

ORIGINAL RESEARCH

Open access

Algorithmic Settling Time: A Conceptual Framework for When Materials AI Outputs Stabilize

Maria Gonzalez^{1*}, Javier Ruiz¹, Lucia Torres², Elena Ruiz¹

Abstract

In the rapidly expanding domain of Artificial Intelligence for Materials Science, researchers routinely train machine learning models until training loss appears to converge. Yet, this practice overlooks a critical and distinct phenomenon: the point at which model outputs themselves cease to change meaningfully with further iterations or data. Algorithmic settling time is introduced here as the number of training iterations, epochs, data points, or active-learning cycles after which predictions for a given input distribution stabilize within a predefined tolerance, independent of loss minimization. This conceptual framework highlights five key factors—data scarcity, feature dimensionality, model complexity, task difficulty, and optimization dynamics—that modulate settling behavior in materials contexts where datasets are sparse, and property landscapes are high-dimensional. A four-component framework for settling-time analysis is proposed, centered on output monitoring, tolerance specification, settling detection, and confidence assessment, offering a principled alternative to ad-hoc early stopping. By foregrounding settling time as an overlooked parameter, this framework promises to enhance reproducibility, reduce computational waste, and improve the reliability of materials predictions ranging from crystal-property regression to generative molecular design, ultimately elevating the epistemic rigor of Materials AI practice.

Keywords Materials AI, Epistemic uncertainty, Active learning, Algorithmic settling time, Prediction stability, Training convergence

*Correspondence:

Maria Gonzalez
maria.gonzalez@gmail.com

¹ Department of Materials Informatics, University of Granada, Granada, Spain

² Department of AI-Based Materials Systems, University of Seville, Seville, Spain

Introduction

Materials AI models are trained for fixed epochs or until loss converges. But loss convergence does not guarantee prediction stability. Model outputs may continue drifting even when the loss is flat. This paper introduces “algorithmic settling time” — when outputs stabilize — as a distinct concept requiring explicit attention [1, 2]. The field of artificial intelligence for materials science has witnessed explosive growth, with models now routinely deployed for property prediction, inverse design, and discovery of novel compounds [3, 4]. Yet beneath the surface of reported training curves lies an under-examined reality: even after loss functions plateau, predictions on unseen materials

queries can keep shifting subtly, sometimes dramatically, as additional data or iterations are supplied. This drift is not merely noise; it reflects the model’s ongoing negotiation with epistemic uncertainty inherent to high-dimensional, data-scarce materials spaces [5, 6].

Current practice treats convergence and stopping as interchangeable, often defaulting to heuristics borrowed from generic deep-learning pipelines [1]. In materials contexts, however, where datasets are orders of magnitude smaller than those in computer vision or natural-language processing and where the underlying physics imposes complex, non-convex landscapes, such heuristics become unreliable [3, 7]. A model may declare itself “converged” on

a validation loss metric while its predictions for band gaps in perovskites or formation energies in intermetallics continue to evolve. The present conceptual framework, therefore, reframes settling time as a first-class object of analysis, separate from both optimization progress and statistical stability. By doing so, it addresses a gap that has persisted across seminal reviews and applications in the field [3, 4, 8].

Figure 1 presents the hierarchical logic of algorithmic settling time, linking the failure of loss-based stopping rules to the construct's formal definition, its principal determinants, the proposed analysis framework, and its task-dependent implications for Materials AI practice.

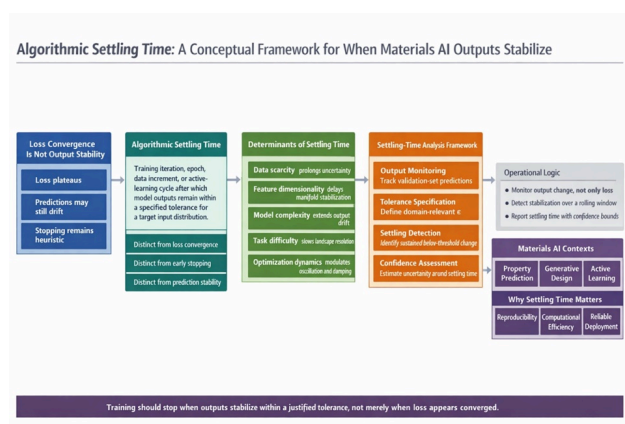


Figure 1. The hierarchical logic of algorithmic settling time, linking the failure of loss-based stopping rules to the construct's formal definition, its principal determinants, the proposed analysis framework, and its task-dependent implications for Materials AI practice.

The motivation is both practical and epistemic. Premature stopping risks publishing unstable predictions that fail under retraining; excessive training wastes scarce computational resources and risks overfitting to noise rather than signal [9, 10]. Moreover, because materials tasks differ radically—ranging from regression on crystalline structures to generative exploration of chemical space—the settling horizon itself becomes task-dependent [5, 11]. This introduction, therefore, sets the stage for a systematic exploration: first by diagnosing the problem of unknown settling, then by defining the concept rigorously, enumerating its governing factors, and finally proposing an actionable framework for its analysis. In doing so, the work aspires to shift Materials AI from a loss-centric paradigm to one that privileges output stability as the ultimate arbiter of readiness [2, 12].

The Problem of Unknown Settling

Why unknown settling matters: (a) premature stopping produces unstable predictions, (b) over-training wastes computational resources, (c) settling time varies across materials tasks, (d) current practices ignore settling entirely. Cite examples from literature where settling was not reported. The absence of explicit attention to settling time creates a cascade of hidden vulnerabilities in materials AI workflows. When models are halted solely based on training-loss plateaus, downstream predictions can remain sensitive to further training, undermining reproducibility across laboratories and hardware platforms [3, 13]. For instance, a graph-network model trained on molecular crystals may appear converged after 200 epochs yet still exhibit drifting formation-energy estimates when additional data points are later incorporated, a phenomenon rarely documented in published studies [5, 14].

This epistemic blind spot is compounded by the extreme data scarcity typical of materials informatics [4, 15]. High-throughput DFT campaigns generate valuable but limited ground-truth labels, and active-learning loops are costly; consequently, practitioners rarely extend training far enough to observe whether outputs have truly settled. Over-training, conversely, consumes GPU-hours that could be redirected toward exploring new chemical spaces, yet without a settling diagnostic, there is no principled way to decide when further computation yields diminishing returns [11, 16]. Task heterogeneity exacerbates the issue: what constitutes adequate settling for a simple regression task on elastic moduli may be entirely insufficient for generative design, where the model continually probes unexplored regions of latent space [7, 17].

Literature surveys reveal that even influential benchmarks and reviews omit any mention of output stabilization metrics [3, 4, 18]. Models are declared successful once validation loss drops below an arbitrary threshold, with no accompanying analysis of whether predictions on held-out test materials would remain invariant under continued training. The result is a literature populated by potentially fragile claims—claims that appear robust in the publication moment but may erode when models are retrained or transferred. Unknown settling therefore functions as a hidden variable that silently modulates the trustworthiness of every materials discovery pipeline [2, 19]. By surfacing this variable, the present framework aims to convert an implicit risk into an explicit, manageable parameter.

Defining Algorithmic Settling Time

Algorithmic settling time—The number of training iterations (epochs, data points, or active learning cycles) after which model outputs for a given input or input distribution stop changing beyond a specified tolerance threshold.

This definition deliberately decouples settling from the optimization objective. Training convergence, as canonically described in foundational texts, concerns the stabilization of a scalar loss function—cross-entropy, mean-squared error, or similar—under gradient descent [1]. Prediction stability, by contrast, quantifies variance across independent runs or bootstrap samples, capturing aleatoric rather than epistemic uncertainty [13, 20]. Algorithmic settling time occupies a third conceptual niche: it tracks the pointwise or distributional invariance of the model’s *outputs themselves* as the training process unfolds. A model may have reached training convergence yet still exhibit unsettled outputs; conversely, outputs may settle while loss continues to exhibit minor fluctuations due to stochastic gradient noise [2, 21].

Table 1 clarifies the theoretical distinctiveness of algorithmic settling time by separating it from convergence, early stopping, prediction stability, reproducibility, uncertainty quantification, and convergence rate.

Table 1. Conceptual Boundaries of Algorithmic Settling Time Relative to Adjacent Machine-Learning Constructs

Construct	Primary question answered	What is being monitored?	Unit of analysis
Training convergence	Has optimization slowed or plateaued?	Scalar loss, gradient norm, or validation error	Optimization trajectory

Early stopping	When should training halt to avoid overfitting?	Validation performance over the patience window	Checkpoint selection rule
Prediction stability	Are predictions similar across seeds, runs, or resamples?	Between-run variance	Replicated model runs
Reproducibility	Can results be recreated across labs, hardware, or retraining?	Agreement of reported findings across replications	Study or pipeline level
Uncertainty quantification	How uncertain is the model about its outputs?	Predictive intervals, ensembles, posterior spread	Prediction or distribution level
Convergence rate	How quickly does the objective decrease?	Speed of loss reduction	Optimization dynamics
Algorithmic settling time	When do outputs stop changing	Successive output changes on a fixed or	Output trajectory across epochs,

beyond a justified tolerance?	task-appropriate probe set	data additions, or AL cycles
-------------------------------	----------------------------	------------------------------

Convergence rate, another related notion, measures how quickly loss decreases but says nothing about the downstream behavior of predictions [1]. In materials AI, where the ultimate deliverable is a predicted property or generated structure rather than an abstract loss value, settling time therefore becomes the more relevant horizon. Conceptually, one can visualize algorithmic settling time as a set of trajectories plotted over iterations: for a fixed validation set of materials queries (e.g., candidate perovskites or alloy compositions), the absolute change in predicted band gap, formation energy, or latent-space coordinate is tracked. These trajectories begin with large excursions and gradually dampen, flattening once the change per iteration falls below a tolerance ϵ (say 0.01 eV for energies). The iteration at which all trajectories remain within ϵ defines the settling time for that tolerance [2].

The definition is intentionally agnostic to architecture—applicable to feed-forward networks, graph neural networks, or diffusion models—and to learning paradigm—supervised, semi-supervised, or active [5, 22]. What matters is the functional invariance of outputs, not the internal weights or gradients. This output-centric lens aligns with the practical needs of materials scientists who must trust a specific numerical prediction or generated candidate, irrespective of whether the loss surface has been fully minimized [3, 23].

Factors Affecting Settling Time

In practice, the temporal stabilization of model outputs in materials AI is governed by a constellation of interacting conditions whose effects are neither additive nor easily separable. A central constraint emerges from the intrinsic scarcity of labeled data, which is a structural feature of the domain rather than a contingent limitation. The reliance on computationally intensive first-principles calculations or resource-heavy experimental synthesis restricts datasets to modest sizes, often several orders of magnitude smaller than those typical in other machine learning contexts [3, 4]. Under these conditions, models are compelled to extrapolate across expansive and sparsely sampled regions of chemical space, and this extrapolative burden sustains epistemic uncertainty deep into training. The

consequence is not merely slower convergence in a loss-based sense, but a prolonged instability in predictions themselves, where successive updates continue to reshape the inferred structure of the problem meaningfully. Even in relatively constrained systems such as oxide perovskites, the persistence of such uncertainty can extend across thousands of epochs before observable quantities—such as lattice parameters—cease to fluctuate in a substantively meaningful way [5, 24].

This dynamic becomes more pronounced when considered alongside the high dimensionality of materials representations, which introduces an additional layer of geometric and statistical complexity. Whether expressed through handcrafted descriptors or learned embeddings, these representations frequently inhabit spaces of considerable dimensional extent, encoding atomic environments, symmetry properties, or graph-based relational structures [11, 25]. Navigating such spaces requires extended exploration before the model identifies stable manifolds that support consistent predictions. Early training phases are therefore dominated by coarse adjustments that reflect incomplete structural understanding, delaying the emergence of output invariance.

The role of model capacity further complicates this trajectory. Highly parameterized architectures possess the expressive power to interpolate training data rapidly, yet this apparent efficiency masks a more subtle dynamic in which predictions on unseen inputs continue to evolve long after conventional convergence criteria are satisfied. This phenomenon, often framed as benign overfitting, acquires a different significance in materials contexts, where the objective extends beyond interpolation toward robust generalization across chemically diverse regimes [1, 26]. Under these conditions, excessive capacity can sustain a prolonged state of adjustment, in which the decision surface remains fluid despite stable loss values, unless explicitly constrained.

The nature of the prediction task introduces an additional axis of variation that shapes settling behavior. Materials properties differ markedly in the complexity of their underlying structure–property relationships, and this variation translates directly into differences in the time required for predictive stabilization. Properties governed by relatively smooth mappings may permit rapid settling within a limited number of training iterations. In contrast, properties characterized by highly non-linear, multi-modal

landscapes—such as topological invariants or anharmonic vibrational free energies—require extensive sampling before their governing patterns become accessible to the model [6, 27]. In more demanding generative settings, particularly those involving metastable phases, complete settling may remain asymptotically unattainable.

Beyond representational and task-specific considerations, the dynamics of optimization exert a decisive influence on how quickly or reliably stabilization occurs. The choice of learning rate, optimization algorithm, and training schedule shapes not only the trajectory of loss reduction but also the temporal behavior of model outputs. Aggressive learning rates can accelerate initial progress while inducing oscillatory dynamics that delay the damping of predictions. In contrast, adaptive methods such as Adam may achieve rapid convergence in objective space yet require extended periods for output stabilization [1, 28]. Techniques such as cyclical learning-rate schedules or second-order optimization further modulate these dynamics, often in ways that remain underreported or insufficiently quantified in materials AI studies. Taken together, these conditions interact multiplicatively, so that combinations such as high dimensionality and data scarcity can extend settling time from modest to prohibitive scales, underscoring the inadequacy of generic heuristics and the need for task-specific analysis [2, 29].

A Framework for Settling Time Analysis

A systematic treatment of settling behavior requires a shift from implicit assumptions toward explicit measurement, grounded in a framework that renders stabilization observable and interpretable. This begins with the continuous monitoring of model outputs rather than exclusive reliance on scalar loss functions. By tracking predictions for a fixed and representative validation cohort and recording the distance between successive outputs, one constructs a temporal signal that captures the evolution of predictive stability itself [2, 13]. Such monitoring reframes convergence as an output-level phenomenon, aligning evaluation with the quantities that ultimately guide scientific decision-making.

This shift necessarily introduces the question of what constitutes meaningful stability, which must be resolved through the specification of a tolerance grounded in domain knowledge. The threshold is not arbitrary but reflects the

resolution at which differences become scientifically consequential, whether in formation energies, band gaps, or lattice parameters [3, 23]. In this sense, the tolerance encodes a decision boundary that links predictive stabilization to downstream experimental or design choices.

Once such a threshold is defined, stabilization can be identified through algorithmic criteria that assess whether output variations remain consistently bounded within the specified tolerance. Rolling-window analyses provide a practical approach, requiring that recent deviations fall below the threshold across all monitored instances, while more advanced techniques such as statistical stationarity tests or Bayesian change-point detection offer robustness against transient fluctuations [2]. These procedures transform settling from a qualitative impression into a quantifiable event.

Yet any estimate of settling time remains contingent on the representativeness of the validation set, necessitating an explicit treatment of uncertainty. By incorporating bootstrap resampling, the framework produces a distribution of settling-time estimates, thereby quantifying epistemic confidence in the inferred stabilization horizon. This probabilistic perspective acknowledges that settling is not a singular point but a range shaped by data variability and sampling limitations.

Conceptually, settling behavior can be visualized as a family of trajectories in which prediction changes decay over training iterations, intersecting with predefined tolerance thresholds at different points depending on model configuration and data regime. The dispersion among these trajectories reveals sensitivity to hyperparameters and highlights the trade-off between computational investment and predictive reliability. In practical terms, this representation provides a principled basis for determining when continued training ceases to yield meaningful gains.

The framework's adaptability across different materials AI contexts further reinforces its utility. In property prediction tasks, it operates on static validation sets drawn from established chemical families; in generative settings, it can be extended to dynamically sampled candidates from evolving latent spaces; and in active learning workflows, it can be reapplied after each data acquisition cycle to reveal how stabilization evolves with dataset growth [5, 7, 11]. Through this integration, settling time is elevated from an implicit byproduct of training to a central diagnostic

variable, strengthening the epistemic rigor of materials AI practice.

Settling Time in Different Contexts

Algorithmic settling time manifests distinctly across the spectrum of materials AI applications, reflecting the underlying objectives and data regimes of each task. In property-prediction contexts that rely on supervised regression, settling time typically emerges after a moderate number of training iterations once the model has internalized stable mappings from structural descriptors to target properties. Graph-network architectures applied to molecules and crystals, for example, often reach output invariance for quantities such as formation energies or band gaps within several hundred epochs because the loss landscape directly constrains predictions toward fixed ground-truth values drawn from DFT or experimental repositories [5, 22]. Yet this horizon is not universal; when training data remain sparse—as is commonplace in solid-state materials informatics—epistemic uncertainty persists longer, requiring extended monitoring to confirm that predicted lattice parameters or elastic moduli no longer drift under additional gradient updates [3, 11]. Practitioners therefore benefit from embedding output-monitoring loops within regression pipelines, ensuring that reported property values represent a truly stabilized model rather than an intermediate snapshot. The implication is practical: regression models can achieve reliable settling relatively early, freeing computational resources for broader chemical-space exploration, provided tolerance thresholds are calibrated to the precision demanded by downstream applications such as screening for photovoltaic candidates.

In generative-design tasks, by contrast, algorithmic settling time behaves more elusively and may never be attained in a complete sense. Models tasked with inverse design—learning continuous latent representations of molecular or crystal structures to propose novel compounds with target functionalities—operate in an open-ended exploratory regime [6, 7]. Here, the objective is not to converge onto fixed labels but to refine a generative distribution that continually samples unseen regions of chemical space. Consequently, generated structures or property distributions can keep evolving long after the variational lower bound or reconstruction loss appears flat, as the model incrementally sharpens its understanding of high-likelihood regions. Settling in this context must therefore be

reframed relative to distributional proxies: for instance, the consistency of sampled band-gap statistics across successive batches of generated candidates or the invariance of decoded structures when the same latent vector is re-sampled. The framework accommodates such fluidity by permitting dynamic validation probes drawn from the evolving manifold itself, allowing detection of when further training ceases to alter the ensemble of proposed materials meaningfully [17, 23]. The practical implication is cautionary: generative workflows risk publishing libraries of candidate structures whose computed properties shift upon retraining, undermining confidence in proposed synthesizable targets. Periodic settling checks at checkpoint intervals become essential to delineate when the generative model has stabilized sufficiently for deployment in high-throughput virtual screening campaigns.

Active-learning contexts introduce yet another regime in which settling time exhibits a resetting character with each oracle cycle. In these iterative pipelines, the model sequentially selects the most informative unlabeled material points, queries an external oracle such as a DFT calculation, augments the training set, and retrains [15, 19]. Each data-acquisition step injects new information that perturbs the previously settled output landscape, effectively restarting the settling clock. Consequently, global settling is never achieved; instead, transient local settling occurs between successive oracle calls, with the required number of retraining epochs lengthening as the active-learning frontier advances into regions of higher complexity or uncertainty. The output-monitoring component of the framework must therefore be reapplied afresh after every cycle, tracking prediction invariance on a fixed held-out set that spans both explored and unexplored chemical families. Early cycles may exhibit rapid settling on coarse structural features, whereas later cycles—confronted with finer distinctions such as defect energetics or metastable polymorphs—demand substantially longer training before outputs stabilize [4, 16]. The implication for practice is efficiency-oriented: active-learning loops can be optimized by terminating retraining once per-cycle settling is detected, thereby minimizing oracle queries and GPU expenditure while still progressively refining the model's epistemic coverage of the materials space. Across these three contexts, algorithmic settling time thus serves as a unifying diagnostic that adapts to the task-specific dynamics of regression, generation, and iterative discovery, ensuring that Materials AI pipelines terminate training on

epistemically defensible grounds rather than arbitrary heuristics.

Table 2 shows that algorithmic settling time is not a single universal horizon but a task-contingent regime whose monitoring strategy, tolerance logic, and reporting standard must vary across regression, generation, active learning, and deployment settings.

Table 2. Task-contingent settling regimes in materials AI: monitoring logic, evidence thresholds, and reporting implications

Materials AI context	Object that should settle	Appropriate probe set	Example tolerance logic
Supervised property prediction	Predicted scalar properties for fixed materials inputs	Static validation cohort spanning chemical families and property range	Domain-calibrated absolute error threshold (for example, energy or band-gap tolerance tied to screening decisions)
Multi-property prediction	Joint output vector across several correlated properties	Balanced validation set covering difficult trade-off regions	Vector norm or property weighted threshold
Generative molecular or crystal design	Distribution of generated candidates, decoded structures, or latent reconstructions.	Repeated probe latents and batch-level distributional samples	Distribution tolerance based on invariant summary statistics or repeated decoding consistency
Active learning for	Predictions between oracle updates within	Fixed held-out set covering,	Per-cycle tolerance relative to

materials discovery	each retraining cycle	explored, and frontier regions	acquisition objective
Low-data rare-materials problems	Predictions in scarcely sampled chemical subspaces	Difficulty-weighted validation subset enriched for rare classes	Stricter uncertainty aware tolerance for high-risk extrapolation
High-dimensional descriptor models	Embedding-informed outputs over complex feature manifolds	Validation cases distributed across descriptor density regions	Threshold scaled to local descriptor complexity or task sensitivity
Deployment-oriented screening pipelines	Ranked shortlist of candidate materials	Decision-focused validation set near ranking cutoffs	Rank-change or top-k stability threshold

Consequences of Premature Stopping

Premature stopping—halting training before algorithmic settling time has been reached—carries four interlocking epistemic and practical consequences that undermine the reliability of materials AI outputs. First, it produces unstable predictions whose values continue to shift upon further training or slight variations in random seeds. A regression model for crystal properties that appears converged on validation loss may still exhibit drifting formation-energy estimates when additional epochs are later supplied, rendering published numbers non-reproducible across independent runs [13, 20]. Second, premature stopping

fosters false confidence: the model is reported as “trained to convergence,” yet its outputs remain sensitive to continued optimization, leading downstream users to treat provisional predictions as definitive when they are in fact transitional. In materials discovery pipelines, this illusion can propagate into experimental prioritization lists that later prove inconsistent [2, 21].

Third, early termination risks missing discoveries by curtailing the model's opportunity to resolve late-emerging patterns that only become visible after the output trajectories have flattened. Complex property landscapes—such as those governing topological or anharmonic behaviors—often reveal subtle correlations only once settling has occurred; stopping beforehand truncates this resolution process and leaves potentially high-value materials candidates undiscovered [6, 24]. Fourth, premature stopping introduces evaluation artifacts whereby test-set performance becomes an accidental function of the chosen stopping iteration rather than an intrinsic model capability. Metrics computed at an unsettled checkpoint may appear favorable simply because the model has not yet overfit or underfit the final stable regime, distorting comparisons across architectures or hyperparameters [1, 28]. Collectively, these consequences highlight that stopping before settling converts an otherwise principled learning process into a source of hidden fragility, emphasizing the necessity of explicit settling diagnostics to safeguard the integrity of materials AI claims.

Relation to Existing Concepts

Algorithmic settling time sits in productive tension with several established ideas in machine learning and materials informatics, clarifying rather than supplanting them. It differs from early stopping, which functions as a pragmatic heuristic to prevent overfitting by monitoring validation loss; early stopping does not interrogate whether outputs themselves have stabilized and may therefore terminate training while predictions remain unsettled [1, 2]. In contrast, settling time offers a principled, output-centric criterion that can coexist with or even override conventional early-stopping rules once tolerance thresholds are satisfied.

The concept also extends beyond traditional convergence diagnostics, which focus almost exclusively on the stabilization of scalar loss functions or gradient norms [1, 21]. While such diagnostics confirm that optimization has slowed, they remain silent on the downstream invariance of

predictions—the very quantity that matters for materials-property tables or generated candidate libraries. Settling time, therefore, supplies the missing link between internal optimization progress and external epistemic reliability.

In relation to reproducibility, algorithmic settling time directly modulates the consistency of published results: models that have not settled yield predictions that vary across retraining runs, eroding the reproducibility that the community increasingly demands [3, 13]. Finally, settling time intersects with uncertainty quantification by serving as a proxy for epistemic uncertainty; unsettled outputs signal that the model's knowledge frontier has not yet been fully resolved, implying higher epistemic variance than a settled counterpart even when aleatoric uncertainty is low [20, 29]. By relating settling time to these neighboring constructs, the framework enriches rather than replaces existing toolkits, positioning output stabilization as the natural culmination of convergence, reproducibility, and uncertainty-aware practice in Materials AI.

Implications for Materials AI Practice

The introduction of algorithmic settling time calls for concrete shifts in how research is conducted, reviewed, and disseminated within the Materials AI community. For authors, three changes are paramount: first, every manuscript must report the measured settling time alongside key predictions, specifying both the tolerance ϵ and the validation cohort used; second, tolerance thresholds must be justified with reference to downstream decision precision (e.g., energy differences relevant to phase stability); third, authors should include a brief settling analysis—perhaps a concise description of output-change trajectories—to demonstrate that results are drawn from a stabilized regime rather than an arbitrary checkpoint [2, 14].

Reviewers, in turn, should adopt a new evaluative lens: manuscripts that omit settling diagnostics or rely solely on loss-convergence claims warrant requests for additional analysis, and results derived from potentially unsettled models should be flagged as provisional until settling evidence is supplied [3, 18]. This heightened scrutiny will elevate the epistemic standard of peer review without imposing undue burden once standardized reporting templates become available.

At the community level, three longer-term initiatives emerge. First, the field should develop public settling-time benchmarks—curated validation sets spanning diverse materials classes against which competing architectures can be compared not only on accuracy but on stabilization speed. Second, reporting standards analogous to those already established for dataset documentation should be extended to include settling-time metadata in supplementary materials or code repositories. Third, systematic studies of settling behavior across sub-domains (oxides, organics, alloys, 2D materials) will reveal domain-specific patterns, informing the design of next-generation architectures and optimizers explicitly tuned for rapid output stabilization [4, 16]. By embedding these practices, materials AI transitions from a loss-centric culture to one that privileges prediction invariance, thereby increasing the trustworthiness of published models and accelerating the reliable translation of computational predictions into experimental realization.

Conclusion

This conceptual framework has articulated algorithmic settling time as the overlooked horizon at which Materials AI outputs cease to evolve meaningfully, distinct from training convergence and prediction stability. By enumerating five governing factors, proposing a four-component analysis protocol, and examining task-specific behaviors together with the risks of premature stopping, the work supplies a principled alternative to ad-hoc heuristics. Loss convergence alone does not guarantee that

predictions are ready for deployment; only when outputs have demonstrably settled within a domain-informed tolerance can researchers claim epistemic closure. The community is therefore urged to adopt settling-time awareness as a standard practice—reporting it, benchmarking it, and designing models with it explicitly in mind—so that the next generation of materials discoveries rests on stabilized, reproducible foundations rather than transient training artifacts.

Acknowledgements

None

Conflict of interest

None

Financial support

None

Ethics statement

None

Received: 17 May 2022 Revised: 07 Jul 2022 Accepted: 20 Aug 2022
Published online: 18 January 2023

Rights and permissions

Open Access This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit <http://creativecommons.org/licenses/by/4.0/>.

References

- Mathew A, Amudha P, Sivakumari S. Deep learning techniques: An overview. In: International Conference on Advanced Machine Learning Technologies and Applications; MLTA 2020. Singapore: Springer; 2020. p. 599-608.
- Xing WW, Zhenjie L, Wang Y. Convergence-aware multi-fidelity Bayesian optimization. In: The Thirteenth International Conference on Learning Representations; 2025.
- Butler KT, Davies DW, Cartwright H, Isayev O, Walsh A. Machine learning for molecular and materials science. *Nature*. 2018;559(7715):547-55.

Schmidt J, Marques MR, Botti S, Marques MA. Recent advances and applications of machine learning in solid-state materials science. *npj Comput Mater.* 2019;5(1):83.

Chen C, Ye W, Zuo Y, Zheng C, Ong SP. Graph networks as a universal machine learning framework for molecules and crystals. *Chem Mater.* 2019;31(9):3564-72.

Zunger A. Inverse design in search of materials with target functionalities. *Nat Rev Chem.* 2018;2(4):0121.

Gómez-Bombarelli R, Wei JN, Duvenaud D, Hernández-Lobato JM, Sánchez-Lengeling B, Sheberla D, et al. Automatic chemical design using a data-driven continuous representation of molecules. *ACS Cent Sci.* 2018;4(2):268-76.

Galan EA, Zhao H, Wang X, Dai Q, Huck WT, Ma S. Intelligent microfluidics: The convergence of machine learning and microfluidics in materials science and biomedicine. *Matter.* 2020;3(6):1893-922.

Choudhary K, Tavazza F. Convergence and machine learning predictions of Monkhorst-Pack k-points and plane-wave cut-off in high-throughput DFT calculations. *Comput Mater Sci.* 2019;166:109-18.
<https://doi.org/10.1016/j.commatsci.2019.05.036>.

Bock FE, Aydin RC, Cyron CJ, Huber N, Kalidindi SR, Klusemann B. A review of the application of machine learning and data mining approaches in continuum materials mechanics. *Front Mater.* 2019;6:110.

Griesemer SD, Xia Y, Wolverton C. Accelerating the prediction of stable materials with machine learning. *Nat Comput Sci.* 2023;3(11):934-45.

Allen AE, Tkatchenko A. Machine learning of material properties: Predictive and interpretable multilinear models. *Sci Adv.* 2022;8(18):eabm7185.

Sutton C, Boley M, Ghiringhelli LM, Rupp M, Vreeken J, Scheffler M. Identifying domains of applicability of machine learning models for materials science. *Nat Commun.* 2020;11(1):4428.

Wang AY, Murdock RJ, Kauwe SK, Oliynyk AO, Gurlo A, Brgoch J, et al. Machine learning for materials scientists: An introductory guide toward best practices. *Chem Mater.* 2020;32(12):4954-65.

Jennings PC, Lysgaard S, Hummelshøj JS, Vegge T, Bligaard T. Genetic algorithms for computational materials discovery accelerated by machine learning. *npj Comput Mater.* 2019;5(1):46.

Suh C, Fare C, Warren JA, Pyzer-Knapp EO. Evolving the materials genome: How machine learning is fueling the next generation of materials discovery. *Annu Rev Mater Res.* 2020;50(1):1-25.

Hasan M, Acar P. Machine learning reinforced microstructure-sensitive prediction of material property closures. *Comput Mater Sci.* 2022;210:110930.

Carmona R, Laurière M. Convergence analysis of machine learning algorithms for the numerical solution of mean field control and games. *Ann Appl Probab.* 2022;32(6):4065-105.

Vasylenko A, Antypov D, Gusev VV, Gaultois MW, Dyer MS, Rosseinsky MJ. Element selection for functional materials discovery by integrated machine learning of elemental contributions to properties. *npj Comput Mater.* 2023;9(1):164.

Bedolla E, Padierna LC, Castaneda-Priego R. Machine learning for condensed matter physics. *J Phys Condens Matter.* 2021;33(5):053001.

Choudhary K, DeCost B, Tavazza F. Machine learning with force-field-inspired descriptors for materials: Fast screening and mapping energy landscape. *Phys Rev Mater.* 2018;2(8):083801.

Choudhary K, DeCost B, Chen C, Jain A, Tavazza F, Cohn R, et al. Recent advances and applications of deep learning methods in materials science. *npj Comput Mater.* 2022;8(1):59.

Kaufmann K, Maryanovsky D, Mellor WM, Zhu C, Rosengarten AS, Harrington TJ, et al. Discovery of high-entropy ceramics via machine learning. *npj Comput Mater.* 2020;6(1):42.

Abdusalamov R, Pandit P, Milow B, Itskov M, Rege A. Machine learning-based structure-property predictions in silica aerogels. *Soft Matter.* 2021;17(31):7350-8.

Esmaeili H, Rizvi R. An accelerated strategy to characterize mechanical properties of polymer composites using the ensemble learning approach. *Comput Mater Sci.* 2023;229:112432.

Kumar D, Chauhan YK, Pandey AS, Srivastava AK, Kumar V, Alsaif F, et al. A novel hybrid MPPT approach for solar PV systems using particle-swarm-optimization-trained machine learning and flying squirrel search optimization. *Sustainability.* 2023;15(6):5575.

Demirtas M, Ilten E, Calgan H. Pareto-based multi-objective optimization for fractional order PI λ speed control of induction motor by using Elman neural network. *Arab J Sci Eng.* 2019;44(3):2165-75.

Lee YS, Jang DW. Optimization of neural network-based self-tuning PID controllers for second order mechanical systems. *Appl Sci.* 2021;11(17):8002.

Al Jlibawi AH, Othman ML, Ishak A, Noor BS, Sajitt AH. Optimization of distribution control system in oil refinery by applying hybrid machine learning techniques. *IEEE Access.* 2021;10:3890-903.