Simulation Running Guide (Modelers / Domain Experts)

Purpose

This document explains how to configure and run flood simulations using the Param Shavak ANUGA pipeline.

This guide is intended for:

- Flood modelers
- Simulation engineers
- HPC job operators

If environment is not installed → See INSTALL.md
If you only want to view results → See USER_GUIDE.md

---

1. Pre-Run Checklist (IMPORTANT)

Before running any simulation:

- ✔ ANUGA installed successfully
- ✔ MPI environment available
- ✔ Simulation inputs ready (DEM, shapefiles, config)
- ✔ GeoServer running (only if auto deployment is needed)

---

2. Step 1 — Load Simulation Environment

Always load MPI environment before running ANUGA.

```
source build/setup_mpi_env.sh
```

This ensures:

- Correct MPI compiler
- Correct library paths
- Correct ANUGA Python environment

---

3. Step 2 — Configure Simulation

Main configuration file: `mahanadi_test_case/settings.toml`

Key Sections

Domain

Controls spatial resolution and mesh size.

Higher resolution → More accuracy → More compute time.

Hydrology

Controls:

- Manning friction
- Boundary conditions
- Flow behavior

Simulation

Controls:

- yield_step → Output frequency
- final_time → Total simulation duration

---

4. Step 3 — Run Simulation

Small Test Runs (Sequential)

```
python3 mahanadi_test_case/simulate.py
```

HPC / Production Runs (MPI Parallel)

Example:

```
mpirun -np 16 python3 mahanadi_test_case/simulate.py
```

Adjust `-np` based on available cores.

---

5. What Happens During Simulation

Pipeline Flow:

```
simulate.py
↓
ANUGA Physics Engine
```

```
↓
.sww Output Generated
↓
bridge.py (auto or manual)
↓
GeoServer Deployment
```

---

6. Step 4 — Post Processing (Bridge)

Normally: simulate.py automatically triggers bridge.py.

Manual Bridge Execution

If you need to redeploy results:

```
python3 mahanadi_test_case/bridge.py <runid>
```

For timeseries generation:

```
python3 mahanadi_test_case/bridge.py <runid> --timeseries
```

---

7. Understanding Outputs

Location: `mahanadi_test_case/anuga_outputs/`

Output Files

File	Description
`.sww`	Raw ANUGA simulation output
`_max_depth.tif`	Maximum flood depth raster
`_meta.json`	Run metadata
`_timeseries/`	Time slice images / rasters

---

8. GeoServer Deployment Requirements

Bridge requires GeoServer to be running.

Start if needed:

```
make geoserver-start
```

Verify:

```
curl -I http://localhost:8080/geoserver
```

Expected: 200 OR 302 response.

---

9. Critical Pitfalls

Mesh Not Updating

Delete: `mesh_cache/`

GeoServer Connection Refused

Start GeoServer OR verify port.

Wrong Coordinate System

Default pipeline assumes: `EPSG:32645 (UTM Zone 45N)`

Update if modeling new region.

---

10. Performance Notes

Increasing mesh resolution increases:

- RAM usage
- Runtime
- Output file size

Parallel MPI scaling improves runtime but increases IO load.

---

11. Typical Workflow Summary

```
Edit settings.toml
↓
source setup_mpi_env.sh
↓
mpirun simulate.py
↓
bridge.py deploys outputs
```

```
↓
View in dashboard
```

Next Step

After successful deployment → Open USER_GUIDE.md to view results.