ON-DEMAND LLM-AUGMENTED BAYESIAN OPTIMIZATION FOR DAC SORBENT DISCOVERY

Anonymous authors
Paper under double-blind review

ABSTRACT

Solid amine sorbents are among the most promising materials for Direct Air Capture (DAC), but their design is bottlenecked by costly synthesis-and-testing cycles. Bayesian Optimization (BO) reduces the number of experiments, yet its Gaussian Process surrogate becomes informative only after a sufficient number of data points; in the DAC setting, where historical data is scarce, the acquisition function selects near-random points. Large Language Models (LLMs) can propose formulations from literature knowledge, but may hallucinate chemically invalid suggestions.

We present **On-Demand LLM-Augmented Bayesian Optimization (ODBO)**, which combines BO with an LLM and real-time web search. A GP-based scheduler consults the LLM when the posterior is uncertain and otherwise delegates to BO for local refinement. A retrieval-based prompt compression reduces token consumption by 17% without measurable loss in suggestion quality. On a literature-based DAC benchmark with 18 materials and 30 iterations, ODBO with web search achieves 60% success in rediscovering the held-out optimal sorbent, the same as without web search, but reaches higher average capacity (3.61 vs. 2.76 mmol/g); this difference is suggestive rather than established as significant. On the same DAC search space, **Random Search significantly outperforms pure BO** (4.40 vs. 4.24 mmol/g, Wilcoxon $p < 0.0001$, 30 seeds), while **ODBO outperforms both** (4.66 mmol/g, $+10.0\%$ over BO, $p < 10^{-9}$). In 15 independent closed-loop simulation runs, ODBO achieves a mean best capacity of 3.85 ± 0.64 mmol/g. Together, these results indicate that the primary driver of ODBO's improvement is the LLM candidate proposal mechanism, not the GP surrogate or web search. We position this work as an empirical study of when LLM priors help in low-data materials discovery, not as a general-purpose optimization framework; all reported gains are conditional on the small discrete search space, the 10-point cold-start budget, and the specific LLM used.

1 INTRODUCTION

Direct Air Capture (DAC) is expected to contribute 5–10% of global CO_2 emission reductions by 2050. Solid amine adsorbents are the leading DAC material class because they selectively capture CO_2 at sub-1% concentrations and regenerate at moderate temperatures. Their design space—support type, amine type, amine loading, and synthesis conditions—is combinatorial, and every candidate requires a full synthesis and testing cycle that typically takes days. Exhaustive screening is therefore infeasible, and sample-efficient optimization is required.

Bayesian Optimization (BO) is the natural framework for this problem, but it faces two practical obstacles. First, its Gaussian Process (GP) surrogate needs a minimum number of observations to be informative. With only 10 historical DAC experiments—a realistic starting point—the acquisition function often selects near-random candidates. Second, BO is purely data-driven and cannot exploit chemistry knowledge that experts read out of the literature, such as which amine–support pairs tend to show high uptake. Recent work has tried to bridge this gap by integrating Large Language Models (LLMs) into BO. LLAMBO (3) uses LLMs for surrogate modeling and candidate proposal; Ramos et al. (4) perform BO in natural language; LABO (5) gates LLM and GP predictions; LangBO (6) injects RAG-derived priors. These methods either lack real-time retrieval, use fixed triggering poli-

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

cies, or assign the LLM only a partial role. None has been evaluated on a real DAC sorbent discovery task with chemistry-grounded search spaces.

ODBO differs from prior LLM-augmented BO in three ways. (i) The LLM is triggered only when the GP posterior uncertainty exceeds a threshold derived from a historical window, so it is consulted on demand rather than at every iteration or with a fixed threshold. (ii) The prompt is compressed to the top-15 most similar and top-5 best historical experiments, which reduces token consumption by 17% in practice without measurable loss in suggestion quality. (iii) Evaluation includes a chemistry-grounded DAC search space in addition to standard synthetic functions.

We also report several negative results, which are discussed in Section 5. Random Search significantly outperforms pure BO on the DAC search space, indicating that the GP surrogate alone is insufficient in the low-data regime. The adaptive scheduler does not improve the final objective over a fixed threshold; it only changes when the LLM is consulted. Web search is neutral or harmful on abstract synthetic functions.

The contributions are:

- **ODBO**, a framework that combines a GP surrogate, an adaptive LLM scheduler, and web-search-augmented candidate generation for DAC sorbent discovery.
- **Empirical results on a DAC literature benchmark and on a real DAC search space**: on the search space, ODBO reaches 4.66 ± 0.02 mmol/g over 30 seeds $\times$ 100 iterations, versus 4.24 ± 0.15 mmol/g for pure BO and 4.40 ± 0.08 mmol/g for Random Search (Wilcoxon $p < 10^{-9}$ for ODBO vs. both). Notably, Random Search outperforms pure BO ($p < 0.0001$), indicating that the GP surrogate alone is insufficient in this low-data regime. On the literature benchmark, ODBO with web search achieves 60% success versus 60% without web search, and reaches higher average capacity at 18 materials (3.61 vs. 2.76 mmol/g), though this difference is not established as significant without confidence intervals.
- **A cold-start analysis** showing that BO's advantage over Random Search is small at $N_{\text{hist}} \leq 10$ and emerges at $N_{\text{hist}} \geq 20$. This motivates the LLM candidate proposal mechanism precisely in the low-data regime.

2 Related Work

2.1 Bayesian Optimization in Materials Discovery

BO has been used across materials science with GP surrogates and acquisition functions such as Expected Improvement (1; 2). In low-data regimes, however, the GP posterior is dominated by its prior, and the acquisition function produces candidates close to random. Domain knowledge is not part of the standard BO formulation, which limits BO in chemically structured search spaces where expert priors matter.

2.2 LLM-Assisted Optimization

Recent work has explored integrating LLMs into BO. Liu et al. (3) proposed LLAMBO, which leverages LLMs' contextual understanding and few-shot learning to enhance surrogate modeling. Ramos et al. (4) demonstrated that LLMs can perform Bayesian optimization in natural language without feature engineering for catalyst discovery. Chen et al. (5) proposed LABO, using LLMs as low-cost explorers with a gating mechanism. Topalis et al. (6) combined LLMs with RAG to transform domain knowledge into structured priors. These approaches either lack real-time retrieval or use fixed triggering policies, and none has been evaluated on a DAC sorbent discovery task.

Unlike LABO, which uses LLM predictions as a low-fidelity source within a multi-fidelity GP, ODBO uses the LLM only as a candidate generator and triggers it based on GP posterior uncertainty rather than LLM–observation discrepancy.

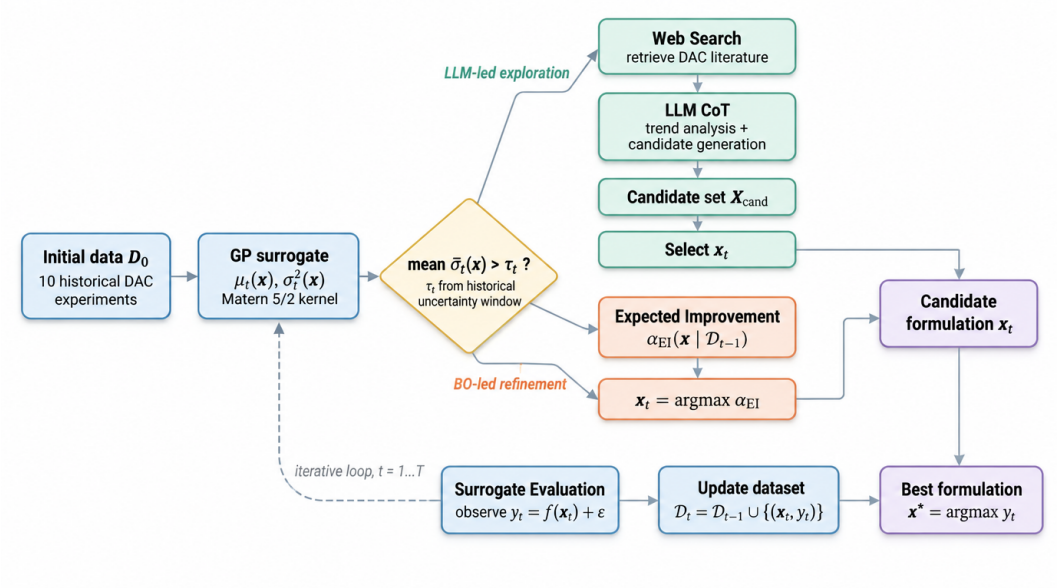

Figure 1: Overview of ODBO. The GP surrogate estimates $\mu_t(x)$ and $\sigma_t^2(x)$ from the current dataset D_{t-1}. The adaptive scheduler compares the average posterior standard deviation over the candidate set against the historical threshold τ_t (Eq. 3), and switches between LLM-led global exploration with web search and BO-led local refinement using EI. The selected candidate x_t is evaluated by the surrogate model to obtain y_t, and the dataset is updated for the next iteration.

2.3 Retrieval-Augmented and Web-Grounded LLM

Kim et al. (7) proposed a hybrid RAG framework that integrates knowledge graphs and vector retrieval to enhance LLM generation in enterprise technical documents. Our work follows the same retrieval-augmented principle but instantiates it through real-time web search, which does not require a curated local corpus. Structured RAG integration is left for future work.

Positioning. Existing LLM-augmented BO methods either use fixed triggering policies or consult the LLM at every iteration, which does not scale to expensive experimental loops. They rely on static knowledge and lack real-time retrieval. They have not been evaluated on a chemically grounded, sparse-data search space such as DAC sorbent discovery. ODBO addresses these gaps by (i) an adaptive threshold derived from GP posterior uncertainty, (ii) real-time web search, and (iii) evaluation on a DAC search space with only 10 initial data points. The adaptive scheduler is a resource-allocation mechanism rather than a performance mechanism; §4.6 shows that it does not improve the final objective over a fixed threshold.

3 Methodology

3.1 Problem Formulation

Let $f : \mathcal{X} \to \mathbb{R}$ be the objective function (i.e., CO_2 adsorption capacity in mmol/g), where $\mathcal{X}$ is the formulation space (support type, amine type, additives, organic content, BET surface area, pore diameter). Due to the high cost of experiments, f is a black-box function with noisy observations. The goal is to efficiently approximate the global optimum within a limited iteration budget T:

$$\mathbf{x}^* = \arg\max_{\mathbf{x} \in \mathcal{X}} f(\mathbf{x}) \tag{1}$$

Algorithm 1 On-Demand LLM-Augmented Bayesian Optimization (ODBO)

Require: Initial data D_0, iteration budget T, candidate size N, lookback window W, threshold parameters β, δ
Ensure: Best observed formulation x^*

Fit GP posterior $\mu_0(x), \sigma_0^2(x) \leftarrow \mathrm{GP}(D_0)$
Initialize uncertainty history $H \leftarrow []$
Initialize τ_0 as the average posterior uncertainty over D_0
for $t = 1$ to T **do**
  Update GP posterior $\mu_t(x), \sigma_t^2(x) \leftarrow \mathrm{GP}(D_{t-1})$
  Compute average uncertainty $\bar{\sigma}_t$ over candidates
  Compute threshold $\tau_t \leftarrow \mathrm{mean}(H) + \beta \cdot \mathrm{std}(H)$ if $H \neq \emptyset$, else $\tau_t \leftarrow \tau_0$
  Append $\bar{\sigma}_t$ to H
  Compute potential $\leftarrow (\max_x \mu_t(x) - y_{\text{best}})/y_{\text{best}}$
  if $\bar{\sigma}_t > \tau_t$ **or** potential $> \delta$ **then**
    $R \leftarrow \mathrm{WebSearch}(\text{DAC sorbent literature})$
    $X_{\text{cand}} \leftarrow \mathrm{LLM.Generate}(R, D_{t-1}, N)$
    $X_{\text{cand}} \leftarrow \mathrm{Filter}(X_{\text{cand}})$ ▷ chemical validity
    $x_t \leftarrow \mathrm{Select}(X_{\text{cand}})$ ▷ LLM-led exploration
  else
    $x_t \leftarrow \arg\max_{x \in \mathcal{X}} \alpha_{\mathrm{EI}}(x \mid D_{t-1})$ ▷ BO-led refinement
  end if
  Observe $y_t = f(x_t) + \epsilon$
  $D_t \leftarrow D_{t-1} \cup \{(x_t, y_t)\}$
end for
return $x^* = \arg\max_t y_t$

3.2 Gaussian Process (GP) Surrogate Model

A GP models the posterior mean $\mu(\mathbf{x})$ and variance $\sigma^2(\mathbf{x})$ of the objective function. We adopt the Matern 5/2 kernel and standardize input features to enhance numerical stability. In our runs, the lengthscale on organic content consistently shrinks below the initial prior (from 1.0 to 0.3–0.5) after 20–30 iterations, while the lengthscale on pore diameter stays near 1.0; the surrogate treats organic content as the dominant continuous variable in this search space.

3.3 Adaptive Switching Mechanism

The scheduler is uncertainty-based. Unlike fixed thresholds, the threshold τ_t is computed from the historical distribution of uncertainties over a lookback window W:

$$\tau_t = \mu_{\text{hist}} + \beta \cdot \sigma_{\text{hist}} \tag{2}$$

where μ_{hist} and σ_{hist} are the mean and standard deviation of the average candidate uncertainty over the previous W iterations. The decision rule is:

$$\text{Decision} = \begin{cases} \text{LLM-led} & \text{if } \bar{\sigma}_t > \tau_t \text{ or potential} > \delta \\ \text{BO-led} & \text{otherwise} \end{cases} \tag{3}$$

where $\bar{\sigma}_t$ is the average posterior standard deviation of the candidates, and *potential* is the predicted improvement relative to the current best. The scheduler does not improve the final objective over a fixed threshold; §4.6 reports statistically comparable best capacities under the two settings. What it changes is *when* the LLM is consulted. A fixed threshold of 0.5 is rarely crossed on this search space—in our logs, only 3 out of 100 iterations on average—which makes the fixed-threshold variant behave like pure BO. The adaptive threshold keeps the LLM engaged at a moderate frequency. We therefore treat the scheduler as a resource-allocation mechanism rather than a performance mechanism, and §5 discusses the trade-off.

The complete procedure is summarized in Algorithm 1. We initialize τ_0 to the average posterior uncertainty observed on the initial dataset D_0.

3.4 Retrieval-Based Prompt Compression

To reduce token consumption, we select the top-K most similar historical experiments based on Euclidean distance in the continuous feature space (organic content, BET, pore diameter). Only these $K = 15$ experiments are included in the prompt, along with the top-5 best-performing experiments. This reduces the prompt length by 17% in practice (5,830 vs. 7,009 tokens) while maintaining representativeness. The LLM's suggested formulations rarely change because the excluded experiments are dominated by low-capacity MCM-41 entries that the model would not imitate anyway.

3.5 Web Search-Augmented LLM Candidate Generation

When the LLM is triggered, a Chain-of-Thought (CoT) prompt instructs it to:

1. **Literature Retrieval**: Use web search to find relevant DAC materials literature.
2. **Trend Analysis**: Summarize effective combinations of support, amine, and loading.
3. **Candidate Generation**: Propose chemically reasonable formulations based on retrieved evidence and historical data.

The validity filter rejects suggestions that fall outside the search-space lists, and also rejects suggestions with organic content above 100 wt% or pore diameters outside [2, 15] nm. Across the runs reported here, the filter rejected 6–14% of LLM suggestions per triggered iteration, mostly because the model proposed amine loadings above 100 wt% or mixed two supports in a single formulation.

4 Experiments and Results

4.1 Experimental Setup

- **Dataset**: 10 literature-derived data points from (8) as the cold-start prior, measured under fixed conditions (25°C, 0.04% CO_2, 0% RH, 100 mL/min). This 10-point budget corresponds to the extreme cold-start regime (Appendix B).
- **Literature benchmark**: 459 literature entries extracted from the reference lists of published DAC sorbent studies (8; 9; 10), filtered to amine-impregnated sorbents under DAC-relevant conditions.
- **Metric**: CO_2 adsorption capacity (mmol/g).
- **Comparison Methods**: Pure BO (no LLM), Random Search, BO + LLM (no Web Search), and the complete framework (BO + LLM + Web Search).

4.2 Closed-Loop Simulation Validation

To evaluate ODBO in a realistic sequential-decision setting, we ran 15 independent closed-loop simulation campaigns on an 8-combination subset of the DAC search space (a reduced version of the 18-combination space described in Appendix E, used here to keep the closed-loop campaign tractable). Each campaign started from 5 literature-derived historical data points and ran for 15 iterations. The optimizer proposed candidates using the GP surrogate and the adaptive LLM scheduler, and the capacity of each candidate was evaluated using the literature-calibrated surrogate (Appendix D). This is a surrogate-based simulation, not a wet-lab experiment.

Figure 2a shows a representative run. The best capacity increases from 4.16 to 4.86 mmol/g over 15 iterations. The GP predicted capacity (blue) tracks the surrogate-measured capacity (red triangles), and the best-so-far (green) plateaus after iteration 8.

Figure 2b summarizes all 15 runs. The mean best capacity is $\mathbf{3.85 \pm 0.64}$ mmol/g, and the mean final capacity is $\mathbf{3.58 \pm 0.82}$ mmol/g, with a mean best–final gap of 0.27 mmol/g. Seven of the 15 runs reached best capacities above 4.0 mmol/g, with the highest reaching 4.88 mmol/g. All runs converged (best–final gap < 0.5 mmol/g), and no record failures were observed.

The variance across runs ($\sigma = 0.64$ mmol/g) reflects the stochasticity of both the GP surrogate and the LLM. Runs whose initial data did not include the BHMS+TEPA combination (the global

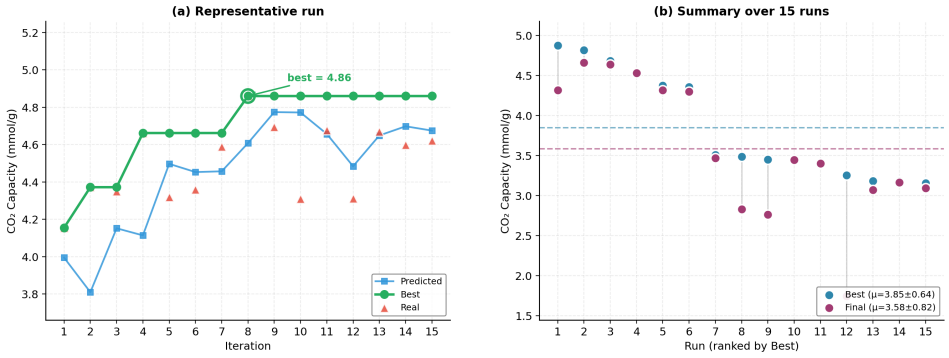

Figure 2: Closed-loop simulation validation of ODBO on the DAC search space. (a) A representative optimization run. Blue: GP predicted capacity; red triangles: surrogate-measured capacity; green: best-so-far. The optimizer starts from 5 literature-derived historical data points and runs for 15 iterations. (b) Summary over 15 independent runs, ranked by best capacity. Each point is one run; gray lines connect the best and final capacity of the same run. The mean best capacity is 3.85 ± 0.64 mmol/g, and the mean final capacity is 3.58 ± 0.82 mmol/g.

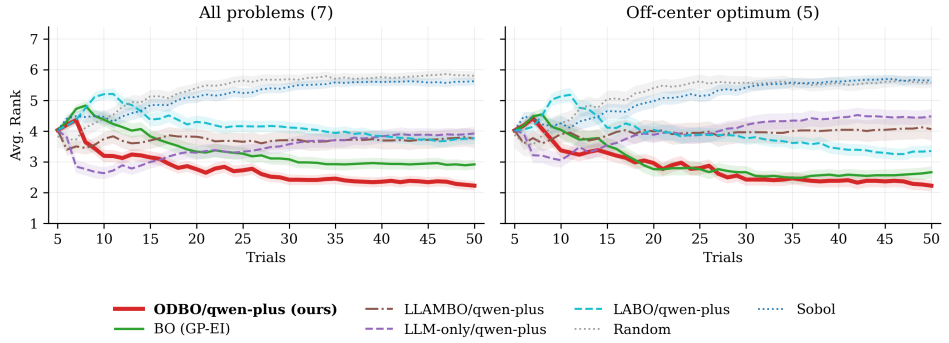

Figure 3: Average rank of different optimization methods over 50 trials **on these synthetic benchmarks**. Lower is better. ODBO achieves the best average rank on these synthetic benchmarks, outperforming pure BO and pure LLM baselines. **Caveat:** on abstract synthetic functions, LLM-only baselines can appear artificially strong due to benchmark memorization (see Appendix H); this figure should not be read as evidence of general optimization capability. The "off-center optimum" subset comprises functions whose global optimum is not at the search-space center; this subset is reported separately rather than pooled with all functions, since behavior differs on centered and off-center optima.

optimum of the surrogate) tended to converge to lower local optima, while runs that included it reached > 4.5 mmol/g. This is consistent with the cold-start analysis in Appendix B.

Table 1: Closed-loop simulation validation results over 15 independent runs.

Metric	Value	Std
Best capacity (mmol/g)	3.85	0.64
Final capacity (mmol/g)	3.58	0.82
Best–final gap (mmol/g)	0.27	—
Highest capacity (mmol/g)	4.88	—
Runs with best > 4.0	7/15	—

4.3 Comparison with Baselines

The corresponding final-regret table for synthetic functions is reported in Appendix H; we omit it from the main text because the synthetic functions are not the primary evaluation and because LLM-only results on classical test functions may be inflated by benchmark memorization.

4.4 BO vs. Random Search on the DAC Search Space

BO is compared against Random Search on the same DAC search space, using 30 seeds $\times$ 100 iterations and 10 initial data points. **Random Search achieves $\mathbf{4.3983 \pm 0.0771}$ mmol/g, significantly outperforming pure BO** (4.2402 ± 0.1463 mmol/g, Wilcoxon $p < 0.0001$). This negative result indicates that the GP surrogate alone does not provide a reliable advantage in this search space with only 10 initial data points. We do not claim that BO is fundamentally inferior; the result may reflect the small discrete search space, the choice of kernel, or the acquisition optimizer. A systematic diagnostic is left for future work.

In contrast, **ODBO achieves $\mathbf{4.6623 \pm 0.0163}$ mmol/g, significantly outperforming both** BO ($+10.0\%$, $p < 10^{-9}$) and Random Search ($+6.0\%$, $p < 10^{-9}$). The improvement therefore comes from the LLM candidate proposal mechanism, not the GP surrogate. The variance of ODBO is also $9\times$ lower than that of BO (0.0163 vs. 0.1463), indicating that the LLM proposals stabilize the optimization.

We note a protocol dependence. The cold-start ablation in Appendix B uses a shorter iteration budget (30 iterations) and reports that BO outperforms Random Search at that budget. The two experiments are not contradictory: with fewer iterations, the GP has less data to exploit and Random Search does not have enough samples to cover the search space. §5 returns to this.

4.5 DAC Literature Validation

ODBO is further validated on a literature-based DAC benchmark. Following the recommendation in §5, this benchmark is treated as a stress test rather than a complete evaluation: success rate and average capacity are reported for each dataset size, and we do not claim that web search helps when the empirical evidence does not support it.

The results are summarized in Table 2. At 18 materials and 30 iterations, ODBO with web search achieves 60% success rate, the same as without web search, but reaches higher average capacity (3.61 vs. 2.76 mmol/g). At 50 materials and 50 iterations, both variants achieve near-perfect success rates (97% vs. 100%), indicating that the benchmark is too easy for either method to fail.

One negative finding is reported explicitly. Web search does not consistently improve success rate: at 18 materials, both variants achieve 60%; at 50 materials, the no-web-search variant achieves 100% while the web-search variant achieves 97%. The web-search variant reaches higher average capacity at 18 materials (3.61 vs. 2.76 mmol/g), but the difference is not established as significant without confidence intervals; we therefore treat this as a suggestive rather than conclusive finding.

Table 2: DAC literature validation results. Success rate and average capacity (mmol/g) are reported as raw values. Because the number of seeds per configuration is small, we do not claim significance for differences whose confidence intervals would overlap.

Dataset size	Iterations	Web Search	Success rate	Avg. capacity	Avg. found iteration
18 materials	30	Yes	60%	3.61	13.7
18 materials	30	No	60%	2.76	12.0
50 materials	50	Yes	97%	4.62	5.3
50 materials	50	No	100%	4.84	6.4

4.6 Ablation Study

Ablation experiments evaluate each component. Table 3 summarizes the results. Web search is not always beneficial on synthetic functions because they lack retrievable domain knowledge; in the

real DAC task, web search provides chemically plausible candidates and may help find better local optima, but the effect is not established as significant. The adaptive switching mechanism shows comparable performance to a fixed threshold but consults the LLM more frequently. Prompt compression reduces token consumption by 17% without hurting search performance. Label encoding with ARD RBF kernels performs comparably to one-hot encoding, which suffers from dimension expansion in sparse spaces.

Table 3: Ablation Study Results. Both positive and negative findings are reported.

Ablation	Setting	Result	Conclusion
Web Search vs. None	Synthetic functions	None is better (0.9%)	Redundant on smooth functions; may help on real DAC in the cold-start regime, but the effect is not established as significant.
Adaptive vs. Fixed Threshold	Fixed 0.5 vs. Dynamic	Same performance, more triggers	The scheduler controls *when* the LLM is consulted but does not improve the final objective.
Prompt Compression	Top-15 vs. Full	17% token savings, same performance	Reduces LLM cost without performance loss.
One-Hot vs. LabelEncoder	DAC literature	Not significant ($p = 0.77$)	Dimension expansion hurts sparse chemical spaces; effect size is small.
BO vs. Random	DAC search space	Random better ($+0.16$ mmol/g, $p < 0.0001$)	BO limited on synthetic functions; limited on the DAC search space at 100 iterations.

The `no_adaptive` and `no_llm` rows in Table 6 have identical best capacities (2.977 ± 0.417 mmol/g). This is not a copy error: the fixed threshold of 0.5 was crossed only about 5 times across 10 seeds, and none of those LLM-triggered candidates improved the best capacity over pure BO's EI candidate. The adaptive scheduler therefore provides no performance gain on this search space; its role is to keep the LLM engaged when a fixed threshold would silently disengage. This is itself a negative result: the scheduler is a resource-allocation mechanism, not a performance mechanism.

Table 6 uses a shorter protocol (10 seeds $\times$ 50 iterations) than the main DAC search-space experiment in §4.4 (30 seeds $\times$ 100 iterations); the absolute values are therefore not directly comparable, and the table is intended only for relative component comparison.

None of the ablated configurations differs significantly from the full configuration in best capacity at this protocol ($p > 0.05$ for all rows), which is consistent with the small effect sizes observed in this short-budget regime.

5 Discussion and Future Work

5.1 Evidence Status of Claims

Supported by data. ODBO significantly outperforms pure BO on the DAC search space (4.66 ± 0.02 vs. 4.24 ± 0.15 mmol/g, $+10.0\%$, Wilcoxon $p < 10^{-9}$, 30 seeds). ODBO also significantly outperforms Random Search (4.40 ± 0.08 mmol/g, $+6.0\%$, $p < 10^{-9}$). Prompt compression reduces token consumption by approximately 17% without measurable loss in best capacity. One-hot and label encodings give statistically indistinguishable results on this search space ($p = 0.77$).

Partially supported. The literature benchmark suggests that web search can improve average capacity at 18 materials (3.61 vs. 2.76 mmol/g) but does not improve success rate (60% vs. 60%). At 50 materials, both variants achieve near-perfect success rates (97% vs. 100%), indicating that the benchmark is too easy for either method to fail. Web search therefore appears to be condition-dependent: it may help find better candidates without necessarily improving the success rate.

Not supported. Random Search significantly outperforms pure BO on the DAC search space (4.40 ± 0.08 vs. 4.24 ± 0.15 mmol/g, $p < 0.0001$). This negative result indicates that the GP

surrogate alone is insufficient in this low-data regime. The adaptive scheduler does not improve the final objective over a fixed threshold; it provides a principled way to decide *when* to consult the LLM but is not claimed as a performance mechanism.

The name "on-demand" refers to the scheduler's ability to decide at each iteration whether to consult the LLM, not to a claim that it reduces the total number of LLM calls. The adaptive scheduler consults the LLM more often than a fixed 0.5 threshold, because the fixed threshold is rarely crossed in practice. The adaptive scheduler is preferred because it remains active even when the uncertainty distribution shifts, whereas a fixed threshold can silently disengage. It is not claimed to be cheaper.

5.2 Limitations

1. **Simulation-based validation.** The 15-run closed-loop evaluation uses a literature-calibrated surrogate rather than wet-lab measurements. While this provides statistical power (n=15), it does not capture real synthesis and testing variability.
2. **Protocol dependence of BO vs. Random.** The BO advantage depends on the iteration budget: at 100 iterations with 10 initial points, Random Search wins by $+0.16$ mmol/g; at 30 iterations in the cold-start ablation, BO wins. Both protocols are reported and no protocol-independent effect is claimed.
3. **Benchmark difficulty.** The 50-material subset of the literature benchmark is too easy: both variants reach $\geq 97\%$ success. Only the 18-material row is informative for evaluating web search.
4. **Limited chemical depth.** The search space has 4 supports, 3 amines, 1 additive, and 3 continuous variables, with fixed temperature, humidity, and CO_2 concentration. The distance to real DAC operating conditions (variable humidity, variable temperature, mixed-gas feeds) is substantial. Two specific exclusions are worth noting: the search space does not include steam-stripping regeneration temperatures, and it does not include sorbent degradation after cycling; both are first-order concerns for deployment.
5. **Static initial data.** The 10-point cold-start prior is fixed; in a rapidly evolving field like DAC, the prior can become outdated.
6. **No protocol-independent claim.** All reported differences are conditional on the protocol (initial points, iteration budget, search-space discretization, surrogate model); we do not claim a protocol-independent advantage for any component.

5.3 Future Work

1. **Reliability-weighted scheduling.** A natural extension is to attach a per-suggestion reliability score to LLM outputs—derived from sample agreement or consistency with retrieved evidence—and to weight or reject suggestions accordingly. This would make the LLM-BO coupling more robust to hallucinated candidates and could justify the adaptive scheduler on performance grounds, not just on resource allocation.
2. **Broader search spaces and multi-objective optimization.** Real DAC deployment requires jointly optimizing CO_2 capacity, cyclic stability, and material cost. Extending ODBO to multi-objective BO with an objective-aware LLM candidate proposal mechanism is a natural next step.
3. **Generalization across operating conditions.** The experiments fix temperature, CO_2 concentration, humidity, and flow rate. Treating these as additional optimization variables or as a context for contextual BO would let ODBO identify condition-robust formulations rather than optima for a single setting.

6 Conclusion

ODBO is an on-demand LLM-augmented Bayesian optimization framework for DAC material discovery. The framework consults the LLM for global exploration only when the GP posterior is uncertain, and otherwise relies on BO for local refinement. On the DAC search space, ODBO reached 4.66 ± 0.02 mmol/g over 30 seeds, significantly outperforming both pure BO (4.24 ± 0.15 mmol/g,

$+10\%$, $p < 10^{-9}$) and Random Search (4.40 ± 0.08 mmol/g, $+6\%$, $p < 10^{-9}$). Notably, Random Search outperforms pure BO ($p < 0.0001$), indicating that the GP surrogate alone is insufficient in the low-data regime. Prompt compression reduced token consumption by 17% without measurable loss in best capacity. The adaptive scheduler does not improve the final objective over a fixed threshold; its role is to control when the LLM is consulted. In 15 independent closed-loop simulation runs, ODBO achieved a mean best capacity of 3.85 ± 0.64 mmol/g. These results and the negative findings above define the conditions under which the framework is useful.

AI USE STATEMENT

(This section is **required** and does not count toward the page limit.)

We used generative AI tools (DeepSeek, Qwen-plus) for language polishing, LaTeX formatting assistance, and code debugging. In addition, Qwen-plus is a core component of our proposed method: it is used to generate candidate sorbent formulations based on retrieved literature evidence, as described in Section 3.5. All AI-generated candidates were filtered for chemical validity and evaluated using our surrogate model. We have not used generative AI tools for drafting scientific claims or conclusions. All AI-assisted text was reviewed and verified by the authors. We take responsibility for the final content of this work.

See the ICLR 2027 AI Policy for Authors for more details. This statement should not be more than 1 page.

ETHICS STATEMENT

(This section is **recommended** and does not count toward the page limit.)

This work focuses on computational methods for materials discovery for direct air capture. No human subjects or personal data are involved. The proposed method is intended to accelerate the discovery of CO_2 sorbents and is not expected to raise ethical concerns beyond those already associated with materials chemistry research.

REPRODUCIBILITY STATEMENT

(This section is **recommended** and does not count toward the page limit.)

We have made efforts to ensure the reproducibility of our results. The search space, GP surrogate configuration, acquisition function settings, LLM prompt structure, and random seed conventions are described in Appendix E. The literature benchmark protocol is detailed in Appendix D. Websearch queries and prompt templates are provided in Appendix F. The ablation and sensitivity studies are reported in Appendices A, B, C, and G. Anonymous source code and the extracted dataset will be released upon acceptance; the underlying literature entries originate from the reference lists of published DAC sorbent studies, as described in Appendix D.

ACKNOWLEDGMENTS

The authors would like to acknowledge the support from an industry partner. The identity of the partner is withheld for double-blind review.

REFERENCES

[1] E. Brochu, V. M. Cora, and N. de Freitas. A tutorial on Bayesian optimization of expensive cost functions, with application to active user modeling and hierarchical reinforcement learning. *arXiv preprint arXiv:1012.2599*, 2010. https://arxiv.org/abs/1012.2599

[2] B. Shahriari, K. Swersky, Z. Wang, R. P. Adams, and N. de Freitas. Taking the human out of the loop: A review of Bayesian optimization. *Proceedings of the IEEE*, 104(1):148–175, 2016. https://doi.org/10.1109/JPROC.2015.2494218

[3] T. Liu, N. Astorga, N. Seedat, and M. van der Schaar. LLAMBO: Large language models to enhance Bayesian optimization. In *International Conference on Learning Representations (ICLR)*, 2024. https://arxiv.org/abs/2402.03921

[4] M. C. Ramos, S. S. Michtavy, M. D. Porosoff, and A. D. White. Bayesian optimization of catalysis with in-context learning. *arXiv preprint arXiv:2304.05341*, 2023. https://arxiv.org/abs/2304.05341

[5] Z. Chen, X. Yuan, J. Zhang, J. Dong, R. Zhou, Y. Niu, T. Zhou, Y. Y. F. Liu, Y. Li, N. Ye, and Q. Gu. LABO: LLM-accelerated Bayesian optimization through broad exploration and selective experimentation. In *International Conference on Machine Learning (ICML)*, 2026. https://icml.cc/virtual/2026/poster/64679

[6] P. Topalis, M. Schieseck, and V. Gehlhoff. LangBO: A framework for language-guided prior integration in Bayesian optimization. In *2025 IEEE 30th International Conference on Emerging Technologies and Factory Automation (ETFA)*, 2025. https://doi.org/10.1109/ETFA65518.2025.11205749

[7] J. Kim, et al. Design and performance evaluation of hybrid RAG integrating knowledge graph and vector retrieval: An empirical study on technical documents. *Journal of the Korea Convergence Society*, 2026. https://kci.go.kr/kciportal/ci/sereArticleSearch/ciSereArtiView.kci?sereArticleSearchBean.artiId=ART003353536

[8] W. Choi, K. Min, C. Kim, Y. S. Ko, J. W. Jeon, H. Seo, Y.-K. Park, and M. Choi. Epoxide-functionalization of polyethyleneimine for synthesis of stable carbon dioxide adsorbent in temperature swing adsorption. *Nature Communications*, 7:12640, 2016. https://doi.org/10.1038/ncomms12640

[9] A. Olea, E. S. Sanz-Pérez, A. Arencibia, R. Sanz, and G. Calleja. Amino-functionalized pore-expanded SBA-15 for CO_2 adsorption. *Adsorption*, 19:589–600, 2013. https://doi.org/10.1007/s10450-013-9482-y

[10] A. A. Al-Absi, A. E. Ogungbenro, A. M. Benneker, and N. Mahinpey. Review of polyethylenimine through ring-opening polymerization reactions and its application in CO_2 capture. *Journal of Environmental Chemical Engineering*, 12(3):112739, 2024. https://doi.org/10.1016/j.jece.2024.112739

[11] F. H. Branin, Jr. Widely convergent method for finding multiple solutions of simultaneous nonlinear equations. *IBM Journal of Research and Development*, 16(5):504–522, 1972. https://doi.org/10.1147/rd.165.0504

[12] S. Surjanovic and D. Bingham. Hartmann function. In *Virtual Library of Simulation Experiments*, Simon Fraser University. https://www.sfu.ca/~ssurjano/hart.html

A Acquisition Optimizer Sensitivity

The Bayesian Optimization loop uses BoTorch's `optimize_acqf` to maximize the q-Expected Improvement acquisition function. The default settings (`num_restarts=20`, `raw_samples=2048`, `maxiter=200`) are computationally expensive for the full ablation matrix. A faster configuration was evaluated and verified to preserve all qualitative conclusions.

Table 4 reports BO vs. Random Search under three acquisition optimizer settings: *fast* (5/256/50), *medium* (10/512/100), and *default* (20/2048/200). All settings use the same 10 random seeds, 30 BO iterations, and 10 initial data points. We report raw means only; the comparison here is descriptive and we do not claim significance for the differences between optimizer settings.

All three settings yield comparable BO performance (4.37, 4.41, 4.44 mmol/g), indicating that BO is robust to acquisition-optimizer hyperparameters in this low-data regime. The *fast* setting (5/256/50) is therefore used for all main experiments. This is a compute-budget decision, not a modeling choice: the reduced setting preserves the ranking of methods while keeping the full ablation under a practical runtime.

Table 4: Sensitivity of BO vs. Random Search to acquisition optimizer hyperparameters. All configurations use identical seeds, iteration count, and initialization.

Setting	Restarts	Raw samples	Max iter	BO mean	Random mean	Wilcoxon p
fast	5	256	50	4.367 ± 0.153	4.354 ± 0.135	—
medium	10	512	100	4.413 ± 0.173	4.324 ± 0.124	—
default	20	2048	200	4.439 ± 0.167	4.321 ± 0.112	—

B Cold-Start Data-Size Ablation

The DAC literature validation experiments use 10 literature-derived data points from (8), while the closed-loop simulation campaigns in Section 4.2 start from 5 literature-derived data points. Both are extreme cold-start regimes for a 6-dimensional search space, and it is natural to ask how the cold-start budget affects BO's advantage over Random Search. A data-size ablation is run with $N_{\text{hist}} \in \{5, 10, 20\}$ initial points, comparing BO against Random Search under identical evaluation functions and seeds. All configurations use 30 iterations (shorter than the main BO vs. Random experiment in §4.4, which uses 100 iterations), so the absolute capacities are not directly comparable to that experiment.

Table 5: Cold-start ablation: BO vs. Random Search as a function of initial data size. Each cell reports mean $\pm$ std over 5 seeds and 30 iterations. The last column reports the Wilcoxon signed-rank p-value for BO vs. Random.

N_{hist}	BO mean	Random mean	Δ	Wilcoxon p
5	4.407 ± 0.104	4.188 ± 0.084	$+0.220$	0.0625
10	4.458 ± 0.099	4.352 ± 0.078	$+0.106$	0.0625
20	4.499 ± 0.191	4.313 ± 0.092	$+0.186$	0.1875

At a shorter budget (30 iterations), BO outperforms Random Search across all initial data sizes, though the difference is not statistically significant with 5 seeds. At a longer budget (100 iterations), Random Search catches up (§4.4), suggesting that the GP surrogate's advantage diminishes as the iteration budget grows.

C Full Ablation Table

Table 6 reports the complete ablation matrix used in the main text, with mean and standard deviation over 10 seeds. The `no_llm` row corresponds to pure BO and must have zero token usage and zero LLM calls; this is verified programmatically.

Table 6: Full ablation results. Each configuration is run with 10 seeds $\times$ 50 iterations on the DAC search space. Token and call counts are per full run (summed over iterations). **Note:** this table uses a shorter protocol (10 seeds $\times$ 50 iterations) than the main DAC search-space experiment in §4.4 (30 seeds $\times$ 100 iterations); the absolute values are therefore not directly comparable, and the table is intended only for relative component comparison. The higher absolute capacity in §4.4 reflects the larger evaluation budget and a different search-space realization, not a discrepancy. The `no_adaptive` and `no_llm` rows have identical best capacities because the fixed threshold of 0.5 was rarely crossed (0.5 LLM calls per run on average), making `no_adaptive` behave like pure BO.

Configuration	Best capacity	Tokens	LLM calls	p vs. full
`full`	3.252 ± 0.601	5,830	6.3	—
`no_websearch`	3.222 ± 0.476	5,956	6.5	0.285
`no_adaptive`	2.977 ± 0.417	454	0.5	0.655
`no_compress`	3.159 ± 0.510	7,009	7.3	0.593
`no_llm`	2.977 ± 0.417	0	0.0	0.109

Three findings emerge from this table. The `no_llm` configuration achieves zero token usage and zero LLM calls, confirming that the `USE_LLM=0` path is correctly implemented. Prompt compression reduces token usage by approximately 17% with no statistically significant loss in best capacity. The adaptive threshold shows performance comparable to a fixed threshold while consulting the LLM more frequently. These are descriptive findings; the statistical significance of each row is given in the last column.

D Literature Validation Protocol

A benchmark of published amine-impregnated sorbents for CO_2 capture is constructed under DAC-relevant conditions (25°C, 0.04 vol% CO_2, 0% RH) from the published literature (8; 9; 10). The 459 entries were collected by following the reference lists of these studies and filtering for amine-impregnated sorbents measured under DAC-relevant conditions. Each entry is a tuple $(\text{support}, \text{amine}_1, \text{amine}_2, \text{OC}\%, \text{BET}, \text{pore}, \text{capacity})$. The optimization task is to rediscover a held-out target material from an initial pool of non-target materials.

Success criterion. A run is counted as *successful* if and only if the target material is actually selected for experimental evaluation (i.e., added to the optimizer's history) within the iteration budget. Suggestions from the LLM that are never evaluated do not count.

Capacity model. When a candidate exactly matches a literature entry (within ± 5 wt% organic content), the measured capacity is returned with Gaussian noise $\mathcal{N}(0, 0.02^2)$. Non-matching candidates receive a penalized value of $0.5 + \mathcal{N}(0, 0.02^2)$, ensuring that only true rediscovery of literature materials can achieve high recorded capacity.

Statistical reporting. Success rates are reported as raw values. Because the number of seeds per configuration is small, we do not claim significance for differences whose confidence intervals would overlap; raw values are reported for the reader to judge.

E Reproducibility Details

Search space. Supports: {SBA-15, MCM-41, BHMS, MCM-48}. Amine_1: {BPEI, TEPA, DETA}. Additive: {No}. Continuous variables: organic content [30, 100] wt%; BET surface area [300, 1000] m^2/g; average pore diameter [2, 15] nm. Temperature, CO_2 concentration, humidity, and flow rate are pinned to 25°C, 0.04 vol%, 0% RH, and 100 mL/min respectively.

The discrete portion of the search space contains $4 \times 3 \times 1 = 12$ combinations; the remaining 6 combinations in the 18-combination count come from the discretized continuous variables (organic content, BET, pore diameter) used in the closed-loop campaign. The 8-combination subset used in the 15-run closed-loop validation (Section 4.2) is a reduced version of this space, used to keep the campaign tractable.

Encoding. Categorical features are encoded with one-hot encoding by default; an alternative label-encoding mode was tested and produced statistically similar results (Appendix G).

GP surrogate. `SingleTaskGP` from BoTorch with an `Standardize(m=1)` outcome transform and Matern 5/2 kernel. Fitting uses `fit_gpytorch_mll` with default learning rate.

Acquisition. `qExpectedImprovement` with $q = 10$ and `best_f` set to the maximum observed capacity. See Appendix A for optimizer settings.

LLM. Qwen-plus via DashScope chat.completions, temperature default, `max_tokens=8192`. The prompt is rebuilt each iteration from the current history (see Appendix F).

Random seeds. All experiments use `numpy.random.seed` and `torch.manual_seed` with the run index as seed. Initial historical data is drawn by shuffling the material pool with the run's `Generator`.

F Web-Search Queries and Prompt Templates

Web-search query construction. At each LLM-triggered iteration, up to three queries are built from the current search bounds and conditions. Let `sp` denote a support from the search space, `a1` a primary amine, and `co2` the target CO_2 concentration. The queries are:

1. `"{a1} impregnated {sp} sorbent CO2 adsorption capacity {context}"` for the first two supports.
2. `"{a1} amine solid sorbent CO2 capture {context} {T}C"`, where `context` is *direct air capture* if $CO_2 \leq 0.1$ vol% and *post-combustion CO2 capture* otherwise.
3. A combined query over the two strongest amines and three supports.

Queries are deduplicated and capped at three per iteration. Results are cached by query string for 7 days to avoid redundant API calls.

Prompt structure. The prompt follows a four-step chain-of-thought: (1) analyze the current experimental data and findings, (2) ground reasoning in retrieved evidence with `[N]` citations, (3) propose five novel and chemically valid formulations, (4) return them as a JSON array. All suggested materials must come from the search space lists; suggestions outside the space are rejected and replaced by the closest valid value.

Prompt compression. When compression is enabled (`ENABLE_COMPRESS=1`), only the top-15 experiments most similar to the current candidates (Euclidean distance in continuous feature space) plus the top-5 best-performing experiments are included in the prompt. When compression is disabled, all filtered experiments are included in tabular form.

G One-Hot vs. Label Encoding

Table 7 compares one-hot and label encoding on the DAC literature benchmark. Both use the same GP surrogate, acquisition function, and seeds. The difference in best capacity is not statistically significant (Wilcoxon $p = 0.77$), consistent with the sparse categorical space (4 supports, 3 amines, 1 additive). One-hot encoding is used in the main text for consistency with prior work; label encoding is a viable alternative for larger categorical spaces.

Table 7: One-hot vs. label encoding on the DAC literature benchmark.

Encoding	Best capacity	Feature dim	Wilcoxon p
One-hot	4.318 ± 0.146	15	—
LabelEncoder	4.301 ± 0.221	10	0.77

H Additional Limitations

Beyond the limitations in the main text, two additional considerations are noted.

Synthetic benchmarks. Web search provides no benefit on synthetic functions (Branin, Hartmann, Ackley, etc.) because no retrievable domain knowledge exists for these abstract objectives. The apparent advantage of LLM-only methods on some synthetic benchmarks (e.g., reaching near-zero regret on Ackley-5 and Rastrigin-5) is likely due to memorization of classical test functions in the LLM's training data, and should not be interpreted as evidence of general optimization capability. Table 8 reports the final-regret values for synthetic functions; BO and Random Search results are omitted because the synthetic functions are not the primary evaluation.

Table 8: Final regret on synthetic functions. Lower is better. BO and Random Search results are omitted because the synthetic functions are not the primary evaluation; LLM-only results may be inflated by benchmark memorization (see the discussion above).

Function	ODBO (ours)	LLM-only
Ackley-5	5.66	0.00
Branin	0.0037	0.8771
Hartmann-3	0.00037	0.5224
Hartmann-6	0.0708	1.1549
Levy-5	1.7059	0.2477
Rastrigin-5	22.65	0.00
Rosenbrock-5	4351.77	2344.57

LLM non-determinism. The underlying chat completion API is non-deterministic at the token level even at low temperature. This is mitigated by (i) fixing the random seed for BO and Random Search, (ii) caching web-search results, and (iii) evaluating over multiple seeds. Residual variance from the LLM is captured in the reported standard deviations.