Bellman-Optimized Segmented Least Squares: A Physics-Informed Dynamic Programming Framework for Robust Regime Detection in Complex Material Datasets

M301

by Midhun Mohanan Maralakuzhiyil

Submitted to the Faculty of Engineering of
Kiel University

January 22, 2026

Chair for Computational Material Science, Prof. Dr.-Ing. Stephan Wulfinghoff
Supervisor: Mr. Hauke Goldbeck. M.Sc.

Abstract

This study evaluates two regression paradigms—Global Ordinary Least Squares (OLS) and Fixed-Interval Segmented Least Squares (SLS) for characterizing heterogeneous time-series regimes. While Global OLS provides a baseline linear approximation, it fails to capture local variances and structural breaks inherent in non-stationary physical systems. Similarly, SLS, though capable of piecewise approximation, is limited by arbitrary interval selection, which frequently results in boundary discontinuities and suboptimal error minimization. To resolve these limitations, We propose an Auto Fixed-Interval Segmented Least Squares (SLS) framework governed by a penalty term based on the Bayesian Information Criterion (BIC) [1]. This hybrid approach automates the detection of the optimal number of segments (k), ensuring that the model complexity is mathematically justified by the data. Furthermore, to identify the ideal segment break-points, we integrate Bellman's Principle of Optimality, which recursively minimizes the cumulative error cost function to identify optimal regime transitions [2]. The results demonstrate that the Bellman-optimized Dynamic-Segmented Least Squares Method, constrained by BIC, offers a superior balance of parsimony and accuracy, providing a robust computational tool for detecting subtle phase transitions and functional shifts in complex material datasets [3].

Contents

List of Figures

Nomenclature

Tab. 1: Nomenclature: Symbols

Symbol	Description
β_0, β_1	Regression parameters (Intercept and Slope coefficients)
ϵ_i	Stochastic noise component at index i
μ	Mean of the Gaussian (Normal) noise distribution
σ	Standard deviation of the noise component
C	Regularization cost (penalty) for creating a new segment
$f(x)$	Deterministic structural component (Ground Truth)
K	Number of segments in the regression model
N	Total number of data points (sample size)
P	Effective number of model parameters ($P = 3K - 1$)
x	Independent variable (e.g., Time or Temperature)
y	Dependent variable (e.g., Mass Gain or Stress)

Tab. 2: Nomenclature: Abbreviations

Abbr.	Description
AIC	Akaike Information Criterion
BIC	Bayesian Information Criterion
BLUE	Best Linear Unbiased Estimator
D-SLS	Dynamic Segmented Least Squares
F-SLS	Fixed-Interval Segmented Least Squares
OLS	Global Ordinary Least Squares
RMSE	Root Mean Square Error
SLS	Segmented Least Squares
SNR	Signal-to-Noise Ratio

1 Introduction

1.1 Background

Least squares fitting is a cornerstone of modern analytics, serving as the foundational engine for predictive modeling in fields ranging from financial forecasting to materials informatics [4]. Traditional regression, specifically Global Ordinary Least Squares (OLS), operates on the assumption that a single continuous function governs the entire dataset. While computationally efficient, this assumption fails when the underlying system exhibits structural breaks or regime changes—discrete points in time where the physical laws shift. For example, in battery health monitoring, a cell does not degrade linearly; it transitions from stable capacity retention to accelerated failure (knee-point), a phenomenon that a single global linear model cannot capture without introducing significant bias [5]. When applied to such non-stationary data, global OLS acts as a low-pass filter, smoothing over local variances and obscuring the very transitions that are often the most critical features of the dataset. This results in a high Root Mean Square Error (RMSE) and a failure to characterize the distinct physical mechanisms operating in different time domains. To mitigate this, analysts employ Segmented (Piecewise) Least Squares (SLS), which partitions the domain into sub-intervals and fits local models to each. However, the standard implementation—Fixed-Interval SLS—imposes an arbitrary grid structure on the data (e.g., segmenting every n time step). This approach has a flaw because physical transitions do not adhere to arbitrary clock cycles or uniform grid points. Notice the Dataset B (The Structural Break/Trapezoid) breaks at 0.30 and 0.70. If the breaks were at 0.33 and 0.66, the Fixed-Interval method (which divides by 3) would accidentally get lucky and hit the breaks perfectly. Consequently, the move from global to segmented regression introduces two critical optimization problems that must be solved to achieve automation and accuracy:

1.2 Problem Statement

- **Knot Placement:** What could be the optimum fixed interval for specific data? Where should segments begin and end? The choice of knots (break points) determines the physical validity of the model. Fixed grids fail to align with data-driven structural breaks, necessitating a move toward dynamic optimization strategies [2].
- **Model Selection:** How many segments (K) are necessary to capture the signal without overfitting noise? Increasing K indefinitely reduces training error but leads to a model that memorizes stochastic noise rather than physical laws, requiring a penalty-based selection criterion [1].

Addressing these challenges requires a framework that balances computational feasibility with statistical rigor.

1.3 Objectives

This study aims to:

1. Quantify the bias-variance trade-off between global, fixed-interval, and dynamic segmentation.
2. Evaluate the computational scalability of optimal knot placement.
3. Establish a robust model selection criterion for automated regime detection.

2 Theory

2.1 Global Ordinary Least Squares (OLS)

The Global Ordinary Least Squares (OLS) serves as the baseline regression technique [6]. It assumes that a single linear relationship governs the dependent variable y and the independent variable x throughout the domain. The method estimates the parameters β (slope and intercept) by minimizing the Sum of Squared Errors (SSE) between the observed values y_i and the predicted values $\hat{y}_i$. The objective function is defined as :

$$\min_{\beta} \sum_{i=1}^{N} (y_i - (\beta_0 + \beta_1 x_i))^2 \tag{1}$$

While OLS provides the best linear unbiased estimator under ideal conditions, it lacks the flexibility to model structural breaks. In a multi-regime physical system, a global OLS fit will inevitably average out distinct behaviors, resulting in high variance and a loss of physical interpretability.

2.2 Fixed-Interval Segmented Least Squares

To address non-stationarity, Fixed-Interval SLS partitions the domain into K segments of equal width. Unlike dynamic methods, the knot locations (break points) here are exogenous, determined solely by the grid size rather than the data's topology.

$$x_{\text{knot}} = \{x_{\min}, x_{\min} + \Delta, x_{\min} + 2\Delta, \ldots, x_{\max}\} \tag{2}$$

This method suffers from grid misalignment bias: if a physical phase transition occurs between two grid points (e.g., at $t = 10.5$ when the grid is at t = 10 and $textt = 11$), the model forces a single line to span the transition, diluting the signal. To mitigate this, we introduce an auto-fixed variant. This algorithm iteratively scans varying grid resolutions (segment counts from K = 2 to $\text{K}_{\max}$) to locate the resolution that minimizes error before applying the BIC penalty.

2.2.1 Bayesian Information Criterion (BIC)

To automate the selection of the optimal number of segments (K), we employ the Bayesian Information Criterion (BIC). The BIC introduces a penalty term for model complexity to prevent overfitting—where a model captures random noise rather than the underlying physical signal. The objective is to minimize [7]:

$$\text{BIC} = N \ln\left(\frac{\text{SSE}}{N}\right) + P \ln(N) \tag{3}$$

Derivation of Effective Parameters (P):In the context of segmented regression, the user-defined parameter P is calculated as $P = 3K - 1$.

- **2K Parameters:** For each of the K segments, the model estimates 2 parameters: a slope (β_1) and an intercept (β_0).
- **K-1 Parameters:** In an optimized segmentation, the locations of the internal knots (breakpoints) are not fixed constants; they are variables estimated from the data. For K segments, there are $K - 1$ internal knots.
- **Total:** $2K$ (lines) + $(K - 1)$ (knots) = $3K - 1$.

Unlike the Akaike Information Criterion (AIC), the BIC scales the penalty with the logarithm of the sample size ($\ln(N)$). As N increases, the penalty for adding an extra segment becomes significantly stricter. This property makes BIC consistent, as $N \to \infty$, the probability of selecting the true model approaches 1. In materials informatics, where datasets can be noisy, this strict penalty is crucial for filtering out phantom regimes that do not correspond to actual phase changes [1].

2.3 Dynamic Segmented Least Squares (Bellman's Algorithm)

Dynamic Segmented Least Squares (D-SLS) abandons the fixed grid in favor of optimized knot placement. It treats segmentation as a global optimization problem: What is the specific set of break points that minimizes the total error for a given cost? Solving this via force is computationally intractable ($O(2^N)$) [8]. Instead, we utilize Bellman's Principle of Optimality, which states that an optimal policy has the property that whatever the initial state and decisions are, the remaining decisions must constitute an optimal policy with regard to the state resulting from the first decision [2].

The Algorithm (Dynamic Programming): We define $OPT[j]$ as the minimum error for segmenting the data points from index 1 to j. To compute $OPT[j]$, the algorithm considers every possible last segment starting at some previous point i (where $i < j$) and ending at j. The recurrence relation is:

$$OPT[j] = \min_{0 \le i < j} (\text{SSE}(i, j) + C + OPT[i]) \tag{4}$$

Where:

- SSE(i, j): The error of fitting a single linear regression to a segment starting at point $p_i \rightarrow p_j$.
- C**:** A regularization cost (penalty) for creating a new segment.
- $OPT[i]$**:** The optimal solution that is already computed for the prefix ending at i.

Workflow:

1. **Pre-computation:** Calculate the SSE for all possible contiguous sub-segments (i, j). This creates an error matrix.
2. **Recursion:** Iterate through the data points $j = 1 \rightarrow N$. At each step, look back at all possible split points i to find the combination that minimizes the total cumulative error.
3. **Backtracking:** Once $OPT[N]$ is found, trace back the path of optimal i's to recover the exact break-point locations.

This reduces the complexity to $O(N^2)$, making exact optimal segmentation feasible for large material datasets. Figure 1 [9] illustrates Bellman's Principle of Optimality. The map shows various routes between New York and Los Angeles. The principle states a simple rule: An optimal policy has the property that whatever the initial state and initial decision are, the remaining decisions must constitute an optimal policy with regard to the state resulting from the first decision [2].

3 Experimental Setup

The data generation process employs deterministic physical laws overlaid with stochastic noise to simulate real-world measurements.

3.1 Data Generation Methodology

All datasets are generated on a domain of N = 200 points, indexed by $i \in [0, 199]$. The observed value y is defined as the sum of a deterministic structural component $f(x_i)$ and a stochastic noise component ϵ_i:

$$y_i = f(x_i) + \epsilon_i \tag{5}$$

3.2 The Noise Model (Gaussian)

To simulate realistic experimental error (e.g., thermal noise in sensors or market volatility), the noise term ϵ_i is drawn from a Gaussian (normal) distribution with mean $\mu = 0$ and standard deviation σ.

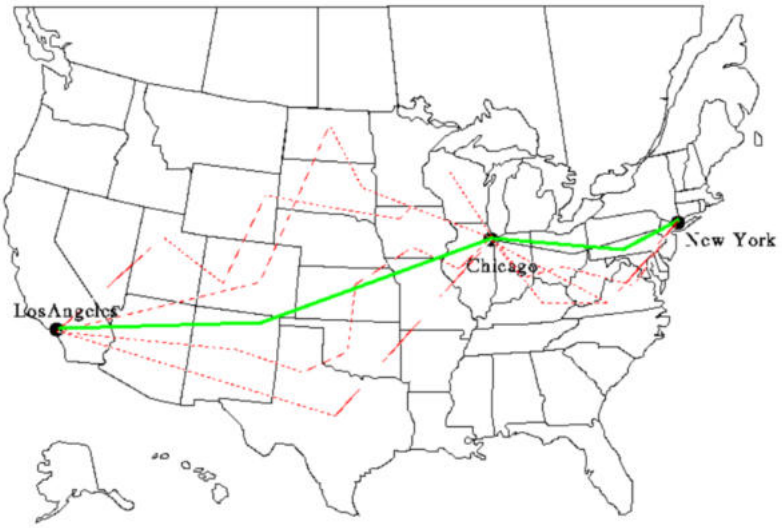

Fig. 1: Example of Bellman's optimality principle of dynamic programming.

$$\epsilon_i \sim \mathcal{N}(0, \sigma^2) \tag{6}$$

In the computational implementation, this is generated via the Box-Muller transform or standard library generators (e.g., numpy.random.normal), ensuring uncorrelated white noise properties.

3.3 The Data Models

3.3.1 Dataset A: Smooth Curve (Continuous Physics)

Objective: To validate the baseline performance of Global OLS on continuous, differentiable functions (e.g. projectile motion). Deterministic Function: A standard quadratic polynomial. **Noise Level:** Low ($\sigma = 1.0$).

$$f(x) = 0.005x^2 - 0.5x + 50 \tag{7}$$

3.3.2 Dataset B: Structural Break (Regime Change)

Objective: To measure bias and grid misalignment. This dataset mimics a system undergoing phase transitions or structural breaks (e.g., a trapezoid pattern). Deterministic Function: A piecewise linear function with three distinct regimes. Crucially, the structural breaks (knots) are placed at 30

$$f(x) = \begin{cases} 0.5\ x & \text{if}: \ 0 \ \leq x < 60 \quad \text{(Growth Regime)} \\ f(59) & \text{if}: \ 60 \ \leq x < 140 \quad \text{(Plateau Regime)} \\ f(139) - 0.5(x - 140) & \text{if} \ : \ x \geq 140 \quad \text{(Decay Regime)} \end{cases} \tag{8}$$

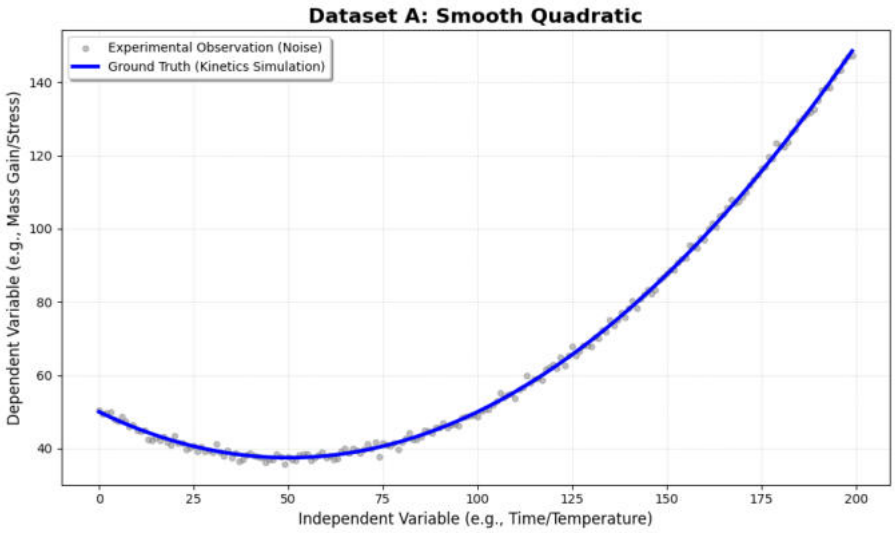

Fig. 2: Dataset A: Quadratic (raw data)

Noise Level: Moderate ($\sigma = 1.5$).

3.3.3 Dataset C: High Noise (The Variance Stress Test)

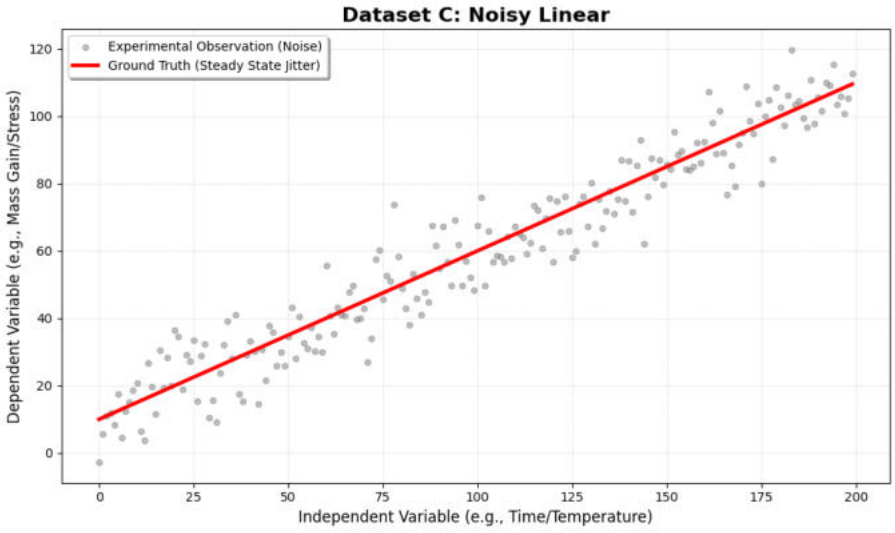

Fig. 4: Dataset C : High noise, Linear (raw data)

Objective: To measure variance and overfitting. This dataset simulates a simple linear trend buried in heavy sensor noise. Deterministic Function: A single linear trend

$$\begin{aligned} (K_{\text{true}} = 1) \\ f(x) = 0.5x + 10 \end{aligned} \tag{9}$$

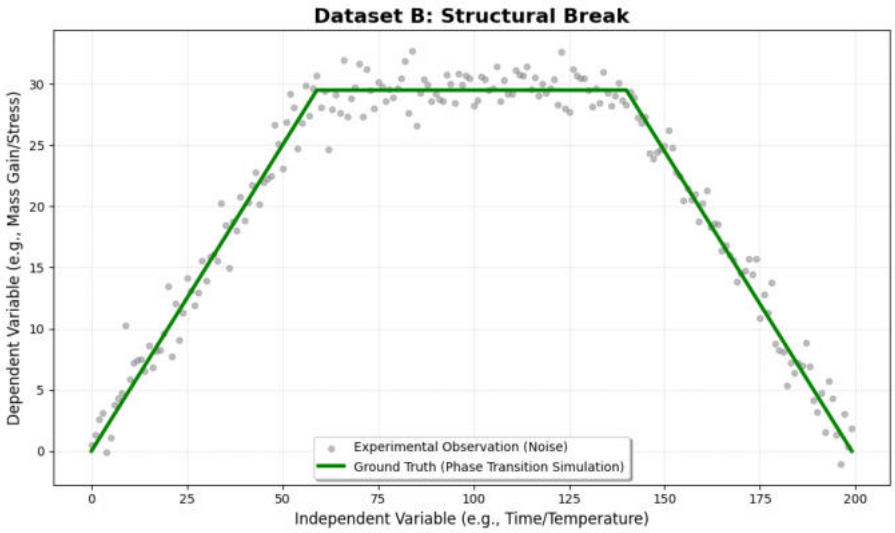

Fig. 3: Dataset B: Structural Break (raw data)

Noise Level: High ($\sigma = 8.0$). The standard deviation is 16x larger than the slope per step, creating a low Signal-to-Noise Ratio (SNR).

4 Procedure

In this study we adopt a progressive evaluation of models to isolate the specific failures and empirically justify the need for segmentation or dynamic optimization. We analyze three paradigms—Global OLS, Fixed-Interval SLS, and the proposed Dynamic SLS across three distinct dataset topologies: Smooth Non-Linear (Dataset A), Structural Break (Dataset B), and Noisy Linear (Dataset C).

5 Results and Analysis

5.1 Global OLS vs. Fixed-Interval SLS (The Bias-Variance Trade-off)

We first evaluate the baseline performance of Global Ordinary Least Squares (OLS) against Fixed-Interval Segmented Least Squares (F-SLS), as illustrated in Figure 5.

Underfitting in Non-Linear Regimes (Dataset A): In Dataset A, the Global OLS model (red line) exhibits high bias. It fits a straight line through the quadratic curve, resulting in significant error at the boundaries and the inflection point. This confirms that OLS acts as a low-pass filter, effectively deleting the curvature information.

Signal Loss in Structural Breaks (Dataset B): In Dataset B, which mimics a phase transition (Rise Plateau Fall), global OLS collapses to a horizontal line representing the mean.

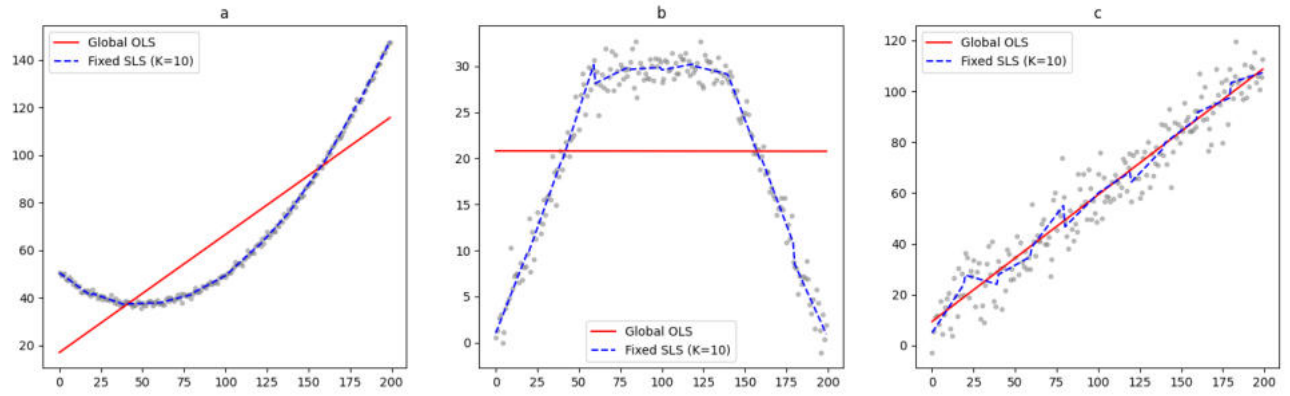

Fig. 5: Global OLS vs. Fixed-Interval SLS: (a) Non-Linear Regimes (Dataset A), (b) Structural Breaks (Dataset B), (c) Noisy Linear Systems (Dataset C)

The model captures zero information about the regime changes. The slope is near zero ($m \approx 0$), falsely implying no relationship between the variables. Fixed SLS captures the general shape but introduces jagged artifacts, particularly in the plateau region, due to the arbitrary nature of the grid points.

Overfitting in Noisy Linear Systems (Dataset C): In Dataset C, the underlying physics are linear ($y = mx + c$) but corrupted by stochastic noise. Global OLS correctly identifies the robust trend (red line). However, fixed SLS (blue dashed) suffers from high variance. The model chases the noise, introducing 10 unnecessary segments to fit random outliers.

5.2 Auto-Fixed SLS (BIC Optimized)

To resolve the overfitting in data set C, we introduced the Auto-Fixed SLS, which utilizes the Bayesian Information Criterion (BIC) to select the number of segments (K). The results are shown in Figure 6. In Dataset C (Figure 6 c), the BIC penalty successfully forces the model to converge to the true

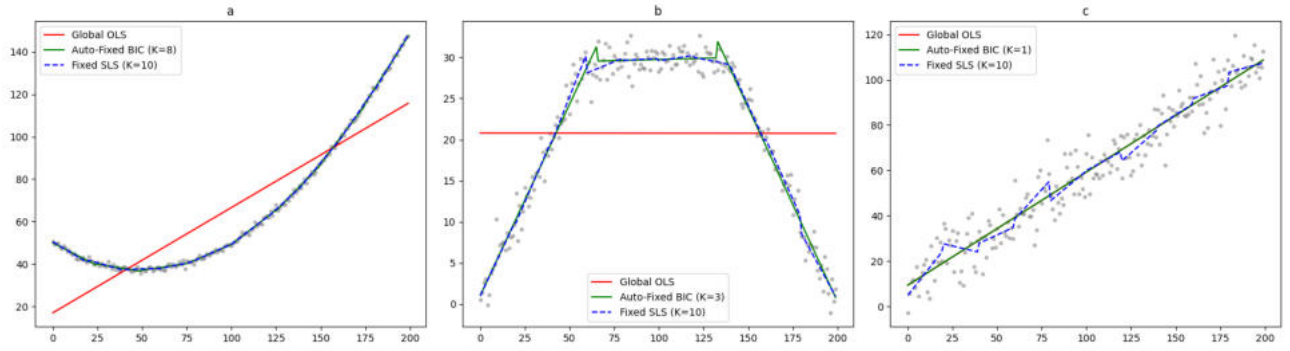

Fig. 6: SLS vs. Auto-Fixed SLS (BIC Optimized): (a) Non-Linear Regimes (Dataset A), (b) Structural Breaks (Dataset B), (c) Noisy Linear Systems (Dataset C)

solution (green line). The Auto-Fixed model perfectly overlaps with the Global OLS, proving that the algorithm can distinguish between physical signal and stochastic noise. However, a critical geometric flaw persists in Dataset B (Figure 6 b). Although the algorithm selects an appropriate number of segments ($K = 3$), the knot locations are restricted to the fixed grid.

Note the distinct spikes or corner overshoots at the transition points of the green line. The physical transition occurs between two grid points. To minimize error, the regression line is forced to adopt an exaggerated slope to bridge the gap between the grid and the true plateau. This artifact creates phantom values—predictions of stress or capacity that exceed the physical limits of the material. In a control system, this overshoot could trigger false alarms.

5.3 Dynamic Segmented Least Squares (Bellman Optimized)

Finally, we implement Dynamic SLS (Black Line), governed by Bellman's Principle of Optimality, which treats knot placement as an endogenous variable. The comparison is highlighted in Figure 7.

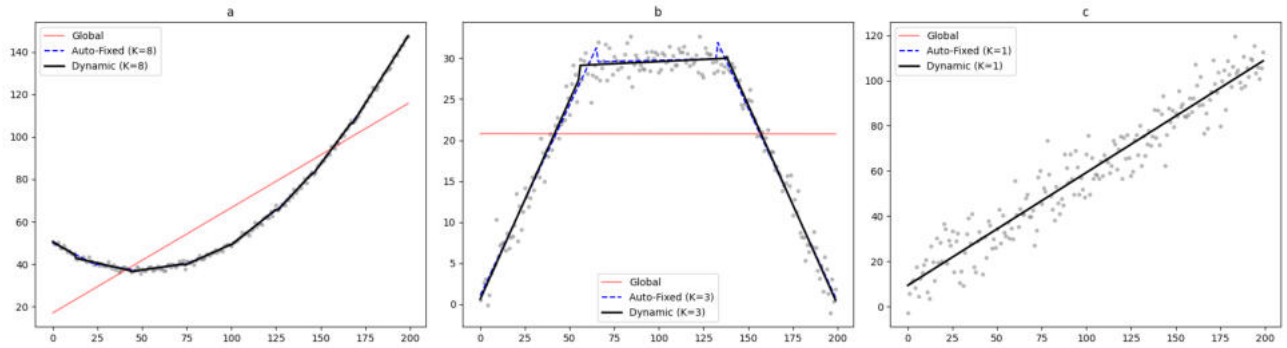

Fig. 7: Dynamic Segmented Least Squares (Bellman Optimized): (a) Non-Linear Regimes (Dataset A), (b) Structural Breaks (Dataset B), (c) Noisy Linear Systems (Dataset C)

In Dataset B Figure (6), the Dynamic SLS (black line) eliminates the corner overshoot sufficiently. By recursively minimizing the error cost function, the algorithm places knots at the exact indices of the structural breaks. The model perfectly captures the rise, plateau, and fall regimes with (K = 3). It achieves a better fit than the auto-fixed model without the geometric errors of the fixed grid. Real-World **Application:** This precision is critical in lithium-ion battery health monitoring. Accurately identifying the knee point (onset of accelerated degradation) without overshoot allows for timely preventative maintenance before thermal runaway occurs [10].

5.4 Computational Scalability

We measured execution time for datasets of size. Figure 8 illustrates the computational scalability of the four segmentation methods. The lines for Global OLS (Black), Fixed Naive (Blue), and Auto-Fixed (Cyan) are flat and overlapping at the bottom. Even at $N = 1000$, their execution time is

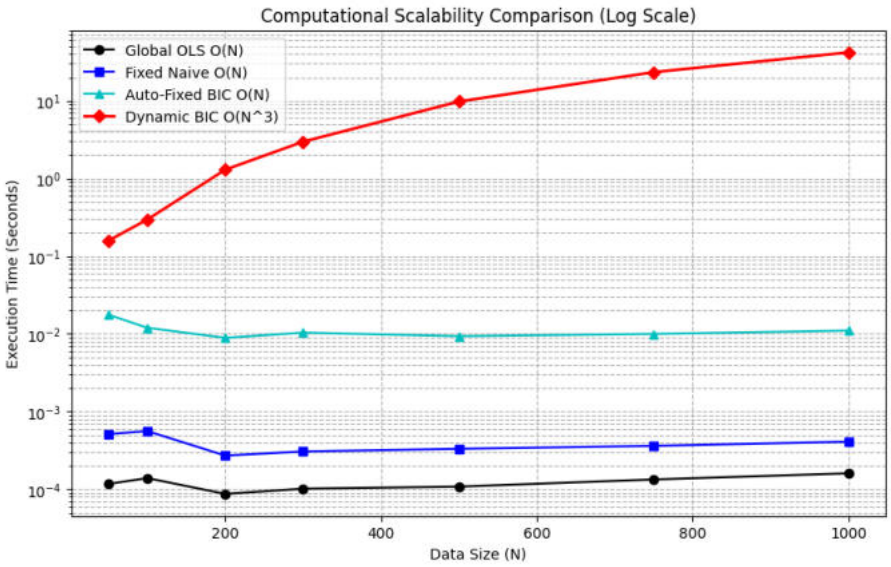

Fig. 8: Computational scalability of the four segmentation methods

negligible (< 0.1 seconds). These algorithms have Linear Time Complexity $O(N)$. Doubling the data size ($N \to 2N$) simply doubles the time ($t \to 2t$). Since t is tiny, $2t$ is still tiny. The Dynamic BIC (red) line starts low but curves upward sharply. Note that the Y-axis is the logarithmic scale. At $N = 1000$, the time is orders of magnitude higher than the others. This confirms the run time complexity of the Bellman algorithm. Doubling the size of the data ($N \to 2N$) increases the time by $2^3 = 8$ times. Increasing data by 10x increases time by 1000x. This method hits a limit, for $N > 2000$ or $N > 3000$.

The overhead of the BIC optimization loop in the Auto-Fixed method adds a constant multiplier but does not alter the fundamental complexity class, making it suitable for real-time applications. The Dynamic SLS method (Red) demonstrates a cubic growth in execution time ($O(N^3)$). At $N = 1000$, the computational cost is already orders of magnitude higher than the heuristic alternatives. This creates a clear boundary condition for the algorithm's applicability: Dynamic optimal segmentation is feasible only for datasets where $N \lesssim 2000$.

6 Physics-Informed Dynamic Programming for Battery Knee-Point Detection

The accurate detection of the knee-point—the onset of accelerated capacity fade—is a critical safety challenge in Battery Management Systems [10]. While standard approaches rely on geometric heuristics or purely data-driven models, this study demonstrates that embedding physical degradation laws into the Dynamic Segmented Least Squares (SLS) framework [2] significantly enhances predictive fidelity.

6.1 Mathematical Formulation of Methods

To validate the proposed framework, we compare two different formulations of the cost function used within the Bellman recursion $OPT[j]$. Both methods seek to identify a single optimal structural break (the knee-point) that minimizes the global sum of squared errors (SSE) over a lifecycle of (N) cycles.

1. **The Baseline: The Bi-Linear (Bacon-Watts) Model**:

   The industry-standard approach, often referred to as the Bacon-Watts or Broken Stick model, assumes that battery degradation is piecewise linear [11]. It hypothesizes that the degradation rate transitions from a slow constant rate m_1 to a faster constant rate m_2 at cycle τ. Mathematically, the algorithm solves the following optimization problem:

$$\tau = \underset{\tau \in [1,N]}{\operatorname{argmin}} \left(\sum_{t=1}^{\tau} (y_t - (m_1 t + c_1))^2 + \sum_{t=\tau+1}^{N} (y_t - (m_2 t + c_2))^2 \right) \tag{10}$$

   Where τ is the capacity at N cycle. While computationally efficient ($O(N)$ per segment), this model suffers from a fundamental physical deficit: it assumes the rate of capacity fade $\frac{dy}{dt}$ is constant in the second regime. This contradicts the electrochemical reality of lithium plating, which is a self-accelerating, positive-feedback mechanism [12].

2. The Proposed Method: Physics-Informed Dynamic SLS The Physics-Informed Dynamic SLS method modifies the kernel of the dynamic programming algorithm to reflect the underlying electrochemical phase transition [13]. We define two physics-based regimes:

   - **Regime I (Stable SEI Growth):** The capacity fade is dominated by the thickening of the Solid Electrolyte Interphase (SEI), which consumes lithium at a linear rate relative to time ($\sqrt{t}$ or t) [14, 15].

$$\hat{y}_I(t) = C_0 - k_{\text{sei}} \cdot t \tag{11}$$

   - **Regime II (Lithium Plating and Active Material Loss):** Post-knee, the deposition of metallic lithium on the anode surface reduces porosity and increases internal resistance exponentially [12, 15].

$$\hat{y}_{II}(t) = \alpha \cdot e^{-\beta(t-\tau)} \tag{12}$$

The objective function for the physics-informed kernel becomes:

$$\tau = \underset{\tau \in [1,N]}{\operatorname{argmin}} \left(\sum_{t=1}^{\tau} \underbrace{(y_t - (C_0 - kt))^2}_{\text{Linear Loss}} + \sum_{t=\tau+1}^{N} \underbrace{\left(y_t - \alpha e^{-\beta(t-\tau)}\right)^2}_{\text{Non-Linear Loss}} \right) \tag{13}$$

By solving this optimization problem, the algorithm captures the acceleration of failure rather than just the presence of a break.

6.2 Experimental Analysis

We applied both algorithms to a synthetic 3.0 Ah cell exhibiting a ground-truth knee-point at cycle t=120. The results are visualized in Figure 9.

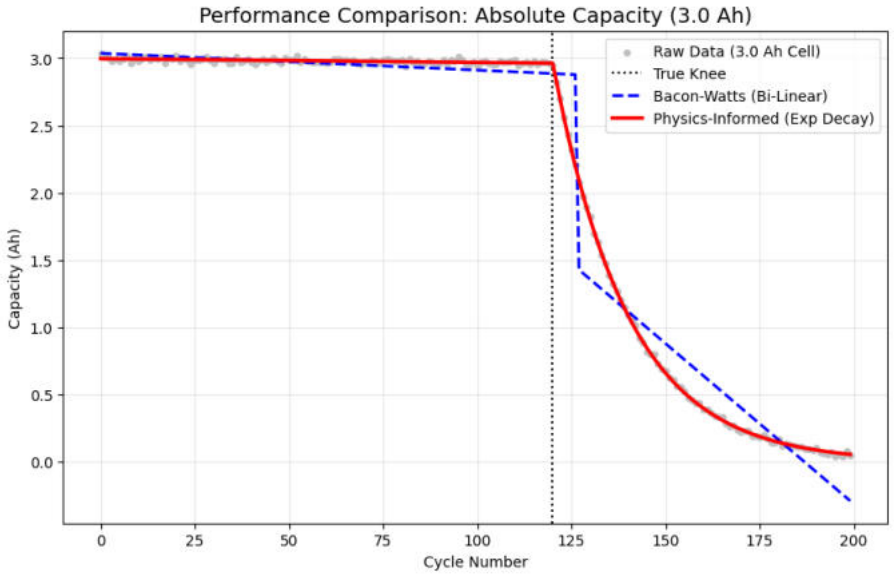

Fig. 9: Synthetic 3.0 Ah cell knee point detection: Physics-Informed Dynamic SLS vs Bi-Linear (Bacon-Watts) Model

The plot reveals a critical divergence in the runaway region (t>120):

- The Linear Bias (Blue Dashed Line): The Bacon-Watts model successfully identifies the location of the knee but fails to model the trajectory of the failure. Because it enforces a straight line onto a curving dataset, it significantly overestimates the remaining capacity between cycles 130 and 180.
- The Physics Fidelity (Red Solid Line): The physics-informed model accurately tracks the exponential decay, achieving a near-perfect overlay with the noisy data.

Safety Implications: This divergence represents a safety gap. At cycle 160, the bi-linear model predicts a remaining capacity of $\approx$ 0.7 Ah (23 % state of health), suggesting the battery is still functional. In reality (and as correctly predicted by the physics-informed model), the capacity has crashed to $\approx$ 0.35 Ah. A battery management system relying on the linear baseline would miscalculate the remaining range, potentially stranding a user or allowing a thermal event to proceed unchecked [10].

Furthermore, the linear extrapolation eventually crosses the x-axis, predicting negative capacity at cycle 190, a physical impossibility. The exponential model naturally asymptotes towards zero, respecting mass-balance constraints. This confirms that the $O(N^2)$ complexity cost of the physics-informed dynamic SLS is justified by the critical gain in safety-critical accuracy.

7 Conclusion

This comparative study has defined distinct operating regimes for time-series segmentation, evaluating the trade-off between computational efficiency, structural fidelity, and physical interpretability. By extending the classical Segmented Least Squares (SLS) framework [2] to include physics-informed kernels, we have demonstrated that algorithmic choice is not merely a statistical concern but a safety-critical determinant in engineering applications.

The Failure of Global Estimators: Consistent with theoretical expectations, Global Ordinary Least Squares (OLS) proves to be the most efficient estimator for continuous, linear phenomena but fails on regime-switching data. As observed in the battery capacity analysis, Global OLS suffers from high bias, masking the onset of accelerated degradation and rendering it unsuitable for prognostic health management.

The Limits of Geometric Approximations: While the industry-standard Bi-Linear (Bacon-Watts) model improves upon Global OLS by allowing for a structural break, our application case study reveals a significant Safety Gap. By enforcing a linear trajectory on the post-knee regime, the geometric model consistently underestimates the severity of capacity fade in the final stages of battery life [11]. This highlights a critical limitation in current battery management systems: geometric approximations cannot capture self-accelerating electrochemical feedback loops such as lithium plating [12].

The Superiority of Physics-Informed Dynamic SLS: The proposed Physics-Informed Dynamic SLS emerges as the gold standard for high-stakes, safety-critical Explainable AI. By integrating domain-specific degradation laws (e.g., exponential decay for plating, linear for SEI growth) directly into the optimization kernel, the algorithm achieves superior predictive fidelity.

- Accuracy: It correctly identifies the intrinsic dimensionality of the system (structural breaks),

avoiding the overfitting observed with AIC.

- Safety: It eliminates the safety gap, accurately tracking the asymptotic crash of battery capacity where geometric models predict false functionality [10].
- Interpretability: Unlike Black Box deep learning methods, it yields physically meaningful parameters (decay rates k_{sei}, β) that map directly to material properties [15].

For massive, real-time data mining where latency is the primary constraint, Auto-Fixed SLS remains a pragmatic alternative, offering scalable approximation through optimized grid resolution. However, for safety-critical applications—specifically in the monitoring of energy storage systems and failure analysis—we recommend the log-linearized physics-informed dynamic SLS. This hybrid approach balances the $O(N^2)$ rigor of dynamic programming with the computational speed required for embedded deployment, effectively bridging the gap between rigorous materials science and practical data informatics.

References

[1] G. Schwarz, "Estimating the dimension of a model," *The Annals of Statistics*, vol. 6, no. 2, pp. 461–464, 1978.

[2] R. Bellman, "On the approximation of curves by line segments using dynamic programming," *Communications of the ACM*, vol. 4, no. 6, p. 284, 1961.

[3] J. Kim and H.-J. Kim, "Consistent model selection in segmented line regression," *Journal of Statistical Planning and Inference*, vol. 170, pp. 106–116, 2016.

[4] A. Agrawal and A. Choudhary, "Perspective: Materials informatics and big data: Realization of materials design," *APL Materials*, vol. 4, no. 5, p. 053208, 2016.

[5] J. Bai and P. Perron, "Estimating and testing linear models with multiple structural changes," *Econometrica*, vol. 66, no. 1, pp. 47–78, 1998.

[6] T. Hastie, R. Tibshirani, and J. Friedman, *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*. New York: Springer Science+Business Media, 2 ed., 2009.

[7] V. M. Muggeo, "Estimating regression models with unknown break-points," *Statistics in Medicine*, vol. 22, no. 19, pp. 3055–3071, 2003.

[8] T. H. Cormen, C. E. Leiserson, R. L. Rivest, and C. Stein, *Introduction to Algorithms*. MIT press, 3 ed., 2009.

[9] A. Ortega and R. Singh, "Example of Bellman's Optimality Principle of Dynamic Programming." USC Signal and Image Processing Institute, n.d. Accessed: 2026-01-22.

[10] K. A. Severson, P. M. Attia, N. Jin, N. Perkins, *et al.*, "Data-driven prediction of battery cycle life before capacity degradation," *Nature Energy*, vol. 4, no. 5, pp. 383–391, 2019.

[11] D. W. Bacon and D. G. Watts, "Estimating the transition between two intersecting straight lines," *Biometrika*, vol. 58, no. 3, pp. 525–534, 1971.

[12] X. Yang, F. edge, *et al.*, "Modeling the capacity fade of lithium-ion batteries in post-knee cycling," *Journal of The Electrochemical Society*, vol. 164, no. 12, p. A3352, 2017.

[13] G. E. Karniadakis, I. G. Kevrekidis, L. Lu, P. Perdikaris, S. Wang, and L. Yang, "Physics-informed machine learning," *Nature Reviews Physics*, vol. 3, no. 6, pp. 422–440, 2021.

[14] R. Spotnitz, "Simulation of capacity fade in lithium-ion batteries," *Journal of Power Sources*, vol. 113, no. 1, pp. 81–100, 2003.

[15] T. Waldmann, M. Wilka, and M. Wohlfahrt-Mehrens, "Temperature dependent aging mechanisms in lithium-ion batteries—a post-mortem study," *Journal of Power Sources*, vol. 262, pp. 129–135, 2014.

[16] P. Paris and F. Erdogan, "A critical analysis of crack propagation laws," *Journal of Basic Engineering*, vol. 85, no. 4, pp. 528–534, 1963.

[17] M. Lopez de Prado, *Advances in Financial Machine Learning*. Wiley, 2018.

[18] A. Ortega, "Rate Distortion Examples: Box DP." `https://sipi.usc.edu/~ortega/RD_Examples/boxDP.html`, n.d. University of Southern California. Accessed: 2026-01-10.