Coupling Language Models with Physics-based Simulation for Synthesis of Inorganic Materials

Edward W. Staley, Tom Arbaugh, Michael Pekala, Alexander New, Christopher D. Stiles, Nam Q. Le
Johns Hopkins Applied Physics Laboratory, Laurel, MD 20723
{*firstname.lastname*}@jhuapl.edu

Gregory Bassen, Wyatt Bunstine, Tyrel McQueen
Johns Hopkins University, Baltimore, MD 21218
mcqueen@jhu.edu

Abstract

Modern generative machine learning (ML) models can propose novel inorganic crystalline materials with targeted properties; however, synthesis planning of these materials remains difficult due to the complexity of the associated physical processes and limited availability of computational tools. We introduce a novel hybrid framework to evaluate Large Language Models (LLMs) in inorganic synthesis planning by combining thermodynamic databases with simplified kinetics models to approximate realistic synthesis conditions. As a case study, we focus on the niobium–oxygen system, which features multiple industrially relevant oxide phases with well-characterized data. In computational simulations, we compare LLM-generated synthesis routes with classical path-planning algorithms, showing that the implicit priors in LLMs can yield more viable strategies. In our evaluation setting, classical search methods serve primarily as a foil rather than a direct competitor. This illustrates the relative complexity of the problem and highlights where the LLM's implicit priors add value.

1 Introduction

The discovery and synthesis of novel inorganic materials underpin advances in energy technologies, electronics, catalysis, and structural applications. Generative ML models have emerged as powerful tools for proposing candidate materials with targeted properties (Wilfong et al. [2025]). Such models can sample vast material design spaces, but the subsequent step of translating a computationally designed material into a feasible synthesis plan presents a significant bottleneck. Traditional approaches for designing synthesis plans, such as exploring energy landscapes, lack reliable methods for determining reaction coordinates and are typically based on a combination of literature sources, intuition, and scientist experience.

We present a preliminary evaluation of LLMs for inorganic synthesis planning using a hybrid framework that augments thermodynamic databases with simplified kinetic models. As a case study, we investigate the binary niobium–oxygen system, a chemically rich space with multiple industrially relevant oxide phases and well-characterized thermodynamics. By comparing LLM-generated synthesis routes with those from a classical path-planning algorithm, we assess the extent to which the embedded scientific priors in LLMs can improve planning quality.

Table 1: The four challenge problems assessed using our physics-based simulator.

ID	Start State		Goal State	
1	300 K	100% Nb	300 K	100% NbO
2	300 K	100% Nb	2200 K	50% NbO, 50% liquid
3	300 K	100% Nb	300 K	50% NbO_2 50% Nb_2O_5
4	300 K	100% Nb_2O_5	300 K	100% NbO

2 Related Work

LLMs, trained on vast corpora spanning scientific literature and general text, offer rich possibilities to augment synthesis planning tools.

Outside of materials applications, there are works that consider the suitability of current generative models for reasoning through problems more generally [Shojaee et al., 2025]. The ability of LLMs to encode prior knowledge raises the possibility of them generating grounded synthesis pathways, and a number of works have considered leveraging generative models to evaluate or determine synthesis pathways. Pu et al. [2024] assessed LLMs in terms of responding to text-based knowledge prompts focusing on synthesis of gold nanoparticles. More direct prediction of synthesis routes, in terms of identification of precursors and conditions, was performed by Prein et al. [2025], including benchmarking against datasets of specific reactions in solid-state synthesis. In this work, we propose a complementary approach using approximate but physically-grounded thermodynamic simulations, rather than sets of reaction-specific data, for quantitative assessment of proposed routes across a broad material system.

3 Methods

3.1 Material System and Problem Statement

Assessment using physics-based simulation in this work relies on the availability of well-validated thermochemical data. We use the binary Nb−O system as a case study; phase diagrams are shown in fig. 2 in the Supplementary Information (SI). This system exhibits several competing solid oxide phases which allow us to pose challenges of varied complexity. Benchmarking synthesis models in well-understood systems, such as Nb−O, will prepare for future extensions to systems without completely assessed phase diagrams, which is ultimately desired for novel materials.

To simulate the kinetics of the system, we adopt a Johnson–Mehl–Avrami (JMA) style approach Mittemeijer [1992] in which phase fractions evolve according to $f(\Delta t) = (f_{\text{eq}} - f_0)\,[1 - \exp(-k\Delta t)]$, where f_0 and f_{eq} are the initial and equilibrium phase fractions, and k is the rate constant given by $k = k_0 \exp\left[-E_a/(RT)\right]$, with activation energy E_a, gas constant R, and temperature T.

Equilibrium phase fractions were computed using the CALPHAD technique with PyCalphad Otis and Liu [2017] and the thermodynamic database of Pérez and Massih [2007], at 10 K intervals from 300 K to 3000 K, with a compositional spacing of 1 at.%. We set $E_a = 95.4$ kJ/mol, based on the experimentally-reported activation energy for diffusion-limited oxidation of Nb Clenny and Rosa [1980], Gulbransen and Andrew [1958], and $k_0 = 100\ \text{s}^{-1}$ to reproduce the reported timescale of the transformation of Nb to Nb_2O_5 at 900 K Clenny and Rosa [1980].

Table 1 shows the four challenge problems that we assessed. Each problem consists of a starting state and a desired goal state, where "state" is defined as a convex combination of realized phases in the material system together with an associated temperature (pressure is assumed fixed at 1 bar). These problems were designed to have a range of difficulty. They vary in features such as requiring temperature variation, single or multiple phases, and/or traversal of "intermediate" phases in the phase diagram. Table 3 in the SI provides further details on the challenges.

In response to each challenge, a respondent must provide a synthesis "recipe," which may consist of any sequence of three options: add precursor, change temperature, or wait. "Add precursor" may choose from the two pure compounds: Nb or O_2. "Change temperature" may set any temperature between 300 K and 3000 K. "Wait" holds the temperature and composition constant while allowing phase fractions to evolve according to thermodynamics and kinetics modeled by the simulator.

Executing the synthesis recipe results in some final state which is then scored by comparing it to the problem goal state. Any solution that does not achieve the final temperature to within 20 K is deemed unresponsive and assigned the maximum error of 1.0. Otherwise, the solution is scored by computing a scaled mean squared error between the goal and realized phase fractions. In the future, other scoring methodologies might be considered, including adding intermediate constraints (such as avoiding a prohibited intermediate phase) or taking more careful account of the geometry of the simplex when evaluating the phase differences.

3.2 Candidate Synthesis Planning Models

We compare two fundamentally different approaches for solving the synthesis planning problem: (1) as a classical path planning problem to help quantify problem difficulty and (2) by leveraging an LLM to explore the inclusion of prior knowledge. There are a variety of classical approaches (e.g. Kocsis and Szepesvári [2006]) that treat the problem as a search space. We elected to utilize A^* search Hart et al. [1968] for its simplicity and because precision to within a small percentage of the goal was acceptable. The search heuristics of A^* are illustrative of techniques that can be guided by mathematical expressions but are hard to shape with abstract priors or domain knowledge. We discretize the state space into 100 bins along each of the phase fraction, element fraction, and temperature dimensions. For a found state in A^* , the cost to reach that state is taken to be the number of steps in the path. Our heuristic for estimating the remaining cost to the goal is the Euclidean distance between two normalized states.

In contrast to classical path planning, the LLM has prior knowledge of materials synthesis and can be given further knowledge through prompting. We also provide the LLM with access to the simulator itself, which it can invoke as a "tool" in generating a solution. This is analogous to domains like mathematics, where LLMs are paired with tools (e.g., theorem provers) to derive new results.

Note that while an LLM-guided synthesis search can incorporate prior knowledge, it is also more computationally expensive, and many-step LLM reasoning is an open area of research Aghajohari et al. [2025], Fei et al. [2023]. We instructed the LLM to continue attempting a problem until it reached a conclusion (fig. 4) and did not otherwise limit its generation length. We let A^* run until it found a solution or exhausted system resources while building the search graph (128GB RAM).

Our LLM is implemented as a ReAct-based agent Yao et al. [2023] using the Declarative Self-improving Python (DSPy) framework Khattab et al. [2024, 2022], which we attach to OpenAI's GPT4o model OpenAI [2024]. GPT4o was selected as a state-of-the-art non-reasoning model for which we could introduce the reasoning process ourselves through prompting. Of course, future work would compare multiple models, including reasoning vs non-reasoning varieties. Given that the performance of in-context learning depends upon the prompt Min et al. [2022], we also consider strategies for designing this prompt in more systematic ways. For this preliminary study, we leveraged DSPy's prompt tuning capabilities; in particular, its `LabeledFewShot` strategy which, given a training data set, randomly selects $k = 4$ new examples to add to the prompt.[1]

4 Results and Discussion

Our primary results[2] are summarized in table 2 and fig. 1; the former shows a comparison of A^* and the LLM agent on the challenge problems while the latter shows the full distribution of scores and provides a comparison of LLM performance pre- and post-prompt tuning. In the case of table 2, "success" is defined as the final state being within some approximate ϵ-ball of the goal state; of course, other criteria might be considered in future work. Since the search space is deterministic, we only run A^* once, resulting in either 100% success or failure. To account for the stochasticity of LLM generation, we run the LLM ten times on each problem. In general, we find that the LLM and A^* both exhibit some variability in performance across the four challenges. The relative effectiveness of the LLM suggests that its scientific priors indeed carry over to our simulation environment. Results from the preliminary prompt tuning (fig. 1) suggests that more careful optimization is important to consider in future work.

---

[1] Note DSPy includes more sophisticated optimization strategies that we propose to explore moving forward. We used random sampling as an initial proof-of-concept to determine whether prompt tuning seemed promising.

[2] More details, including the best solutions found by the LLM for each challenge, are provided in the SI.

Table 2: Comparison of A* and LLM (no prompt tuning) results. "States visited" is the rough magnitude of state transitions simulated during the application of the method.

	Challenge 1		Challenge 2		Challenge 3		Challenge 4	
Method	Success Rate	States Visited	Success Rate	States Visited	Success Rate	States Visited	Success Rate	States Visited
A*	100%	O(10^5)	0%	$\geq$ O(10^7)	0%	$\geq$ O(10^7)	100%	O(10^7)
LLM	20%	O(10)	0%	O(10)	0%	O(10)	40%	O(10)

Challenges 1 and 4 were both solved by A* and the LLM. However, A* explored orders of magnitude more states than the LLM. Challenges 2 and 3 include a target with multiple phases, and was more difficult for both approaches. On challenge 2, the LLM obtained the correct phases, but not the correct fractions. On challenge 3, the LLM was only able to find solutions after prompt tuning. A* ran out of system resources before finding a solution to challenges 2 and 3. A key challenge in applying A* is that, while distance to the goal is primarily measured in phase fractions, the available simulation options only include elemental composition, temperature, and time. The relationship between these spaces is provided by the simulation itself and difficult to estimate a priori, making classic search strategies (even those leveraging heuristics) impractical. In contrast, the LLM is able to attempt sensible approaches for this domain by leveraging training data and prompt information, rather than an explicit notion of distance. Examples of LLM outputs are provided in 5 and 6.

5 Conclusions

This work provides three main results. First, we have demonstrated a strategy for using simple physics-based simulation as a virtual "testbed" to assess LLMs' ability to propose material synthesis plans. Second, we identified synthesis problems where the LLM's implicit priors appear to facilitate finding solutions more efficiently than naive search methods. Third, there are still very simple synthesis problems which both A* and the example LLM fail to solve; however, this framework suggests straightforward ways to ground LLM predictions by invoking physics-based tools. Both our simulation and our prompt optimization strategy are nascent and there are many opportunities to make this approach even more applicable to practical synthesis pathway planning.

Acknowledgments and Disclosure of Funding

EWS, TA, MP, CDS, and NQL gratefully acknowledge internal financial support from the Johns Hopkins Applied Physics Laboratory's Independent Research & Development (IR&D) Program for funding portions of this work. We would like to thank the reviewers for their valuable feedback.

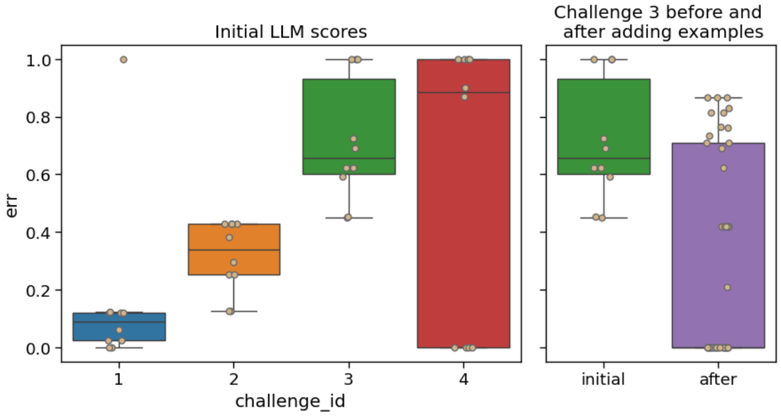


Figure 1: LLM performance on the 4 challenge problems (CP). (left) performance for the original, manually curated prompt; (right) performance on CP 3 before (green) and after (purple) prompt tuning. We ran the prompt tuning procedure 5 times (with different seeds), hence there are more purple data points than green.

References

Milad Aghajohari, Kamran Chitsaz, Amirhossein Kazemnejad, Sarath Chandar, Alessandro Sordoni, Aaron Courville, and Siva Reddy. The Markovian Thinker, 2025. URL `https://arxiv.org/abs/2510.06557`.

James T Clenny and Casimir J Rosa. Oxidation kinetics of niobium in the temperature range of 873 to 1083 K. *Metallurgical Transactions A*, 11(8):1385–1389, 1980.

Weizhi Fei, Xueyan Niu, Pingyi Zhou, Lu Hou, Bo Bai, Lei Deng, and Wei Han. Extending context window of large language models via semantic compression, 2023. URL `https://arxiv.org/abs/2312.09571`.

Earl A Gulbransen and Kenneth F Andrew. Oxidation of niobium between 375 c and 700 c. *Journal of The Electrochemical Society*, 105(1):4, 1958.

Peter E. Hart, Nils J. Nilsson, and Bertram Raphael. A formal basis for the heuristic determination of minimum cost paths. *IEEE Transactions on Systems Science and Cybernetics*, 4(2):100–107, 1968. doi: 10.1109/TSSC.1968.300136.

Omar Khattab, Keshav Santhanam, Xiang Lisa Li, David Hall, Percy Liang, Christopher Potts, and Matei Zaharia. Demonstrate-search-predict: Composing retrieval and language models for knowledge-intensive NLP. *arXiv preprint arXiv:2212.14024*, 2022.

Omar Khattab, Arnav Singhvi, Paridhi Maheshwari, Zhiyuan Zhang, Keshav Santhanam, Sri Vardhamanan, Saiful Haq, Ashutosh Sharma, Thomas T. Joshi, Hanna Moazam, Heather Miller, Matei Zaharia, and Christopher Potts. DSPy: Compiling declarative language model calls into self-improving pipelines. In *The Twelfth International Conference on Learning Representations*, 2024.

Levente Kocsis and Csaba Szepesvári. Bandit based Monte-Carlo planning. In Johannes Fürnkranz, Tobias Scheffer, and Myra Spiliopoulou, editors, *Machine Learning: ECML 2006*, pages 282–293, Berlin, Heidelberg, 2006. Springer Berlin Heidelberg. ISBN 978-3-540-46056-5.

Sewon Min, Xinxi Lyu, Ari Holtzman, Mikel Artetxe, Mike Lewis, Hannaneh Hajishirzi, and Luke Zettlemoyer. Rethinking the role of demonstrations: What makes in-context learning work? *arXiv preprint arXiv:2202.12837*, 2022.

EJ Mittemeijer. Analysis of the kinetics of phase transformations. *Journal of Materials Science*, 27 (15):3977–3987, 1992.

H Okamoto. Nb−O (niobium oxygen). *Binary Alloy Phase Diagrams, II Ed, Ed. TB Massalski*, 3: 2748–2749, 1990.

OpenAI. ChatGPT (gpt-4o), 2024. URL `https://chat.openai.com/`. Large language model, version GPT-4o. Accessed: 2025-08-18.

Richard Otis and Zi-Kui Liu. pycalphad: CALPHAD-based computational thermodynamics in Python. *Journal of Open Research Software*, Jan 2017. doi: 10.5334/jors.140.

R Jerlerud Pérez and Ali R Massih. Thermodynamic evaluation of the Nb–O–Zr system. *Journal of nuclear materials*, 360(3):242–254, 2007.

Thorben Prein, Elton Pan, Janik Jehkul, Steffen Weinmann, Elsa A Olivetti, and Jennifer LM Rupp. Language models enable data-augmented synthesis planning for inorganic materials. *arXiv preprint arXiv:2506.12557*, 2025.

Yingming Pu, Liping Huang, Tao Lin, and Hongyu Chen. Leveraging large language models for explaining material synthesis mechanisms: The foundation of materials discovery. In *AI for Accelerated Materials Design-NeurIPS 2024*, 2024.

Parshin Shojaee, Iman Mirzadeh, Keivan Alizadeh, Maxwell Horton, Samy Bengio, and Mehrdad Farajtabar. The illusion of thinking: Understanding the strengths and limitations of reasoning models via the lens of problem complexity. *arXiv preprint arXiv:2506.06941*, 2025.

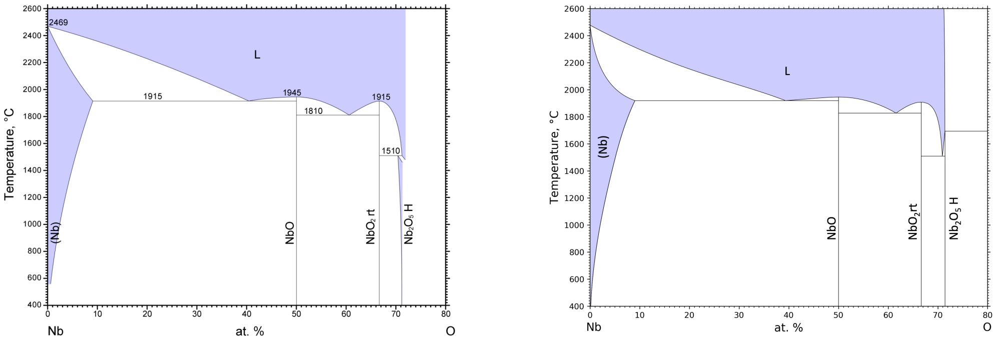
Brandon Wilfong, Alexander New, Gregory Bassen, Wyatt Bustine, Michael J Pekala, Nam Q Le, Karun K Rao, Elizabeth A Pogue, Eddie Gienger, Michał J Winiarski, et al. Ternary materials discovery using human-in-the-loop generative machine learning. *RSC advances*, 15(24):19126–19131, 2025.

Shunyu Yao, Jeffrey Zhao, Dian Yu, Nan Du, Izhak Shafran, Karthik Narasimhan, and Yuan Cao. React: Synergizing reasoning and acting in language models. In *International Conference on Learning Representations (ICLR)*, 2023.

A Supplementary Information

A.1 Phase diagrams

For the binary Nb−O system used in this work as a case study, two phase diagrams are shown in fig. 2. The experimental phase diagram is reproduced from Okamoto [1990], and we computed the phase diagram with the CALPHAD method as implemented in PyCalphad using the database of Pérez and Massih [2007]. The same database was used with PyCalphad to compute the phase fractions that would occur at thermodynamic equilibrium, which serve as input to the JMA-style approximation to the kinetics described in Section 3.1.



(a) Experimental.

(b) Computational (CALPHAD).

Figure 2: Phase diagrams for Nb−O. Observe the good agreement between the experimental (a) and computational estimates (b) of the phases. "L" denotes the liquid phase, "rt" denotes the rutile NbO_2 phase, "H" denotes the monoclinic Nb_2O_5 phase, and "at.%" denotes atomic percent.

A.2 Details of challenge problems

The four synthesis challenges used to assess A* and LLMs in terms of their ability to generate synthesis plans are enumerated in Table 3. Additionally, the search space for A* normalized all temperatures by dividing by 3000K such that all dimensions of the search space fell within [0-1]. The phase fractions are naturally in this range without modification.

A.3 Additional results

Figure 3 provides a visualization of the states explored by one invocation of the A* algorithm.

Table 4 provides examples of successful synthesis pathways identified by the LLM for each challenge. Through pretraining knowledge, prompted hints, and strategies uncovered by the ReAct technique, the LLM is able to attempt synthesis pathways that are potentially much closer to viable than those prioritized by A*, even though they may appear quite roundabout when plotted in some choice of two dimensions. For example, successful synthesis in the first challenge involves temporarily raising temperature and later lowering it, which is hard to influence via an A* heuristic but straightforward to convey to an LLM, or even for an LLM with pretraining knowledge to consider without user interventions.

Table 3: Challenge problems related to the Nb-O oxide system posed to algorithms in this work.

ID	Start State	Goal State	Example Solution	Comments
1	$T_0 = 300$ K 100% Nb	$T_f = 300$ K 100% NbO	`add`(50 at.% O) `settemp`(1300 K) `wait`(60 min) `settemp`(300 K)	Direct composition change toward single oxide phase that is “adjacent” in composition space.
2	$T_0 = 300$ K 100% Nb	$T_f = 2200$ K 50% NbO, 50% liquid	`add`(46 at.% O) `settemp`(2200 K) `wait`(60 min)	Requires multiple target phases and temperature change.
3	$T_0 = 300$ K 100% Nb	$T_f = 300$ K 50% NbO_2, 50% Nb_2O_5	`add`(58 at.% O) `settemp`(1300 K) `wait`(60 min) `settemp`(300 K)	Requires multiple target phases that are “non-adjacent” to starting phase.
4	$T_0 = 300$ K 100% Nb_2O_5	$T_f = 300$ K 100% NbO	`add`(22 at.% Nb) `settemp`(1300 K) `wait`(60 min) `settemp`(300 K)	Single oxide phase that is “non-adjacent.”

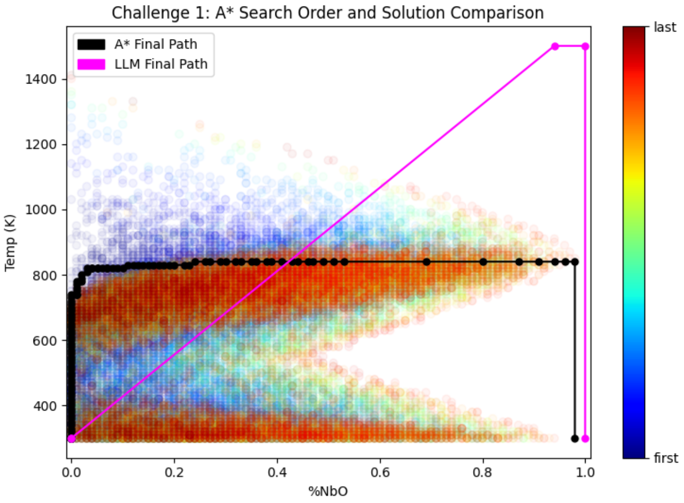


Figure 3: Progress of A* search on the first challenge, projected onto the temperature dimension and the percent NbO dimension (the goal state is 100 % NbO at 300 K). Marker color denotes order in which states were explored with blue being earliest and red the latest. The final synthesis path of A* is in black, while the best plan found by the LLM for this challenge is in magenta.

Table 4: Best synthesis pathways found by the LLM for each challenge.

Id	Best LLM Solution	Comments
1	`add`(50 at.% O), `settemp`(1500 K), `wait`(30 min), `settemp`(300 K), `wait`(10 min)	Exactly solves challenge 1.
2	`add`(47 at.% O), `settemp`(1000 K), `wait`(70 min), `settemp`(1500 K), `wait`(70 min), `settemp`(2200 K), `wait`(10 min)	Achieves correct temperature but 61% NbO, 39% liquid instead of the goal: 49% NbO, 51% liquid.
3	`add`(65 at.% O), `settemp`(1700 K), `wait`(120 min), `settemp`(1900 K), `wait`(120 min), `settemp`(300 K), `wait`(3 min)	Achieves correct temperature but far from correct material phases.
4	`add`(50 at.% Nb), `settemp`(2000 K), `wait`(240 min), `settemp`(300 K), `wait`(10 min)	Exactly solves challenge 4.

A.4 Additional LLM details

The LLM's prompt consists of a context string, which provides the bulk of the detail, and the challenge problem itself, which defines the start and goal states. The context string is shown in fig. 4. For all of our runs we used a LLM temperature setting of 0.1.

LLM Prompt

You should consider raising temperatures or holding them over long periods (hours), as well as fast temperature changes (seconds). Even if you get close, try a few more options to see if you can get closer.

Action sequences are specified as a list of dictionaries, with one dictionary for each action. There are three types of actions: temperature change actions, composition change actions, and wait actions.

* Temperature change actions increase or decrease temperature over the provided amount of time.
* Composition change actions introduce raw precursors (Nb or O) in order to realize the specific percentages. Note the arguments to this action are the final percentages to realize, NOT the amount to be added.
* Wait actions specify a duration of time where there is no change to composition or temperature. This may be needed (with a large duration) if you want to approach thermal equilibrium.

Valid keys for action dictionary are ”Nb”, ”O”, ”Temperature”, and ”Time”, representing the fraction of niobium in the overall composition, the fraction of oxygen in the overall composition, the temperature, and the time taken from the last step.

* Composition entries may be omitted if they can be inferred from the implied sum-to-1 constraint.
* Temperature actions should not specify any raw material changes; these must be changed separately.
* Use your knowledge of materials science and the underlying phase diagram when devising the plan. A successful plan may not necessarily be a straightforward trajectory.
* All times are in units of seconds and all temperatures are in units of Kelvin.
* Feel free to try as many examples as you like in the process of searching for a solution. Don't give up until you can't think of any other ways to try to solve the problem.

Example: (1) raise temperature to 1500K over 10 seconds, (2) wait for 60 minutes, then (3) add precursors so that the fraction of oxygen is 80% of the mixture:

[{”Temperature”:1500, ”Time”:10.0}, {”Wait”: 3600}, {”O”:0.8, ”Nb”:0.2}]

Example: (1) add raw precursors achieve 70% Nb then (2) heat to 2100K over 5 seconds, and finally (3) cool back down to 300K over one minute:

[{”Nb”:0.7}, {”Temperature”:2100, ”Time”:5.0}, {”Temperature”:300, ”Time”:60.0}]

Your final answer must be a synthesis plan in the above form. You must provide a synthesis plan, even if you don't think it will solve the problem correctly.

Figure 4: LLM context string.

Example LLM Output on Challenge 4 (Successful)

Generate a synthesis plan for the following problem:
Start state: TEMPERATURE: 300, COMPOSITION: [NB: 0.29, O: 0.71], PHASE FRACTIONS: [NB2O5: 1.0].
Goal state: TEMPERATURE: 300, PHASE FRACTIONS: [NBO: 1.0].

... LLM Context inserted here ...

Thought 1: To transform Nb2O5 into NBO, we need to consider the phase transformations that occur with temperature changes. Nb2O5 can be reduced to NBO by controlling the temperature and possibly the composition. Initially, I will try a synthesis plan that involves heating the material to a high temperature to facilitate the reduction and then cooling it back to the desired temperature. I will start by heating to 1500K over 10 seconds, wait for 3600 seconds to allow the reaction to occur, and then cool back to 300K over 60 seconds.
Tool 1: *simulate([Temperature: 1500, Time: 10.0], [Wait: 3600], [Temperature: 300, Time: 60.0])*
Observation 1: *final_temp=300; final_phases=[NB2O5: 0.9099999999999995, BCC_A2: 0.0, NBO: 0.0, HCP_A3: 0.0, GAS: 0.0, ION: 0.0, LIQUID: 0.0, NBO2: 0.09000000000000052, FCC_A1: 0.0]*
Thought 2: The initial attempt did not produce the desired NBO phase... *(truncated)*.

... Truncated for Readability ...

Thought 4: The third attempt also failed to produce the desired NBO phase, with the majority of the material remaining as Nb2O5 and a small amount converting to NBO2. This suggests that the conditions are still not optimal for the conversion to NBO. I will try a different strategy by adjusting the composition to increase the niobium content, which might facilitate the formation of NBO. I will add raw precursors to achieve 50% Nb, then heat to 1700K over 10 seconds, wait for 7200 seconds, and finally cool back to 300K over 60 seconds.
Tool 4: *simulate([Nb: 0.5, O: 0.5], [Temperature: 1700, Time: 10.0], [Wait: 7200], [Temperature: 300, Time: 60.0])*
Observation 4: *final_temp=300; final_phases=NB2O5: 0.0, BCC_A2: 2.2024444095574645e-06, NBO: 0.9999917981555904, HCP_A3: 0.0, GAS: 0.0, ION: 0.0, LIQUID: 0.0, NBO2: 5.9994000001725e-06, FCC_A1: 0.0*
Thought 5: The final attempt successfully produced the desired NBO phase with a phase fraction of approximately 0.99999. This indicates that the strategy of adjusting the composition to increase the niobium content, followed by heating to 1700K and holding for 7200 seconds, was effective in achieving the goal state. The synthesis plan is now complete.

Response: The synthesis plan to convert Nb2O5 to NBO involved several attempts to find the optimal conditions for the phase transformation. Initially, attempts to simply heat the material to high temperatures and wait did not yield the desired NBO phase. This indicated that the temperature and duration were not sufficient or appropriate for the conversion. Adjusting the composition to increase the niobium content proved to be a successful strategy. By achieving a 50% Nb composition, heating to 1700K, and holding for 7200 seconds, the desired NBO phase was successfully formed with a phase fraction of approximately 0.99999. This approach balanced the composition and thermal conditions to facilitate the desired phase transformation.
Answer: [Nb: 0.5, O: 0.5], [Temperature: 1700, Time: 10.0], [Wait: 7200], [Temperature: 300, Time: 60.0]

Figure 5: Successful attempt on challenge 4 by the LLM using the ReAct framework. The output has been manually formatted for readability and to fit the page. The plan produced by the LLM achieves the correct phase and the correct final temperature.

Example LLM Output on Challenge 4 (Unsuccessful)

Generate a synthesis plan for the following problem:
Start state: TEMPERATURE: 300, COMPOSITION: [NB: 0.29, O: 0.71], PHASE FRACTIONS: [NB2O5: 1.0].
Goal state: TEMPERATURE: 300, PHASE FRACTIONS: [NBO: 1.0].

... LLM Context inserted here ...

Thought 1: To transform Nb2O5 into NBO, we need to consider the phase transformations and the necessary conditions for these transformations. Typically, reducing Nb2O5 to NBO involves a reduction process, which might require a high temperature to facilitate the reaction. We should start by increasing the temperature significantly to see if this promotes the desired phase change. After heating, we can hold the temperature to allow the system to reach equilibrium. Let's start by heating the system to a high temperature and then holding it for a while.
Tool 1: *simulate([Temperature: 1800, Time: 60], [Wait: 7200])*
Observation 1: *final_temp=1800; final_phases=[NB2O5: 0.07840216105964366, GAS: 0.0, NBO: 0.0, LIQUID: 0.0, NBO2: 2.0829399531378196e-05, BCC_A2: 0.0, FCC_A1: 0.0, ION: 0.9215770095408249, HCP_A3: 0.0]*
Thought 2: The initial attempt to heat the system to 1800K and hold it for 2 hours did not result in the formation of NBO.... *(truncated).*

... Truncated for Readability ...

Thought 4: The stepwise temperature increase with intermediate holds still resulted in the complete conversion to the ION phase, with no formation of NBO. This suggests that the current approach of only adjusting temperature is not effective in achieving the desired phase transformation. We may need to consider altering the composition to favor the formation of NBO. Let's try adding Nb to increase its proportion in the composition and see if this helps in forming NBO.
Tool 4: *simulate([Nb: 0.5], [Temperature: 1500, Time: 60], [Wait: 3600], [Temperature: 1800, Time: 60], [Wait: 7200], [Temperature: 2100, Time: 60], [Wait: 14400])*
Observation 4: *final_temp=2100; final_phases=[NB2O5: 0.0, GAS: 0.0, NBO: 0.9999889654813701, LIQUID: 0.0, NBO2: 0.0, BCC_A2: 2.3775424105629224e-06, FCC_A1: 0.0, ION: 8.656976219446085e-06, HCP_A3: 0.0]*
Thought 5: The addition of Nb to increase its proportion in the composition, followed by a stepwise temperature increase, successfully resulted in the formation of NBO with a phase fraction of approximately 0.99999. This indicates that the synthesis plan is effective in achieving the desired phase transformation from Nb2O5 to NBO. The final synthesis plan involves adjusting the composition and carefully controlling the temperature profile.

Response: The initial attempts to transform Nb2O5 into NBO by only adjusting the temperature were unsuccessful, as they resulted in the formation of the ION phase instead. This indicated that the conditions were not suitable for the desired phase transformation. By increasing the proportion of Nb in the composition, the conditions became favorable for the formation of NBO. The successful synthesis plan involved adding Nb to adjust the composition, followed by a stepwise temperature increase with intermediate holds. This approach allowed the system to reach the desired phase transformation, achieving a phase fraction of NBO close to 1.0.
Answer: [Nb: 0.5], [Temperature: 1500, Time: 60], [Wait: 3600], [Temperature: 1800, Time: 60], [Wait: 7200], [Temperature: 2100, Time: 60], [Wait: 14400]

Figure 6: Unsuccessful attempt on challenge 4 by the LLM using the ReAct framework. The output has been manually formatted for readability and to fit the page. While this plan achieves the desired phase, it forgets to account for the goal temperature and ends with the material heated to 2100K.