

DYNA REFERENCE SERIES

Dyna User Guide

Simulation, fitting, tutorials, notebooks, and reproducible workflows

DynaRef 6.10.1

Institut Néel, CNRS · Grenoble, France

Generated from the maintained Dyna documentation and runnable examples

Contents

Use this guide sequentially for a first project, or jump directly to the cookbook and reference sections.

Contents	2
1. Orientation	4
What a calculation contains	4
Choose the right interface	4
Installation and launch	4
2. Sample and experiment model	5
Layer order and units	5
Scan definitions	5
Polarization	5
Formalism guide	5
3. First calculation	6
Run the prepared workflow	6
Run the same model from Python	6
Build a scan directly	6
4. Energy-scan tutorial	7
Fixed angle	7
Fixed Q	7
5. Fitting measured data	8
Input data	8
Run a single fit	8
Run a shared multfit	8
Interpretation checklist	8
6. Reproducible studies	9
Recommended directory	9
Record with every study	9
Reproduction commands	9

7. Cookbook	10
List and load samples	10
Plot a curve	10
Compare two formalisms	10
Sweep a parameter	10
Choose an export	10
8. Notebook and example catalog	11
Example directories	11
9. Troubleshooting and publication checklist	12
Common problems	12
Before publication	12
Further reference	12

1. Orientation

Dyna models specular X-ray reflection and transmission from layered samples. It supports ordinary, resonant, magnetic, and anisotropic workflows through a desktop application and scriptable interfaces.

What a calculation contains

- A sample ordered from the incident medium (usually Vacuum) to the substrate.
- An angular scan at fixed energy, or an energy scan at fixed angle or fixed Q.
- Incident and analyzed polarization states.
- A numerical formalism appropriate to the physics and numerical demands.

Practical note. The desktop application, Python API, command line, fitting workflows, and notebooks use the same calculation path.

Choose the right interface

Goal	Interface
Explore interactively	Desktop application: dyna
Reproducible batch calculation	YAML plus dyna-sim
Sweep or post-process arrays	Python API or notebook
Refine one or many scans	Fit/multifit workflow or GUI
Share a complete editable study	YAML project plus data files

Installation and launch

```
pip install DynaRef
dyna                # desktop application
dyna-sim --help     # scriptable workflows
```

For development, install the repository in editable mode. The package published on PyPI is named DynaRef; the Python import and application command are named dyna.

2. Sample and experiment model

Layer order and units

Layers are listed from Vacuum to substrate. Thickness, roughness, and wavelength-derived quantities use ångström conventions; angles are degrees, energy is electronvolts, and canonical density is formula units per cubic ångström.

- Vacuum must be the first layer.
- The last layer is the semi-infinite substrate and conventionally has zero thickness.
- Multiplicity compactly represents repeated layers.
- Roughness is an interface model parameter, not a universal replacement for interdiffusion.

Scan definitions

Scan	Varies	Held constant
Angular	Incidence angle	Photon energy
Energy, fixed angle	Photon energy	Incidence angle
Energy, fixed Q	Photon energy and angle	Momentum transfer Q

Scan endpoints are inclusive when the interval is an exact multiple of the step. Angular resolution, sample length, and beam thickness affect post-processing and should be recorded with the model.

Polarization

Configuration names are unpolarized, circ-plus, circ-minus, sigma, and pi. Use unpolarized as the analyzed state when the outgoing beam is not polarization selected.

Formalism guide

Family	Use
Parratt	Simple non-magnetic reflectivity
Elzo	Fast magnetic/resonant default
Stepanov	Thick or numerically demanding stacks
Magnetic Parratt	Supported transverse approximation
CF2 Parratt	Supported anisotropic case
Berreman variants	Exact validation; generally slower

Practical note. For publication, record the exact family and scan variant. Re-run final parameters with a second appropriate formalism when practical.

3. First calculation

The bundled W/Si workflow is the fastest route from installation to a reproducible curve.

Run the prepared workflow

```
dyna-sim examples/workflows/stepanov-angle.yaml \
  --output first-curve.csv
```

The CSV contains angle and calculated intensity. Plot intensity on a logarithmic vertical axis to reveal multilayer structure.

Run the same model from Python

```
from dyna import load_config

config = load_config('examples/workflows/stepanov-angle.yaml')
result = config.run()
print(result.x.shape, result.intensity.min(), result.intensity.max())
result.save('first-curve.npz')
```

Build a scan directly

```
from dyna import Experiment, Formalism, ScanSettings
from dyna import load_builtin_sample, run_reflectivity

experiment = Experiment()
experiment.Formalism = Formalism.STEPANOV_ANGLE
experiment.angle = ScanSettings(0.1, 5.0, 0.01)
experiment.energy.start = 700.0
result = run_reflectivity(load_builtin_sample('w-si'), experiment)
```

The result holds x, intensity, optional reversed intensity, amplitudes, and interpolated structure factors. CSV stores observable columns; NPZ preserves all numerical arrays.

4. Energy-scan tutorial

An energy scan must state what stays constant. Fixed angle and fixed Q are different experimental trajectories even though both use photon energy on the x axis.

Fixed angle

```
formalism: stepanov-energy
energy_scan_mode: fixed-angle
fixed_angle: 5.0
energy: {start: 680.0, stop: 740.0, step: 0.5}
```

Fixed Q

```
formalism: stepanov-energy
energy_scan_mode: fixed-q
fixed_q: 0.06
energy: {start: 680.0, stop: 740.0, step: 0.5}
```

The requested Q must be geometrically reachable throughout the interval. The notebook `06_energy_scans.ipynb` compares the curves and reports the angle trajectory implied by fixed Q.

Practical note. Always label the geometry in plots and archived metadata. Matching energy axes do not imply matching momentum-transfer paths.

5. Fitting measured data

Dyna can refine one sample against one curve or share parameters across several scans. Optimization finds a numerical minimum; it does not prove the physical model is unique.

Input data

```
x    intensity
x    intensity    sigma_intensity
```

The optional third column contains positive finite one-standard-deviation intensity uncertainties. When present for every fitted dataset, Dyna can report chi-square and reduced chi-square statistics.

Run a single fit

```
dyna-sim fit examples/workflows/fit-angle.yaml \
  --summary fit-summary.json \
  --output fitted.csv
```

Run a shared multifit

```
dyna-sim multifit examples/workflows/fe-si-four-scan-multifit.yaml \
  --summary multifit-summary.json \
  --output-dir fitted-curves
```

Interpretation checklist

- Plot data, model, and residuals separately.
- Check whether fitted values are pinned to bounds.
- Repeat from physically plausible alternative starting values.
- Inspect covariance and strong correlations when available.
- Fit complementary scans together when they constrain the same sample.

Practical note. Local covariance intervals assume a near-linear identifiable solution. Parameter bounds are not confidence intervals.

6. Reproducible studies

Recommended directory

```
study/  
  README.md  
  config/  
    simulation.yaml  
    fit.yaml  
  data/  
    measured.dat  
  results/  
    simulation.npz  
    fit-summary.json  
    fitted.csv
```

Record with every study

- Exact YAML workflow or project and every input data file.
- Dyna version and selected formalism.
- Units, polarization convention, and experimental geometry.
- Commands used to generate numeric outputs and figures.
- Checksums or persistent identifiers for externally archived large data.

Reproduction commands

```
dyna-sim config/simulation.yaml --output results/simulation.npz  
dyna-sim fit config/fit.yaml --summary results/fit-summary.json \  
  --output results/fitted.csv
```

Use relative data paths, rebuild figures from saved numeric results, and test the study from a clean environment before sharing.

7. Cookbook

List and load samples

```
from dyna import list_samples, load_built_in_sample
print(list_samples())
layers = load_built_in_sample('fe-si')
```

Plot a curve

```
import matplotlib.pyplot as plt
from dyna import load_config
result = load_config('examples/workflows/stepanov-angle.yaml').run()
plt.semilogy(result.x, result.intensity)
plt.xlabel('Angle (degrees)'); plt.ylabel('Reflectivity')
plt.show()
```

Compare two formalisms

```
from dataclasses import replace
from dyna import Formalism, load_config
base = load_config('examples/workflows/stepanov-angle.yaml')
fast = replace(base, formalism=Formalism.ELZO_ANGLE).run()
stable = replace(base, formalism=Formalism.STEPANOV_ANGLE).run()
```

Sweep a parameter

Run `examples/workflows/run_parameter_sweep.py`. It creates independent immutable configurations and writes one table with x plus an intensity column for every W thickness.

Choose an export

Format	Contents	Best for
CSV	x and observable intensity columns	Plotting and interchange
NPZ	Observables, amplitudes, factors	Complete numerical follow-up

8. Notebook and example catalog

Launch the interactive course with jupyter lab examples/notebooks.

Notebook	Topic
01	Basic reflectivity
02	Elzo and Stepanov comparison
03	Kerr and Faraday effects
04	Load a YAML project
05	Fit experimental or synthetic data
06	Fixed-angle and fixed-Q energy scans
07	Multiple reflectivity calculations
08	Immutable parameter sweeps
09	Result inspection and exports

Example directories

Directory	Contents
projects	Complete GUI/headless YAML studies
samples	Reusable sample-only YAML
workflows	Runnable configs and scripts
data	Small inputs for examples and tests
results	Reference outputs and plotting material

Installed packages include the collection. Export an editable copy with `dyna-sim examples --export`. Dyna refuses to overwrite changed exported examples unless `--force` is deliberately supplied.

9. Troubleshooting and publication checklist

Common problems

Symptom	Check first
Invalid sample	Vacuum first; substrate last; at least two layers
Missing scattering factors	Material name matches an available table
NaN or unstable curve	Density, limits, formalism, and extreme thickness
Fit stuck at a bound	Starting model and physical justification of bounds
Unexpected low-angle shape	Footprint, sample length, beam thickness, resolution

Before publication

- Retain exact machine-readable inputs and numeric outputs.
- State Dyna version, formalism, scan geometry, and polarization.
- Inspect residuals rather than reporting only the objective.
- Discuss parameter correlations and non-uniqueness.
- Cite Dyna and the numerical method used.

Further reference

The complete source documentation includes the reflectivity primer, formalism decision guide, transmissivity guide, YAML schema, scientific references, API documentation, pyGTM validation notes, and developer architecture.

Project: <https://gitlab.com/dynadevgroup/Dyna>

Documentation: <https://pydyna.readthedocs.io>

License: GPL-2.0-or-later