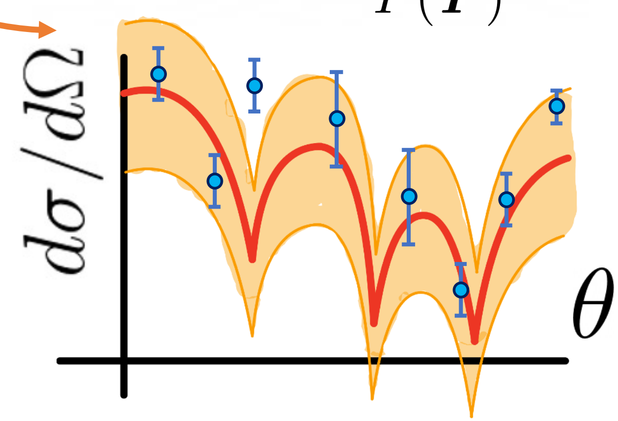
5,000,000
parameter samples

Bayesian statistics

$$P(\alpha|\mathbf{Y}) = \frac{P(\mathbf{Y}|\alpha)P(\alpha)}{P(\mathbf{Y})}$$

$$f(\alpha, x) \rightarrow \hat{f}(\alpha, x)$$

"Exact" "Emulated"



Emulators

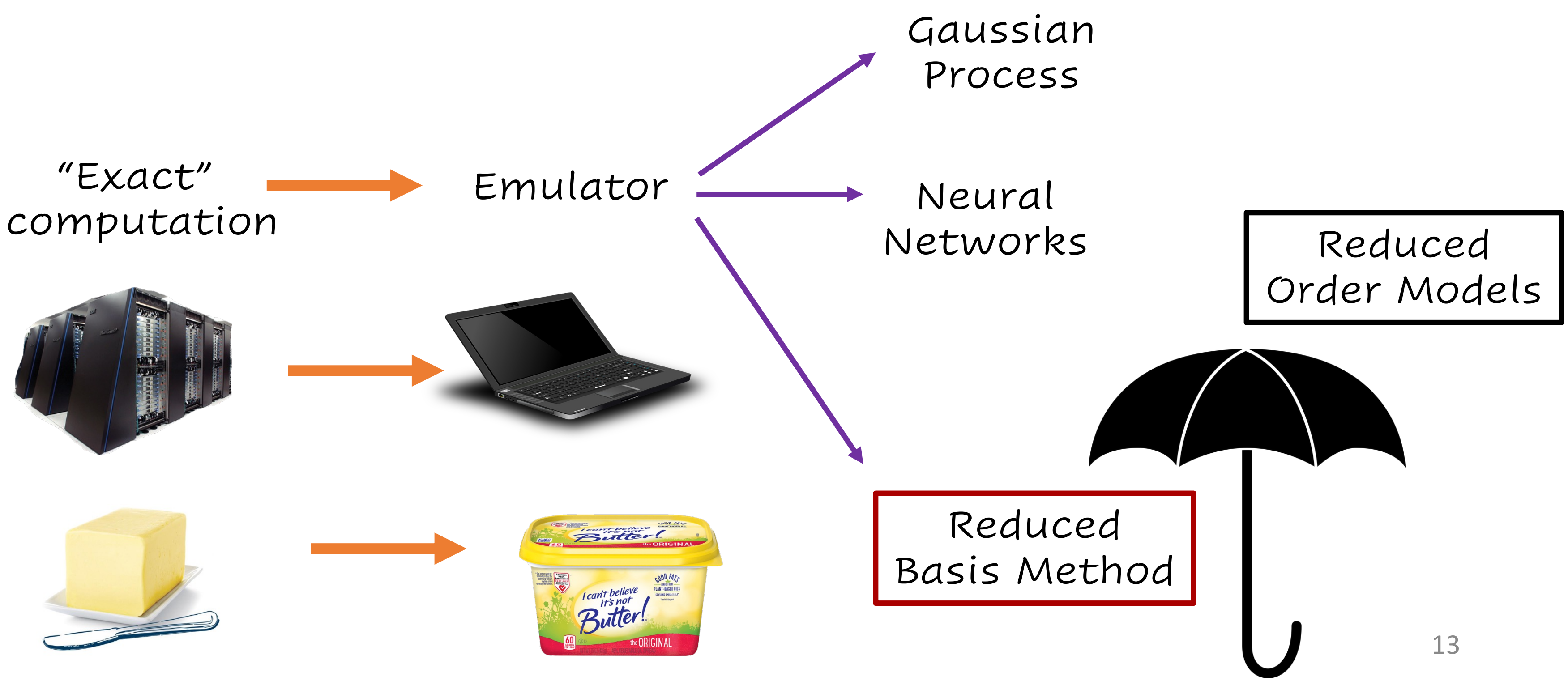
"Exact" computation → Emulator

Emulator → Gaussian Process

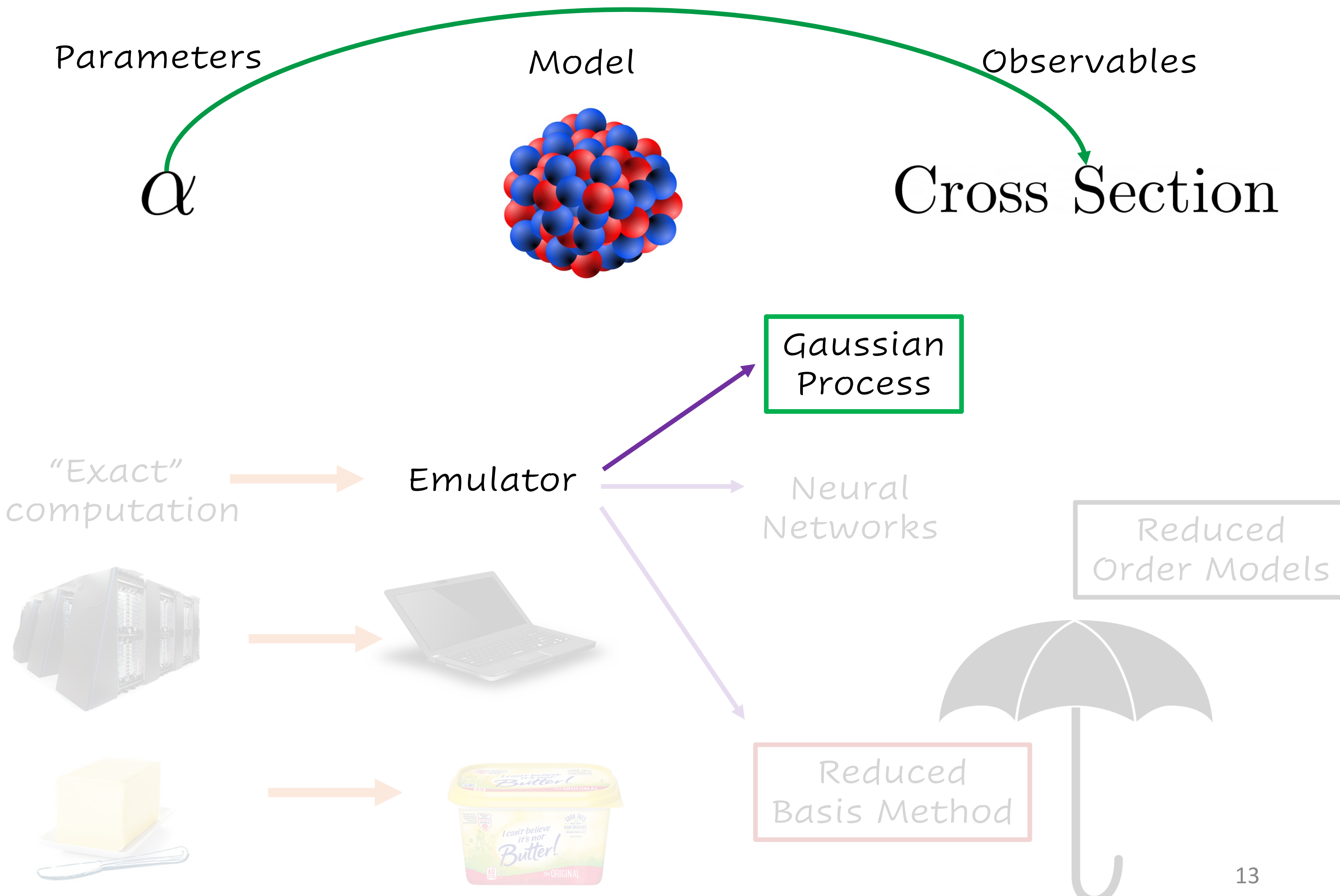
Emulator → Neural Networks

Emulator → Reduced Basis Method

Emulators


"Exact" computation
Emulator
Gaussian Process
Neural Networks
Reduced Order Models
Reduced Basis Method
I can't believe it's not Butter!
the ORIGINAL

Emulators


Parameters
Model
Observables
α
Cross Section
Gaussian Process
"Exact" computation
Emulator
Neural Networks
Reduced Order Models
Reduced Basis Method

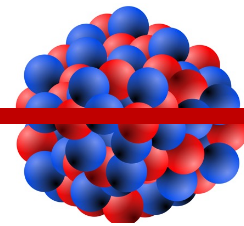
Emulators

Parameters

Model

Observables

α



Cross Section

$$\left(-\frac{d^2}{dr^2}+\frac{\ell(\ell+1)}{r^2}+U(r,\alpha)-p^2\right)\phi(r)=0$$

an
Process

"Exact" computation

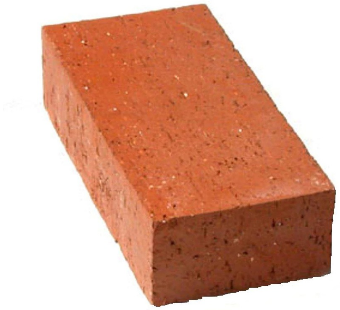
Emulator

Neural Networks

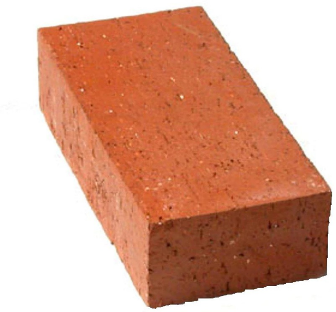
Reduced Order Models

Reduced Basis Method

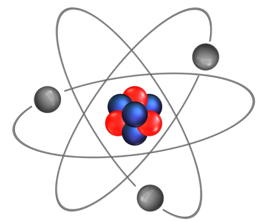
The Reduced Basis Method



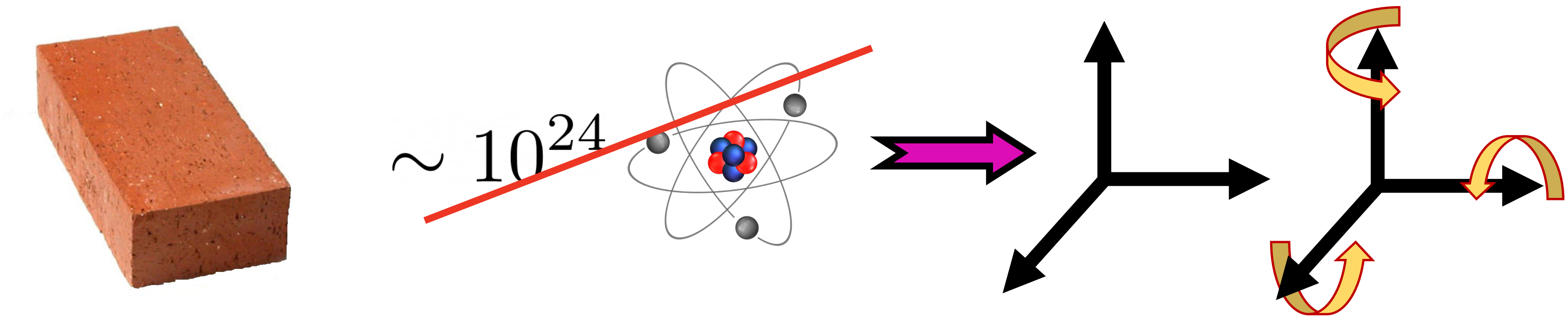
The Reduced Basis Method



$\sim 10^{24}$

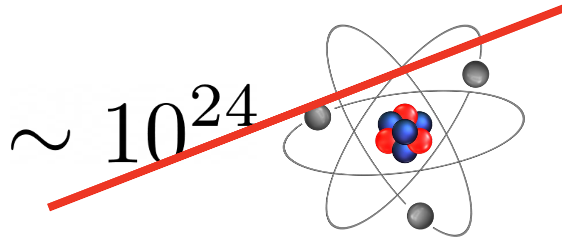


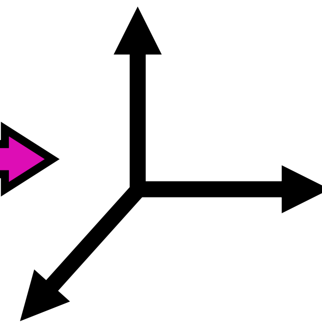
The Reduced Basis Method

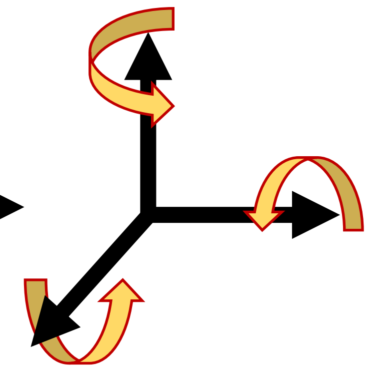


3 translations + 3 rotations

The Reduced Basis Method



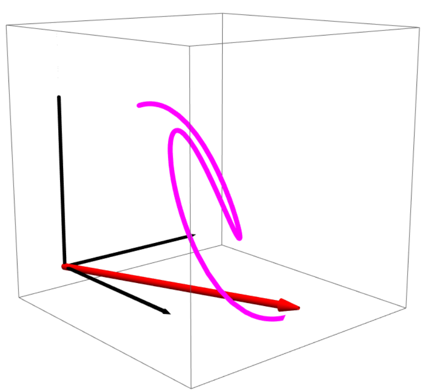


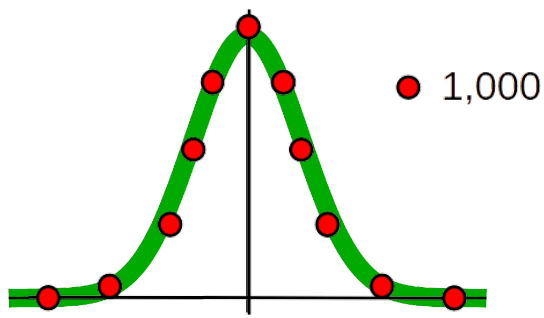


3 translations + 3 rotations

parameters

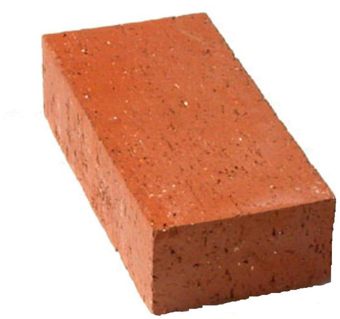
$$\mathcal{H}_\alpha \phi(x) = \lambda \phi(x)$$

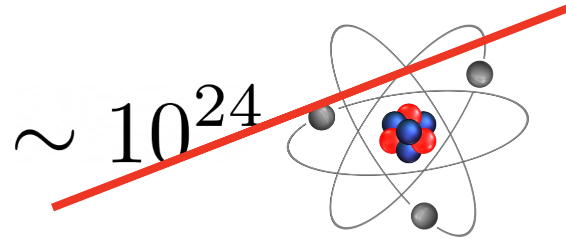


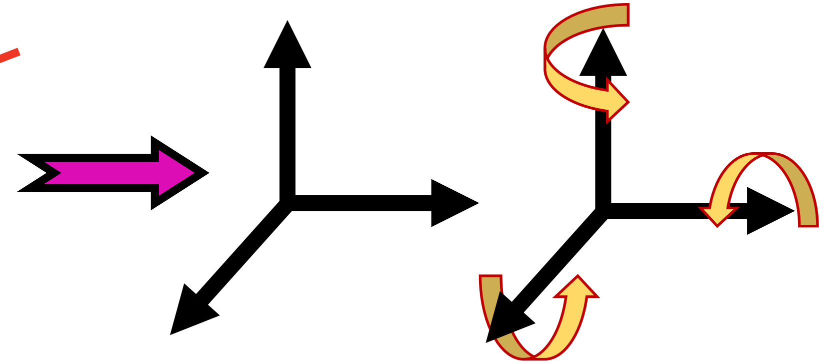


Finite element

The Reduced Basis Method



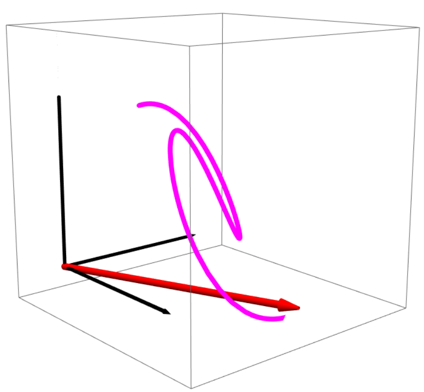


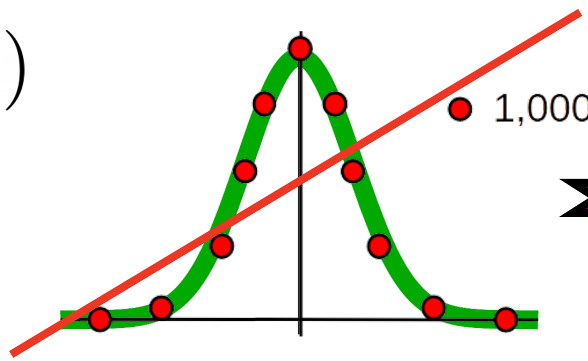


3 translations + 3 rotations

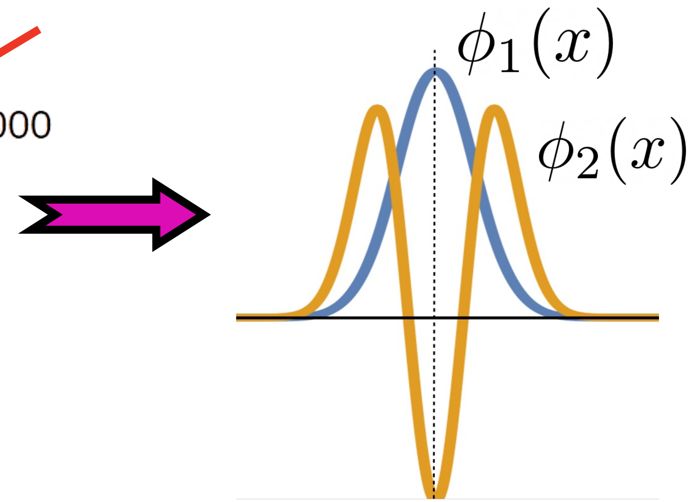
parameters

$$\mathcal{H}_\alpha \phi(x) = \lambda \phi(x)$$



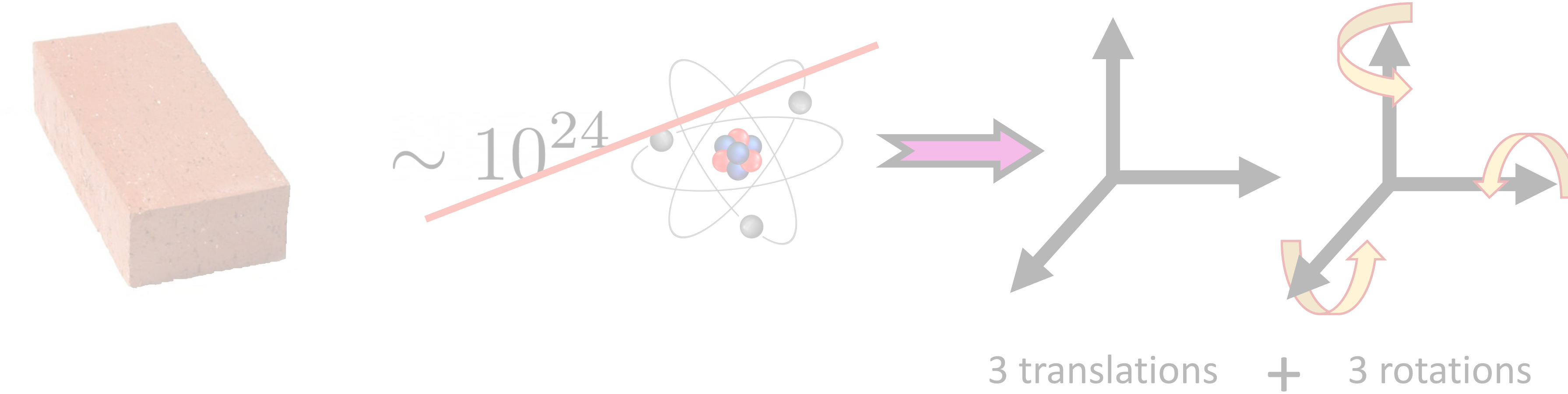


Finite element



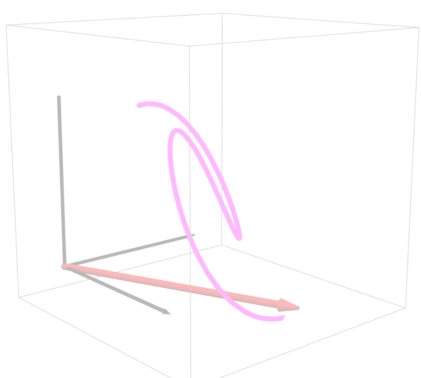
$$\phi(x) \approx a_1\phi_1(x) + a_2\phi_2(x)$$

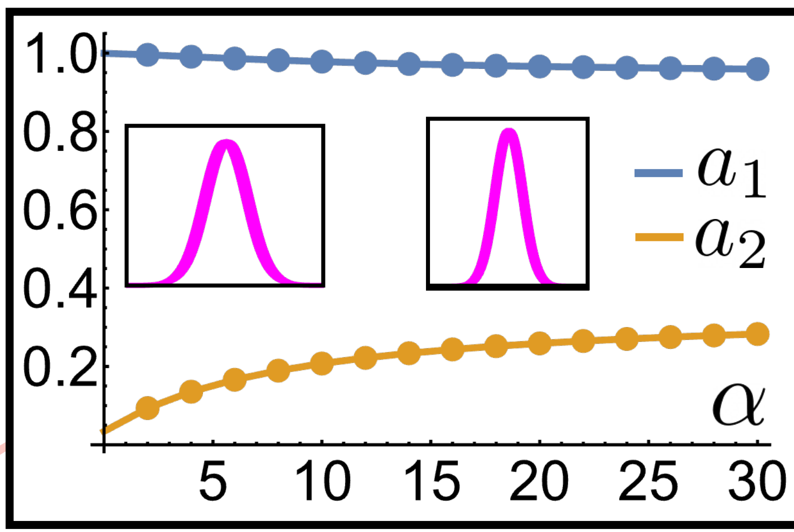
The Reduced Basis Method

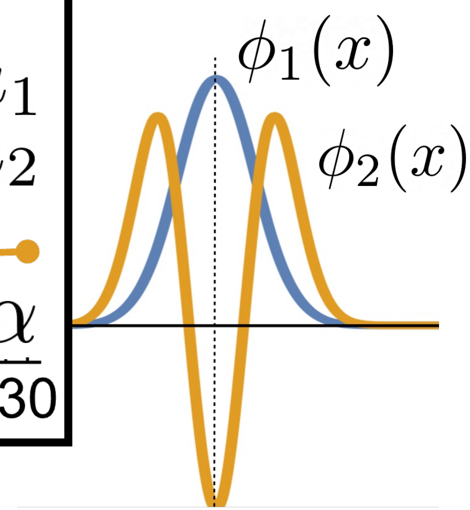


parameters

$$\mathcal{H}_\alpha \phi(x) = \lambda \phi(x)$$

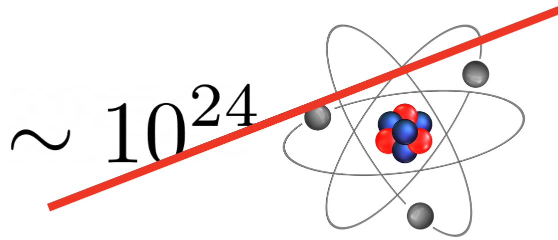


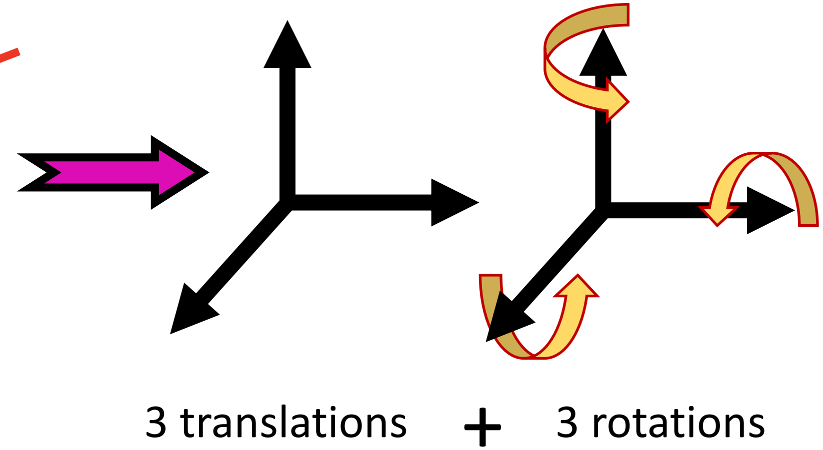


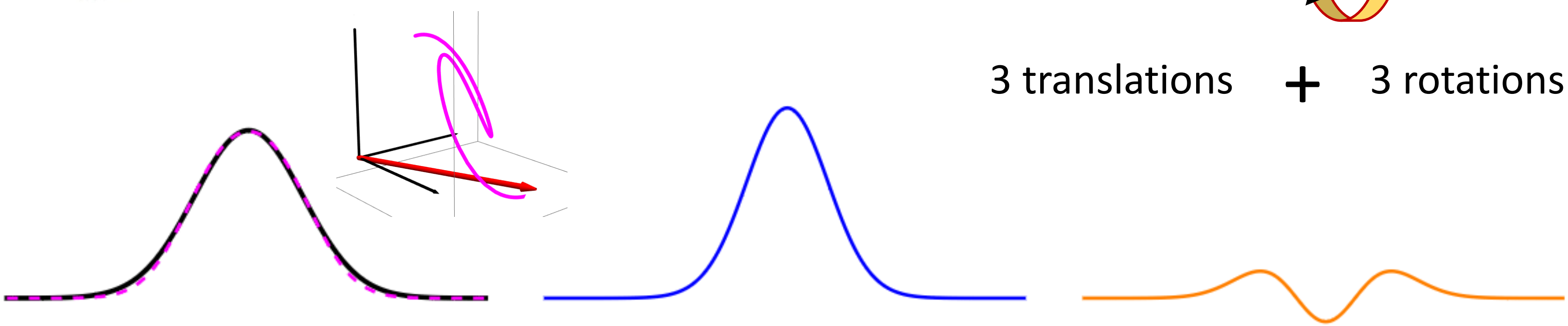


$$\phi(x) \approx a_1\phi_1(x) + a_2\phi_2(x)$$

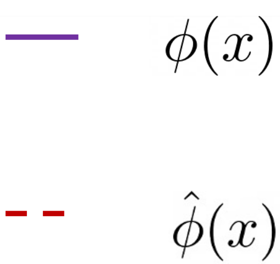
The Reduced Basis Method

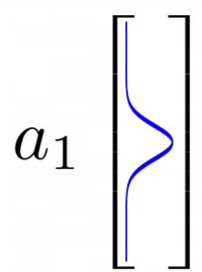


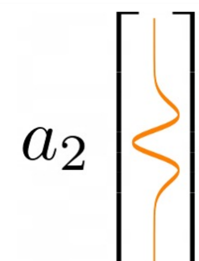




$$\phi(x) \approx a_1\phi_1(x) + a_2\phi_2(x)$$







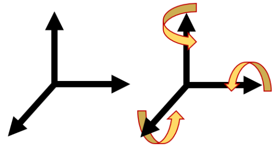
Changing the trapping strength α

The Reduced Basis Method

$$F_\alpha[\phi(x)] = 0$$

General differential equation

$$F_\alpha(\phi) = \left(-\frac{d^2}{dr^2} + \frac{\ell(\ell+1)}{r^2} + U(r,\alpha) - p^2\right)\phi(r) = 0$$

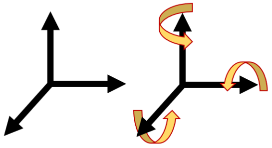


The Reduced Basis Method

$$F_\alpha[\phi(x)] = 0$$

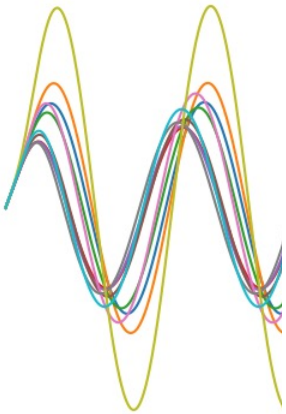
General differential equation

$$F_\alpha(\phi) = \left(-\frac{d^2}{dr^2} + \frac{\ell(\ell+1)}{r^2} + U(r,\alpha) - p^2\right)\phi(r) = 0$$

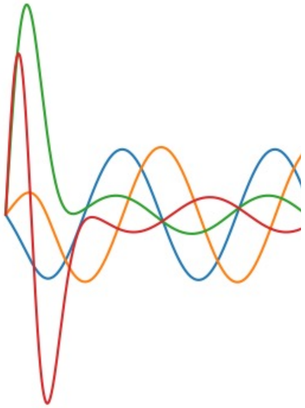


1) Choose a basis

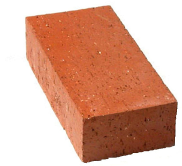
$$\hat{\phi}(x) = \phi_0 + \sum_k^n a_k \phi_k(x)$$



Principal Component Analysis



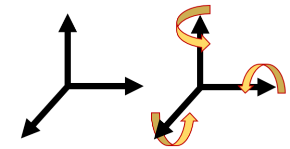
The Reduced Basis Method



$$F_\alpha[\phi(x)] = 0$$

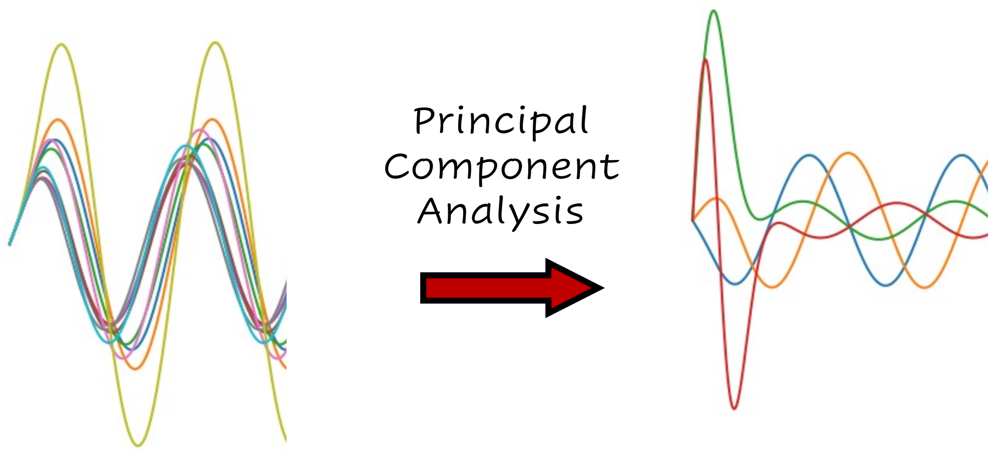
General differential equation

$$F_\alpha(\phi) = \left(-\frac{d^2}{dr^2} + \frac{\ell(\ell+1)}{r^2} + U(r,\alpha) - p^2\right)\phi(r) = 0$$



1) Choose a basis

$$\hat{\phi}(x) = \phi_0 + \sum_k^n a_k \phi_k(x)$$



2) Project onto judges

$$j = \{1, n\} \quad \langle \psi_j | F_\alpha[\hat{\phi}(x)] \rangle = 0$$

One equation
per coefficient

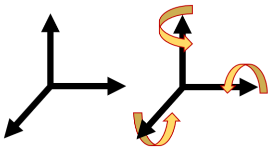
WHY?
(ask me)

The Reduced Basis Method

$$F_\alpha[\phi(x)] = 0$$

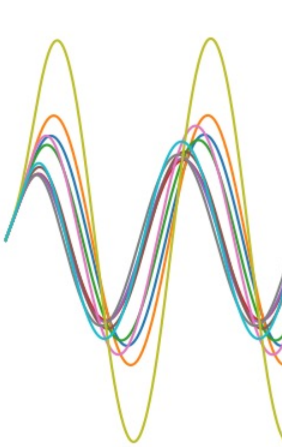
General differential equation

$$F_\alpha(\phi) = \left(-\frac{d^2}{dr^2} + \frac{\ell(\ell+1)}{r^2} + U(r,\alpha) - p^2\right)\phi(r) = 0$$

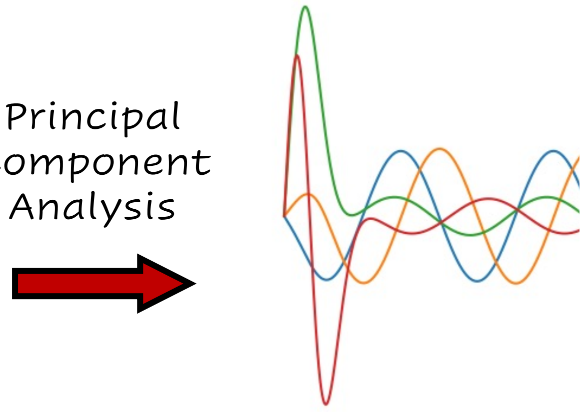


1) Choose a basis

$$\hat{\phi}(x) = \phi_0 + \sum_k^n a_k \phi_k(x)$$



Principal Component Analysis



2) Project onto judges

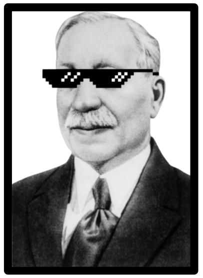
$$j = \{1, n\} \quad \langle \psi_j | F_\alpha[\hat{\phi}(x)] \rangle = 0$$

One equation per coefficient

WHY?
(ask me)

Galerkin

Training and Projecting



https://doi.org/10.1103/PhysRevC.106.054322

The Reduced

https://kylegodbey.github.io/nuclear-rbm

jupyter {book}

Reduced Basis Methods in Nuclear Physics

https://www.frontiersin.org/articles/10.3389/fphy.2022.1092931/full

BUQEYE Guide to Projection-Based Emulators in Nuclear Physics

C. Drischler [1,2,*], **J. A. Melendez** [3], **R. J. Furnstahl** [3], **A. J. Garcia** [3], **and Xilin Zhang** [2]

[1] *Department of Physics and Astronomy & Institute of Nuclear and Particle Physics, Ohio University, Athens, OH 45701, USA*

[2] *Facility for Rare Isotope Beams, Michigan State University, MI 4*

[3] *Department of Physics, The Ohio State University, Columbus, O*

1) Choose a basis

$$\hat{\phi}(x) = \phi_0 + \sum^{n} a_k \phi_k(x)$$

Principal Component Analysis

$\phi_2(x)$

Bayesian Analysis of Nuclear Dynamics

and friends!

Model reduction methods for nuclear emulators

J A Melendez[1], **C Drischler**[2], **R J Furnstahl**[1,*], **A J Garcia**[1] **and Xilin Zhang**[2]

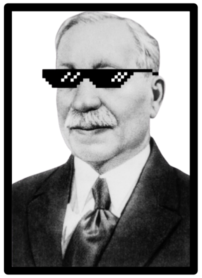
[1] Department of Physics, The Ohio State University, Columbus, OH 43210, United States of America

[2] Facility for Rare Isotope Beams, Michigan State University, MI 48824, United States of America

https://doi.org/10.1088/1361-6471/ac83dd

Galerkin

Training and Projecting



https://doi.org/10.1103/PhysRevC.106.054322

One equation per coefficient

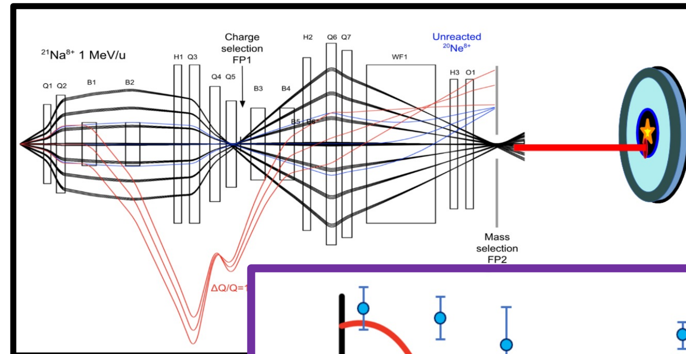
WHY? (ask me)

https://kylegodbey.github.io/nuclear-rbm

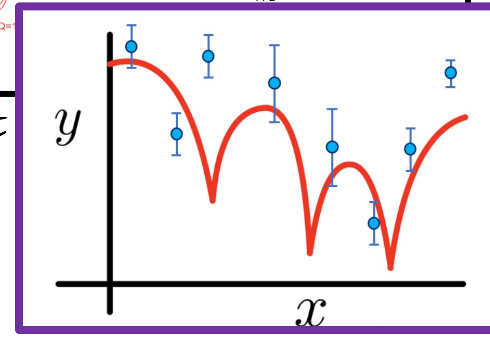
jupyter {book}

Reduced Basis Methods in Nuclear Physics

Motivation



Experiment



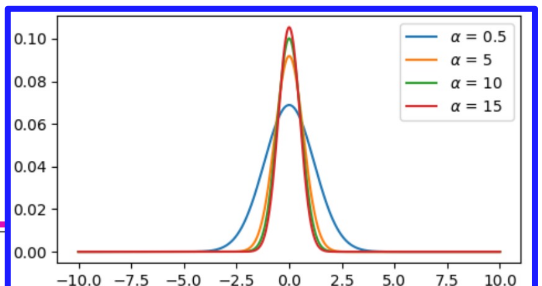
Theory

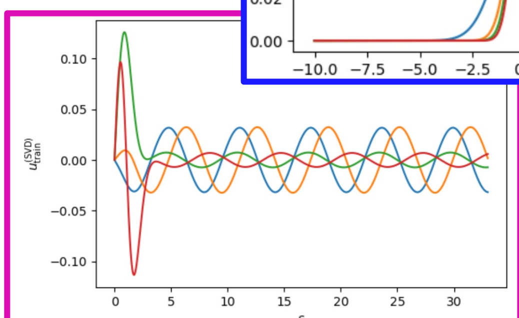
Code

```
#We select four basis and obtain the following gorgeous functions:
nbasis = 4

fig, ax = plt.subplots(dpi=100)
fig.patch.set_facecolor('white')

for i in range(nbasis):
    ax.plot(s_mesh, U[:, i])

ax.set_xlabel(r'$s$')
ax.set_ylabel(r'$u_{\rm train}^{({\rm SVD})}$');
```

Examples

Bound Systems



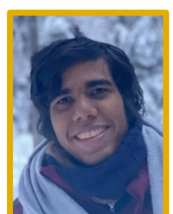


Scattering

(with Empirical Interpolation Method)

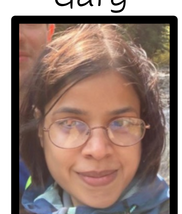
Kyle Beyer

Edgard Bonilla



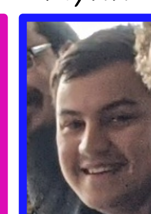
Kyle Godbey

Ruchi Garg



Daniel Odell

Eric Flynn



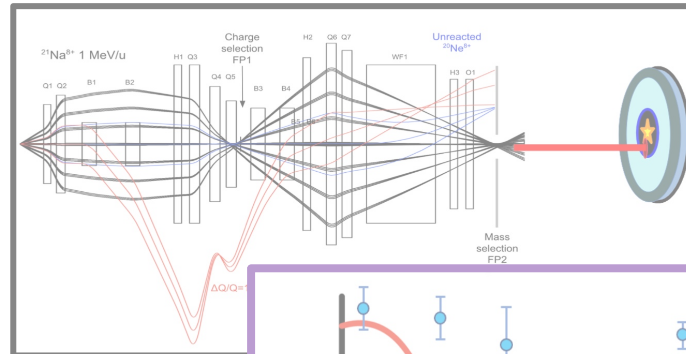
RBM Jupyter Book

https://kylegodbey.github.io/nuclear-rbm https://kylegodbey.github.io/nuclear-rbm

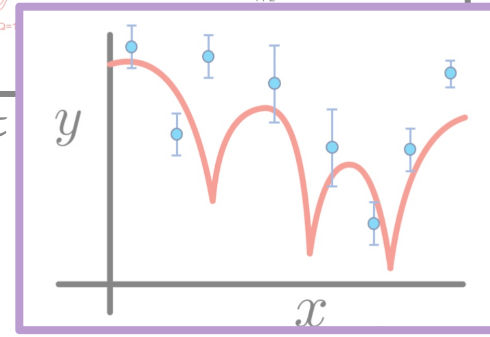
jupyter {book}

Reduced Basis Methods in Nuclear Physics

Motivation



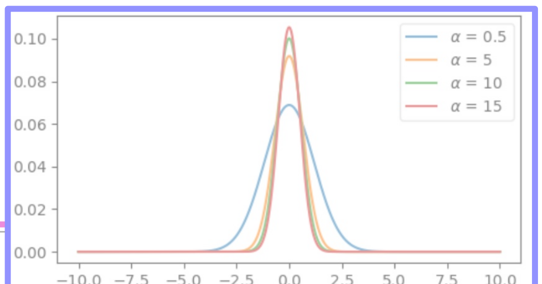
Experiment

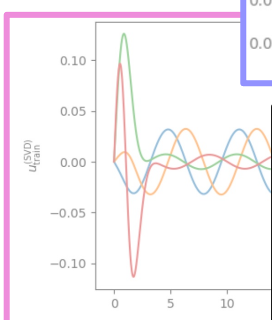


Theory

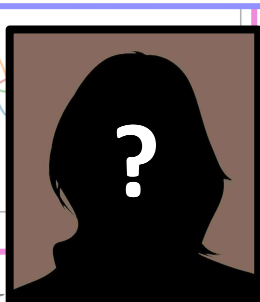
Examples

Bound Systems





Scatt



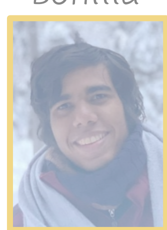
you?

Code

```
#We select four basis and obtain the following gorgeous functions:
nbasis = 4

fig, ax = plt.subplots(dpi=100)
fig.patch.set_facecolor('white')

for i in range(nbasis):
    ax.plot(s_mesh, U[:, i])

ax.set_xlabel(r'$s$')
ax.set_ylabel(r'$u_{\rm train}^{({\rm SVD})}$');
```

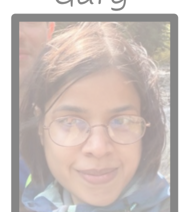
Kyle Beyer

Edgard Bonilla



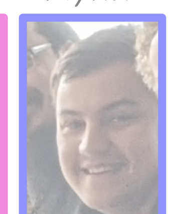
Kyle Godbey

Ruchi Garg



Daniel Odell

Eric Flynn

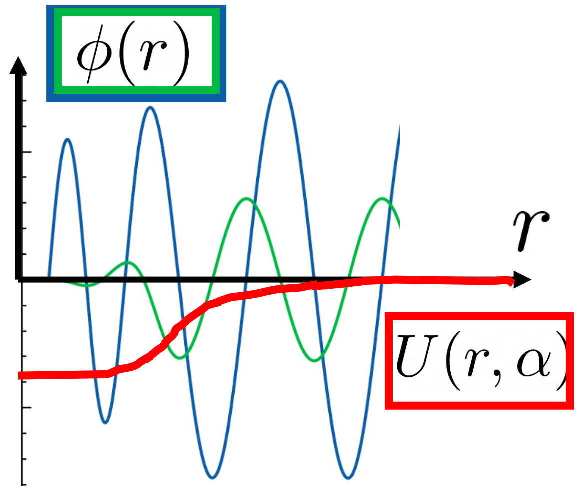


RBM Jupyter Book

https://kylegodbey.github.io/nuclear-rbm

Reduced Basis Method

$$F_\alpha(\phi) = \left(-\frac{d^2}{dr^2} + \frac{\ell(\ell+1)}{r^2} + U(r,\alpha) - p^2\right)\phi(r) = 0$$

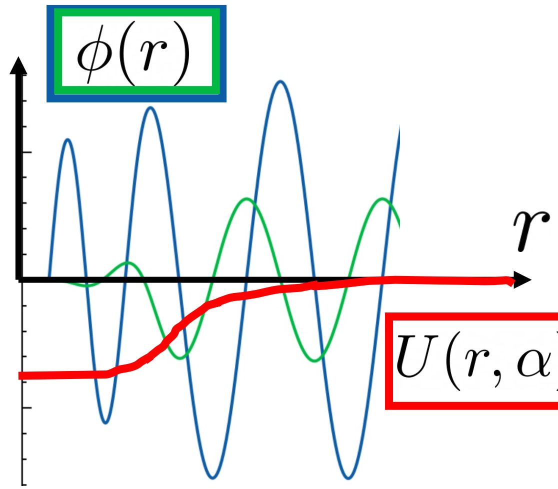


High Fidelity

RBM

Reduced Basis Method

$$F_\alpha(\phi) = \left(-\frac{d^2}{dr^2} + \frac{\ell(\ell+1)}{r^2} + U(r,\alpha) - p^2\right)\phi(r) = 0$$



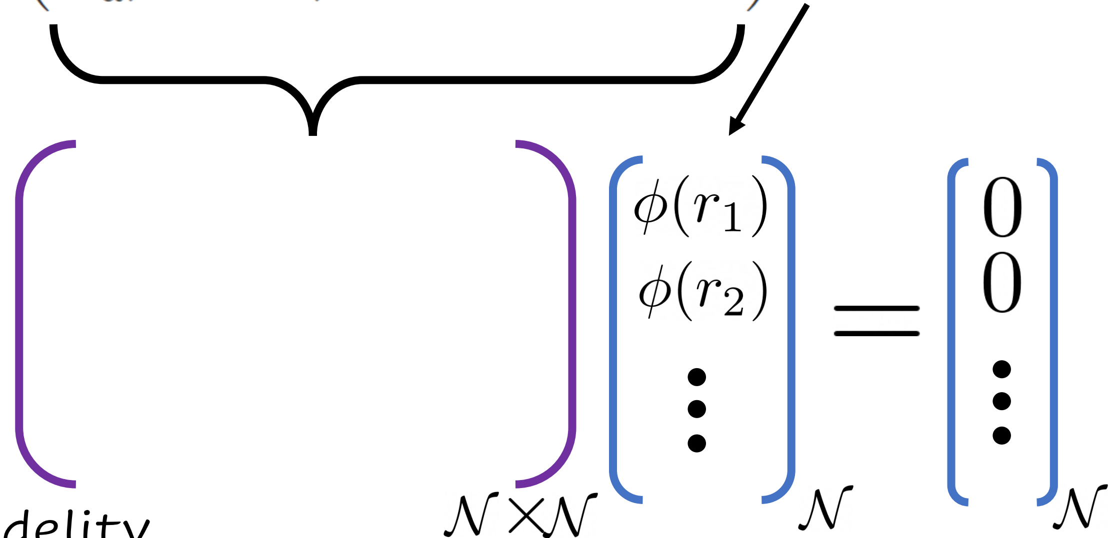
$$\begin{bmatrix} & & \\ & & \\ & & \end{bmatrix}_{\mathcal{N}\times\mathcal{N}} \begin{bmatrix} \phi(r_1) \\ \phi(r_2) \\ \vdots \end{bmatrix}_{\mathcal{N}} = \begin{bmatrix} 0 \\ 0 \\ \vdots \end{bmatrix}_{\mathcal{N}}$$

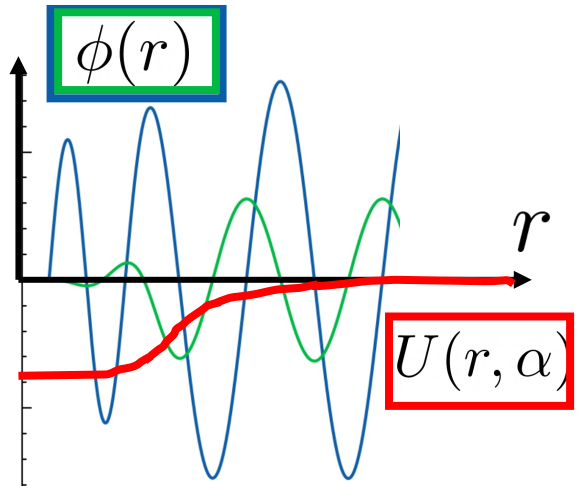
High Fidelity

RBM

Reduced Basis Method

$$F_\alpha(\phi) = \left(-\frac{d^2}{dr^2} + \frac{\ell(\ell+1)}{r^2} + U(r,\alpha) - p^2\right)\phi(r) = 0$$





High Fidelity

RBM

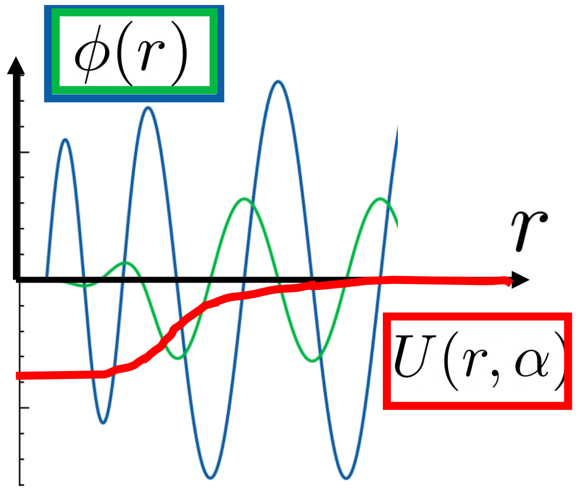
$$\hat{\phi}(r) = \phi_0(r) + \sum_k^n a_k \phi_k(r)$$

$$j = \{1, n\}$$

$$\langle \psi_j | F_\alpha[\hat{\phi}(r)] \rangle = 0$$

Reduced Basis Method

$$F_\alpha(\phi) = \left(-\frac{d^2}{dr^2} + \frac{\ell(\ell+1)}{r^2} + U(r,\alpha) - p^2\right)\phi(r) = 0$$



$$\Big(\quad\Big)_{\mathcal{N}\times\mathcal{N}} \begin{bmatrix}\phi(r_1)\\ \phi(r_2)\\ \vdots\end{bmatrix}_{\mathcal{N}} = \begin{bmatrix}0\\ 0\\ \vdots\end{bmatrix}_{\mathcal{N}}$$

High Fidelity

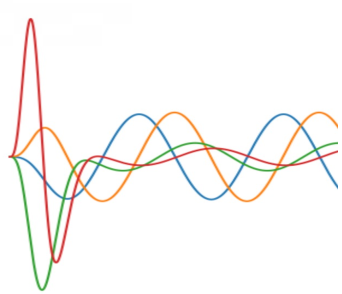
RBM

$$\hat{\phi}(r) = \phi_0(r) + \sum_k^n a_k \phi_k(r)$$

Free solution

$U(r,\alpha) = 0$

Principal components



$j = \{1, n\}$

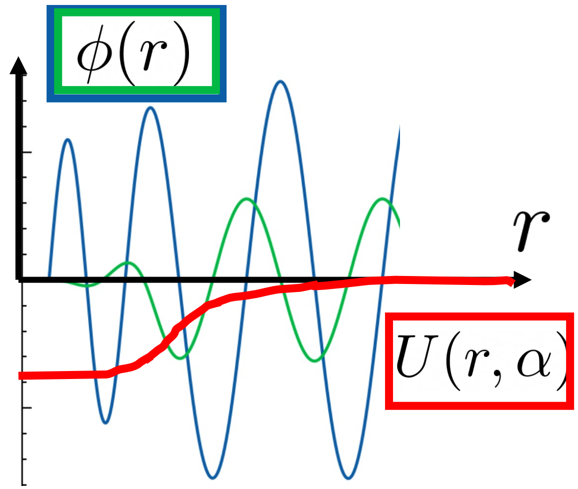
$$\langle \psi_j | F_\alpha[\hat{\phi}(r)] \rangle = 0$$

Equations for the coefficients $a_k(\alpha)$

$$\int \psi_j(r) F_\alpha[\hat{\phi}(r)] dr = 0$$

Reduced Basis Method

$$F_\alpha(\phi) = \left(-\frac{d^2}{dr^2} + \frac{\ell(\ell+1)}{r^2} + U(r,\alpha) - p^2\right)\phi(r) = 0$$



$$\left[\quad\right]_{\mathcal{N}\times\mathcal{N}} \begin{bmatrix}\phi(r_1)\\ \phi(r_2)\\ \vdots\end{bmatrix}_{\mathcal{N}} = \begin{bmatrix}0\\0\\ \vdots\end{bmatrix}_{\mathcal{N}}$$

High Fidelity

RBM

$$\hat{\phi}(r) = \phi_0(r) + \sum_k^n a_k\phi_k(r)$$

$$\langle\psi_j|F_\alpha[\hat{\phi}(r)]\rangle = 0 \qquad j = \{1, n\}$$

$$\hat{\phi} = \begin{bmatrix}a_1\\ a_2\\ \vdots\end{bmatrix}_n$$

$$n\begin{cases} 3.1a_1 - 4.2a_2 + 10.1\alpha a_1 + \ldots = 0 \\ -2.1a_1 + 2.2a_2 + 0.3\alpha a_1 + \ldots = 0 \\ \vdots \end{cases}$$

Reduced Basis Method

$$F_\alpha(\phi) = \left(-\frac{d^2}{dr^2} + \frac{\ell(\ell+1)}{r^2} + U(r,\alpha) - p^2\right)\phi(r) = 0$$

$$\Big(\qquad\qquad\Big)_{\mathcal{N}\times\mathcal{N}} \begin{bmatrix} \phi(r_1) \\ \phi(r_2) \\ \vdots \end{bmatrix}_{\mathcal{N}} = \begin{bmatrix} 0 \\ 0 \\ \vdots \end{bmatrix}_{\mathcal{N}}$$

High Fidelity

RBM

$$\hat{\phi}(r) = \phi_0(r) + \sum_k^n a_k \phi_k(r) \qquad j = \{1, n\} \quad \langle \psi_j | F_\alpha[\hat{\phi}(r)] \rangle = 0$$

$$\hat{\phi} = \begin{bmatrix} a_1 \\ a_2 \\ \vdots \end{bmatrix}_n$$

$$n \begin{cases} 3.1a_1 - 4.2a_2 + 10.1\alpha a_1 + \ldots = 0 \\ -2.1a_1 + 2.2a_2 + 0.3\alpha a_1 + \ldots = 0 \\ \vdots \end{cases}$$

Reduced Basis Method

Empirical Interpolation Method

$$F_\alpha(\phi) = \left(-\frac{d^2}{dr^2} + \frac{\ell(\ell+1)}{r^2} + \boxed{U(r,\alpha)} - p^2\right)\phi(r) = 0$$

$$\langle\psi_j|F_\alpha[\hat{\phi}(r)]\rangle = \int \psi_j(r)F_\alpha[\hat{\phi}(r)]dr = 0$$

Reduced Basis Method

Empirical Interpolation Method

$$F_\alpha(\phi) = \left(-\frac{d^2}{dr^2} + \frac{\ell(\ell+1)}{r^2} + \boxed{U(r,\alpha)} - p^2\right)\phi(r) = 0$$

$$\langle \psi_j | F_\alpha[\hat{\phi}(r)] \rangle = \int \psi_j(r) F_\alpha[\hat{\phi}(r)] dr = 0$$

$$U(r,\alpha) = \alpha \frac{1}{(1+e^r)}$$

Reduced Basis Method

Empirical Interpolation Method

$$F_\alpha(\phi) = \left(-\frac{d^2}{dr^2} + \frac{\ell(\ell+1)}{r^2} + U(r,\alpha) - p^2\right)\phi(r) = 0$$

$$\langle \psi_j | F_\alpha[\hat{\phi}(r)] \rangle = \int \psi_j(r) F_\alpha[\hat{\phi}(r)] dr = 0$$

$$U(r,\alpha) = \alpha \frac{1}{(1+e^r)} \Rightarrow \alpha \underbrace{\int \psi_j(r) \frac{1}{(1+e^r)} \hat{\phi}(r) dr}_{\mathcal{N}}$$

Reduced Basis Method

Empirical Interpolation Method

$$F_\alpha(\phi) = \left(-\frac{d^2}{dr^2} + \frac{\ell(\ell+1)}{r^2} + U(r,\alpha) - p^2\right)\phi(r) = 0$$

$$\langle \psi_j | F_\alpha[\hat{\phi}(r)] \rangle = \int \psi_j(r) F_\alpha[\hat{\phi}(r)] dr = 0$$

$$U(r,\alpha) = \alpha \frac{1}{(1+e^r)} \Rightarrow \alpha \underbrace{\int \psi_j(r) \frac{1}{(1+e^r)} \hat{\phi}(r) dr}_{\mathcal{N}}$$

$$U(r,\alpha) = \frac{1}{(1+e^{\alpha r})} \Rightarrow$$

Reduced Basis Method

Empirical Interpolation Method

$$F_\alpha(\phi) = \left(-\frac{d^2}{dr^2} + \frac{\ell(\ell+1)}{r^2} + \boxed{U(r,\alpha)} - p^2\right)\phi(r) = 0$$

$$\langle \psi_j | F_\alpha[\hat{\phi}(r)] \rangle = \int \psi_j(r) F_\alpha[\hat{\phi}(r)] dr = 0$$

$$U(r,\alpha) = \alpha \frac{1}{(1+e^r)} \Rightarrow \alpha \underbrace{\int \psi_j(r) \frac{1}{(1+e^r)} \hat{\phi}(r) dr}_{\mathcal{N}}$$

$$U(r,\alpha) = \frac{1}{(1+e^{\alpha r})} \Rightarrow \overbrace{\int \psi_j(r) \frac{1}{(1+e^{\alpha r})} \hat{\phi}(r) dr}^{\mathcal{N}}$$

Reduced Basis Method

Empirical Interpolation Method

$$F_\alpha(\phi) = \left(-\frac{d^2}{dr^2} + \frac{\ell(\ell+1)}{r^2} + U(r,\alpha) - p^2\right)\phi(r) = 0$$

$$U(r,\alpha) = \frac{1}{(1+e^{\alpha r})} \approx \sum_i^m b_i(\alpha) f(r)$$

Reduced Basis Method

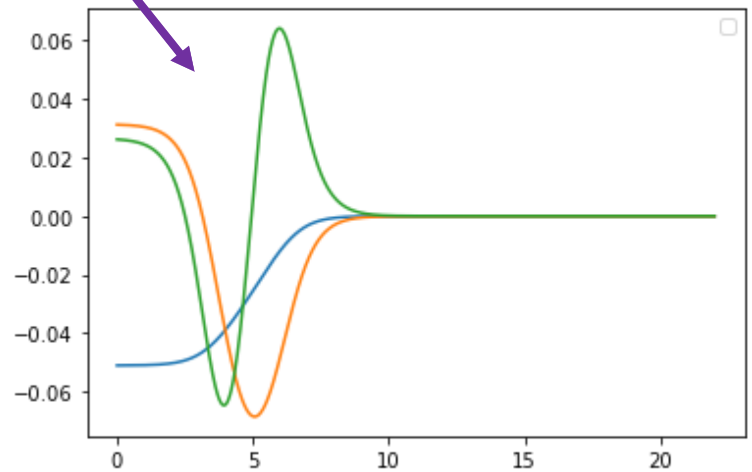
Empirical Interpolation Method

$$F_\alpha(\phi) = \left(-\frac{d^2}{dr^2} + \frac{\ell(\ell+1)}{r^2} + U(r,\alpha) - p^2\right)\phi(r) = 0$$

$$U(r,\alpha) = \frac{1}{(1+e^{\alpha r})} \approx \sum_i^m b_i(\alpha) f(r)$$

Principal components of $U(r,\alpha)$

Obtained by interpolation



$$U(r_j,\alpha) - \sum_i^m b_i(\alpha) f(r_j) = 0$$

$$j = \{1, m\}$$

Reduced Basis Method

Empirical Interpolation Method

$$F_\alpha(\phi) = \left(-\frac{d^2}{dr^2} + \frac{\ell(\ell+1)}{r^2} + \boxed{U(r,\alpha)} - p^2\right)\phi(r) = 0$$

$$U(r,\alpha) = \frac{1}{(1+e^{\alpha r})} \approx \sum_i^m b_i(\alpha) f(r)$$

$$\int \psi_j(r) U(r,\alpha) \hat{\phi}(r) dr$$

$$\approx \sum_i^m b_i(\alpha) \int \psi_j(r) f(r) \hat{\phi}_k(r) dr$$

Reduced Basis Method

Empirical Interpolation Method

$$F_\alpha(\phi) = \left(-\frac{d^2}{dr^2} + \frac{\ell(\ell+1)}{r^2} + U(r,\alpha) - p^2\right)\phi(r) = 0$$

$$U(r,\alpha) = \frac{1}{(1+e^{\alpha r})} \approx \sum_i^m b_i(\alpha) f(r)$$

$$\int \psi_j(r) U(r,\alpha) \hat{\phi}(r) dr$$

Computed only once

$$\approx \sum_i^m b_i(\alpha) \int \psi_j(r) f(r) \hat{\phi}_k(r) dr$$

Reduced Basis Method

Empirical Interpolation Method

$$F_\alpha(\phi) = \left(-\frac{d^2}{dr^2} + \frac{\ell(\ell+1)}{r^2} + U(r,\alpha) - p^2\right)\phi(r) = 0$$

Optical Potential

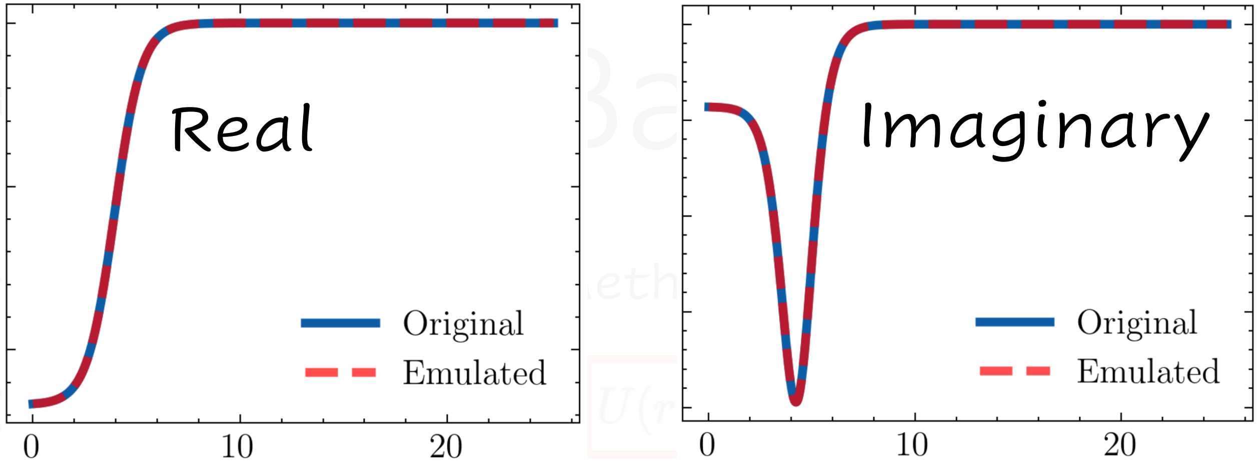
$$U(r,\alpha) = -V_v\left[1+\exp\left(\frac{r-R_v}{a_v}\right)\right] - iW_v\left[1+\exp\left(\frac{r-R_w}{a_w}\right)\right] - i4a_dW_d\frac{d}{dr}\left[1+\exp\left(\frac{r-R_d}{a_d}\right)\right]$$

$$\int \psi_j(r)U(r,\alpha)\hat{\phi}(r)dr$$

Computed only once

$$\approx \sum_i^m b_i(\alpha)\int \psi_j(r)f(r)\hat{\phi}_k(r)dr$$

```python
op = rose.InteractionEIM(
    optical_potential,
    7,
    MU,
    energy,
    bounds,
    is_complex = True,
#     explicit_training = True,
    n_basis = 10
)
```



Optical Potential

$$U(r,\alpha) = -V_v\left[1+\exp\left(\frac{r-R_v}{a_v}\right)\right] - iW_v\left[1+\exp\left(\frac{r-R_w}{a_w}\right)\right]$$

$$-i4a_dW_d\frac{d}{dr}\left[1+\exp\left(\frac{r-R_d}{a_d}\right)\right]$$

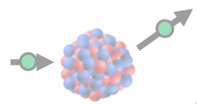
```
op = rose.InteractionEIM(
    optical_potential,
    7,
    MU,
    energy,
    bounds,
    is_complex = True,
#     explicit_training = True,
    n_basis = 10
)
```

Computed only once

$$\approx \sum_{i}^{m} b_i(\alpha) \int \psi_j(r) f(r) \hat{\phi}_k(r) dr$$

Outline

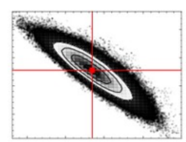
Main ideas

Nuclear Scattering 

Reduced Basis Method 

Main results

Accuracy and time 

Going Bayesian with Surmise 

Tutorials

1) Emulating without ROSE 

2) Emulating with ROSE

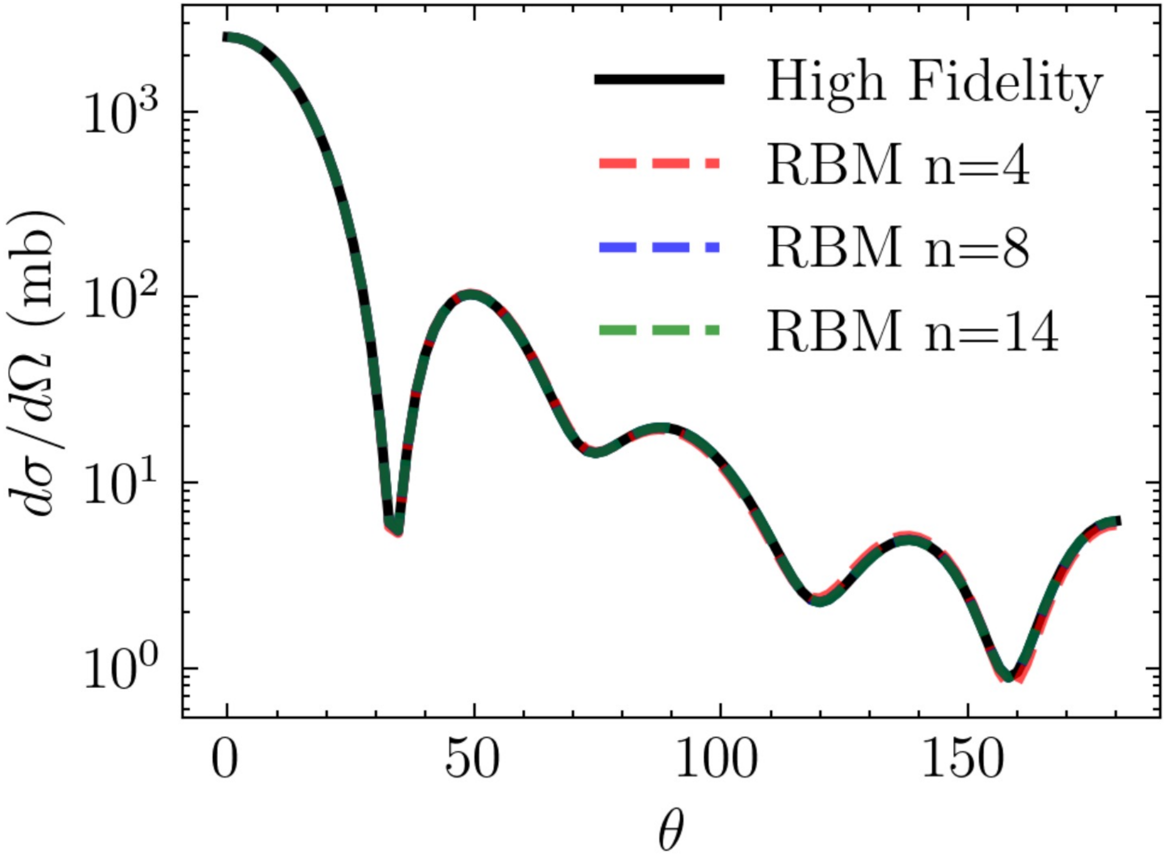
It works!

5,000 times faster than the high fidelity*!

1 milliseconds

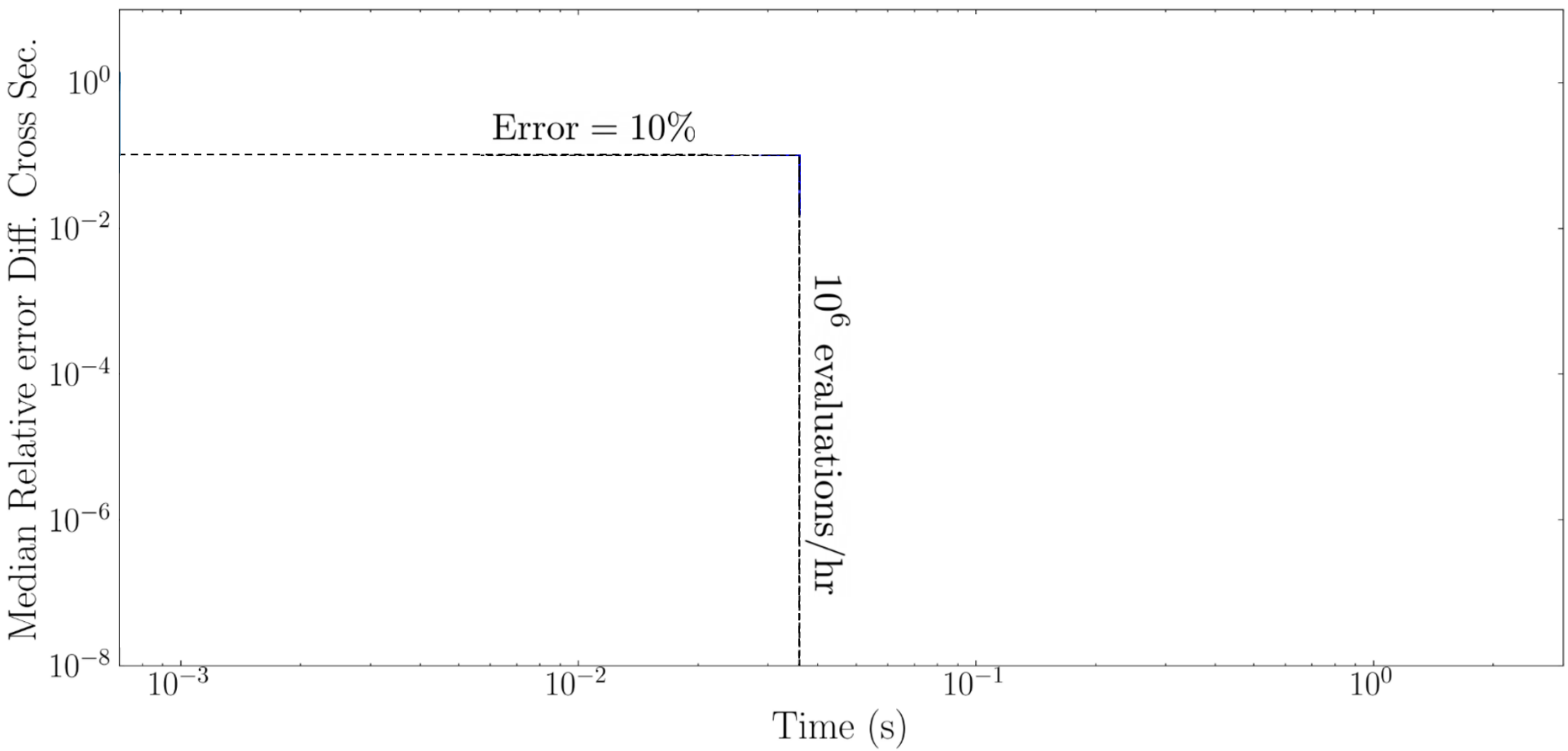
vs

5 seconds

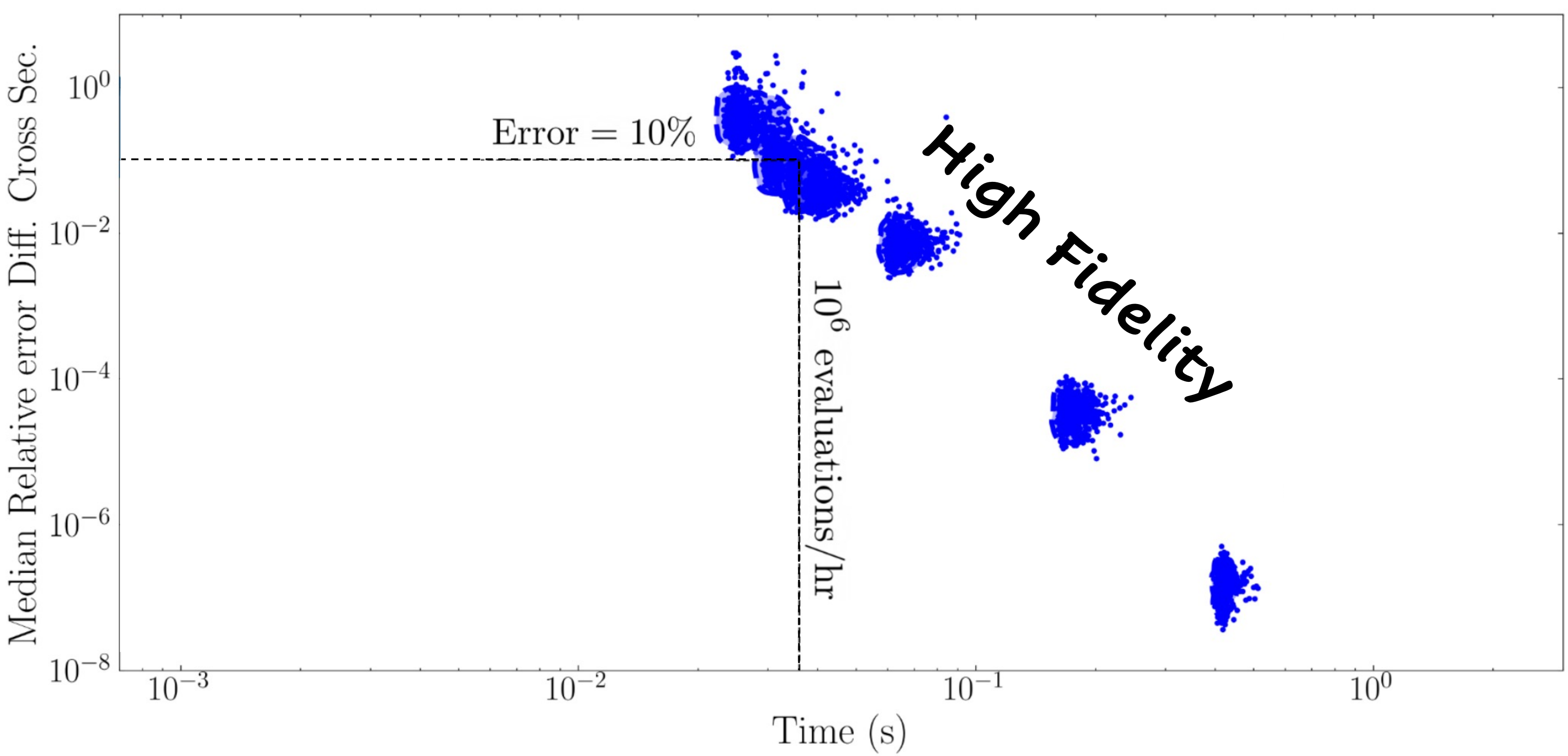


*Performance of the reduced basis method applied to nuclear scattering in relation to the high-fidelity solver is subject to variability depending on the actual efficiency of the high-fidelity solver, leading one to wonder if this cautionary disclaimer, hidden within the depths of small print, might inadvertently spur contemplation upon the complexities and interdependencies of computational methods and their inherent limitations within the domain of nuclear physics. Certain conditions may apply to your region, please consult your nearest computational Doctor.

Computation Accuracy vs Time

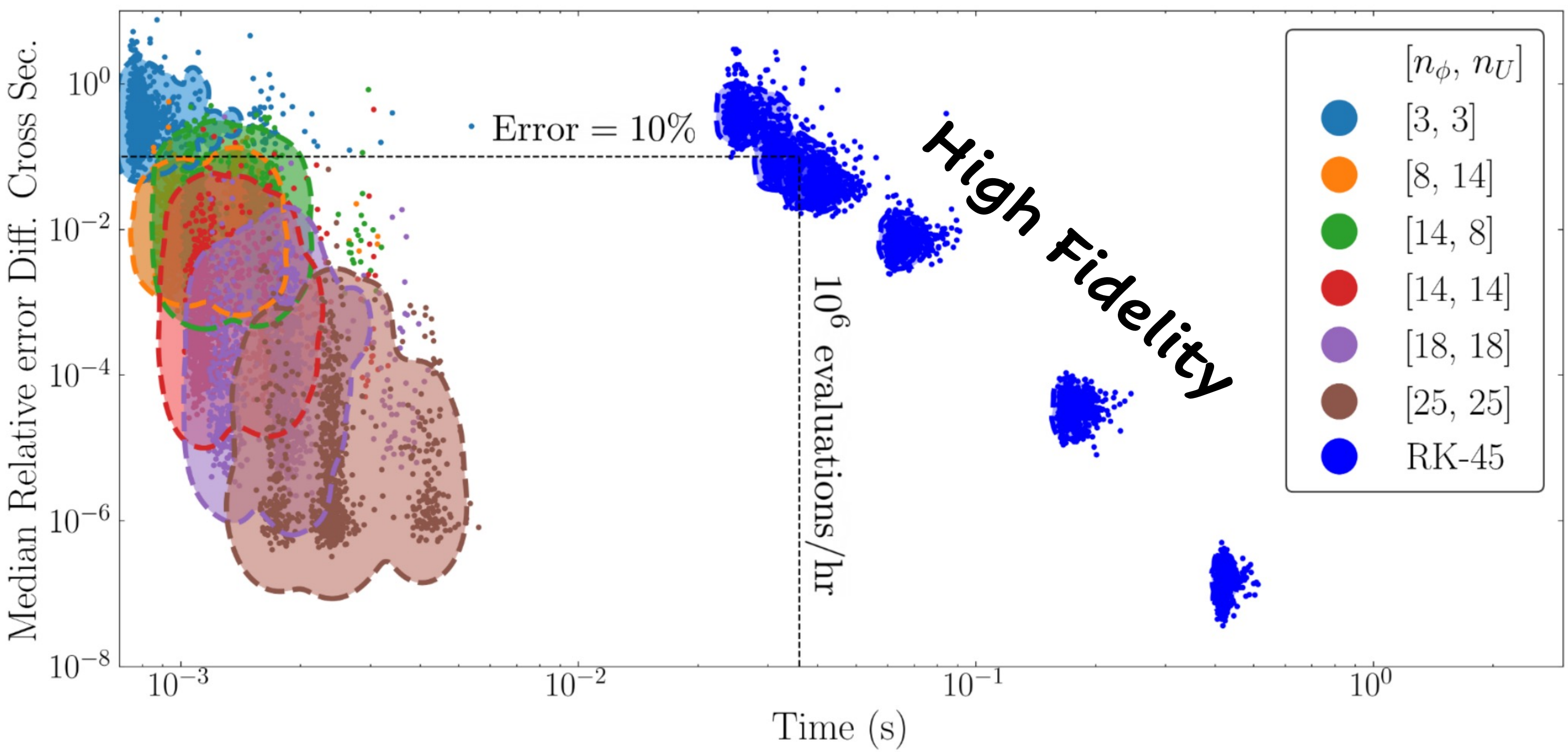


Computation Accuracy vs Time

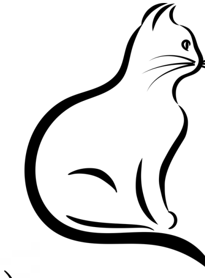


Computation Accuracy vs Time

ROSE



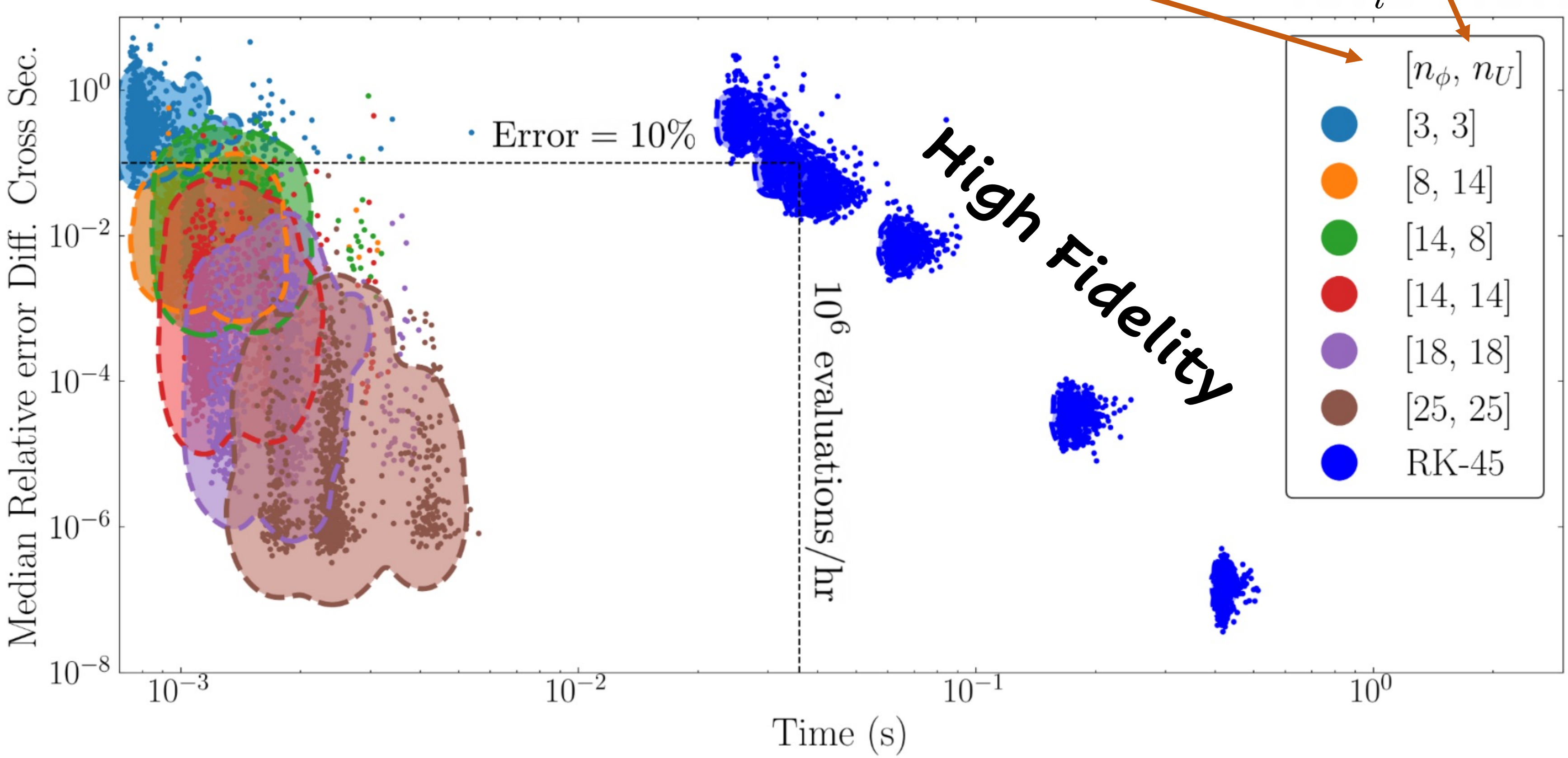
Computation Accuracy vs Time



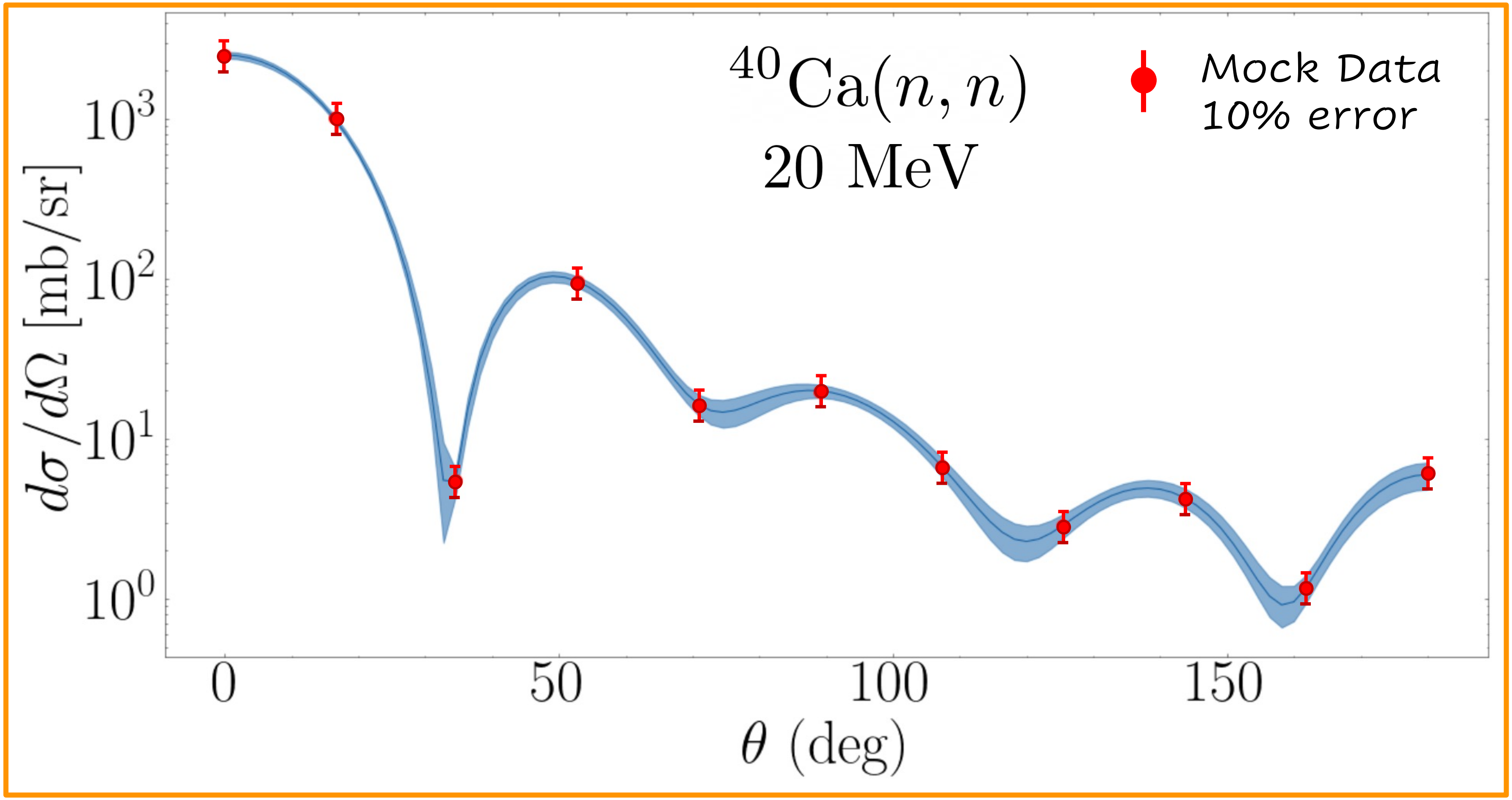
$U(r,\alpha) \approx \sum_i^m b_i(\alpha) f(r)$

$\hat{\phi}(x) = \phi_0 + \sum_k^n a_k \phi_k(x)$

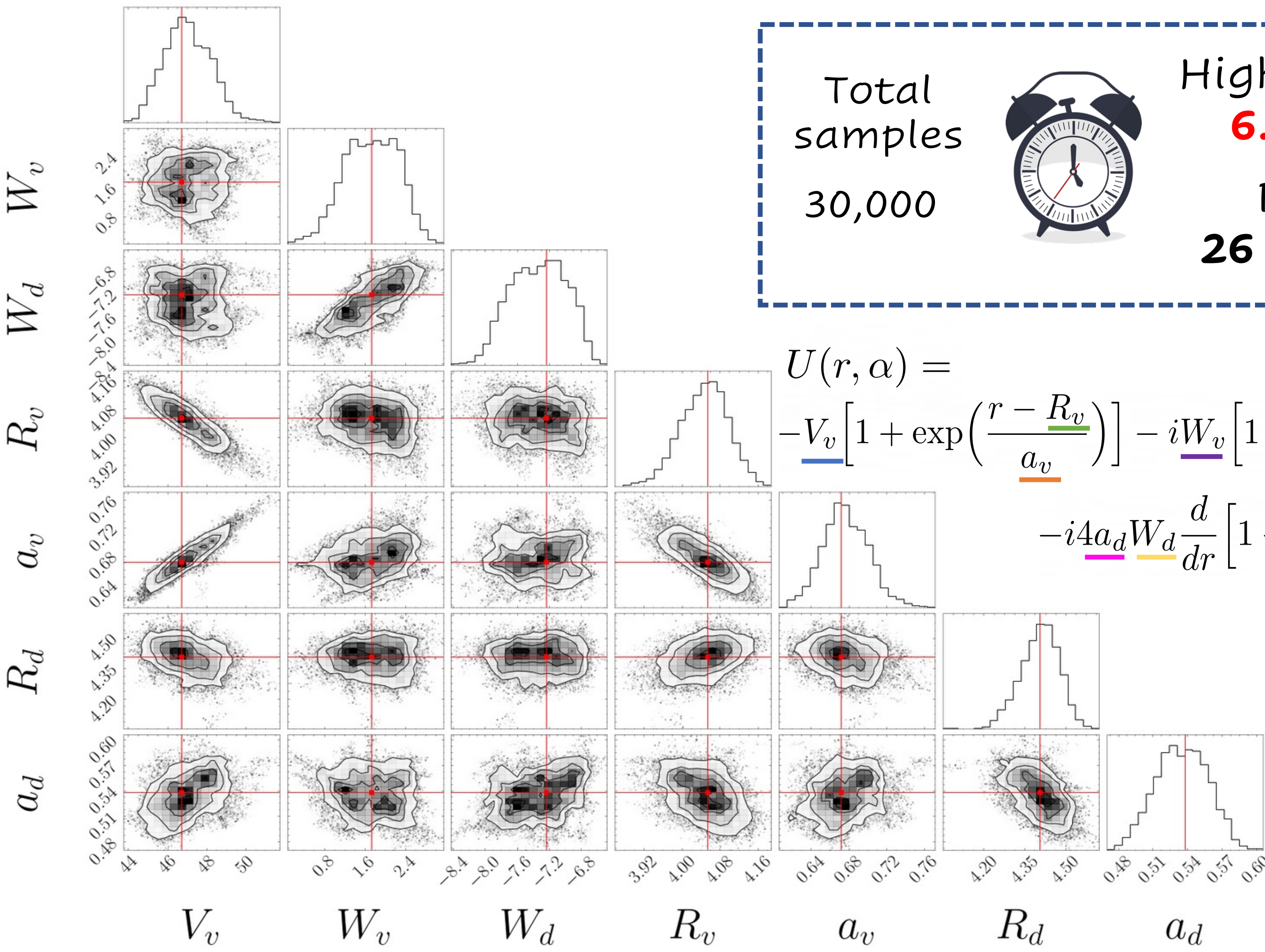
ROSE



Going Bayesian with Surmise



Going Bayesian with Surmise



Total samples

30,000

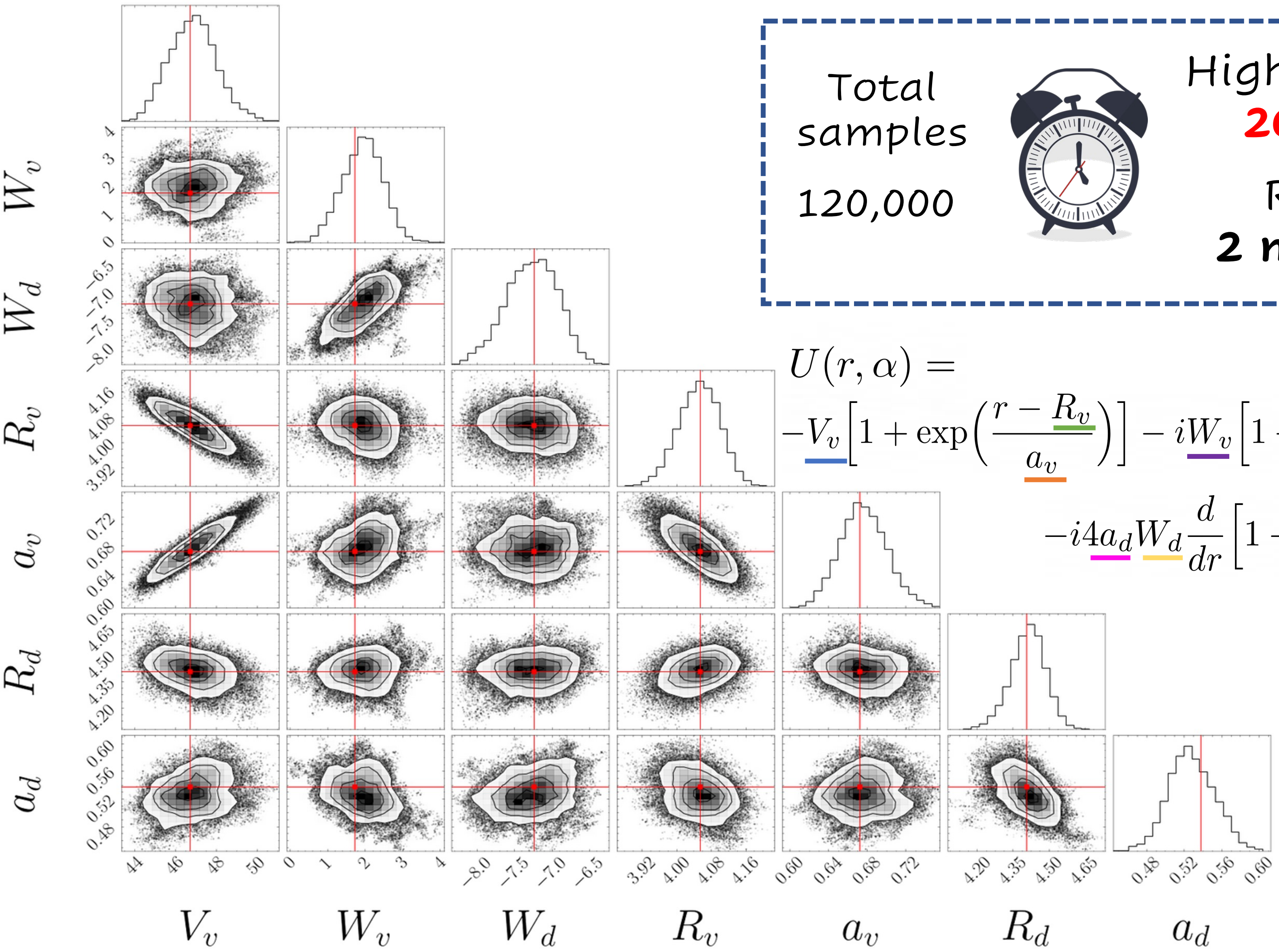
High Fidelity: **6.5 hours**

ROSE: **26 seconds**

$$U(r,\alpha) = -V_v\left[1+\exp\left(\frac{r-R_v}{a_v}\right)\right] - iW_v\left[1+\exp\left(\frac{r-R_v}{a_v}\right)\right] - i4a_d W_d \frac{d}{dr}\left[1+\exp\left(\frac{r-R_d}{a_d}\right)\right]$$

```
Vv0 = 46.72
Wv0 = 1.723
Rv0 = 4.053
av0 = 0.671
Wd0 = -7.23
Rd0 = 4.405
ad0 = 0.537
```

Going Bayesian with Surmise



Total samples

120,000

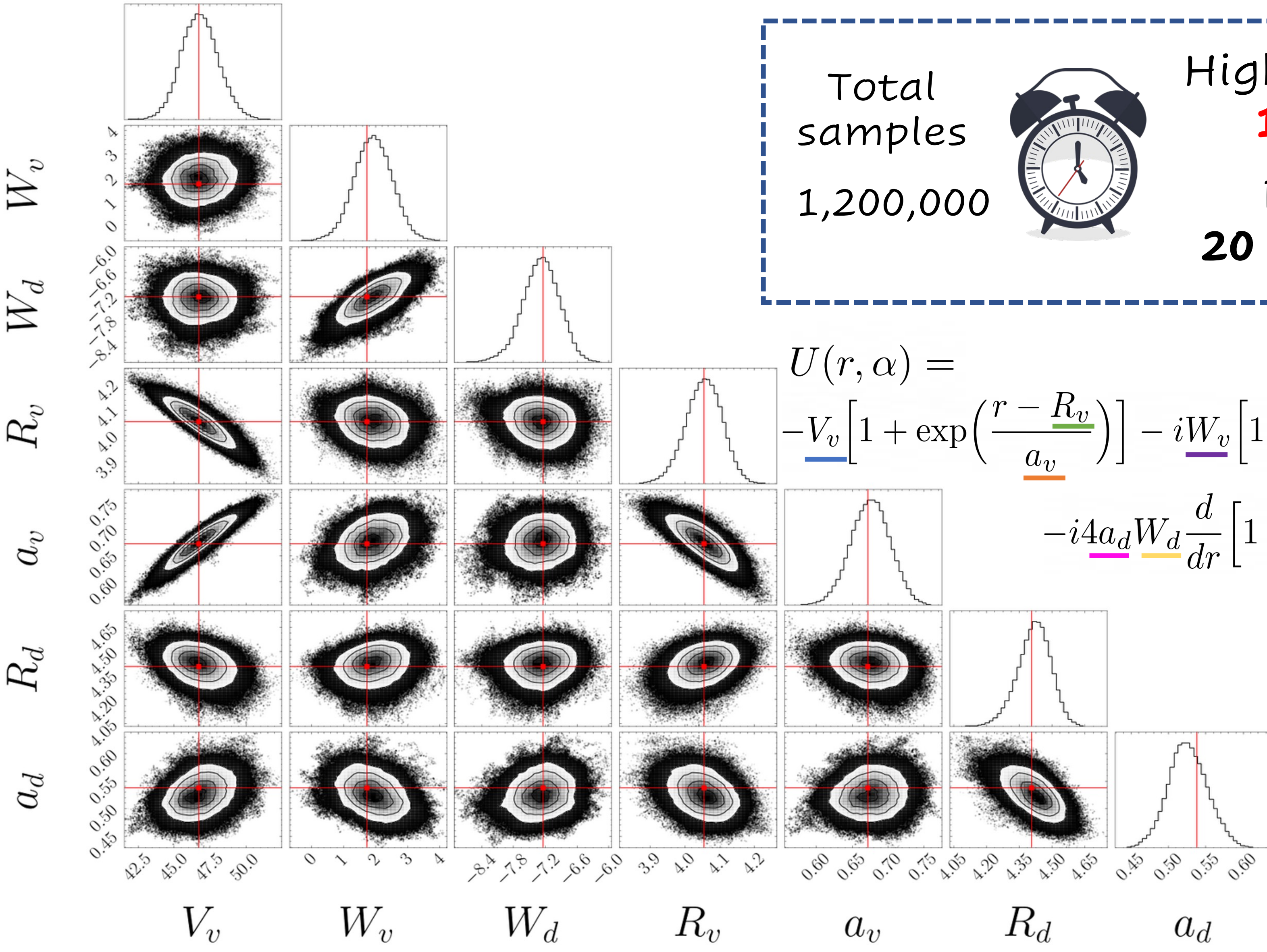
High Fidelity:
26 hours

ROSE:
2 minutes

$$U(r,\alpha) =$$
$$-V_v\left[1+\exp\left(\frac{r-R_v}{a_v}\right)\right] - iW_v\left[1+\exp\left(\frac{r-R_v}{a_v}\right)\right]$$
$$-i4a_dW_d\frac{d}{dr}\left[1+\exp\left(\frac{r-R_d}{a_d}\right)\right]$$

```
Vv0 = 46.72
Wv0 = 1.723
Rv0 = 4.053
av0 = 0.671
Wd0 = -7.23
Rd0 = 4.405
ad0 = 0.537
```

Going Bayesian with Surmise



Total samples

1,200,000

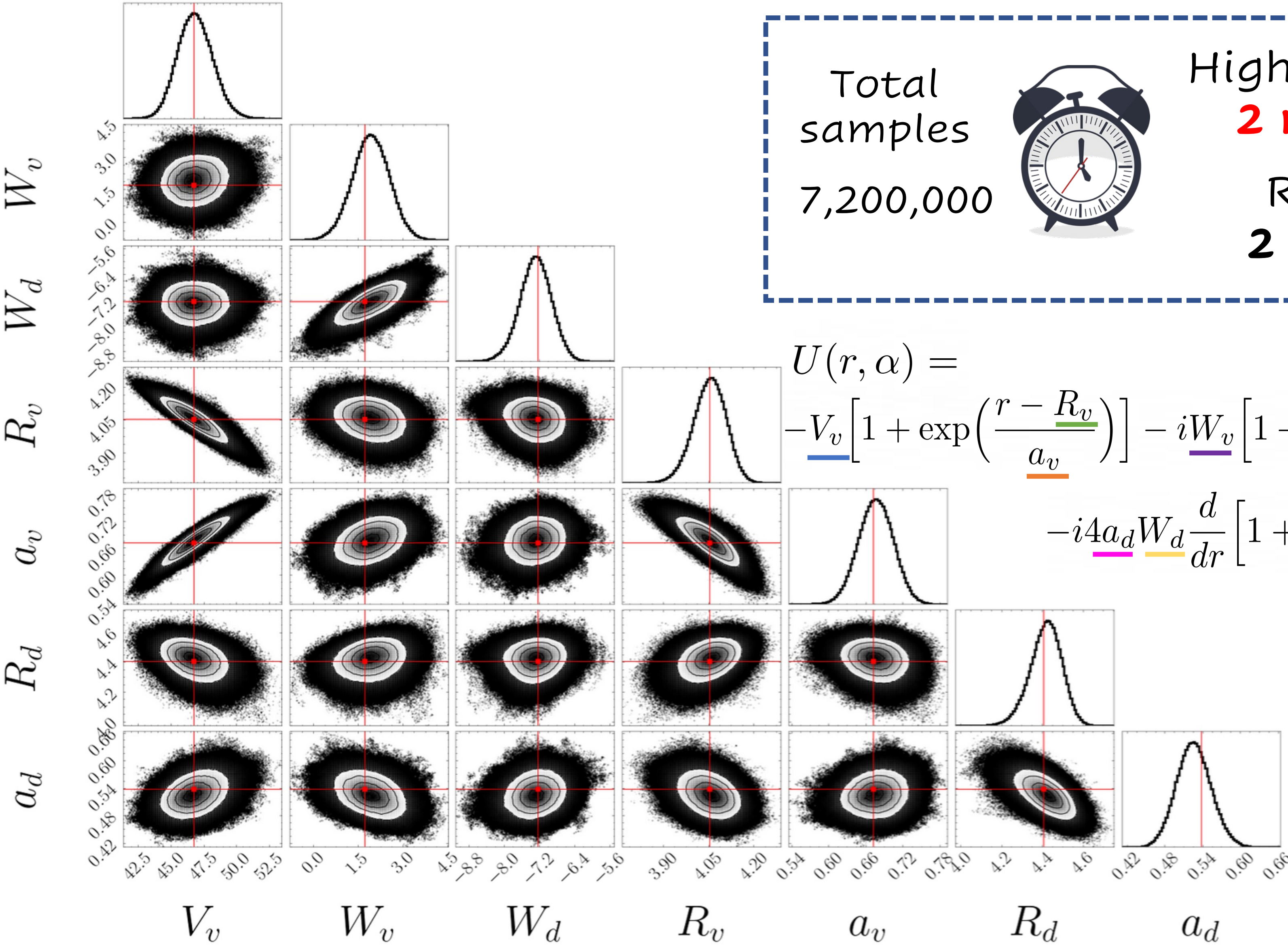
High Fidelity: **11 days**

ROSE: **20 minutes**

$$U(r,\alpha) = -V_v\left[1+\exp\left(\frac{r-R_v}{a_v}\right)\right] - iW_v\left[1+\exp\left(\frac{r-R_v}{a_v}\right)\right] - i4a_dW_d\frac{d}{dr}\left[1+\exp\left(\frac{r-R_d}{a_d}\right)\right]$$

```
Vv0 = 46.72
Wv0 = 1.723
Rv0 = 4.053
av0 = 0.671
Wd0 = -7.23
Rd0 = 4.405
ad0 = 0.537
```

Going Bayesian with Surmise



Total samples
7,200,000

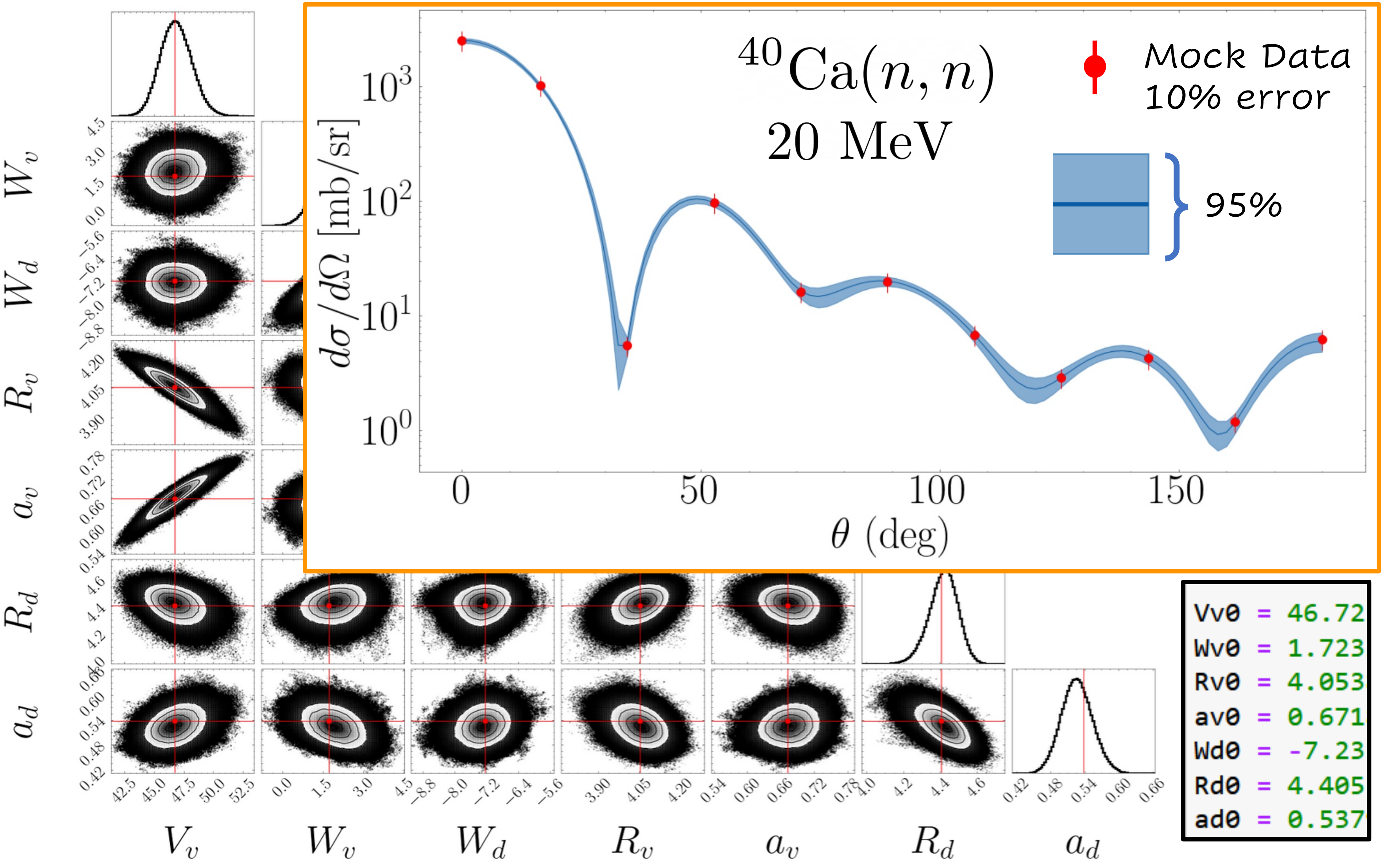
High Fidelity:
2 months

ROSE:
2 hours

$$U(r,\alpha) = -V_v\left[1+\exp\left(\frac{r-R_v}{a_v}\right)\right] - iW_v\left[1+\exp\left(\frac{r-R_v}{a_v}\right)\right] - i4a_d W_d \frac{d}{dr}\left[1+\exp\left(\frac{r-R_d}{a_d}\right)\right]$$

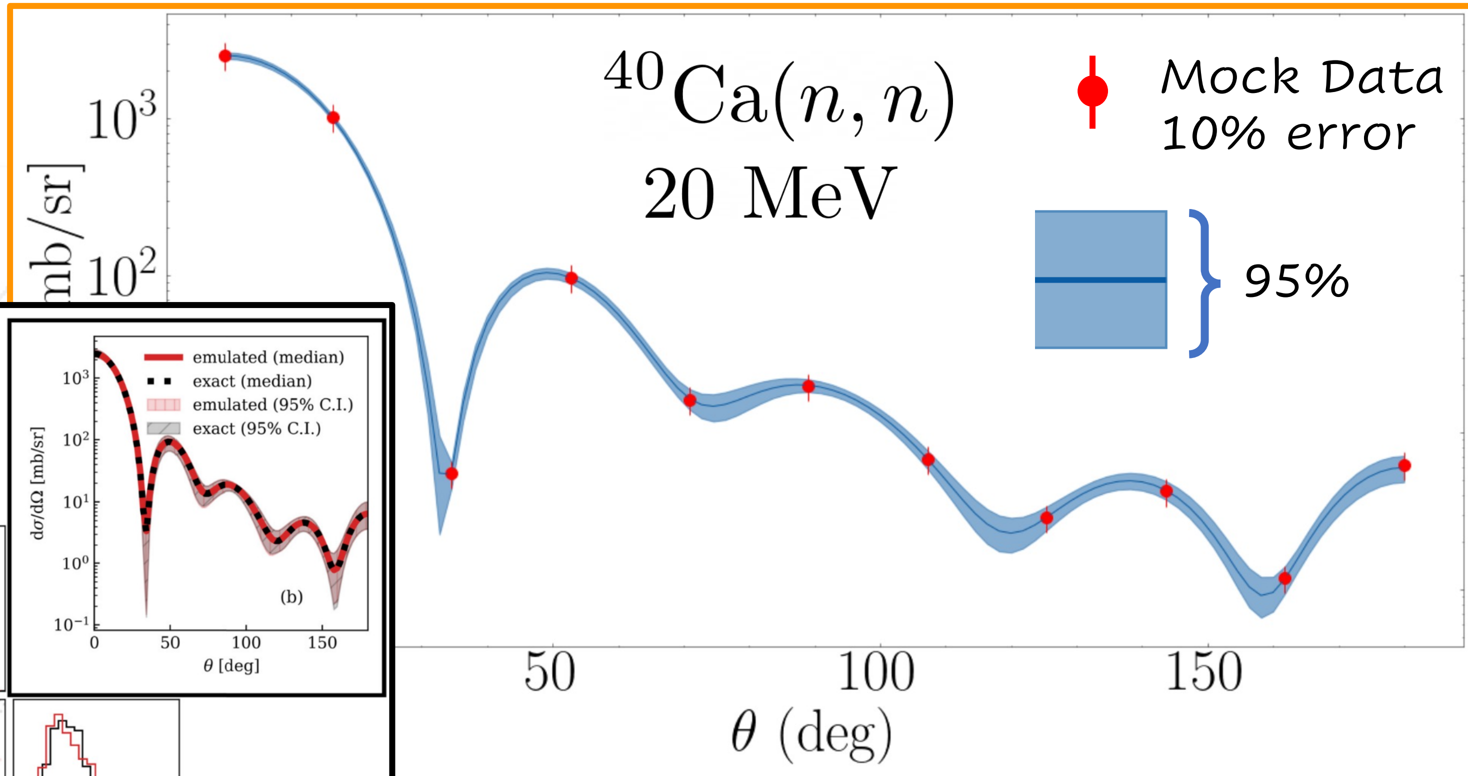
```
Vv0 = 46.72
Wv0 = 1.723
Rv0 = 4.053
av0 = 0.671
Wd0 = -7.23
Rd0 = 4.405
ad0 = 0.537
```

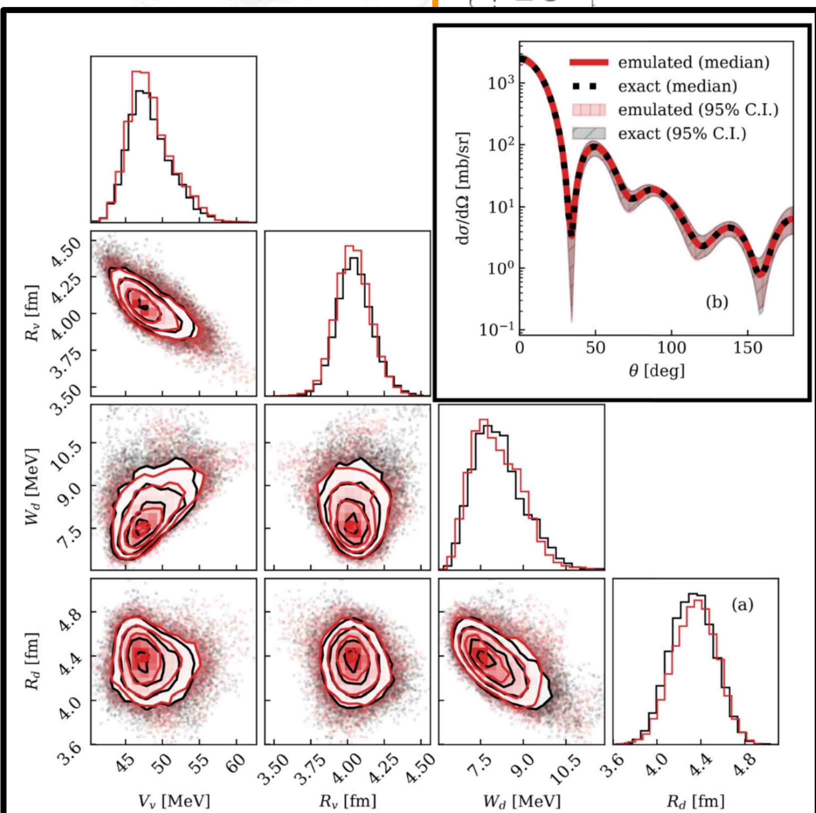
Going Bayesian with Surmise




```
Vv0 = 46.72
Wv0 = 1.723
Rv0 = 4.053
av0 = 0.671
Wd0 = -7.23
Rd0 = 4.405
ad0 = 0.537
```

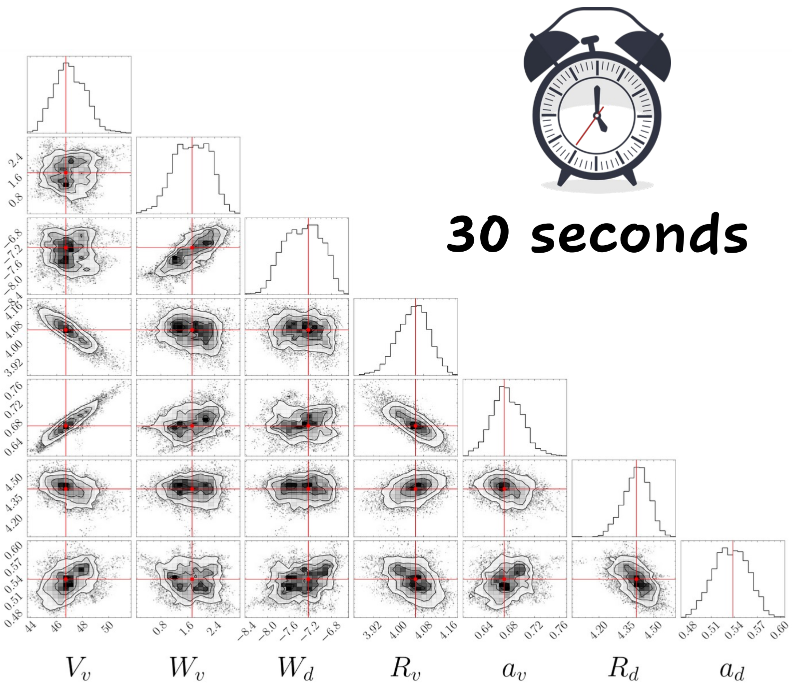
Going Bayesian with Surmise

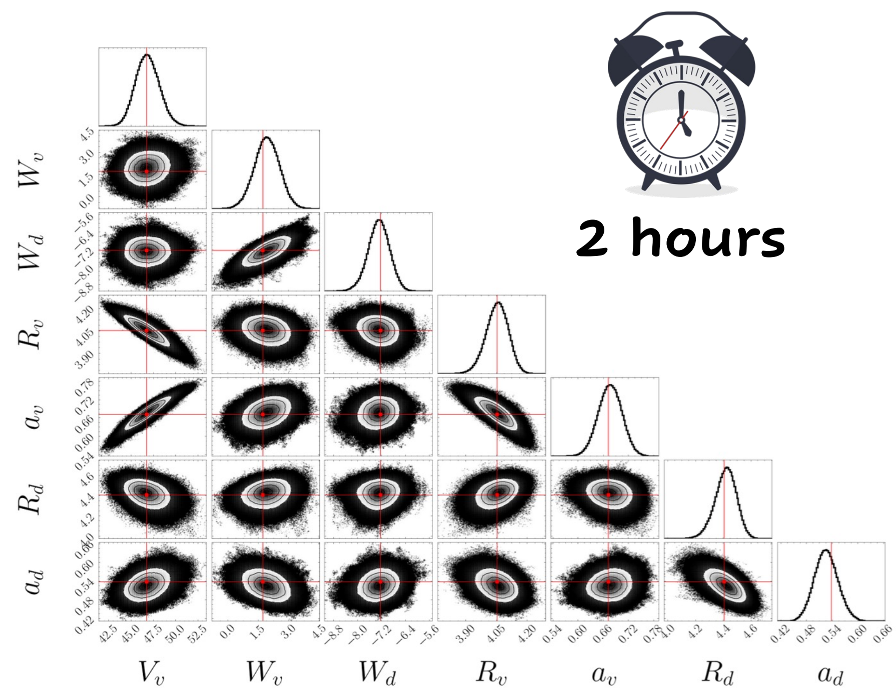




EC paper, Christian and friends (2021)

Going Bayesian with Surmise

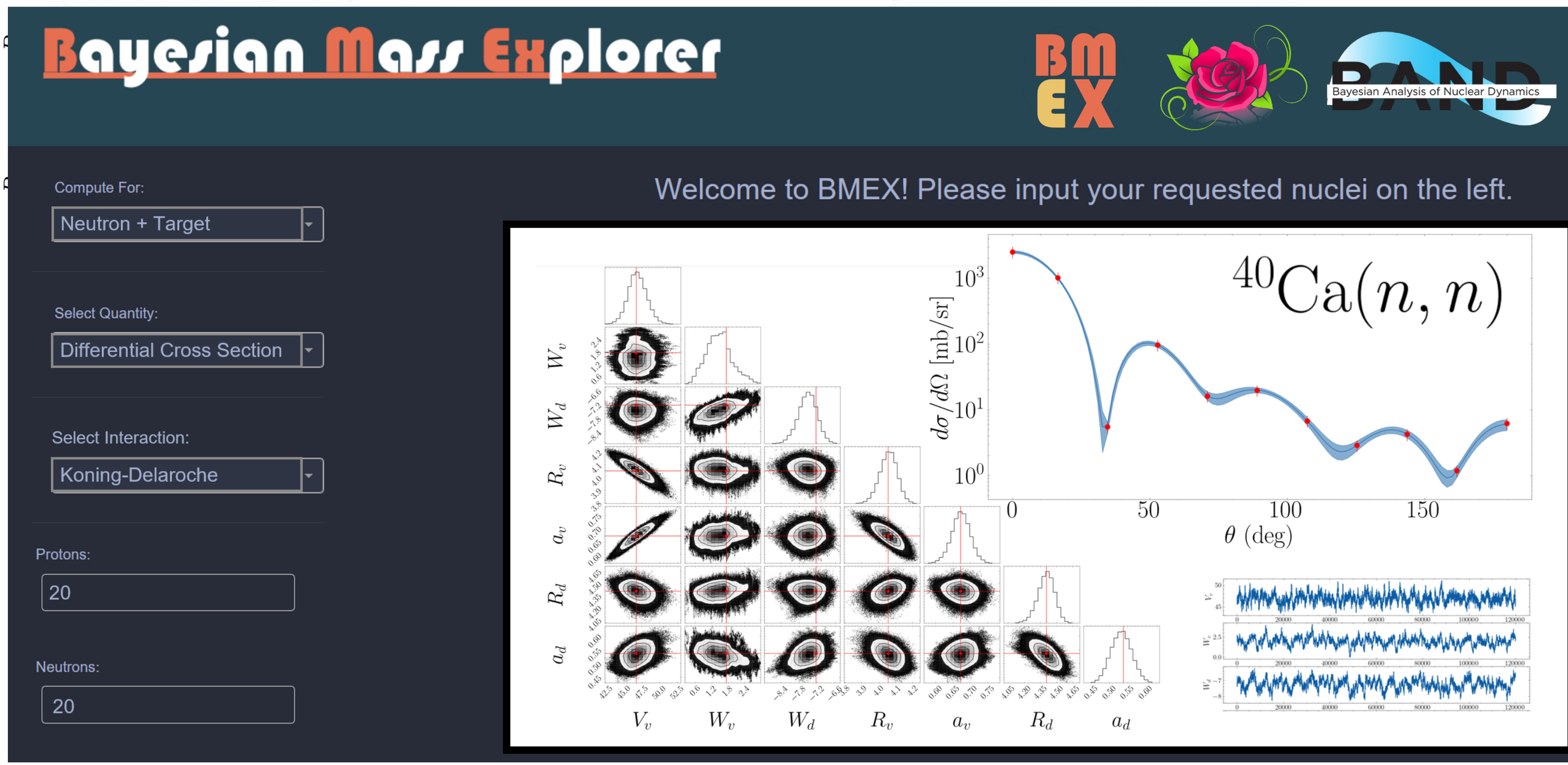




Going Bayesian with Surmise

30 seconds

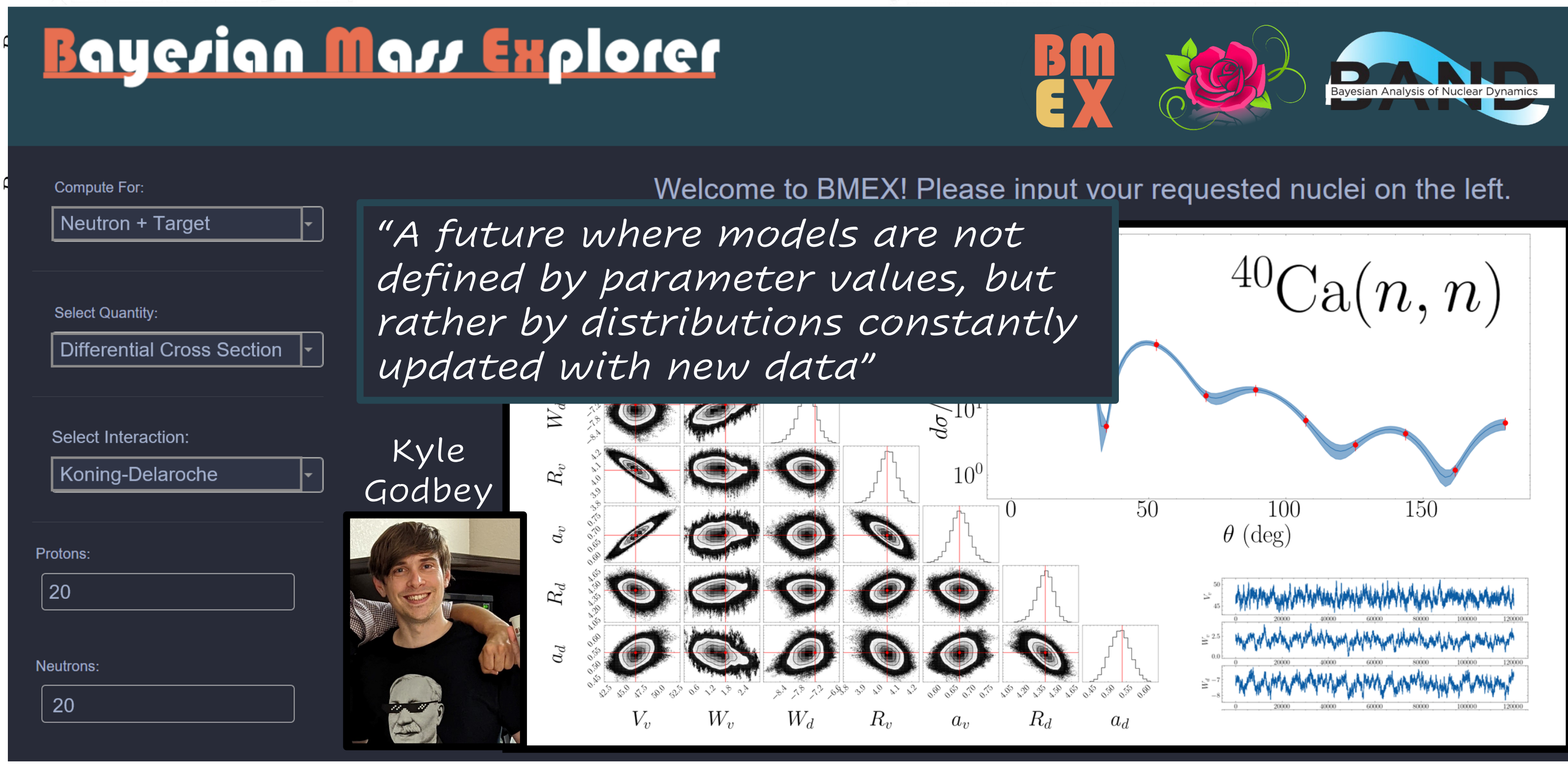
2 hours



Going Bayesian with Surmise

30 seconds

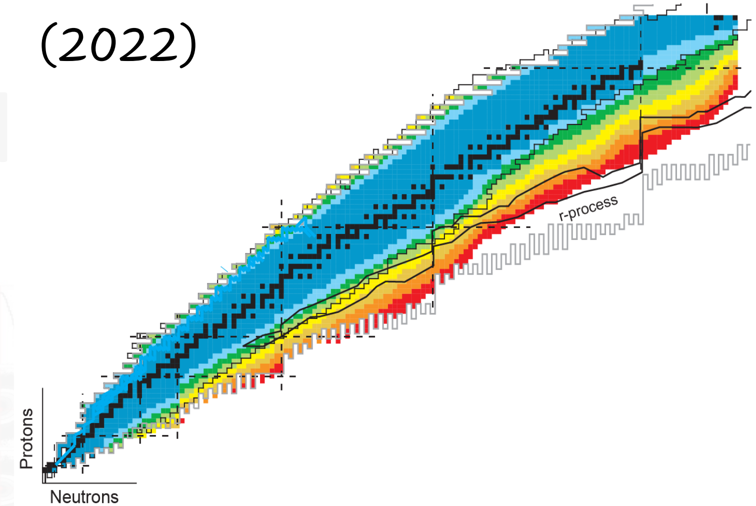
2 hours

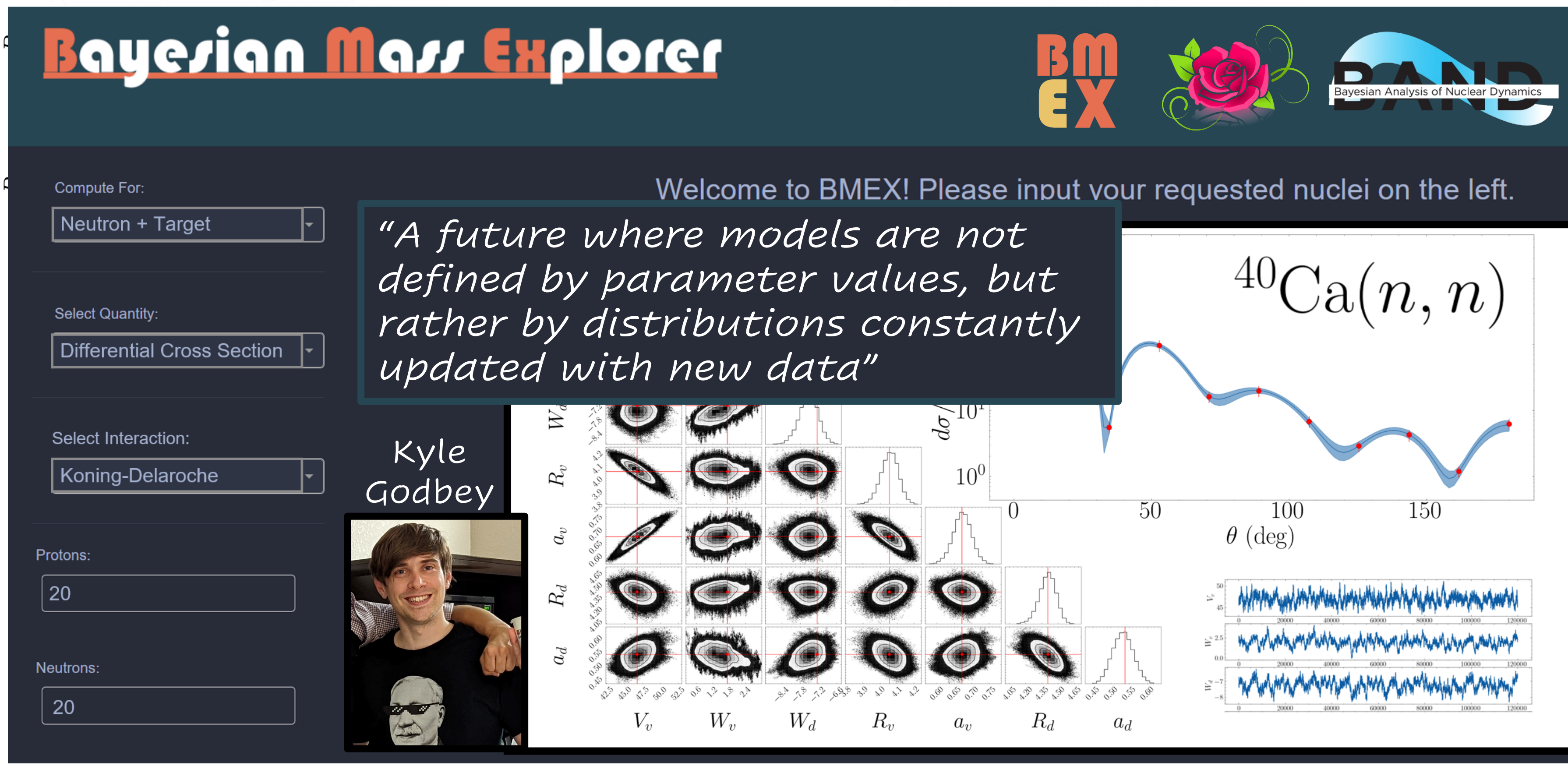


Optical potentials for the rare-isotope beam era (2022)

In regions of the nuclear chart away from stability, which represent a frontier in nuclear science over the coming decade and which will be probed at new rare-isotope beam facilities worldwide, there is a targeted ***need to quantify and reduce theoretical reaction model uncertainties****, especially with respect to nuclear optical potentials.*

30 seconds





"A future where models are not defined by parameter values, but rather by distributions constantly updated with new data"

Kyle Godbey

Coming soon

to an arxiv near you

Title under revision…

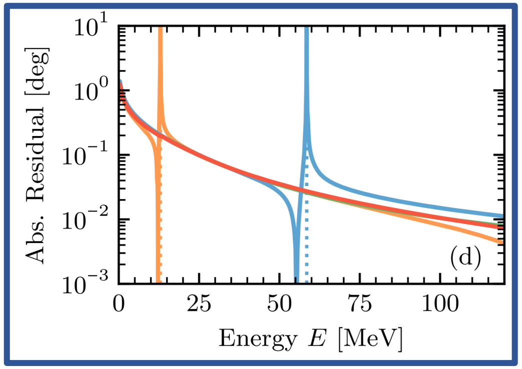
Presenting ROSE, a Reduced Order Scattering Emulator

D. Odell,[1,*] P. Giuliani,[2,3] M. Catacora-Rios,[2,4] M. Chan,[5] E. Bonilla,[6] K. Godbey,[2] R. J. Furnstahl,[7] and F. M. Nunes[2,4,†]

[1]*Department of Physics and Astronomy, Institute of Nuclear and Particle Physics, Ohio University, Athens, OH, United States*
[2]*Facility for Rare Isotope Beams, Michigan State University, East Lansing, Michigan 48824, USA*
[3]*Department of Statistics and Probability, Michigan State University, East Lansing, Michigan 48824, USA*
[4]*Department of Physics and Astronomy, Michigan State University, East Lansing, MI 48824-1321, USA*
[5]*Department of Industrial Engineering and Management Sciences, Northwestern University, Evanston, IL 60208, United States of America*
[6]*Department of Physics, Stanford University, Stanford, California 94305, USA*
[7]*Department of Physics, The Ohio State University, Columbus, OH 43210, USA*

(Dated: May 22, 2023)

- Neutrons and protons
- Emulating across energies
- ``Anomalies" seem gone



Reduced Basis Methods for emulation and the ROSE software package

Daniel Odell (Ohio), Pablo Giuliani (MSU/FRIB)

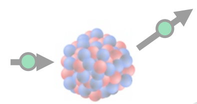
22 May 2023 12:15

Whitaker 218 (Washington University in St. Louis)

Information and Statistics for Nuclear Experiment and Theory workshop (ISNET-9)

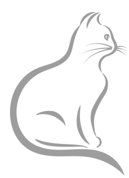
Outline

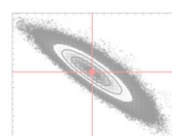
Main ideas

Nuclear Scattering 

Reduced Basis Method 

Main results

Accuracy and time 

Going Bayesian with Surmise 

Tutorials

1) Emulating without ROSE

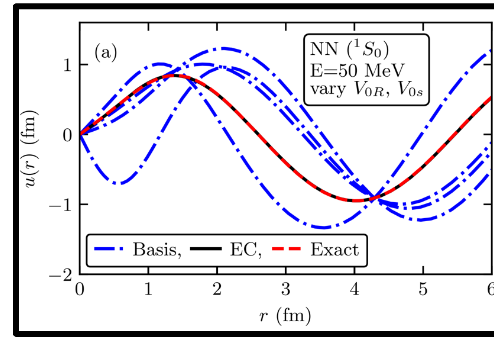
2) Emulating with ROSE

https://doi.org/10.1016/j.physletb.2020.135719

Tutorial 1

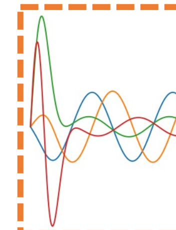
Minnesota Potential

$$V(r) = V_{0R}\, e^{-\kappa_R r^2} + V_{0s}\, e^{-\kappa_s r^2}$$



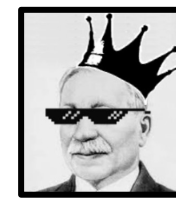
1) Build a basis from principal component analysis and the free solution

$$\hat{\phi}(x) = \phi_0 + \sum_{k}^{n} a_k \phi_k(x)$$

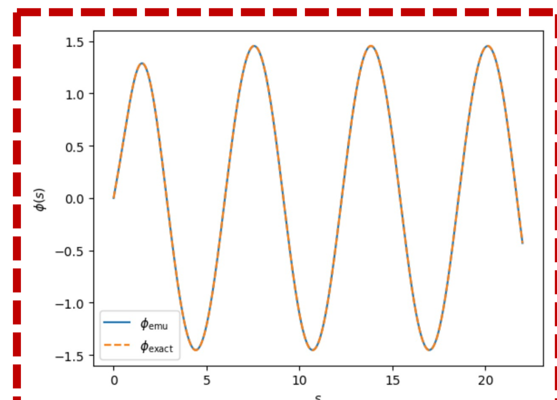


2) Build the Galerking projection equations

$$\boldsymbol{A}\boldsymbol{a} = \boldsymbol{c} \begin{cases} c_j = -\langle \psi_j | F_\alpha | \phi_0 \rangle \\ A_{j,k} = \langle \psi_j | F_\alpha | \phi_k \rangle \end{cases}$$

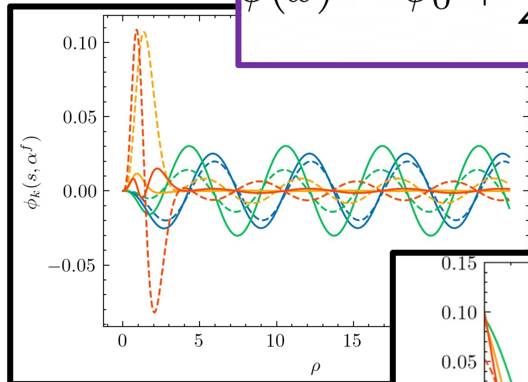


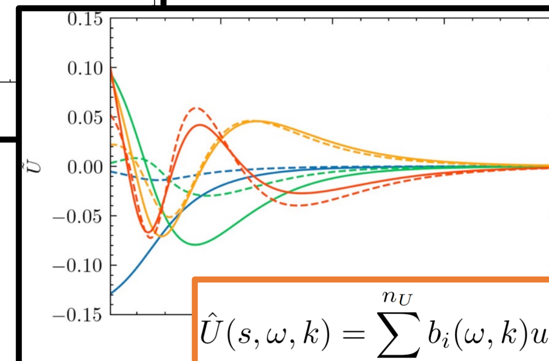
3) Use emulator for a new parameter value and celebrate!



Tutorial 2

$$\hat{\phi}(x) = \phi_0 + \sum_{k}^{n} a_k \phi_k(x)$$

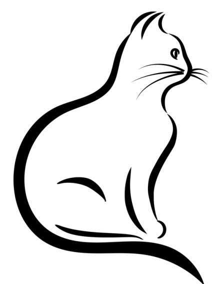




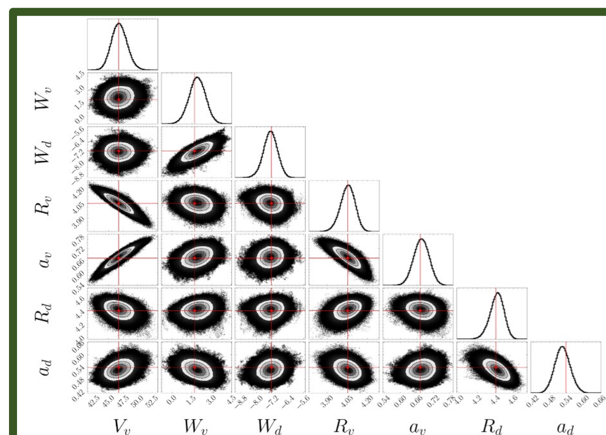
$$\hat{U}(s, \omega, k) = \sum_{i=1}^{n_U} b_i(\omega, k) u_i(s)$$

1) Build basis for the wave functions and the potential

2) Build many emulators and let them fight in a CAT plot



3) Select the fittest and go Bayesian with surmise



From zero to Bayes in minutes