4_quantifying_uncertainty_due_to_heterogeneous_material_fields-solution

September 29, 2025

1 Quantifying uncertainty due to heterogeneous material fields

In this tutorial we'll have a look at prescribing heterogenous material parameter fields using 4C in order to do an uncertainty quantification on the resulting Cauchy stresses.

2 The 4C model

In this example, we'll have a look at a nonlinear solid mechanics problem. The momentum equation is given by

$$\ddot{u} = \nabla \cdot \sigma + b \text{ in } \Omega \times T$$

where u are the displacements, σ the Cauchy stress and b volumetric body forces. For the examples, $b = 0$, and we prescribe Dirichlet boundary conditions

$$u_x = 2.5\frac{t}{T_s} \text{ on } \Gamma_r = \{x = 12.5\} \times T$$

$$u_y = 2.5\frac{t}{T_s} \text{ on } \Gamma_t = \{y = 12.5\} \times T$$

$$u_y = 0 \text{ on } \Gamma_b = \{y = -12.5\} \times T$$

$$u_x = 0 \text{ on } \Gamma_l = \{x = -12.5\} \times T$$

where $T_s = 40$ and $T = [0, T_s]$. For $t = 0$ we set $u = 0$.

The domain Ω is a two-dimensional hyperelastic isochoric membrane, assuming plane stress with strain energy function

$$\Psi = \frac{\mu}{2}(J^{-2/3}I_C - 3)$$

where μ is the shear modulus, I_C is the first invariant of the Cauchy-Green tensor and J the determinant of the deformation gradient.

The problem is discretized in space using linear hexahedral finite elements and finite differences in time.

Note: This tutorial is based on this QUEENS test.

3 Uncertainty quantification

The goal of this example is to compute the expectation of the Cauchy stress tensor for $t = T_s$

$$\overline{\sigma} = \int \sigma(\mu) p(\mu) d\mu$$

where the shear modulus μ is a random field. This scenario lends itself to study the effect of the heterogenous constitutive material field on the Cauchy stress.

Using the law of the unconscious statistician, we can reformulate the problem as

$$\overline{\sigma} = \int \sigma(\mu) p(\mu) d\mu = \int \sigma(\mu(\theta)) p(\theta) d\theta$$

where θ are parameters describing the heterogenous material field $\mu(\theta)$.

3.1 The random field

Let's define a random field:

$$\mu(\theta) = 0.1 + \exp\left(-0.5 \frac{(x-\theta_1)^2 + (y-\theta_2)^2}{16}\right)$$

We set the distribution of the latent variable to be a multivariate Normal $p(\theta) = \mathcal{N}(\theta|0, 5I)$ with mean value 0 and a variance of 5.

[ ]:
```python
import numpy as np


def random_field_function(theta, positions):
    diff = positions.copy()[:, :2]

    diff[:, 0] = diff[:, 0] - theta[0]
    diff[:, 1] = diff[:, 1] - theta[1]
    return 0.1 + np.exp(-0.5 * (np.sum(diff**2, axis=1)) / 16)


from queens.distributions import Normal

theta_distribution = Normal(np.array([0, 0]), 5 * np.eye(2))
```

4 Monte Carlo integration

Since the integral above can not be computed analytically, we employ Monte Carlo integration

$$\overline{\sigma} = \int \sigma(\mu(\theta))p(\theta)d\theta \approx \frac{1}{N}\sum_{s=0}^{N} \sigma(\mu(\theta^{(s)}))$$

where $\theta^{(s)}$ are independent and identically distributed samples drawn from $p(\theta)$. For this examples we'll set $N = 100$, so 100 4C runs will be done per QUEENS run.

```
[ ]: # Let's get the domain from the 4C input file
from queens_interfaces.fourc.random_material_preprocessor import (
    create_jinja_json_template,
)
import pyvista as pv
pv.set_jupyter_backend("static")

# Open the mesh for the random field
mesh = pv.read("membrane_20.e")[0][0]

# Compute the cell centers
centers = mesh.cell_centers().points

# Construct the RF information
mu_rf_parameters = {
    "coords": centers,
    "keys": [f"MUE_{i}" for i in range(1, 1 + len(centers))],
}

# Contains the random field values
material_file_template = "material.json"

# Create the material file for 4C
create_jinja_json_template(
    "MUE",
    np.arange(1, len(centers) + 1),
    mu_rf_parameters["keys"],
    material_file_template,
)


from utils.random_field import CustomRandomField

random_field = CustomRandomField(
    mu_rf_parameters,  # Parameter names and coordinates
    theta_distribution,  # Latent variables
    random_field_function,  # Expansion, transformation from theta to mu
)
```

Now that we defined a random field, let's look at some samples!

[ ]:
```python
# Let's plot some samples

from utils.plot_input_random_field import plot_field

latent_samples = random_field.draw(3)
mu_samples = random_field.expanded_representation(latent_samples)

plotter = pv.Plotter(shape=(1, 3))

for i, mu_s in enumerate(mu_samples):
    plotter.subplot(0, i)
    plot_field(mu_s, plotter, f"Sample field mu {i}\n")

plotter.show()
```

As we can see, we set the material parameter to be constant in each element. We can think about this random field of a domain that has some type of defect where the mechanical properties are different.

[8]:
```python
# Let's do Monte Carlo integration

from queens.parameters import Parameters
from queens.data_processors import PvdFile
from queens.schedulers import Local
from queens_interfaces.fourc.driver import Fourc
from queens.models import Simulation
from queens.iterators import MonteCarlo
from queens.global_settings import GlobalSettings
from queens.main import run_iterator
from queens.utils.io import load_result
import pathlib

# Set the paths docker
home = pathlib.Path("/home/a11brebu")
fourc_executable = "/data/a11brebu/4C_build/4C"
input_template = "coarse_plate_dirichlet_template_herakles.4C.yaml"

if __name__ == "__main__":
    pathlib.Path("monte_carlo_4C").mkdir(exist_ok=True)
    with GlobalSettings(
        "monte_carlo_random_field_4C", output_dir="monte_carlo_4C"
    ) as gs:
        parameters = Parameters(MUE=random_field)

        # Extract the Cauchy stresses at the last time step
        data_processor = PvdFile(
            field_name="element_cauchy_stresses_xyz",
            file_name_identifier="output-structure.pvd",
            file_options_dict={},
            point_data=False,
            time_steps=[-1],
        )

        # How to run 4C
        driver = Fourc(
            parameters=parameters,
            input_templates={
                "input_file": input_template,  # Input file for 4C
                "material_file": material_file_template,  # File containing the␣
↪random field
            },
            executable=fourc_executable,
            data_processor=data_processor,
        )

        # Schedule parallel simulations
        scheduler = Local(num_procs=1, num_jobs=4, experiment_name=gs.
↪experiment_name)

        # our model
        model = Simulation(scheduler=scheduler, driver=driver)
        iterator = MonteCarlo(
            seed=1,
            num_samples=100,
            result_description={"write_results": True, "plot_results": False},
            model=model,
            parameters=parameters,
            global_settings=gs,
        )

        run_iterator(iterator, gs)

        results = load_result(gs.result_file("pickle"))
```

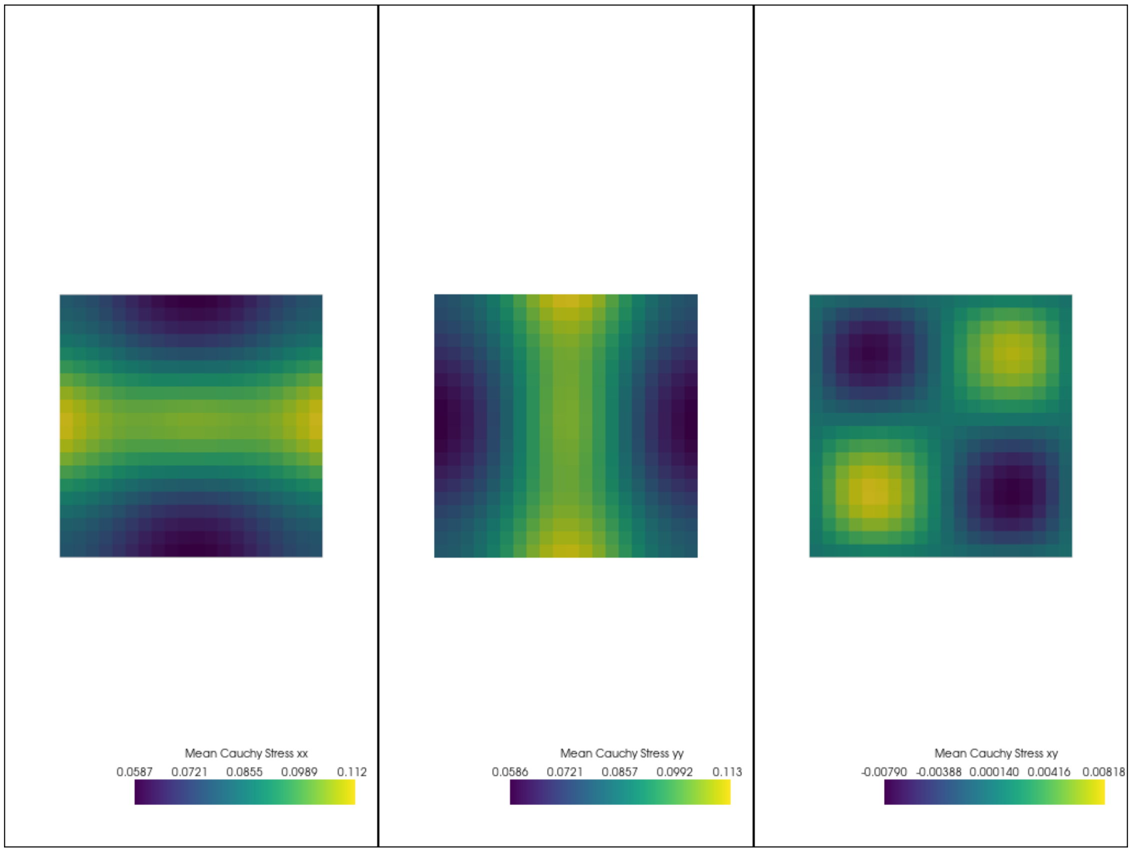
```
100%|     | 100/100 [01:32<00:00,  1.37it/s]


+--------------------------------------------------------------------------------
+
|                          Batch summary for jobs 0 - 99
|
|- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
-|
| number of jobs                 : 100
|
| number of parallel jobs        : 4
```

```
|
| number of procs              : 1
|
| total elapsed time           : 9.235e+01s
|
| average time per parallel job : 3.694e+00s
|
+--------------------------------------------------------------------------------
+

100%|     | 100/100 [01:32<00:00,  1.08it/s]

Time for CALCULATION: 92.41592502593994 s



2025-09-29 16:15:19,241 - distributed.scheduler - INFO - Retire worker addresses
(stimulus_id='retire-workers-1759162519.2409308') (0, 1, 2, 3)
2025-09-29 16:15:19,243 - distributed.nanny - INFO - Closing Nanny at
'tcp://127.0.0.1:35445'. Reason: nanny-close
2025-09-29 16:15:19,245 - distributed.nanny - INFO - Nanny asking worker to
close. Reason: nanny-close
2025-09-29 16:15:19,246 - distributed.nanny - INFO - Closing Nanny at
'tcp://127.0.0.1:37751'. Reason: nanny-close
2025-09-29 16:15:19,248 - distributed.nanny - INFO - Nanny asking worker to
close. Reason: nanny-close
2025-09-29 16:15:19,247 - distributed.worker - INFO - Stopping worker at
tcp://127.0.0.1:44275. Reason: nanny-close
2025-09-29 16:15:19,249 - distributed.nanny - INFO - Closing Nanny at
'tcp://127.0.0.1:44053'. Reason: nanny-close
2025-09-29 16:15:19,248 - distributed.worker - INFO - Removing Worker plugin
shuffle
2025-09-29 16:15:19,250 - distributed.nanny - INFO - Nanny asking worker to
close. Reason: nanny-close
2025-09-29 16:15:19,250 - distributed.worker - INFO - Stopping worker at
tcp://127.0.0.1:45553. Reason: nanny-close
2025-09-29 16:15:19,250 - distributed.worker - INFO - Removing Worker plugin
shuffle
2025-09-29 16:15:19,250 - distributed.core - INFO - Connection to
tcp://127.0.0.1:41717 has been closed.
2025-09-29 16:15:19,251 - distributed.nanny - INFO - Closing Nanny at
'tcp://127.0.0.1:42613'. Reason: nanny-close
2025-09-29 16:15:19,252 - distributed.nanny - INFO - Nanny asking worker to
close. Reason: nanny-close
2025-09-29 16:15:19,252 - distributed.worker - INFO - Stopping worker at
tcp://127.0.0.1:39675. Reason: nanny-close
2025-09-29 16:15:19,243 - distributed.nanny - INFO - Closing Nanny at
```

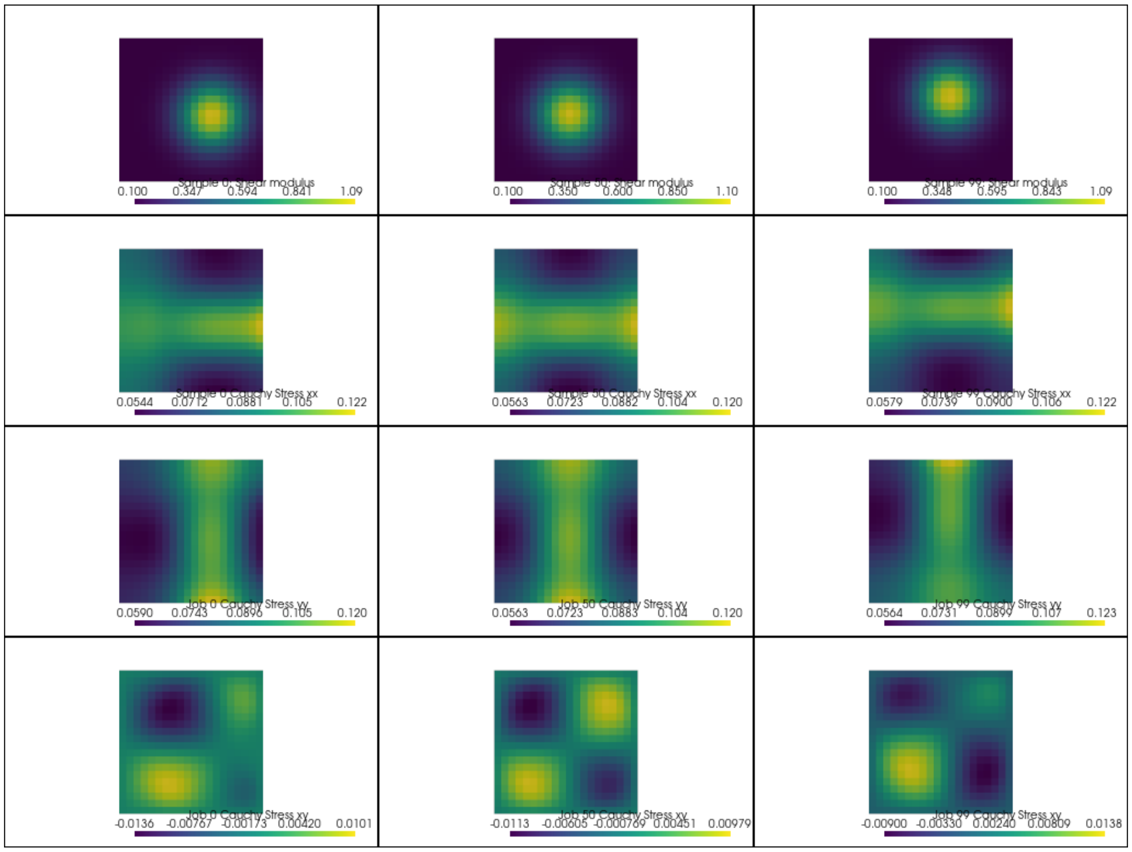
```
'tcp://127.0.0.1:35445'. Reason: nanny-close
2025-09-29 16:15:19,245 - distributed.nanny - INFO - Nanny asking worker to
close. Reason: nanny-close
2025-09-29 16:15:19,246 - distributed.nanny - INFO - Closing Nanny at
'tcp://127.0.0.1:37751'. Reason: nanny-close
2025-09-29 16:15:19,248 - distributed.nanny - INFO - Nanny asking worker to
close. Reason: nanny-close
2025-09-29 16:15:19,247 - distributed.worker - INFO - Stopping worker at
tcp://127.0.0.1:44275. Reason: nanny-close
2025-09-29 16:15:19,249 - distributed.nanny - INFO - Closing Nanny at
'tcp://127.0.0.1:44053'. Reason: nanny-close
2025-09-29 16:15:19,248 - distributed.worker - INFO - Removing Worker plugin
shuffle
2025-09-29 16:15:19,250 - distributed.nanny - INFO - Nanny asking worker to
close. Reason: nanny-close
2025-09-29 16:15:19,250 - distributed.worker - INFO - Stopping worker at
tcp://127.0.0.1:45553. Reason: nanny-close
2025-09-29 16:15:19,250 - distributed.worker - INFO - Removing Worker plugin
shuffle
2025-09-29 16:15:19,250 - distributed.core - INFO - Connection to
tcp://127.0.0.1:41717 has been closed.
2025-09-29 16:15:19,251 - distributed.nanny - INFO - Closing Nanny at
'tcp://127.0.0.1:42613'. Reason: nanny-close
2025-09-29 16:15:19,252 - distributed.nanny - INFO - Nanny asking worker to
close. Reason: nanny-close
2025-09-29 16:15:19,252 - distributed.worker - INFO - Stopping worker at
tcp://127.0.0.1:39675. Reason: nanny-close
2025-09-29 16:15:19,253 - distributed.worker - INFO - Removing Worker plugin
shuffle
2025-09-29 16:15:19,253 - distributed.core - INFO - Connection to
tcp://127.0.0.1:41717 has been closed.
2025-09-29 16:15:19,253 - distributed.nanny - INFO - Worker closed
2025-09-29 16:15:19,255 - distributed.worker - INFO - Stopping worker at
tcp://127.0.0.1:44961. Reason: nanny-close
2025-09-29 16:15:19,256 - distributed.core - INFO - Received 'close-stream' from
tcp://127.0.0.1:40564; closing.
2025-09-29 16:15:19,255 - distributed.worker - INFO - Removing Worker plugin
shuffle
2025-09-29 16:15:19,255 - distributed.core - INFO - Connection to
tcp://127.0.0.1:41717 has been closed.
2025-09-29 16:15:19,256 - distributed.nanny - INFO - Worker closed
2025-09-29 16:15:19,257 - distributed.core - INFO - Received 'close-stream' from
tcp://127.0.0.1:40548; closing.
2025-09-29 16:15:19,258 - distributed.core - INFO - Connection to
tcp://127.0.0.1:41717 has been closed.
2025-09-29 16:15:19,258 - distributed.nanny - INFO - Worker closed
2025-09-29 16:15:19,259 - distributed.scheduler - INFO - Remove worker addr:
tcp://127.0.0.1:44275 name: 0 (stimulus_id='handle-worker-
cleanup-1759162519.2598548')
2025-09-29 16:15:19,261 - distributed.scheduler - INFO - Remove worker addr:
tcp://127.0.0.1:45553 name: 1 (stimulus_id='handle-worker-
cleanup-1759162519.2613401')
2025-09-29 16:15:19,261 - distributed.nanny - INFO - Worker closed
2025-09-29 16:15:19,262 - distributed.core - INFO - Received 'close-stream' from
tcp://127.0.0.1:40558; closing.
2025-09-29 16:15:19,264 - distributed.scheduler - INFO - Remove worker addr:
tcp://127.0.0.1:39675 name: 2 (stimulus_id='handle-worker-
cleanup-1759162519.2648122')
2025-09-29 16:15:19,266 - distributed.core - INFO - Received 'close-stream' from
tcp://127.0.0.1:40580; closing.
2025-09-29 16:15:19,268 - distributed.scheduler - INFO - Remove worker addr:
tcp://127.0.0.1:44961 name: 3 (stimulus_id='handle-worker-
cleanup-1759162519.2680366')
2025-09-29 16:15:19,269 - distributed.scheduler - INFO - Lost all workers
2025-09-29 16:15:19,270 - distributed.batched - INFO - Batched Comm Closed <TCP
(closed) Scheduler connection to worker local=tcp://127.0.0.1:41717
remote=tcp://127.0.0.1:40558>
Traceback (most recent call last):
  File "/data/a11brebu/miniforge3/envs/queens-ukacm-gacm/lib/python3.11/site-
packages/distributed/batched.py", line 115, in _background_send
    nbytes = yield coro
             ^^^^^^^^^^
  File "/data/a11brebu/miniforge3/envs/queens-ukacm-gacm/lib/python3.11/site-
packages/tornado/gen.py", line 783, in run
    value = future.result()
            ^^^^^^^^^^^^^^^
  File "/data/a11brebu/miniforge3/envs/queens-ukacm-gacm/lib/python3.11/site-
packages/distributed/comm/tcp.py", line 263, in write
    raise CommClosedError()
distributed.comm.core.CommClosedError
2025-09-29 16:15:19,531 - distributed.nanny - INFO - Nanny at
'tcp://127.0.0.1:35445' closed.
2025-09-29 16:15:19,559 - distributed.nanny - INFO - Nanny at
'tcp://127.0.0.1:37751' closed.
2025-09-29 16:15:19,591 - distributed.nanny - INFO - Nanny at
'tcp://127.0.0.1:44053' closed.
2025-09-29 16:15:19,596 - distributed.nanny - INFO - Nanny at
'tcp://127.0.0.1:42613' closed.
2025-09-29 16:15:19,598 - distributed.scheduler - INFO - Closing scheduler.
Reason: unknown
2025-09-29 16:15:19,600 - distributed.scheduler - INFO - Scheduler closing all
comms
```

[9]:
```
# Let's plot the mean Cauchy stresses
```

```python
input_samples = results["input_data"]
cauchy_stresses = results["raw_output_data"]["result"]


plotter = pv.Plotter(shape=(1, 3))

plotter.subplot(0, 0)
plot_field(
    color_bar_title=f"Mean Cauchy Stress xx\n",
    field=results["mean"][:, 0],
    plotter=plotter,
)

plotter.subplot(0, 1)
plot_field(
    color_bar_title=f"Mean Cauchy Stress yy\n",
    field=results["mean"][:, 1],
    plotter=plotter,
)

plotter.subplot(0, 2)
plot_field(
    color_bar_title=f"Mean Cauchy Stress xy\n",
    field=results["mean"][:, 3],
    plotter=plotter,
)

plotter.show()
```



[10]:
```python
# Let's have a look at some outputs

plotter = pv.Plotter(shape=(4, 3))

job_ids = [0, 50, 99]
for i, job_id in enumerate(job_ids):
    plotter.subplot(0, i)
    plot_field(
        color_bar_title=f"Sample {job_id}: Shear modulus",
        field=random_field.expanded_representation(input_samples[job_id, :]),
        plotter=plotter,
    )
    plotter.subplot(1, i)
    plot_field(
        color_bar_title=f"Sample {job_id} Cauchy Stress xx",
        field=cauchy_stresses[job_id, :][:, 0],
        plotter=plotter,
    )
    plotter.subplot(2, i)
    plot_field(
        color_bar_title=f"Job {job_id} Cauchy Stress yy",
        field=cauchy_stresses[job_id, :][:, 1],
        plotter=plotter,
    )
    plotter.subplot(3, i)
    plot_field(
        color_bar_title=f"Job {job_id} Cauchy Stress xy",
        field=cauchy_stresses[job_id, :][:, 3],
        plotter=plotter,
    )
plotter.show()
```



Nice, we started 4C using QUEENS and prescribed a heterogeneous constitutive parameters! Note that compared to the previous examples, we only had to adapt the parameters! From driver to scheduler to model to iterator, nothing changed, highlighting the modularity of QUEENS.

5 Gaussian random fields

There are various ways of defining a random field, a common one being Gaussian random field (also known as Gaussian process)! The main aspect of Gaussian random fields $f(x, y)$, is that at every location x, y of the domain, the random field is normally distributed:

$f(x,y) \sim \mathcal{N}(f|\mu_f(x,y), K((x,y),(x',y')))$

Here $\mu_f(x,y)$ is the mean value function and K the covariance function. The latter one, describes the properties of random field, such as smoothness, lengthscales, etc.

Gaussian random fields can be sampled using the Kosambi–Karhunen–Loève theorem:

$$f^{(s)}(x,y) = \mu_f(x,y) + \sum_{k=1}^{\infty} \sqrt{\lambda_k}\xi_k^{(s)} e_k(x,y)$$

where λ_k and e_k are the k-th eigenvalues and eigenfunctions of K, where $\xi^{(s)}$ are samples of Gaussian random variables with zero mean and identity covariance matrix. This approach is similar to proper orthogonal decomposition.

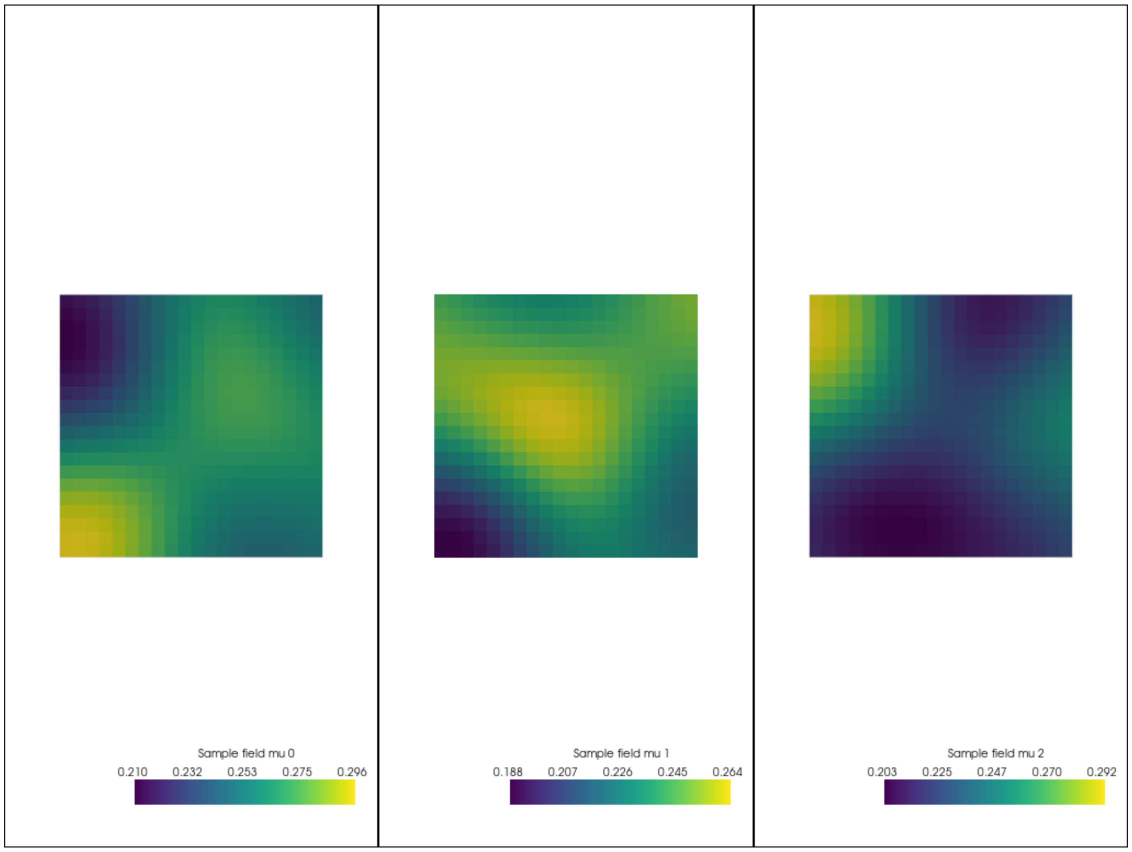
For these, we truncate this series at some value k_t. We select the covariance function

$$K((x,y),(x',y')) = \sigma_G^2 \exp\left(-\frac{(x-x')^2+(y-y')^2}{2l^2}\right)$$

where we set the length scale $l = 8.0$ and the output variance $\sigma_G^2 = 0.03$. The mean function is assumed constant as $\mu_f(x,y) = 0.25$. The number of eigenfunctions k_t is chosen such that the explained variance equals 0.95, i.e. $\frac{\sum_{k=1}^{k_t}\lambda_k}{\sum_{j=1}^{\infty}\lambda_j} \approx 0.95$

Enough theory, let's look at some samples

[11]:
```
from queens.parameters.random_fields import KarhunenLoeve

np.random.seed(42)

random_field = KarhunenLoeve(
    corr_length=8.0,
    std=0.03,
    mean=0.25,
    explained_variance=0.95,
    coords=mu_rf_parameters,
)

latent_samples = random_field.draw(3)
mu_samples = random_field.expanded_representation(latent_samples)

plotter = pv.Plotter(shape=(1, 3))

for i, mu_s in enumerate(mu_samples):
    plotter.subplot(0, i)
    plot_field(mu_s, plotter, f"Sample field mu {i}\n")

plotter.show()
```



You can see how the samples are distinct from the previous examples. They are more 'wiggly', yet still one can see a correlation structure. This is precisely what the length scale and variance does!

Try it out yourself, decrease/increase the parameters and look at the samples.

What do these parameters do?

Answer

According to the The Kernel Cookbook by David Duvenaud:

- The lengthscale determines the length of the 'wiggles' in your function. In general, you won't be able to extrapolate more than l units away from your data.
- The output variance σ_G^2 determines the average distance of your function away from its mean. Every kernel has this parameter out in front; it's just a scale factor.

Look at the cutoff value k_t (hint: `random_field.dimension`). How does this change depending on the length scale?

Answer

With smaller length scales, the 'wiggle'-frequency increases, hence more components are needed to explain the desired variance in eigenvalues of 95%.

Nice, now let us repeat the Monte Carlo experiment using the Gaussian random field describe above!

```
[12]: if __name__ == "__main__":
          with GlobalSettings(
              "monte_carlo_random_field_4C", output_dir="monte_carlo_4C"
          ) as gs:
              parameters = Parameters(MUE=random_field)

              # Extract the Cauchy stresses at the last time step
              data_processor = PvdFile(
                  field_name="element_cauchy_stresses_xyz",
                  file_name_identifier="output-structure.pvd",
                  file_options_dict={},
                  point_data=False,
                  time_steps=[-1],
                  files_to_be_deleted_regex_lst=[
                      "output.control",
                      "*.mesh.s0",
                      "output-vtk-files/*",
                  ],  # These files are deleted
              )

              # How to run 4C
              driver = Fourc(
                  parameters=parameters,
                  input_templates={
                      "input_file": input_template,  # Input file for 4C
                      "material_file": material_file_template,  # File containing the␣
       ↪random field
                  },
                  executable=fourc_executable,
                  data_processor=data_processor,
              )

              # Schedule parallel simulations
              scheduler = Local(num_procs=1, num_jobs=4, experiment_name=gs.
       ↪experiment_name)

              # our model
              model = Simulation(scheduler=scheduler, driver=driver)
              iterator = MonteCarlo(
                  seed=1,
                  num_samples=100,
                  result_description={"write_results": True, "plot_results": False},
                  model=model,
                  parameters=parameters,
                  global_settings=gs,
              )
```

```python
        run_iterator(iterator, gs)

        results = load_result(gs.result_file("pickle"))

# Let's plot the mean Cauchy stresses

input_samples = results["input_data"]

plotter = pv.Plotter(shape=(1, 3))

plotter.subplot(0, 0)
plot_field(
    color_bar_title=f"Mean Cauchy Stress xx\n",
    field=results["mean"][:, 0],
    plotter=plotter,
)

plotter.subplot(0, 1)
plot_field(
    color_bar_title=f"Mean Cauchy Stress yy\n",
    field=results["mean"][:, 1],
    plotter=plotter,
)

plotter.subplot(0, 2)
plot_field(
    color_bar_title=f"Mean Cauchy Stress xy\n",
    field=results["mean"][:, 3],
    plotter=plotter,
)

plotter.show()
```

```
                     *
                   * | *
                  * \|/ *
             * * * \|O|/ * * *
              \o\o\o|O|o/o/o/
              (<><><>O<><><>)
               '==========='

:'#######::'##::::'##:'########:'########:'##::: ##::'######::
'##… ##: ##:::: ##: ##…:: ##…:: ###:: ##:'##… ##:
 ##:::: ##: ##:::: ##: ##::::::: ##::::::: ####: ##: ##:::..::
 ##:::: ##: ##:::: ##: ######::: ######::: ## ## ##:. ######::
 ##:'## ##: ##:::: ##: ##…:::: ##…:::: ##. ####::… ##:
 ##:.. ##:: ##:::: ##: ##::::::: ##::::::: ##:. ###:'##::: ##:
```

```
          : ##### ##:. #######:: ########: ########: ##::. ##:. ######::
          :…:..:::…:::…::…::..::::..:::…:::


           A general purpose framework for Uncertainty Quantification,
            Physics-Informed Machine Learning, Bayesian Optimization,
                    Inverse Problems and Simulation Analytics


+--------------------------------------------------------------------------------
-----------------+
|                                            Parameters
|
|- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
- - - - - - - - - |
| self : <queens.parameters.parameters.Parameters object at 0x7eab521e6010>
|
| MUE  : <queens.parameters.random_fields.karhunen_loeve.KarhunenLoeve object at
0x7eac01443490> |
+--------------------------------------------------------------------------------
-----------------+



+--------------------------------------------------------------------------------
---------------------+
|                                               PvdFile
|
|- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
- - - - - - - - - - - |
| self                         : <queens.data_processors.pvd_file.PvdFile
object at 0x7eab486a3cd0> |
| field_name                   : 'element_cauchy_stresses_xyz'
|
| file_name_identifier         : 'output-structure.pvd'
|
| file_options_dict            : {}
|
| files_to_be_deleted_regex_lst : ['output.control', '*.mesh.s0', 'output-vtk-
files/*']               |
| time_steps                   : [-1]
|
| block                        : 0
|
| point_data                   : False
|
+--------------------------------------------------------------------------------
---------------------+
+-----------------------------------------------------------------------------
+
|                                      Local
|
|- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
-|
| self            : <queens.schedulers.local.Local object at 0x7eab486b0350>
|
| experiment_name : 'monte_carlo_random_field_4C'
|
| num_jobs        : 4
|
| num_procs       : 1
|
| restart_workers : False
|
| verbose         : True
|
+-----------------------------------------------------------------------------
+


2025-09-29 16:16:15,444 - distributed.scheduler - INFO - State start
2025-09-29 16:16:15,448 - distributed.scheduler - INFO -   Scheduler at:
tcp://127.0.0.1:35277
2025-09-29 16:16:15,449 - distributed.scheduler - INFO -   dashboard at:
http://127.0.0.1:8787/status
2025-09-29 16:16:15,450 - distributed.scheduler - INFO - Registering Worker
plugin shuffle
2025-09-29 16:16:15,461 - distributed.nanny - INFO -         Start Nanny at:
'tcp://127.0.0.1:32985'
2025-09-29 16:16:15,464 - distributed.nanny - INFO -         Start Nanny at:
'tcp://127.0.0.1:36053'
2025-09-29 16:16:15,467 - distributed.nanny - INFO -         Start Nanny at:
'tcp://127.0.0.1:43269'
2025-09-29 16:16:15,472 - distributed.nanny - INFO -         Start Nanny at:
'tcp://127.0.0.1:46741'
2025-09-29 16:16:15,769 - distributed.worker - INFO -       Start worker at:
tcp://127.0.0.1:42565
2025-09-29 16:16:15,769 - distributed.worker - INFO -          Listening to:
tcp://127.0.0.1:42565
2025-09-29 16:16:15,769 - distributed.worker - INFO -           Worker name:
0
2025-09-29 16:16:15,769 - distributed.worker - INFO -          dashboard at:
127.0.0.1:44969
2025-09-29 16:16:15,769 - distributed.worker - INFO - Waiting to connect to:
tcp://127.0.0.1:35277
```

```
2025-09-29 16:16:15,769 - distributed.worker - INFO -
-------------------------------------------------
2025-09-29 16:16:15,769 - distributed.worker - INFO -               Threads:
1
2025-09-29 16:16:15,769 - distributed.worker - INFO -                Memory:
125.85 GiB
2025-09-29 16:16:15,769 - distributed.worker - INFO -       Local Directory:
/tmp/dask-scratch-space/worker-yiz58x_g
2025-09-29 16:16:15,769 - distributed.worker - INFO -
-------------------------------------------------
2025-09-29 16:16:15,770 - distributed.worker - INFO -       Start worker at:
tcp://127.0.0.1:34427
2025-09-29 16:16:15,771 - distributed.worker - INFO -          Listening to:
tcp://127.0.0.1:34427
2025-09-29 16:16:15,771 - distributed.worker - INFO -           Worker name:
1
2025-09-29 16:16:15,771 - distributed.worker - INFO -          dashboard at:
127.0.0.1:33639
2025-09-29 16:16:15,771 - distributed.worker - INFO - Waiting to connect to:
tcp://127.0.0.1:35277
2025-09-29 16:16:15,771 - distributed.worker - INFO -
-------------------------------------------------
2025-09-29 16:16:15,771 - distributed.worker - INFO -               Threads:
1
2025-09-29 16:16:15,771 - distributed.worker - INFO -                Memory:
125.85 GiB
2025-09-29 16:16:15,771 - distributed.worker - INFO -       Local Directory:
/tmp/dask-scratch-space/worker-9rhaxhdz
2025-09-29 16:16:15,771 - distributed.worker - INFO -
-------------------------------------------------
2025-09-29 16:16:15,771 - distributed.worker - INFO -       Start worker at:
tcp://127.0.0.1:39547
2025-09-29 16:16:15,771 - distributed.worker - INFO -          Listening to:
tcp://127.0.0.1:39547
2025-09-29 16:16:15,771 - distributed.worker - INFO -           Worker name:
2
2025-09-29 16:16:15,771 - distributed.worker - INFO -          dashboard at:
127.0.0.1:45867
2025-09-29 16:16:15,771 - distributed.worker - INFO - Waiting to connect to:
tcp://127.0.0.1:35277
2025-09-29 16:16:15,771 - distributed.worker - INFO -
-------------------------------------------------
2025-09-29 16:16:15,771 - distributed.worker - INFO -               Threads:
1
2025-09-29 16:16:15,771 - distributed.worker - INFO -                Memory:
125.85 GiB
2025-09-29 16:16:15,771 - distributed.worker - INFO -       Local Directory:
/tmp/dask-scratch-space/worker-irjcrkas
```

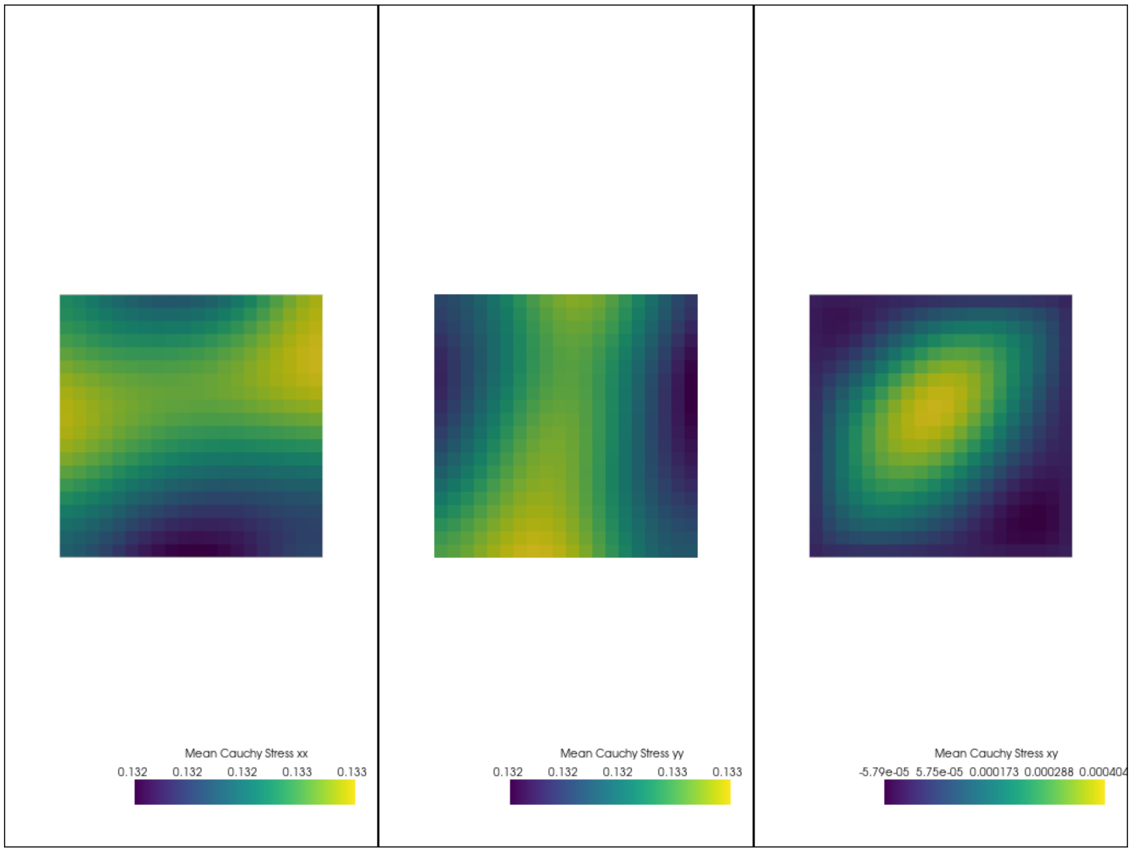
```
2025-09-29 16:16:15,771 - distributed.worker - INFO -
-------------------------------------------------
2025-09-29 16:16:15,779 - distributed.scheduler - INFO - Register worker addr:
tcp://127.0.0.1:42565 name: 0
2025-09-29 16:16:15,779 - distributed.worker - INFO -       Start worker at:
tcp://127.0.0.1:44103
2025-09-29 16:16:15,779 - distributed.worker - INFO -          Listening to:
tcp://127.0.0.1:44103
2025-09-29 16:16:15,779 - distributed.worker - INFO -           Worker name:
3
2025-09-29 16:16:15,779 - distributed.worker - INFO -          dashboard at:
127.0.0.1:41817
2025-09-29 16:16:15,779 - distributed.worker - INFO - Waiting to connect to:
tcp://127.0.0.1:35277
2025-09-29 16:16:15,779 - distributed.worker - INFO -
-------------------------------------------------
2025-09-29 16:16:15,779 - distributed.worker - INFO -               Threads:
1
2025-09-29 16:16:15,779 - distributed.worker - INFO -                Memory:
125.85 GiB
2025-09-29 16:16:15,779 - distributed.worker - INFO -       Local Directory:
/tmp/dask-scratch-space/worker-b_dlufcz
2025-09-29 16:16:15,779 - distributed.worker - INFO -
-------------------------------------------------
2025-09-29 16:16:15,782 - distributed.scheduler - INFO - Starting worker compute
stream, tcp://127.0.0.1:42565
2025-09-29 16:16:15,782 - distributed.worker - INFO - Starting Worker plugin
shuffle
2025-09-29 16:16:15,782 - distributed.worker - INFO -         Registered to:
tcp://127.0.0.1:35277
2025-09-29 16:16:15,782 - distributed.worker - INFO -
-------------------------------------------------
2025-09-29 16:16:15,783 - distributed.core - INFO - Starting established
connection to tcp://127.0.0.1:57116
2025-09-29 16:16:15,783 - distributed.core - INFO - Starting established
connection to tcp://127.0.0.1:35277
2025-09-29 16:16:15,786 - distributed.scheduler - INFO - Register worker addr:
tcp://127.0.0.1:34427 name: 1
2025-09-29 16:16:15,788 - distributed.scheduler - INFO - Starting worker compute
stream, tcp://127.0.0.1:34427
2025-09-29 16:16:15,788 - distributed.worker - INFO - Starting Worker plugin
shuffle
2025-09-29 16:16:15,788 - distributed.core - INFO - Starting established
connection to tcp://127.0.0.1:57120
2025-09-29 16:16:15,788 - distributed.worker - INFO -         Registered to:
tcp://127.0.0.1:35277
2025-09-29 16:16:15,788 - distributed.worker - INFO -
-------------------------------------------------
```

```
2025-09-29 16:16:15,788 - distributed.core - INFO - Starting established
connection to tcp://127.0.0.1:35277
2025-09-29 16:16:15,790 - distributed.scheduler - INFO - Register worker addr:
tcp://127.0.0.1:39547 name: 2
2025-09-29 16:16:15,791 - distributed.scheduler - INFO - Starting worker compute
stream, tcp://127.0.0.1:39547
2025-09-29 16:16:15,792 - distributed.worker - INFO - Starting Worker plugin
shuffle
2025-09-29 16:16:15,792 - distributed.worker - INFO -         Registered to:
tcp://127.0.0.1:35277
2025-09-29 16:16:15,792 - distributed.core - INFO - Starting established
connection to tcp://127.0.0.1:57130
2025-09-29 16:16:15,792 - distributed.worker - INFO -
-------------------------------------------------
2025-09-29 16:16:15,793 - distributed.core - INFO - Starting established
connection to tcp://127.0.0.1:35277
2025-09-29 16:16:15,796 - distributed.scheduler - INFO - Register worker addr:
tcp://127.0.0.1:44103 name: 3
2025-09-29 16:16:15,797 - distributed.scheduler - INFO - Starting worker compute
stream, tcp://127.0.0.1:44103
2025-09-29 16:16:15,798 - distributed.core - INFO - Starting established
connection to tcp://127.0.0.1:57134
2025-09-29 16:16:15,797 - distributed.worker - INFO - Starting Worker plugin
shuffle
2025-09-29 16:16:15,798 - distributed.worker - INFO -         Registered to:
tcp://127.0.0.1:35277
2025-09-29 16:16:15,798 - distributed.worker - INFO -
-------------------------------------------------
2025-09-29 16:16:15,798 - distributed.core - INFO - Starting established
connection to tcp://127.0.0.1:35277
2025-09-29 16:16:15,855 - distributed.scheduler - INFO - Receive client
connection: Client-9fdc67a3-9d4f-11f0-8810-be3af2b6059f
2025-09-29 16:16:15,856 - distributed.core - INFO - Starting established
connection to tcp://127.0.0.1:57148

To view the Dask dashboard open this link in your browser:
http://127.0.0.1:8787/status

+--------------------------------------------------------------------------------
+
|                                   Simulation
|
|- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
-|
| self      : <queens.models.simulation.Simulation object at 0x7eab48508d90>
|
| scheduler : <queens.schedulers.local.Local object at 0x7eab486b0350>
|
| driver    : <queens_interfaces.fourc.driver.Fourc object at 0x7eab487ef850>
|
+--------------------------------------------------------------------------------
+


+--------------------------------------------------------------------------------
----------+
|                                       MonteCarlo
|
|- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
- - - - -|
| self              : <queens.iterators.monte_carlo.MonteCarlo object at
0x7eab523e4c10> |
| model             : <queens.models.simulation.Simulation object at
0x7eab48508d90>     |
| parameters        : <queens.parameters.parameters.Parameters object at
0x7eab521e6010> |
| global_settings   : <queens.global_settings.GlobalSettings object at
0x7eab48771450>   |
| seed              : 1
|
| num_samples       : 100
|
| result_description : {'write_results': True, 'plot_results': False}
|
+--------------------------------------------------------------------------------
----------+


+--------------------------------------------------------------------------------
+
|                                git information
|
|- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
-|
| commit hash       : 8b58f1998eecf6465541aeba0b8dd7871f1b10f8
|
| branch            : main
|
| clean working tree : False
|
+--------------------------------------------------------------------------------
+

MonteCarlo for experiment: monte_carlo_random_field_4C

Starting Analysis…
```

```
100%|     | 100/100 [01:46<00:00,  2.99s/it]

+--------------------------------------------------------------------------------
+
|                       Batch summary for jobs 0 - 99
|
|- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
-|
| number of jobs               : 100
|
| number of parallel jobs      : 4
|
| number of procs              : 1
|
| total elapsed time           : 1.061e+02s
|
| average time per parallel job : 4.245e+00s
|
+--------------------------------------------------------------------------------
+

100%|     | 100/100 [01:46<00:00,  1.06s/it]

Time for CALCULATION: 106.68641519546509 s

2025-09-29 16:18:03,008 - distributed.scheduler - INFO - Retire worker addresses
(stimulus_id='retire-workers-1759162683.0083067') (0, 1, 2, 3)
2025-09-29 16:18:03,011 - distributed.nanny - INFO - Closing Nanny at
'tcp://127.0.0.1:32985'. Reason: nanny-close
2025-09-29 16:18:03,012 - distributed.nanny - INFO - Nanny asking worker to
close. Reason: nanny-close
2025-09-29 16:18:03,014 - distributed.nanny - INFO - Closing Nanny at
'tcp://127.0.0.1:36053'. Reason: nanny-close
2025-09-29 16:18:03,015 - distributed.nanny - INFO - Nanny asking worker to
close. Reason: nanny-close
2025-09-29 16:18:03,017 - distributed.nanny - INFO - Closing Nanny at
'tcp://127.0.0.1:43269'. Reason: nanny-close
2025-09-29 16:18:03,019 - distributed.nanny - INFO - Nanny asking worker to
close. Reason: nanny-close
2025-09-29 16:18:03,017 - distributed.worker - INFO - Stopping worker at
tcp://127.0.0.1:42565. Reason: nanny-close
2025-09-29 16:18:03,017 - distributed.worker - INFO - Removing Worker plugin
shuffle
2025-09-29 16:18:03,019 - distributed.worker - INFO - Stopping worker at
```

```
tcp://127.0.0.1:34427. Reason: nanny-close
2025-09-29 16:18:03,021 - distributed.nanny - INFO - Closing Nanny at
'tcp://127.0.0.1:46741'. Reason: nanny-close
2025-09-29 16:18:03,020 - distributed.worker - INFO - Removing Worker plugin
shuffle
2025-09-29 16:18:03,020 - distributed.core - INFO - Connection to
tcp://127.0.0.1:35277 has been closed.
2025-09-29 16:18:03,023 - distributed.nanny - INFO - Nanny asking worker to
close. Reason: nanny-close
2025-09-29 16:18:03,023 - distributed.worker - INFO - Stopping worker at
tcp://127.0.0.1:39547. Reason: nanny-close
2025-09-29 16:18:03,023 - distributed.worker - INFO - Removing Worker plugin
shuffle
2025-09-29 16:18:03,023 - distributed.core - INFO - Connection to
tcp://127.0.0.1:35277 has been closed.
2025-09-29 16:18:03,025 - distributed.nanny - INFO - Worker closed
2025-09-29 16:18:03,026 - distributed.worker - INFO - Stopping worker at
tcp://127.0.0.1:44103. Reason: nanny-close
2025-09-29 16:18:03,026 - distributed.core - INFO - Connection to
tcp://127.0.0.1:35277 has been closed.
2025-09-29 16:18:03,026 - distributed.worker - INFO - Removing Worker plugin
shuffle
2025-09-29 16:18:03,027 - distributed.core - INFO - Received 'close-stream' from
tcp://127.0.0.1:57116; closing.
2025-09-29 16:18:03,027 - distributed.nanny - INFO - Worker closed
2025-09-29 16:18:03,029 - distributed.core - INFO - Received 'close-stream' from
tcp://127.0.0.1:57120; closing.
2025-09-29 16:18:03,029 - distributed.core - INFO - Connection to
tcp://127.0.0.1:35277 has been closed.
2025-09-29 16:18:03,030 - distributed.nanny - INFO - Worker closed
2025-09-29 16:18:03,031 - distributed.core - INFO - Received 'close-stream' from
tcp://127.0.0.1:57130; closing.
2025-09-29 16:18:03,034 - distributed.scheduler - INFO - Remove worker addr:
tcp://127.0.0.1:42565 name: 0 (stimulus_id='handle-worker-
cleanup-1759162683.0347621')
2025-09-29 16:18:03,033 - distributed.nanny - INFO - Worker closed
2025-09-29 16:18:03,037 - distributed.scheduler - INFO - Remove worker addr:
tcp://127.0.0.1:34427 name: 1 (stimulus_id='handle-worker-
cleanup-1759162683.0370104')
2025-09-29 16:18:03,038 - distributed.scheduler - INFO - Remove worker addr:
tcp://127.0.0.1:39547 name: 2 (stimulus_id='handle-worker-
cleanup-1759162683.038709')
2025-09-29 16:18:03,043 - distributed.core - INFO - Received 'close-stream' from
tcp://127.0.0.1:57134; closing.
2025-09-29 16:18:03,045 - distributed.scheduler - INFO - Remove worker addr:
tcp://127.0.0.1:44103 name: 3 (stimulus_id='handle-worker-
cleanup-1759162683.0458865')
2025-09-29 16:18:03,047 - distributed.scheduler - INFO - Lost all workers
```

```
2025-09-29 16:18:03,434 - distributed.nanny - INFO - Nanny at
'tcp://127.0.0.1:43269' closed.
2025-09-29 16:18:03,441 - distributed.nanny - INFO - Nanny at
'tcp://127.0.0.1:46741' closed.
2025-09-29 16:18:03,457 - distributed.nanny - INFO - Nanny at
'tcp://127.0.0.1:32985' closed.
2025-09-29 16:18:03,473 - distributed.nanny - INFO - Nanny at
'tcp://127.0.0.1:36053' closed.
2025-09-29 16:18:03,476 - distributed.scheduler - INFO - Closing scheduler.
Reason: unknown
2025-09-29 16:18:03,477 - distributed.scheduler - INFO - Scheduler closing all
comms
```



6 Why QUEENS?

- Hide complexity in handling complex simulation codes (same as in the last example)
- QUEENS hides also the complexity of working with random fields:
  - Monte Carlo iterator does not care that it is a random field
  - We can still use the native 4C driver even though we now have 2 input files (one for the simulation parameters, one for the random field)

6.1 Let's play around

Let your creativity flow and try out stuff.

6.1.1 Inspirations

- Create your own random field definition, maybe a sine wave? (Keep in mind the parameter has to remain positive!)
- Look some simulation output of the Monte Carlo run (change to the data processor attribute `files_to_be_deleted_regex_lst` to `None`)
- Rerun the Monte Carlo analysis with different length scales
- Rerun the Monte Carlo analysis with different number of samples
- Plot the variance estimate from the Monte Carlo run
- Change length scale and look at the Monte Carlo mean value for multiple runs with the same number of samples