

GMI Verification Package – Build & Reproducibility Instructions

Audience: Reviewers and Reproducibility Leads

Target OS: Windows Server 2022, Ubuntu 22.04 LTS, macOS 14

1) Exact Environment Specs

- Coq: 8.20.1
- OCaml: 4.14.1
- mathcomp-analysis: 2.2.0
- Python: 3.11.7
- Python libs (pinned): numpy==2.1.*, scipy==1.13.*, pandas==2.2.*
- Shell tools: git >= 2.42, sha256sum (or shasum -a 256 on macOS)

2) Preflight Checks

- Verify PowerShell ≥ 5.1 on Windows: `$PSVersionTable.PSVersion`
- Verify Python: `python --version`
- Verify Coq: `coqc --version`
- Verify sha256 tool: `sha256sum --version` (or `shasum -a 256 --version`)

3) One-Command Verification

Windows (PowerShell)

```
```powershell
./verify_integrity.ps1
```
```

Linux/macOS (bash)

```
```bash
pwsh ./verify_integrity.ps1
```
```

4) Build Formal Proofs (parallel)

Windows (PowerShell)

```
```powershell
$jobs = [Math]::Max(1, [int]([Environment]::ProcessorCount) - 1)
cd proofs\v3.0.0\coq
make clean
make -j $jobs
```
```

Linux/macOS (bash)

```
```bash
cd proofs/v3.0.0/coq
make clean
make -j "$((nproc)-1)"
```
```

5) Run Empirical Validation with Provenance

Windows (PowerShell)

```
```powershell
cd proofs\v3.0.0\empirical_validation
python run_full_validation_suite.py --config
configs\gmi_validation_20251110.yaml --seed 42 --run-id
GMI_VAL_20251110_220933
```
```

Linux/macOS (bash)

```
```bash
```

```
cd proofs/v3.0.0/empirical_validation
python3 run_full_validation_suite.py --config
configs/gmi_validation_20251110.yaml --seed 42 --run-id
GMI_VAL_20251110_220933
```
```

6) Reproduce Quick Test

```
```bash
bash
cd proofs/v3.0.0/empirical_validation
python3 run_quick_test.py --seed 42
```

```powershell
PowerShell
cd proofs\v3.0.0\empirical_validation
python .\run_quick_test.py --seed 42
```
```

7) Expected Results

- Compilation: ~4-6 s; 7 theorems QED
- Mean deviation: < 0.1; Max deviation: < 0.2
- Zero violations within envelope; artifacts written under results/

8) Troubleshooting and Determinism

- If hashes fail, re-extract package to a clean path and rerun `verify_integrity.ps1`
- Deterministic seeds ensure reproducibility; confirm config path and Run ID in logs
- Cross-platform differences are mitigated by pinned toolchain; rebuild if `coqc` version differs