Readme for Code used in

*Predicting the unobserved: a statistical mechanics framework for*

*non-equilibrium material response with quantified uncertainty*

Shenglin Huang, Ian R. Graham, Robert A. Riggleman, Paulo Arratia,

Steve Fitzgerald, and Celia Reina

1. **Overview**

This file explains all codes used in

*Predicting the unobserved: a statistical mechanics framework for non-equilibrium material response with quantified uncertainty* (JMPS 2022) by Shenglin Huang, Ian R. Graham, Robert A. Riggleman, Paulo Arratia, Steve Fitzgerald, and Celia Reina

to generate Figs. 1-7 and Fig. 13 in the paper. Specifically, Sections 2-4 introduce the steps for generating the figures: material extrapolation and uncertainty quantification for pulling experiment of 1D mass-spring chain (Section 2), uncertainty quantification of diffusion in periodic box (Section 3) and plotting figures from the generated data (Section 4). Finally, Section 5 describes the details for all the folders and files for the code.

1. **Material Extrapolation for 1D Mass-Spring Chain Governed by Over-damped Langevin Dynamics**

(in folder **“MassSpringChain1D”**)

* 1. **Material Extrapolation**

Run the MATLAB file “MassSpringChain1D\_Simulation.m” with proper input. It predicts the non-equilibrium behavior of the pulling experiment of 1D mass-spring chain governed by overdamped Langevin dynamics for target system/process 2 from simulated system/process 1. The difference between the simulated system 1 and target system are described by their interatomic potentials and pulling velocity. The file will output a MATLAB data file “DataLang\_XXXX.mat” in folder “Result”, a MATLAB figure file “FigLang\_XXXX.fig” in folder “Result/figures” and two movies “hist\_Ib\_XXXX.mp4” and “hist\_Pb\_XXXX.mp4” in folder “Result/movies”, where XXXX contains the information of the simulation. When the parameter “PfactorOutput” is set as 1, the file may also output a MATLAB data file “DataPb\_XXXX.mat” in folder “Result” that contains some trajectory information.

* 1. **Uncertainty Quantification**

Run the MATLAB file “MassSpringChain1D\_UQ.m” with proper input. It provides the uncertainty quantification (UQ) for the simulation described in Sec. 1.1. It requires the MATLAB data file “DataLang\_XXXX.mat” in folder “Result” as the input data file for validation purpose. Notice that this input data is not necessary if one only requires the UQ and removes the validation portion. This file will output a MATLAB data file “DataEst\_XXXX.mat” in folder “Result” and a MATLAB figure file “FigEst\_XXXX.mat” in folder “Result/figures”.

1. **Material Extrapolation for Diffusion in Periodic Box Governed by Over-damped Langevin Dynamics**
   1. **Material Extrapolation**

All simulation steps are run using the “md-lv” binary. Build instructions can be found within the “md-lv” folder. Scripts for automating the production steps of these simulations can be found in the “scripts” subfolder, and Jupyter notebooks used for post-processing analysis are found in “notebooks”. The basic command-line steps were employed to generate the datasets.

* Within “scripts”, run “python gen\_equilibrated\_states.py”. This small python script generates force-quenched states for the Hertzian and Lennard-Jones potentials. These are saved in a human-readable JSON format for simple inspection and modification.
* Again within “scripts”, assuming one is working on a system with a SLURM workload manager, run “sbatch gen\_val\_data.sh” and “sbatch gen\_bias\_pred\_data.sh” to generate the the validation and bias-prediction datasets respectively. Within these scripts the logic is fairly straightforward, where the SLURM manager launches a couple hundred python scripts that further handles the logic of calling the “md-lv” binary to generate the simulation outputs. These files are stored in HDF5 format.
* The ‘md-lv’ is also able to produce “XYZ” files readible by programs like Ovito. An example of this usage can be found in “scripts/run\_xyz.py”.
  1. **Uncertainty Quantification** (in folder **“PeriodicBoxDiffusion”**)

This section provides the uncertainty quantification for material extrapolation for the diffusion in periodic box governed by over-damped Langevin dynamics. Notice that this UQ provides the results for multiple target systems from one simulated system. It requires the MATLAB data file from the material extrapolation simulation as input for validation.

1. Copy the MATLAB data file “XXXX.mat” generated in Section 2.1 in folder “Result/YYYY”, where YYYY describes the interested system such as “2D\_Hertzian” and “2D\_LJ”.
2. Run the MATLAB file “PeriodicBoxDiffusion\_UQ.m” with proper input. It provides the uncertainty quantification for the simulation described in Sec. 2.1. It requires the MATLAB data file “XXXX.mat” in folder “Result/YYYY” as the input data file for detail description of system and validation. This file will output *NV* MATLAB data files “DataEst\_ZZZZ.mat” and *NV* MATLAB figure files “FigEst\_ZZZZ.mat” in folder “Result/YYYY”, where *NV* is the number of target systems (given one simulated system) for UQ and ZZZZ contains the information of the simulation and chosen target system.
3. **Plot Figures in the Paper**
   1. **Plot Figures except the Predictions for 2D Glassy System** (in folder **“PlotFigures”**)

This section provides the Python code “PlotFigures\_PredictingUnobserved\_1.ipynb” for plotting all the figures in the paper except the predictions for 2D glassy system. Notice that the code was designed for working on Google Colab.

1. Copy the code “PlotFigures\_PredictingUnobserved\_1.ipynb” into the root folder, named as “RRRR” here, on Google Colab.
2. Copy following data files into folder “RRRR/DataFiles”:
   * + Data files for plotting the histograms of
       - DataPb\_N1\_R100000\_k0\_1\_4k0\_0\_eta5\_kT0.0001\_v0\_0.01\_dt0.001.mat
       - DataLang\_N1\_R100000\_k0\_1\_4k0\_0\_eta5\_kT0.0001\_v0\_0.01\_dt0.001.mat
       - DataPb\_N1\_R100000\_k0.5\_1\_4k0\_0\_eta5\_kT0.0001\_v0.01\_0.01\_dt0.001.ma
       - DataLang\_N1\_R100000\_k0.5\_1\_4k0\_0\_eta5\_kT0.0001\_v0.01\_0.01\_dt0.001.mat
       - DataPb\_N2\_R100000\_k0\_1\_4k0\_0\_eta5\_kT0.0001\_v0\_0.015\_dt0.001.mat
       - DataLang\_N2\_R100000\_k0\_1\_4k0\_0\_eta5\_kT0.0001\_v0\_0.015\_dt0.001.mat
       - DataPb\_N3\_R100000\_k0\_1\_4k0\_0\_eta5\_kT0.0001\_v0\_0.02\_dt0.001.mat
       - DataLang\_N3\_R100000\_k0\_1\_4k0\_0\_eta5\_kT0.0001\_v0\_0.02\_dt0.001.mat'
     + Data files for plotting the results for linear and nonlinear UQ
       - DataEst\_N1\_R100000\_k0\_1\_4k0\_0\_eta5\_kT0.0001\_v0\_0.01\_dt0.001\_NG100\_xr0.mat
       - DataLang\_N1\_R100000\_k0\_1\_4k0\_0\_eta5\_kT0.0001\_v0\_0.01\_dt0.001.mat (appeared before)
     + Data files for plotting the results 1D mass-spring chain
       - DataLang\_N10\_R100000\_k0.5\_1\_4k0\_100\_eta5\_kT0.0001\_v0.01\_0.01\_dt0.001.mat
       - DataEst\_N10\_R100000\_k0.5\_1\_4k0\_100\_eta5\_kT0.0001\_v0.01\_0.01\_dt0.001\_NG100\_xr0.mat
       - DataLang\_N10\_R100000\_k1\_1\_4k100\_100\_eta5\_kT0.0001\_v0\_0.01\_dt0.001.mat
       - DataEst\_N10\_R100000\_k1\_1\_4k100\_100\_eta5\_kT0.0001\_v0\_0.01\_dt0.001\_NG100\_xr0.mat
       - DataLang\_N10\_R100000\_k0\_1\_4k0\_100\_eta5\_kT0.0001\_v0\_0.01\_dt0.001.mat
       - DataEst\_N10\_R100000\_k0\_1\_4k0\_100\_eta5\_kT0.0001\_v0\_0.01\_dt0.001\_NG100\_xr0.mat
     + Data files for plotting the UQ of 2D glassy system
       - hertz.mat
       - DataEst\_N10\_2DH\_R1e+07\_Vm10\_12.5893\_eta5\_kT0.1\_T100\_NG100\_xr0.mat
       - DataEst\_N10\_2DH\_R1e+07\_Vm10\_100\_eta5\_kT0.1\_T100\_NG100\_xr0.mat
       - hertz\_lowt.mat
       - DataEst\_N10\_2DH\_R1e+07\_Vm1\_1.25893\_eta5\_kT0.01\_T10\_NG100\_xr0.mat
       - DataEst\_N10\_2DH\_R1e+07\_Vm1\_10\_eta5\_kT0.01\_T10\_NG100\_xr0.mat
       - lj.mat
       - DataEst\_N10\_2DLJ\_R1e+07\_Vm0.1\_0.125893\_eta5\_kT0.1\_T10\_NG100\_xr0.mat
       - DataEst\_N10\_2DLJ\_R1e+07\_Vm0.1\_1\_eta5\_kT0.1\_T10\_NG100\_xr0.mat
       - lj\_lowt.mat
       - DataEst\_N10\_2DLJ\_R1e+07\_Vm0.01\_0.0125893\_eta5\_kT0.01\_T10\_NG100\_xr0.mat
       - DataEst\_N10\_2DLJ\_R1e+07\_Vm0.01\_0.1\_eta5\_kT0.01\_T10\_NG100\_xr0.mat
3. Create folder “RRRR/Figures” for saving figures.
4. Run the code “PlotFigures\_PredictingUnobserved\_1.ipynb”. It will generate all the figures in the paper except the predictions for 2D glassy system.
   1. **Plot Figures for the Prediction of 2D Glassy System**

From code ran during material exploration of the 2D glassy system, all figures can be generated by running the jupyter notebook “notebooks/make\_paper\_figures.ipynb”. Without modification to the files here, the data produced above should be properly fetched and processed without intervention.

1. **File Description**

* **MassSpringChain1D** (folder)

The folder for doing material extrapolation and corresponding uncertainty quantification for pulling experiment of 1D mass-spring chain governed by over-damped Langevin dynamic. The material extrapolation refers to predicting the ensemble averages of macroscopic observables of a non-equilibrium process of target system with interatomic potential (V2) and constant pulling velocity (vp\_2) from the simulation of a different non-equilibrium process of simulated system with interatomic potential (V1) and constant pulling velocity (vp\_1).

* + **MassSpringChain1D\_Simulation.m**

The MATLAB file for material extrapolation for pulling experiment of 1D mass-spring chain.

* + - **Input:** Parameters of the simulation, including:
      * N % Number of particles
      * eta % Viscosity
      * kBT % Temperature (in energy scale)
      * vp\_1 % Pulling velocity for simulated system
      * vp\_2 % Pulling velocity for target system
      * NR % Number of realizations
      * T % Total time for process
      * dt % Time step for Langevin dynamics
      * type % Definition of the interatomic potential type. Each type of

requires corresponding .m files for calculation. Only **'quartic'** (quartic potential) is available in the current code.

* + - * PfactorOutput % Whether to save or not to save (‘Pfactor’) for all

realization in output file. Value 0 for not saving ‘Pfactor’ and 1 for saving it.

* + - * k2\_1, k4\_1 % Parameters for quartic potential of the simulated system when

setting **type = ‘quartic’**.

* + - * k2\_2, k4\_2 % Parameters for quartic potential of the target system when

setting **type = ‘quartic’**.

* + - **Output:**
      * A MATLAB data file “DataLang\_XXXX.mat” in folder “Result”. It contains systems parameters and some statistical data (such as ensemble averages or variance) of the non-equilibrium process, where XXXX contains the information of the simulation. The included variables are listed as:
      * N, eta, kBT, vp\_1, vp\_2, NR, T, dt, type

% Parameters defined as input.

* + - * k2\_1, k4\_1, k2\_2, k4\_2

% Parameters defined as input, available when

**type = ‘quartic’**.

* + - * Nt % Number of time steps.
      * time % Time grid for Langevin dynamics.
      * r\_1, Wave\_1, Force\_1

% Ensemble averages of displacement, work and external

force for the simulated system 1.

* + - * r\_2, Wave\_2, Force\_2

% Ensemble averages of displacement, work and external

force for the target system 2.

* + - * r\_bias, Wave\_bias, Force\_bias

% Predicted ensemble averages of displacement, work and

external force for the target system form the simulated system.

* + - * normalization % The summation of (Pfactor) over all realizations.
      * mean\_factor, var\_factor

% Observed mean and variance of (factor).

* + - * mean\_Pfactor, var\_Pfactor

% Observed mean and variance of (Pfactor).

* + - * Pfactor\_out % The output (Pfactor) with coarser time grid,

available when **PfactorOutput = 1**.

* + - * time\_Pb, index\_Pb

% The coarser time grid and corresponding index for

Pfactor\_out, available when **PfactorOutput = 1**.

* + - * calTime\_Lang % Computational time cost for the Langevin simulation.
      * A MATLAB figure file “FigLang\_XXXX.fig” in folder “Result/figures”. It contains the figures of the prediction for macroscopic observables and statistics of the process.
      * Two movies “hist\_Ib\_XXXX.mp4” and “hist\_Pb\_XXXX.mp4” in folder “Result/movies”. They show the time evolution of the histogram for (factor) and (Pfactor).
      * An *optional* MATLAB data file “DataPb\_XXXX.mat” in folder “Result” when setting the parameter **PfactorOutput = 1**. It contains the time evolution of (Pfactor) for each realization with a courser time step, which can be used for postprocessing of the statistics of . The included variables are listed as:
      * PlotProp % Whether to show or not to show the linear UQ (linear

variance propagation) result. Value 1 for applying linear UQ and value 0 for not to do so. Notice that the linear UQ should be applied only when the simulated system has no interaction (**V1 = 0**, Brownian motion)

* + - * N % Number of particles
  + **MassSpringChain1D\_UQ.m**

The MATLAB file for the uncertainty quantification of the material extrapolation for pulling experiment of 1D mass-spring chain.

* + - **Input:** Parameters of the simulation, including:
      * PlotProp % Whether to show or not to show the linear UQ (linear variance

propagation) result. Value 1 for applying linear UQ and value 0 for not to do so. Notice that the linear UQ should be applied only when the simulated system has no interaction (V1 = 0, Brownian motion)

* + - * N % Number of particles
      * eta % Viscosity
      * kBT % Temperature (in energy scale)
      * vp\_1 % Pulling velocity for simulated system
      * vp\_2 % Pulling velocity for target system
      * NR % Number of realizations
      * T % Total time for process
      * dt % Time step for Langevin dynamics
      * type % Definition of the interatomic potential type. Each type of

requires corresponding .m files for calculation. Only 'quartic' (quartic potential) is available in the current code.

* + - * k2\_1, k4\_1 % Parameters for quartic potential of the simulated system when

setting **type = ‘quartic’**.

* + - * k2\_2, k4\_2 % Parameters for quartic potential of the target system when

setting **type = ‘quartic’**.

* + - * type\_xr % Type of reference trajectory for quadratic potential expansion.

Two choices are available in current code: **'0'** for expanding at zero displacement and **'linear'** for expanding at displacement field that is linear in space under given pulling velocity.

* + - * Nt\_est % Number of estimation time points for UQ
      * Nt\_grid\_est % Number of time discretization for UQ at given estimation time

point , which uniformly discretize the time interval .

* dt\_pre % Timestep for linear UQ when **PlotProp = 1** (optional).
  + - **Output:**
      * A MATLAB data file “DataEst\_XXXX.mat” in folder “Result”. It contains the data for UQ:
      * Nt\_est, Nt\_grid\_est, type\_xr

% Parameters defined as input.

* + - * dt\_pre % Parameters defined as input, available when **PlotProp = 1**

(optional).

* + - * Nt\_pre % Number of time steps (estimation points) for linear UQ,

available when **PlotProp = 1** (optional).

* + - * time\_pre % Estimation time points for linear UQ, available when

**PlotProp = 1** (optional).

* + - * time\_est % Time mesh for estimation time points in UQ
      * dt\_est % Timestep for UQ at different estimation time points
      * time\_grid\_est % Time grid that discretizes the interval from 0 to the

estimation time points.

* + - * xrt % The reference trajectory for quadratic potential expansion.
      * cy\_t, cvy\_t, detAvy\_t, detAy\_t, b\_A\_inv\_b\_y\_t, bv\_Av\_inv\_bv\_y\_t

% Some parameters during the UQ calculation.

* + - * mean\_Pfactor\_2nd, var\_Pfactor\_2nd

% Estimated mean and variance of (Pfactor) based on

nonlinear UQ.

* + - * Nfactor % The observed finite average of (Pfactor).
      * err\_Nfactor % The observed deviation of Nfactor from 1.
      * err\_Nfactor\_2nd % The estimated deviation of Nfactor based on UQ.
      * var\_Pfactor\_prop % Estimated variance of P\_bias (Pfactor) based on linear

UQ, available when **PlotProp = 1** (optional).

* + - * err\_Nfactor\_prop % Estimated deviation of Nfactor based on linear UQ,

available when **PlotProp = 1** (optional).

* + - * calTime\_Est % Computational time cost for UQ estimation.
      * A MATLAB figure file “FigEst\_XXXX.mat” in folder “Result/figures”. It contains the figures for UQ.
  + **DQuarticPotential.m**

The MATLAB code file for calculating the gradient of system’s potential with respect to the displacements for all particles *together with the protocol (lambda)* when the interatomic potential is quartic. It is used in “MassSpringChain1D\_Simulation.m” when choosing **type = ‘quartic’**. It takes

* + **DV\_x\_Quartic.m**

The MATLAB code file for calculating the gradient of system’s potential with respect to the displacements for all particles *(without the protocol lambda) at multiple times* when the interatomic potential is quartic. It is used in “MassSpringChain1D\_UQ.m” when choosing **type = ‘quartic’**.

* + **D2V\_x\_Quartic.m**

The MATLAB code file for calculating the double gradient of system’s potential with respect to the displacements for all particles *(without the protocol lambda) at multiple times* when the interatomic potential is quartic. It is used in “MassSpringChain1D\_UQ.m” when choosing **type = ‘quartic’**.

* + **xr\_t\_linear.m**

The MATLAB code file for calculating the reference trajectory that is linear in space with given protocol (lambda) at different times for quadratic potential expansion. It is used in “MassSpringChain1D\_ UQ.m” when choosing **type\_xr = 'linear'**.

* **PeriodicBoxDiffusion** (folder)

The folder for doing uncertainty quantification of the material extrapolation for diffusive particles in a periodic box governed by over-damped Langevin dynamic. The material extrapolation refers to predicting the ensemble averages of macroscopic observables of a non-equilibrium process of target system with interatomic potential (V2) from the simulation of a different non-equilibrium process of simulated system with interatomic potential (V1). Particularly, there is only one simulated system but can exist multiple target systems.

* + **PeriodicBoxDiffusion\_UQ.m**

The MATLAB file for the uncertainty quantification of the material extrapolation for diffusive particles in a periodic box.

* + - **Input:** Parameters of the simulation, including:
      * type % Definition of the interatomic potential type. Each type of

requires corresponding .m files for calculation. ‘Hertzian' (Hertzian potential) and ‘LJ’ (Lenard-Jones potential) are available in the current code and both options allow two species particles A & B.

* + - * FolderName & InputFileName

% Name of the folder for the data file generated from previous

simulation of material extrapolation. Notice that this data file should include system parameters such as (num\_parts) particle number, (visc) viscosity, (beta) inverse temperature, (L) box size, (dim) dimension of the system, (dVs) ratios of the potential amplitude differences to potential amplitude of system, process related variables such as (time) simulation time, (init\_pos) initial position for all particles, (Ib\_mean & Ib\_var) mean and variance of . Notice that the parameter names are different with the conventions in the MATLAB code and the paper.

* + - * Vm\_1 % Potential amplitude for V1 when choosing **type = ‘Hertzian’**.
      * rad\_A, rad\_B % Radius of particle A and B, respectively, when choosing

**type = ‘Hertzian’**.

* + - * Vm\_AA\_1, Vm\_AB\_1, Vm\_BB\_1

% Potential amplitude between A-A, A-B, B-B, respectively, for

V1 when choosing **type = ‘LJ’**.

* + - * sigma\_AA, sigma \_AB, sigma \_BB

% Interatomic length between A-A, A-B, B-B, respectively, for

V1 when choosing **type = ‘LJ’**.

* + - * index\_A % Indices for particle A.
      * iV\_index % The index/indices of target systems/potentials for applying UQ.
      * type\_xr % Type of reference trajectory for quadratic potential expansion.

The only choice available in current code is **'0'** for expanding at zero displacement (expanding at initial positions).

* + - * Nt\_est % Number of estimation time points for UQ
      * Nt\_grid\_est % Number of time discretization for UQ at given estimation time

point , which uniformly discretize the time interval .

* + - **Output:**
      * *NV* MATLAB data files “DataEst\_XXXX.mat” in folder “Result”, where *NV* is the number of target systems for applying UQ. It contains the data for UQ:
      * Nt\_est, Nt\_grid\_est, type\_xr

% Parameters defined as input.

* + - * time\_est % Time mesh for estimation time points in UQ
      * dt\_est % Timestep for UQ at different estimation time points
      * time\_grid\_est % Time grid that discretizes the interval from 0 to the

estimation time points.

* + - * xrt % The reference trajectory for quadratic potential expansion.
      * cy\_t, cvy\_t, detAvy\_t, detAy\_t, b\_A\_inv\_b\_y\_t, bv\_Av\_inv\_bv\_y\_t

% Some parameters during the UQ calculation.

* + - * mean\_Pfactor\_2nd, var\_Pfactor\_2nd

% Estimated mean and variance of (Pfactor) based on

nonlinear UQ.

* + - * Nfactor % The observed finite average of (Pfactor).
      * err\_Nfactor % The observed deviation of Nfactor from 1.
      * err\_Nfactor\_2nd % The estimated deviation of Nfactor based on UQ.
      * calTime\_Est % Computational time cost for UQ estimation.
      * *NV* MATLAB figure files “FigEst\_XXXX.mat” in folder “Result/figures”. It contains the figures for UQ.
  + **DV\_x\_Hertzian.m**

The MATLAB code file for calculating the gradient of system’s potential with respect to the displacements for all particles *at multiple times* when the interatomic potential is Hertzian. It is used in “PeriodicBoxDiffusion\_UQ.m” when choosing **type = ‘Hertzian’**.

* + **D2V\_x\_Hertzian.m**

The MATLAB code file for calculating the double gradient of system’s potential with respect to the displacements for all particles *at multiple times* when the interatomic potential is Hertzian. It is used in “PeriodicBoxDiffusion\_UQ.m” when choosing **type = ‘Hertzian’**.

* + **DV\_x\_LJ.m**

The MATLAB code file for calculating the gradient of system’s potential with respect to the displacements for all particles *at multiple times* when the interatomic potential is Lenard-Jones. It is used in “PeriodicBoxDiffusion\_UQ.m” when choosing **type = ‘LJ’**.

* + **D2V\_x\_LJ.m**

The MATLAB code file for calculating the double gradient of system’s potential with respect to the displacements for all particles *at multiple times* when the interatomic potential is Lenard-Jones. It is used in “PeriodicBoxDiffusion\_UQ.m” when choosing **type = ‘LJ’**.

* **PlotFigures** (folder)

This section provides the Python code “PlotFigures\_PredictingUnobserved\_1” for plotting Figs. 1-7 and Fig. 13 in the paper. Notice that the code was designed for working on Google Colab.

The folder for doing uncertainty quantification of the material extrapolation for diffusive particles in a periodic box governed by over-damped Langevin dynamic. The material extrapolation refers to predicting the ensemble averages of macroscopic observables of a non-equilibrium process of target system with interatomic potential (V2) from the simulation of a different non-equilibrium process of simulated system with interatomic potential (V1). Particularly, there is only one simulated system but can exist multiple target systems.

* + **PlotFigures\_PredictingUnobserved\_1.****ipynb**

The Python code for plotting Figs. 1-7 and Fig. 13 in the paper, i.e., except the predictions for 2D glassy system. It was designed for working on Google Colab. Notice that change may needed to be made for the root folder name of the files, called “FolderNameRoot” in the code.

* **JMPS\_Data/DataFiles** (folder)

Includes all the data files for the accepted JMPS paper.

* **JMPS\_Data/DataFiles** (folder)

Includes all the figures in the accepted JMPS paper.