TMCMC×TSM-ROM (linearization management + analytical derivatives/JIT) Implementation Notes

Keisuke Nishioka
Project: IKM_Hiwi / tmcmc

January 17, 2026

One-page overview (for paper / talk)

What this is

We combine **Transitional Markov Chain Monte Carlo (TMCMC)** with a **TSM-ROM** (Taylor-series reduced-order model) and **linearization-point management** to make Bayesian inference feasible for an expensive forward model while keeping results auditable.

Key contributions (what is new here)

- **TMCMC with explicit stage control**: ESS-targeted $\Delta\beta$ updates, resampling, and K-step mutation.
- **TSM-ROM with linearization point updates**: turn linearization OFF for robust exploration; turn it ON later and update θ_0 to stay accurate near the posterior mass.
- **Reproducibility by construction**: each run persists configuration, likelihood definition, diagnostics tables, and logs; no "information leakage" is required for inference.

Reproducibility recipe (one command)

Run the full pipeline (experiment + `REPORT.md`):

```
python tmcmc/run_pipeline.py --mode paper --seed 123 --run-id paper_M1_seed123_fixed
--models M1 --lock-paper-conditions --use-paper-analytical
```

Paper-fixed conditions (enforced by `--lock-paper-conditions`):

- Observation noise: `sigma_obs = 0.01`
- Relative covariance: `cov_rel = 0.005`
- Conservative β jumps (max_delta_beta capped)

Audit checklist for a valid posterior run:

- β **reaches 1.0** (`subprocess.log`: "beta reached 1.0")
- **likelihood definition is persisted** (`likelihood_meta_*.json`)
- **diagnostics tables exist** (`diagnostics_tables/*.csv`)

Run artifacts (what to cite / archive)

Artifact	Purpose
`config.json`	full config + seeds (re-run exactly)
`likelihood_meta_*.json`	explicit likelihood definition (audit)
`diagnostics_tables/*.csv`	β schedule, acceptance, ROM error, θ_0 history
`subprocess.log`, `pipeline.log`	end-to-end provenance + failure diagnosis
`figures/*.png`	posterior + fits (paper-ready visuals)

Paper-fixed run reference

For paper comparison, use a run with `--lock-paper-conditions` and record the `run-id`. Figures can be referenced from `tmcmc/_runs/<run-id>/figures/*.png` (no embedding needed). Example: `tmcmc/_runs/paper_M1_seed123_fixed/figures/`

1 Purpose

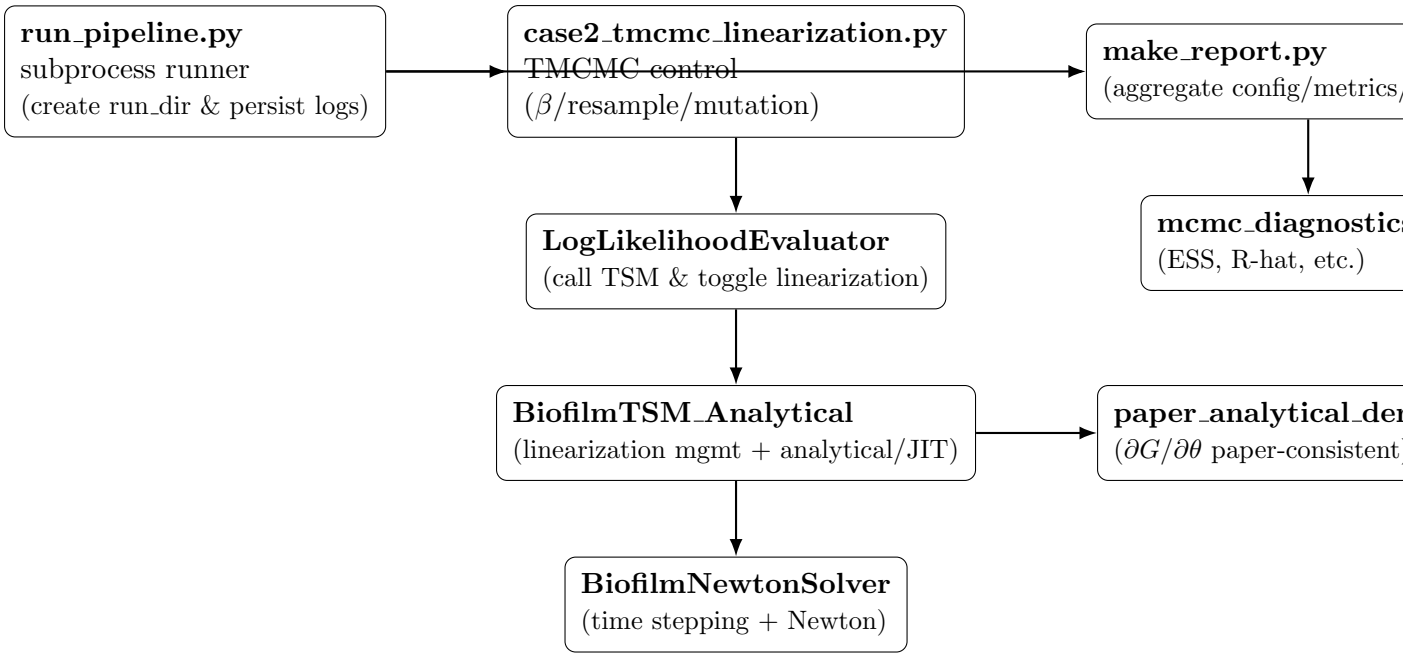
This document summarizes the **program flow**, **module boundaries**, and the key drivers of **reproducibility**, **performance**, and **inference accuracy** for the Case II runner whose entry point is `tmcmc/case2_tmcmc_linearization.py`.

2 Key modules (execution-critical dependency set)

Minimal set of modules that matter for actual runs (import-traceable):

- Entry / experiment control: `tmcmc/case2_tmcmc_linearization.py`
- Configuration: `tmcmc/config.py`
- Physical model + base TSM: `tmcmc/improved1207_paper_jit.py`
- TSM (linearization point mgmt + analytical derivatives/JIT): `tmcmc/demo_analytical_tsm_with_lineari`
- Analytical derivatives (paper mode): `tmcmc/paper_analytical_derivatives.py`
- Diagnostics: `tmcmc/mcmc_diagnostics.py`
- $\theta \to (A, b)$ patch (complex-step readiness): `tmcmc/bugfix_theta_to_matrices.py`
- Post-run reporting: `tmcmc/make_report.py`, pipeline wrapper: `tmcmc/run_pipeline.py`

2.1 Module map (concept)



Dominant cost: solve_tsm / run_deterministic / $x^{(1)}$ (sensitivities)

3 End-to-end flow

3.1 Pipeline wrapper

`run_pipeline.py` creates a `run_dir` and then runs:

1. `case2_tmcmc_linearization.py` (the experiment)
2. `make_report.py` (generate `REPORT.md` from `run_dir`)

Combined stdout/stderr are tee'd into `subprocess.log` under `run_dir`.

3.2 TMCMC loop (high level)

Conceptual stage loop (implemented in `run_TMCMC`):

1. Initialize particles from prior
2. For stages $s = 0, 1, \ldots$:
   - Update $\beta_s \to \beta_{s+1}$ based on target ESS ratio (with min/max $\Delta\beta$)
   - Evaluate likelihood for each particle (calls TSM)
   - Update weights $\to$ normalize $\to$ compute ESS $\to$ resample
   - Mutation (MCMC steps) to restore diversity
   - Enable linearization (later stages) and update linearization point θ_0 at intervals
3. Critical: **ensure β reaches 1.0** (look for "β reached 1.0" in logs)

3.3 TSM (linearization management + analytical/JIT)

`BiofilmTSM_Analytical.solve_tsm()` supports:

- **Linearization OFF** (early exploration): full non-linear TSM evaluation
- **Linearization ON** (later stages): speed up via $x(\theta) \approx x(\theta_0) + \frac{\partial x}{\partial \theta}\big|_{\theta_0}(\theta - \theta_0)$
- Linearization point update: `update_linearization_point(`$\theta_{\texttt{new}}$`)` invalidates caches and recomputes $x^{(0)}(\theta_0)$ and $x^{(1)}$

3.4 Physical solver (Newton + time stepping)

`BiofilmNewtonSolver.run_deterministic()` integrates the deterministic trajectory; each step solves a Newton system based on residual Q and Jacobian J. Time-dependent antibiotics are supported via `alpha_schedule` (e.g., `M3_val`).

4 Mathematical validation (Hamilton principle → strong form → implemented residual)

We added an explicit math-level validation for the physical model implementation in `tmcmc/improved1207_pape`
The goal is **not** to prove the full continuum theory, but to show the following: **the paper's strong form equations (biofilm_simulation, eqs. (16)–(18)) reduced to a material point and discretized by implicit Euler are exactly the same as the residual Q solved by the code**.

4.1 Paper definitions (model skeleton)

Let ϕ_i be volume fractions, ψ_i be fractions of living cells, and $\bar{\phi}_i = \phi_i\psi_i$ be living biomass. The holonomic volume constraint is

$$f(\phi) = \sum_{l=0}^{n} \phi_l - 1 = 0,$$

enforced via a Lagrange multiplier γ. The free energy density and dissipation potential (paper eqs. (10),(14)) are:

$$\Psi(\phi, \psi) = -\frac{1}{2}c^* \bar{\phi}^{\mathsf{T}} A\, \bar{\phi} + \frac{1}{2}\alpha^* \psi^{\mathsf{T}} B\, \psi, \tag{1}$$

$$\Delta_s(\dot{\bar{\phi}}, \dot{\phi}) = \frac{1}{2}\dot{\bar{\phi}}^{\mathsf{T}} \eta\, \dot{\bar{\phi}} + \frac{1}{2}\dot{\phi}^{\mathsf{T}} \eta\, \dot{\phi}. \tag{2}$$

4.2 Strong form (16)–(18) and mapping to residual Q

From the Hamilton principle evaluation, the paper gives (for each species i):

$$0 = -c^*\psi_i(A\bar{\phi})_i + \eta_i(\dot{\phi}_i\psi_i^2 + \bar{\phi}_i\dot{\psi}_i + \dot{\phi}_i) + \gamma, \tag{3}$$

$$0 = -c^*\phi_i(A\bar{\phi})_i + \alpha^* b_i\psi_i + \eta_i(\dot{\psi}_i\phi_i^2 + \bar{\phi}_i\dot{\phi}_i), \tag{4}$$

$$0 = \sum_{l=0}^{n} \phi_l - 1. \tag{5}$$

The implementation uses a material-point model and implicit Euler time discretization ($\dot{x} \approx (x^{n+1} - x^n)/\Delta t$), and solves $Q(g^{n+1}) = 0$ at each step via Newton. Key one-to-one correspondences are:

- Interaction term $(A\bar{\phi})_i$: `Interaction = A @ (phi * psi)`

- Constraint: `Q[9] = sum(phi) + phi0 - 1`
- γ **must not appear in the** ψ **equations** because the constraint depends only on ϕ (a mathematical requirement)

Additionally, the code includes a **barrier/penalty term** (coefficient K_p) to enforce $0 < \phi, \phi_0, \psi < 1$ as discussed in the paper.

4.3 Automated regression checks

To keep this math-level agreement from regressing, we added pytest checks:

- With barrier disabled ($K_p = 0$), the discretized strong form matches the implemented residual Q
- The ψ equations are independent of γ

Tests live in `tmcmc/test_hamilton_model_consistency.py`; the detailed note is `tmcmc/HAMILTON_VALIDATION.`

5 Accuracy drivers

The **likelihood definition** is the top driver (e.g., σ_{obs}, variance model). In particular, including/excluding $\mathbf{Cov}(\bar{\phi}, \bar{\psi})$ **in** $\mathbf{Var}(\bar{\phi}\bar{\psi})$ can change inference materially. Keep it audit-able in `likelihood_meta_*.json`.

Why Cov matters (short intuition). Even if the observable is the product $\bar{\phi}\bar{\psi}$, the uncertainty model depends on whether fluctuations in $\bar{\phi}$ and $\bar{\psi}$ are treated as correlated. Ignoring correlation can lead to systematic under/over-confidence in the likelihood and thus a materially different posterior.

Note on paper condition mismatch Runs with `--sigma-obs 0.02` will generally not match paper figures that often use 0.01 (not a bug; it changes quantitative fit).

6 Reproducibility (audit artifacts)

Minimum artifacts to keep under `run_dir`:

- `config.json`: run configuration, seeds, TMCMC/model params
- `likelihood_meta_*.json`: explicit likelihood definition
- `diagnostics_tables/*.csv`: β schedule, acceptance, ROM error, θ_0 history
- `subprocess.log / pipeline.log`: progress + "β reached 1.0"

7 Performance drivers

Rule of thumb:

$$\text{total time} \approx (\#\text{likelihood evaluations}) \times (\text{cost per TSM evaluation}) \tag{6}$$

Dominant components:

- **Largest**: `BiofilmTSM_Analytical.solve_tsm()`

- **Largest**: `BiofilmNewtonSolver.run_deterministic()` + `compute_Q_vector()` + `compute_Jacobian_ma`
- **Large**: sensitivity generation $x^{(1)}$ (especially when linearization is off)
- **Medium**: TMCMC mechanics (mutation/resample/β update)
- **Small**: plotting and I/O (can grow depending on settings)

8 Critical checks

- $\beta = 1$ **reached**: otherwise you did not reach the posterior
- **NaN/Inf**: ensure no NaN/Inf in solve_tsm / Newton
- **Complex-step readiness**: `theta_to_matrices` preserves complex dtype
- **Analytical derivative validation**: verify against complex-step reference

9 Common failure modes (symptom → likely cause → fix)

- β **never reaches 1.0** → too strict ESS target / too few stages → increase `--n-stages` or relax `--target-ess-ratio`, check min/max $\Delta\beta$.
- **Low acceptance / frozen mutation** → proposal too narrow or linearization too aggressive → increase mutation scale / steps, delay linearization threshold, cap $\|\Delta\theta_0\|$.
- **ROM error spikes after update** → θ_0 jump too large → reduce update interval or step cap, add ROM-gated enabling.
- **Posterior too narrow/wide** → likelihood variance model mismatch → audit `likelihood_meta_*.json` (especially Cov handling) and σ_{obs}.

10 Example figures (auto-picked best run)

Best run id: `m1_check_np100_ns15`. The following figures are included *if present* under `tmcmc/_runs/m1_check_np100_ns15`.

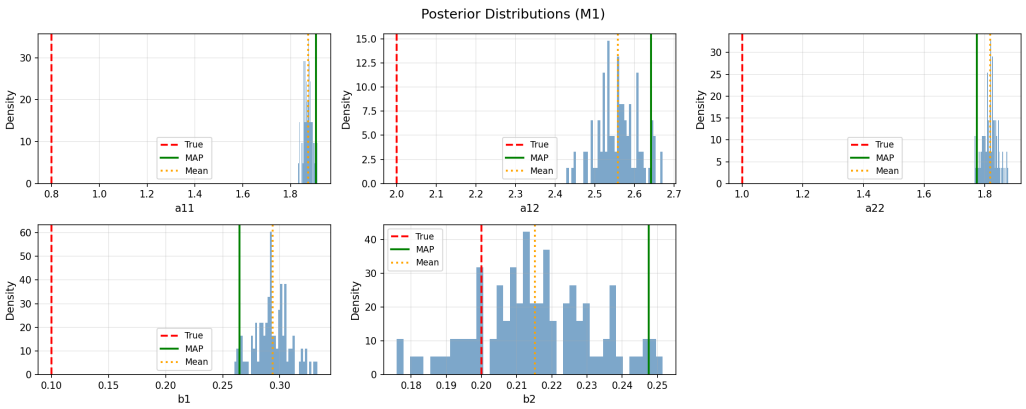


Figure 1: Posterior for M1 (example).

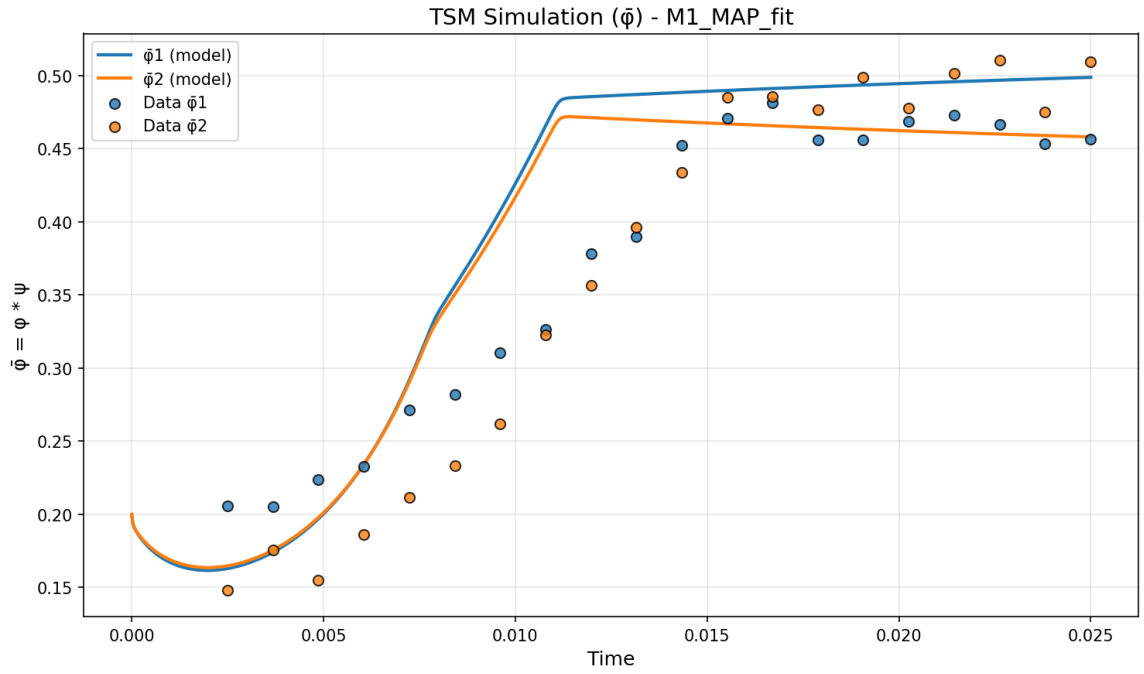


Figure 2: MAP fit vs data for M1 (example).

11 Figure ideas (ready-to-use)

1. TMCMC β schedule (per chain)
2. ROM error at linearization update events (pre/post)
3. $\|\Delta\theta_0\|$ history (stability of updates)
4. Cost–accuracy tradeoff (FOM evals or wall-time vs MAP error)
5. Posterior plots (M1/M2/M3/M3_val) aligned to paper figures

12 Appendix: one-sentence summary

TMCMC stabilizes the transition from prior to posterior while periodically updating the TSM-ROM linearization point, reducing expensive FOM evaluations without sacrificing estimation accuracy near the MAP.

13 Related Work

This work integrates TMCMC (Transitional Markov Chain Monte Carlo) with TSM-ROM (Time-Separated Stochastic Mechanics reduced-order model) to enable efficient Bayesian inference under hybrid uncertainty. We organize related research by category below.

13.1 TMCMC (Transitional Markov Chain Monte Carlo)

TMCMC was proposed by Ching & Chen (2007) as an MCMC method that achieves gradual transition from prior to posterior via β tempering[5]. Compared to conventional MCMC methods, TMCMC is tune-free and naturally provides estimates of model evidence. Betz et al. (2016) proposed observations and improvements to TMCMC, demonstrating practical performance gains[6].

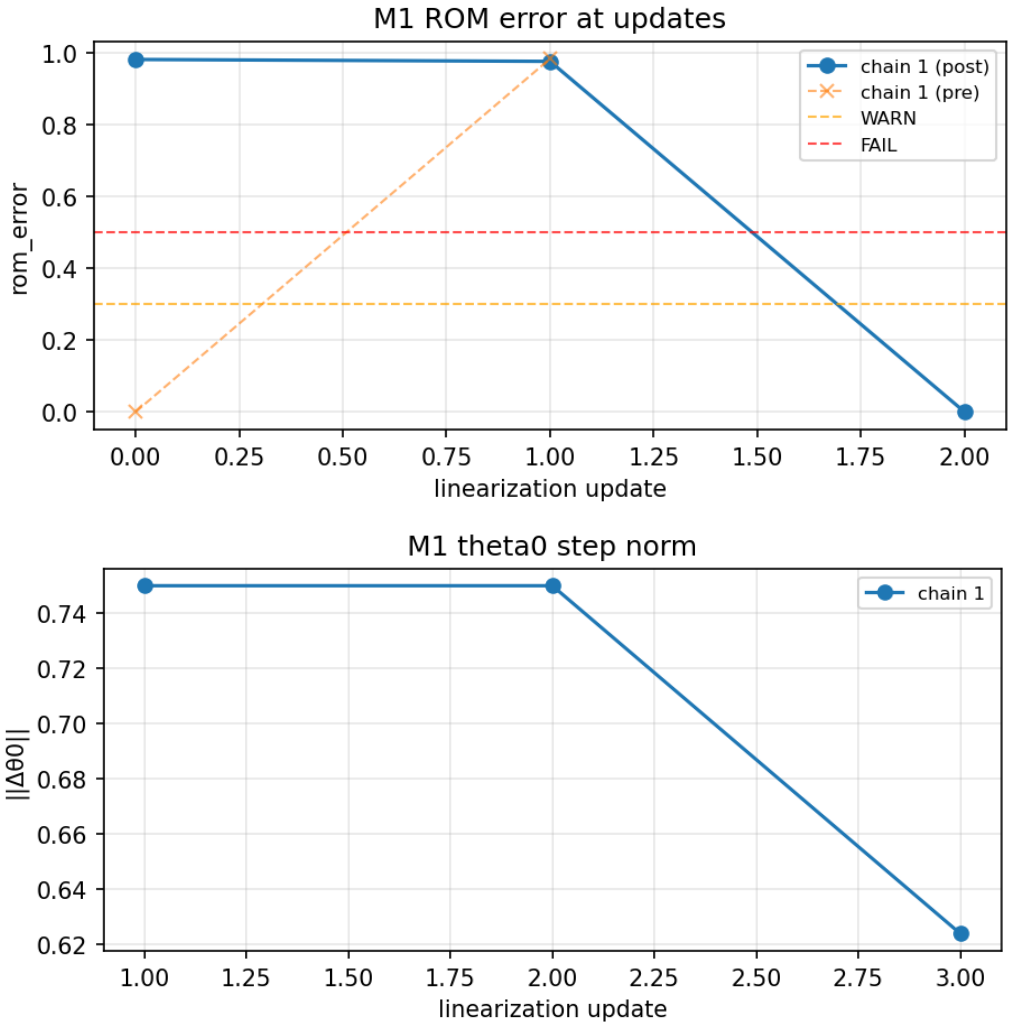


Figure 3: ROM error and $\|\Delta\theta_0\|$ history (optional diagnostics).

Recent extensions include:

- **X-TMCMC** (Angelikopoulos et al., 2015): Integrates Kriging surrogate models to reduce computational cost[9].
- **Generalized TMCMC** (Lu et al., 2021): Addresses inefficiencies in tempering schedules for broad applicability[10].
- **BASIS** (Wu et al., 2017): An unbiased version of TMCMC that fixes bias issues[11].
- **CTMCMC** (Ma et al., 2025): Uses copula functions in proposal distributions for complex, high-dimensional distributions[12].

13.2 TSM-ROM (Time-Separated Stochastic Mechanics)

TSM was proposed by Geisler & Junker (2023) as an efficient uncertainty propagation method that separates time-dependence from stochasticity[7]. Compared to Monte Carlo methods, TSM can estimate expectation and variance with few deterministic simulations. Geisler et al. (2025) presented a comprehensive TSM framework and demonstrated applicability to inelastic material models[8].

Key features of TSM:

- Applicable to complex material models with internal variable evolution via separation of time-dependence and stochasticity
- Significant computational cost reduction via linear or low-order Taylor expansion
- Efficiency achieved through approximation in stochastic parameter space rather than spatial DOF reduction

13.3 Hybrid Uncertainty Quantification

Bayesian inference under hybrid uncertainty (epistemic + aleatory) faces computational challenges with conventional double-loop procedures. Beck & Katafygiotis (1998) established the statistical framework for Bayesian model updating[26]. Kennedy & O'Hagan (2001) proposed a hybrid uncertainty framework that represents model inadequacy as a statistical discrepancy term[27].

Fritsch et al. (2025) achieved Bayesian updating of bacterial biofilms under hybrid uncertainty using TSM-ROM as a surrogate model[1]. This work extends that research by integrating TMCMC with TSM-ROM and achieving both accuracy and efficiency through dynamic linearization point updates.

13.4 Reduced-Order Models for Uncertainty Quantification

ROM methods for uncertainty quantification are an important research area for computational cost reduction. Benner et al. (2015) provided a survey of projection-based ROM methods for parametric dynamical systems[28]. Peherstorfer et al. (2018) provided a survey of multifidelity methods for UQ[29].

Polynomial Chaos Expansion (Xiu & Karniadakis, 2002) is a method that efficiently performs UQ via expansion in stochastic parameter space[30]. TSM is similar to this approach but is particularly applicable to inelastic material models through separation of time-dependence and stochasticity.

13.5 Bayesian Inference and MCMC Methods

The foundation of MCMC methods dates back to the Metropolis algorithm (Metropolis et al., 1953) and the Metropolis-Hastings algorithm (Hastings, 1970)[31, 32]. Sequential Monte Carlo (Del Moral et al., 2006) provides a similar approach to TMCMC via population-based sampling[33].

13.6 Surrogate Models and Emulators

As alternatives to expensive physical models, surrogate models such as Gaussian Process Regression (Rasmussen & Williams, 2006) and Bayesian Emulation (Conti & O'Hagan, 2010) are used[34, 35]. This work uses TSM-ROM as a surrogate model, achieving both accuracy and efficiency through analytical sensitivity computation compared to GP-based methods.

13.7 Adaptive Surrogate Models and Active Learning

Recent work has focused on adaptive surrogate models with active learning. Villani et al. (2024) proposed adaptive GP surrogates using KL divergence-based acquisition criteria, reducing forward model evaluations[13]. Xu et al. (2024) proposed GP surrogates for multimodal posteriors combined with ensemble smoothers[14]. Meles et al. (2025) achieved two orders of magnitude computational savings through sequential surrogate refinement with posterior-guided training[15]. Scheurer et al. (2025) proposed UA-SABI (Uncertainty-Aware Surrogate-based Amortized Bayesian Inference) with explicit surrogate uncertainty modeling[16].

13.8 Adaptive Tempering Schedules and ESS-based Methods

Research on ESS-based adaptive tempering schedules has advanced. Zhao & Pillai (2024) optimized temperature ladders using policy gradients with ACT/ESS as metrics[17]. Peña & Jenkins (2025) proposed reddemcee, achieving adaptive tempering based on multiple objectives including ESS-based metrics[18]. Li et al. (2024) analyzed the optimality of swap acceptance rate ≈ 0.234 across dimensions and tuning regimes[19]. Wang et al. (2025) combined normalizing flows with ESS-based adaptive annealing schedules, achieving approximately $10\times$ computational savings[20].

13.9 Adaptive Linearization Point Updates in ROMs

Research on adaptive linearization point updates in ROMs has also advanced. Farhat et al. (2020) achieved in-situ adaptive reduction with libraries of local ROBs and online updates[21]. MORe DWR (2024) proposed incremental POD with dual-weighted residual (DWR) error estimators, updating linearization points incrementally[22]. Goal-oriented adaptive sampling (2025) proposed sampling new linearization points based on error to maintain ROM validity[23]. Interpolated adaptive linear ROM (2025) proposed dynamically adjusting linearization mappings via Grassmann interpolation[24].

13.10 Biofilm Modeling with Bayesian Inference

Applications of Bayesian inference to biofilm modeling have also progressed. A 2023 study performed Bayesian estimation of viscoelastic parameters for *Pseudomonas aeruginosa* biofilms using MCMC to estimate posterior distributions[25]. These studies demonstrate the applicability of our biofilm model approach.

13.11 Positioning of This Work

This work extends and integrates existing research in the following ways:

- **Integration of TMCMC and TSM-ROM**: Combines both methods to enable efficient Bayesian inference under hybrid uncertainty
- **Dynamic linearization point updates**: Maintains first-order approximation accuracy of TSM-ROM by updating linearization points near MAP in later TMCMC stages (related to adaptive ROM research)
- **ESS-targeted $\Delta\beta$ control**: Adaptive tempering schedule based on ESS targets rather than fixed schedules (related to adaptive tempering research)
- **Practical improvements**: Implements practical enhancements such as K-step mutation and observation-based linearization point selection
- **Surrogate uncertainty consideration**: Monitors TSM-ROM error and controls timing of linearization point updates (related to UA-SABI research)

14 References

References

[1] Fritsch, L., Geisler, H., Grashorn, J., Klempt, F., Soleimani, M., Broggi, M., Junker, P., Beer, M. *Bayesian updating of bacterial microfilms under hybrid uncertainties with a novel surrogate model.* (Local PDF: `tmcmc/Bayesian updating of bacterial microfilms under hybrid uncertainties with a novel surrogate model - Kopie.pdf`).

[2] Klempt, F., Geisler, H., Soleimani, M., Junker, P. *A continuum multi-species biofilm model with a novel interaction scheme.* arXiv:2509.01274v1, 2025-09-01. (Local PDF: tmcmc/biofilm_simulation.pdf).

[3] Junker, P., Balzani, D. *An extended Hamilton principle as unifying theory for coupled problems and dissipative microstructure evolution.* Continuum Mechanics and Thermodynamics, 33(4), 1931--1956 (2021). DOI: 10.1007/s00161-021-01017-z. (Local PDF: tmcmc/hamiltonian.pdf).

[4] Heine, N., Bittroff, K., Szafrański, S. P., Duitscher, M., Behrens, W., Vollmer, C., Mikolai, C., Kommerein, N., Debener, N., Frings, K., Heisterkamp, A., Scheper, T., Torres-Mapa, M. L., Bahnemann, J., Stiesch, M., Doll-Nikutta, K. *Influence of species composition and cultivation condition on peri-implant biofilm dysbiosis in vitro.* Frontiers in Oral Health, 6:1649419 (2025). DOI: 10.3389/froh.2025.1649419. (Local PDF: tmcmc/Influence of species composition and cultivation condition on peri-implant biofilm dysbiosis in vitro.pdf).

[5] Ching, J., Chen, Y.-C. *Transitional Markov Chain Monte Carlo Method for Bayesian Model Updating, Model Class Selection, and Model Averaging.* Journal of Engineering Mechanics, 133(7), 816--832 (2007). DOI: 10.1061/(ASCE)0733-9399(2007)133:7(816).

[6] Betz, W., Papaioannou, I., Straub, D. *Transitional Markov Chain Monte Carlo: Observations and Improvements.* Journal of Engineering Mechanics, 142(5), 04016016 (2016). DOI: 10.1061/(ASCE)EM.1943-7889.0001066.

[7] Geisler, H., Junker, P. *Time-separated stochastic mechanics for the simulation of viscoelastic structures with local random material fluctuations.* Computer Methods in Applied Mechanics and Engineering, 407, 115916 (2023). DOI: 10.1016/j.cma.2023.115916.

[8] Geisler, H., Erdogan, C., Nagel, J., Junker, P. *A new paradigm for the efficient inclusion of stochasticity in engineering simulations: Time-separated stochastic mechanics.* Computational Mechanics, 75(1), 211--235 (2025). DOI: 10.1007/s00466-024-02500-5.

[9] Angelikopoulos, P., Papadimitriou, C., Koumoutsakos, P. *X-TMCMC: Adaptive Kriging for Bayesian Inverse Problems.* Computer Methods in Applied Mechanics and Engineering, 289, 409--428 (2015). DOI: 10.1016/j.cma.2015.02.011.

[10] Lu, D., Khalil, M., Catanach, T. et al. *Generalized Transitional Markov Chain Monte Carlo for Bayesian Inverse Problems.* arXiv preprint arXiv:2112.02180 (2021).

[11] Wu, S., Angelikopoulos, P., Papadimitriou, C., Koumoutsakos, P. *BASIS: Bayesian Annealed Sequential Importance Sampling.* Journal of Computational Physics, 335, 289--299 (2017). DOI: 10.1016/j.jcp.2017.01.010.

[12] Ma, P., Zhang, Y., Cai, E., Luo, M., Guo, X. *Copula-based Transitional Markov Chain Monte Carlo for Bayesian Model Updating.* Reliability Engineering & System Safety, 265, 110772 (2025). DOI: 10.1016/j.ress.2025.110772.

[13] Villani, M. et al. *Adaptive Gaussian Process Regression for Bayesian Inverse Problems*. arXiv preprint arXiv:2404.19459 (2024).

[14] Xu, W. et al. *Adaptive Gaussian Process for Multi-modal Bayesian Inverse Problems*. arXiv preprint arXiv:2409.15307 (2024).

[15] Meles, G. A. et al. *Bayesian Full Waveform Inversion with Sequential Surrogate Model Refinement*. arXiv preprint arXiv:2505.03246 (2025).

[16] Scheurer, M. et al. *Uncertainty-Aware Surrogate-based Amortized Bayesian Inference (UA-SABI)*. arXiv preprint arXiv:2505.08683 (2025).

[17] Zhao, T., Pillai, N. S. *Policy Gradients for Optimal Parallel Tempering MCMC*. arXiv preprint arXiv:2409.01574 (2024).

[18] Peña, J., Jenkins, D. *reddemcee: Adaptive Parallel Tempering Ensemble Sampler*. arXiv preprint arXiv:2509.24870 (2025).

[19] Li, Y., Wang, Y., Dou, Z., Rosenthal, J. S. *Temperature Spacing via Swap Acceptance* $\approx$ *0.234*. arXiv preprint arXiv:2408.06894 (2024).

[20] Wang, Y., Chi, C., Dinner, A. R. *Normalizing Flows + Annealing with ESS-based Adaptive Schedule*. arXiv preprint arXiv:2505.03652 (2025).

[21] Farhat, C. et al. *In-situ Adaptive Reduction of Nonlinear Multiscale Structural Dynamics*. arXiv preprint arXiv:2004.00153 (2020).

[22] *Adaptive Space-Time Model Order Reduction with Dual-Weighted Residual (MORe DWR)*. Archive of Applied Mechanics (2024). DOI: 10.1186/s40323-024-00262-6.

[23] *Goal-oriented Adaptive Sampling for Projection-based ROMs*. Computers & Structures (2025).

[24] *Interpolated Adaptive Linear ROM for Deformation Dynamics*. arXiv preprint arXiv:2509.25392 (2025).

[25] *Viscoelastic Properties of Pseudomonas aeruginosa Biofilms: Bayesian Estimation*. Biophysical Reports, 3, 100130 (2023). DOI: 10.1016/j.bpr.2023.100130.

[26] Beck, J. L., Katafygiotis, L. S. *Updating Models and Their Uncertainties. I: Bayesian Statistical Framework*. Journal of Engineering Mechanics, 124(4), 455--461 (1998). DOI: 10.1061/(ASCE)0733-9399(1998)124:4(455).

[27] Kennedy, M. C., O'Hagan, A. *Bayesian Calibration of Computer Models*. Journal of the Royal Statistical Society: Series B, 63(3), 425--464 (2001). DOI: 10.1111/1467-9868.00294.

[28] Benner, P., Gugercin, S., Willcox, K. *A Survey of Projection-Based Model Reduction Methods for Parametric Dynamical Systems*. SIAM Review, 57(4), 483--531 (2015). DOI: 10.1137/130932715.

[29] Peherstorfer, B., Willcox, K., Gunzburger, M. *Survey of Multifidelity Methods for Uncertainty Quantification, Propagation, and Optimization*. SIAM Review, 60(3), 550--591 (2018). DOI: 10.1137/16M1082469.

[30] Xiu, D., Karniadakis, G. E. *The Wiener--Askey Polynomial Chaos for Stochastic Differential Equations*. SIAM Journal on Scientific Computing, 24(2), 619--644 (2002). DOI: 10.1137/S1064827501387826.

[31] Metropolis, N., Rosenbluth, A. W., Rosenbluth, M. N., Teller, A. H., Teller, E. *Equation of State Calculations by Fast Computing Machines*. The Journal of Chemical Physics, 21(6), 1087--1092 (1953). DOI: 10.1063/1.1699114.

[32] Hastings, W. K. *Monte Carlo Sampling Methods Using Markov Chains and Their Applications*. Biometrika, 57(1), 97--109 (1970). DOI: 10.1093/biomet/57.1.97.

[33] Del Moral, P., Doucet, A., Jasra, A. *Sequential Monte Carlo Samplers*. Journal of the Royal Statistical Society: Series B, 68(3), 411--436 (2006). DOI: 10.1111/j.1467-9868.2006.00553.x.

[34] Rasmussen, C. E., Williams, C. K. I. *Gaussian Processes for Machine Learning*. MIT Press (2006). ISBN: 026218253X.

[35] Conti, S., O'Hagan, A. *Bayesian Emulation of Complex Multi-Output and Dynamic Computer Models*. Journal of Statistical Planning and Inference, 140(3), 640--651 (2010). DOI: 10.1016/j.jspi.2009.08.006.