

# Geometry, Quadrature, and Trainability in Battery Models: A Unifying View for Efficient, Stable, and Differentiable Simulation

August 29, 2025

## Abstract

We argue that the stability and trainability of physics-based lithium-ion battery models (from ECM/SPM to DFN/P2D) are governed by a single organizing theme: *geometry-aware quadrature*. In time, Gaussian collocation underlies stiff ODE/DAE integrators; in space, the correct inner products (e.g. the  $r^2$  measure in spherical particles) dictate which bases and quadrature rules are optimal. We distill practical recipes for parameter identification and optimal charging that pair stiff adaptive solvers with fixed Gauss loss quadrature, and we outline a research program—including adjoint-aware mesh refinement and spectral-element particle solvers—aimed at advancing accuracy-per-DOF and gradient stability. Throughout, we provide rigorous but accessible derivations and cite the foundational literature.

## 1 Thesis and scope

Battery models are multi-scale and multi-physics. Yet, from the perspective of numerical analysis and optimization, they share a common structure:

$$\text{(Dynamics)} \quad \mathcal{M}(y; \theta) \dot{y} = f(t, y, u; \theta), \quad 0 = g(t, y, u; \theta) \text{ (DAE)} \quad (1)$$

$$\text{(Objective)} \quad J = \int_0^T \ell(y(t), u(t), \text{data}(t)) dt + \mathcal{R}(\theta, u). \quad (2)$$

Two choices dominate stability & learnability: (i) the *time integrator*, which is fundamentally a quadrature/collocation device, and (ii) the *spatial quadrature* that respects the PDE geometry (e.g. spherical, cylindrical). Our theme is that *geometry-matched quadrature* yields sparse, well-conditioned discretizations and steady gradients for learning.

## 2 Quadrature, collocation, and stiff ODE/DAE solvers

Many high-order implicit Runge–Kutta (IRK) schemes arise from *collocation* at Gaussian nodes; Gauss–Legendre gives order  $2s$  with  $s$  stages, Radau IIA gives  $2s - 1$  and  $L$ -stability—a workhorse for stiff problems and DAEs citehairerwanner. In DAEs, enforcing the residual at collocation nodes ensures both differential and algebraic parts are satisfied to high order; index issues are addressed by index reduction and consistent initialization.

**Proposition 1** (Loss quadrature decoupled from solver adaptivity). *Let the forward solve of (1) be performed by an adaptive stiff method (e.g. BDF or Radau IIA). Approximate  $J$  in (2) on each time subinterval by a fixed  $m$ -point Gauss–Legendre rule, independent of internal solver steps. Then, the gradient  $\nabla_{\theta} J$  computed by discrete adjoints is (i) invariant to internal step rejections and (ii) Lipschitz in small step-size perturbations, provided  $\ell$  is  $C^1$  and the solver is consistent. Sketch: the quadrature error of  $J$  is controlled uniformly by Gauss–Legendre; discrete adjoints differentiate a fixed quadrature stencil rather than a solver-dependent mesh, eliminating step-jitter terms.*

### 3 Battery models as DAEs and where the weights come from

The Doyle–Fuller–Newman (DFN/P2D) framework couples solid/electrolyte diffusion and charge transport and is typically semi-discretized to an index-1 DAE solved by BDF-type methods; excellent overviews appear in [?, 6, 7, 8]. For the solid particle (radius  $R$ ) the weak form of radial diffusion reads, for test  $v$ ,

$$\int_0^R v \partial_t c r^2 dr = -D_s \int_0^R r^2 \partial_r c \partial_r v dr + [D_s r^2 v \partial_r c]_0^R. \quad (3)$$

The *natural inner product* is  $\langle u, v \rangle = \int_0^R uv r^2 dr$ , i.e. weight  $r^2$ . This geometry dictates the numerics: either (i) **FVM** that integrates  $r^2$  exactly over shells (conservative), or (ii) **spectral/SEM** based on Jacobi polynomials with  $(\alpha, \beta) = (0, 2)$  and Gauss–Jacobi quadrature in  $r$  for high accuracy per DOF [3, 4]. PyBaMM exposes these measures directly in its integral operators Cartesian, cylindrical, spherical) and implements FVM and FEM backends [11, 12].

### 4 FVM vs. spectral-element for particle diffusion

**FVM**: midpoint/trapezoidal-like formulas on control volumes yield sparse matrices and exact conservation; with spherical geometry, volumes scale like  $r^2 dr$  by construction. **SEM**: choosing basis  $\{P_k^{(0,2)}\}$  on  $[0, 1]$  and Gauss–Jacobi nodes delivers near-spectral accuracy for smooth signals (e.g. EIS, pulses). The choice is problem-driven: smooth inputs favor SEM; strong nonlinearities or non-smooth data favor FVM with mesh grading or AMR. Recent AMR work for P2D/P3D demonstrates significant efficiency gains with adaptive refinement and immersed interfaces [13, 14].

### 5 Optimization and identification: adjoints + quadrature

Optimal charging and parameter ID share the same form: minimize (2) subject to (1). Two numerically robust routes:

- (a) **Differentiate through a DAE solver with sensitivities**: use IDAS (SUNDIALS) for forward/adjoint sensitivities on DAEs; pair with fixed Gauss loss quadrature (Prop. 1) [9, 10].
- (b) **Direct collocation**: transcribe on Gauss/Radau nodes per interval, enforcing defects at the same nodes and integrating  $J$  with matching weights (sparse NLP; high order) [1].

For ECM/SPM where ODE reduction is valid, differentiable ODE solvers suffice; for DFN-level DAEs, DAE-aware adjoints are essential.

### 6 Actionable recipes (rigorous yet practical)

**R1: Gauss loss on fixed grids.** Simulate with stiff adaptive time stepping; evaluate  $J$  on a fixed Gauss–Legendre grid per observation interval. This decouples gradient calculations from solver adaptivity and reduces variance in training.

**R2: Geometry-matched space.** For particle diffusion, use either (i) FVM with exact shell volumes and graded meshes towards  $r = R$ , or (ii) SEM on few elements with Gauss–Jacobi  $(0, 2)$  nodes; both respect the  $r^2$  measure.

**R3: Use DAE sensitivities.** For DFN-class models, rely on IDAS adjoints; avoid forcing DAEs into ODE-only frameworks for training.

**R4: Collocation for constrained problems.** For optimal charging with path/terminal constraints, use Gauss/Radau collocation; same nodes for dynamics and cost simplify KKT structure.

**R5: Flatness-informed parametrizations.** Where appropriate (e.g. reduced/circuit models), parameterize trajectories via flat outputs to limit search spaces; see differential flatness foundations [2].

## 7 Placeholders for numerical experiments (to be filled with results)

**E1 (Accuracy-per-DOF):** FVM vs. SEM on spherical diffusion under sinusoidal/pulsed currents. Metrics: error vs. DOF; Newton iterations; wall-time.

*Placeholder for tables/plots.*

**E2 (Gradient stability):** Train SPM parameters from synthetic data; compare (i) Gauss loss vs. (ii) endpoint-sampled loss. Measure gradient variance and convergence.

*Placeholder for tables/plots.*

**E3 (DAE sensitivities):** DFN parameter ID using IDAS adjoints vs. ODE-only autodiff; compare accuracy, stability, and runtime.

*Placeholder for tables/plots.*

**E4 (Adjoint-driven AMR):** Goal-oriented refinement using dual-weighted residuals on P2D; track effect on voltage error and cost functional.

*Placeholder for tables/plots.*

**E5 (Control transcription):** Optimal fast-charge with thermal/voltage constraints via Radau collocation; compare to simulate-then-optimize.

*Placeholder for tables/plots.*

## 8 Publication-oriented mathematical notes

- **Well-posedness:** Clarify index-1 structure after semi-discretization; state assumptions for existence uniqueness and consistent initialization.
- **Error estimates:** Provide a priori error bounds for SEM in the  $r^2$ -weighted  $H^1$  norm; cite generalized Jacobi bases; outline  $hp$ -strategy.
- **Adjoint consistency:** Show that discrete adjoints of the collocated NLP coincide (to order) with continuous adjoints, ensuring correct gradients for ID/control.
- **AMR indicators:** Specify dual-weighted residuals tied to *voltage* or *cost* as goals; prove estimator reliability/efficiency on 1D/1D×radius P2D.
- **Reproducibility:** Release code and parameter sets; document tolerances and stopping criteria; include unit-consistent nondimensionalization.

## 9 Conclusion

Treating time and space through the lens of *geometry-aware quadrature* yields discretizations that are both physically faithful and optimization-friendly. This perspective unifies solver stability, gradient quality, and accuracy-per-DOF. The proposed experiments target practical gains (shorter training times, fewer DOFs) while retaining rigor (error and sensitivity control).

**Acknowledgments and data/code availability.** Placeholder.

## References

- [1] E. Hairer and G. Wanner, *Solving Ordinary Differential Equations II: Stiff and Differential-Algebraic Problems*, Springer (1996/2010). See Radau IIA & collocation discussions. Available overview: [https://www.researchgate.net/publication/230800481\\_Solving\\_Ordinary\\_Differential\\_Equations\\_II](https://www.researchgate.net/publication/230800481_Solving_Ordinary_Differential_Equations_II).
- [2] M. Fliess, J. Lévine, P. Martin, P. Rouchon (1995), *Flatness and defect of non-linear systems: introductory theory and examples*. PDF: <https://cas.minesparis.psl.eu/~rouchon/publications/PR1995/IJC95.pdf>.
- [3] L. N. Trefethen (2000), *Spectral Methods in MATLAB*, SIAM. See Gauss/Clenshaw–Curtis discussion. <https://epubs.siam.org/doi/pdf/10.1137/1.9780898719598.bm?download=true>.
- [4] B.-Y. Guo and J. Shen (2006), *Optimal spectral-Galerkin methods using generalized Jacobi polynomials*, J. Sci. Comput. Preprint. [https://www.math.purdue.edu/~shen7/pub/short\\_GJP.pdf](https://www.math.purdue.edu/~shen7/pub/short_GJP.pdf).
- [5] Z. Chen *et al.* (2022), *Porous Electrode Modeling and its Applications to Li-Ion Batteries*, Adv. Energy Materials. <https://juser.fz-juelich.de/record/910992/files/Advanced%20Energy%20Materials%20-%202022%20-%20Chen%20-%20Porous%20Electrode%20Modeling%20and%20its%20Applications%20to%20Li%E2%80%90Ion%20Batteries.pdf>.
- [6] A. M. Bizeray *et al.* (2015), *Model-based battery management system design*, arXiv:1506.08689. <https://arxiv.org/pdf/1506.08689>.
- [7] R. B. Smith, J. C. Bazan, M. Z. Bazant (2017), *Multiphase Porous Electrode Theory*, J. Electrochem. Soc. <https://dspace.mit.edu/bitstream/handle/1721.1/134525/J.%20Electrochem.%20Soc.-2017-Smith-E3291-310.pdf>.
- [8] R. Drummond *et al.* (2020), *A feedback interpretation of the Doyle–Fuller–Newman model*, IEEE CSS Tech. Rep. [https://eprints.whiterose.ac.uk/id/eprint/209085/8/cst\\_paper.pdf](https://eprints.whiterose.ac.uk/id/eprint/209085/8/cst_paper.pdf).
- [9] SUNDIALS IDAS user guide: *Implicit DAE solver with sensitivity analysis*. [https://sundials.readthedocs.io/en/latest/idas/Introduction\\_link.html](https://sundials.readthedocs.io/en/latest/idas/Introduction_link.html).
- [10] SUNDIALS overview: CVODE/IDA/IDAS and sensitivity capabilities. <https://sundials.readthedocs.io/>.
- [11] PyBaMM docs: integral operators and measures (Cartesian, cylindrical, spherical). [https://docs.pybamm.org/en/v25.8.0/source/api/expression\\_tree/unary\\_operator.html](https://docs.pybamm.org/en/v25.8.0/source/api/expression_tree/unary_operator.html).

- [12] PyBaMM docs: discretisation and spatial methods (FVM, FEM backends). [https://docs.pybamm.org/en/v25.4.2/source/api/spatial\\_methods/index.html](https://docs.pybamm.org/en/v25.4.2/source/api/spatial_methods/index.html).
- [13] J. Li *et al.* (2025), *An immersed interface AMR algorithm for Li-ion battery simulations I: Development of a fast P2D solver*, J. Appl. Phys. (early view). <https://pubs.aip.org/aip/jap/article/138/4/045001/3356255/An-immersed-interface-Adaptive-Mesh-Refinement>.
- [14] J. Li *et al.* (2025), *An immersed interface AMR algorithm for Li-ion battery simulations II: Multi-dimensional extension and separator modeling*, J. Appl. Phys. (early view). <https://pubs.aip.org/aip/jap/article/138/4/045002/3356263/An-immersed-interface-Adaptive-Mesh-Refinement>.