2026 IEEE International Conference on Robotics and Automation (ICRA 2026)
June 1-5, 2026. Vienna, Austria

Streaming Loop-Closure Selection under Memory Constraints in Graph-SLAM

Reza Vafaee† and Usman A. Khan†

Abstract— **Graph-based SLAM models robot poses as vertices and relative-pose measurements (odometry and loop-closures) as edges. Odometry edges are always kept to preserve connectivity, while loop-closure edges reduce drift but cannot all be stored due to memory or computation limits. Our challenge is to decide online which closures to keep under a strict budget, when the full set of measurements cannot be stored or centralized. Prior work instead addresses an offline problem that assumes access to the complete pose graph and optimizes a log-determinant (D-optimality) surrogate. In the online regime, an additional difficulty arises because the odometry backbone grows over time and the utility of each loop-closure changes as the graph evolves. We formulate this problem as streaming submodular maximization with a time-varying log-determinant objective. We propose a one-pass preemptive greedy policy that operates with exactly k memory slots for loop-closures. We show that, under arbitrary arrival order, it achieves a uniform constant-factor guarantee on the log-determinant improvement beyond an odometry-only baseline, relative to the hindsight-optimal size-k solution. On benchmark data, the proposed method closely matches offline greedy despite the conservative bound, showing that principled streaming selection can recover most of the benefit of loop-closures while respecting resource limits.**

I. Introduction

Simultaneous localization and mapping (SLAM) is a core capability for autonomous navigation, allowing robots to estimate their trajectory while building a consistent map [1]. Modern SLAM back-ends often adopt graph-based formulations in which poses are vertices and relative measurements are edges [2]. This representation yields a sparse least-squares estimator with well-understood accuracy but scalability remains a bottleneck. As exploration proceeds, the number of edges grows quickly, which increases both memory demands and computational cost. On resource-constrained platforms or in long-term missions, it is therefore essential to decide which measurements to retain and which to discard [3]–[6].

To improve scalability, recent work has linked spectral properties of the pose-graph Laplacian to estimation performance, yielding principled criteria for measurement selection. Algebraic connectivity has been used to sparsify pose graphs while bounding error [3], [7], and D-optimality has been tied to the reduced Laplacian, turning the log-determinant into a surrogate for accuracy [4]. Later studies showed that the weighted Laplacian tracks the Fisher information [5] and derived Cramér–Rao bounds relating A-, T-, and D-optimal designs to graph structure [6]. More recent work examined spectral sparsification with tradeoffs that limit edge over-concentration [8].

†The authors are with the Department of Computer Science, Boston College, Chestnut Hill, MA 02467, USA. {vafaee, khaus}@bc.edu. UAK holds concurrent appointments as a Professor at Boston College and as an Amazon Scholar with Amazon Robotics. This paper describes work performed at Boston College and is not associated with Amazon.

Despite these advances, most formulations are offline and assume that all measurements can be stored locally or centralized [3]–[5]. In practice, storage is limited and centralization is often infeasible due to bandwidth and latency. Robots therefore face resource-constrained conditions in which only a bounded subset of measurements can be kept and decisions must be made online without access to the full pose graph. Odometry, which connects consecutive poses, is always retained to preserve connectivity. The challenge lies in loop-closure measurements–relative pose constraints between nonconsecutive poses when the robot revisits a location–which appear intermittently and must be accepted or discarded immediately under a memory budget of k slots reserved for loop-closures. The goal is to maintain estimation quality despite this strict constraint [1], [2].

The link between estimation uncertainty and the log-determinant of the weighted Laplacian [5] allows loop-closure selection to be cast as streaming submodular maximization, since the log-determinant is both monotone and submodular [9]. In streaming optimization, elements are processed sequentially in their arrival order, and some models allow multiple passes over the same stream. In loop-closure selection, each candidate is processed once upon arrival (no revisits) and must be accepted or discarded immediately, and therefore our focus is on strict one-pass algorithms with guarantees under arbitrary arrival order. Earlier methods instead allowed bounded revisits to past elements rather than enforcing this strict one-pass constraint [10].

Among one-pass streaming algorithms with provable guarantees under arbitrary arrival order, Sieve-Streaming [11] is a canonical example. It attains a $(1/2-\varepsilon)$ approximation to the hindsight-optimal solution, but it requires $O(k\log k/\varepsilon)$ memory, which exceeds the strict k-slot constraint relevant to loop-closure storage in SLAM. In fact, with only k slots and adversarial order, there is a fundamental $1/2$ barrier for one-pass submodular maximization [12]. Better guarantees are only available under additional structure, such as random-order streams that can beat $1/2$ [12], preemptive models that allow constant-factor competitiveness under strict feasibility [13], or i.i.d. streams that can approach $(1-1/e)$, where e is Euler's number, with $O(k)$ memory [14]. These results are derived for static objectives, whereas in our SLAM setting the utility varies over time because odometry continuously augments the backbone, which changes the marginal value of loop-closures. This temporal drift is a key departure from the classical streaming model and motivates our analysis.

Loop-closure candidates arrive sequentially, indexed by t, while odometry edges accumulate between arrivals and strengthen the backbone, typically reducing loop-closure marginals. Under a strict budget of k stored loop-closures, each candidate must be accepted or discarded immediately. This departs from offline SLAM selection and from classical

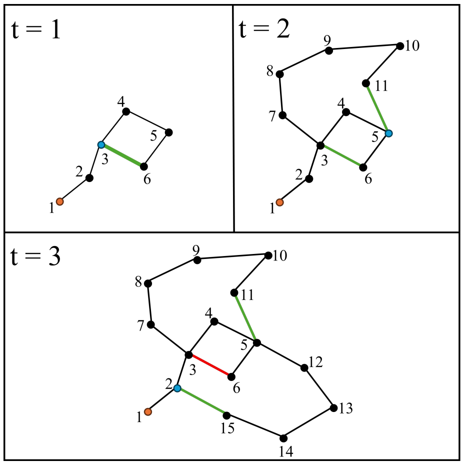

Fig. 1. Toy example of streaming loop-closure selection with a budget of $k = 2$ at successive loop-closure arrivals ($t = 1, 2, 3$). Green edges denote retained loop-closures, red edges indicate discarded ones, and black edges represent odometry. The figure highlights how the pose graph grows between arrivals and how loop-closure decisions must adapt online.

streaming guarantees that assume a static objective, motivating a one-pass policy with exactly k slots for a time-varying log-determinant utility.

Beyond sparsification, measurement selection has also been considered in active SLAM. Swarm-SLAM prioritized inter-robot loop-closures using algebraic connectivity [15]. OpenStreetMap-guided exploration incorporated D-optimality with map priors [16], and semantic mapping combined mutual information from semantic labels with spectral metrics [17]. Exploration over topo-metric priors also applied D-optimality [18], while multi-robot loop-edge selection was cast as submodular maximization with approximation guarantees [19]. Resource-aware collaborative detection was similarly formulated as submodular maximization under communication and computation budgets, with greedy algorithms providing provable bounds [20]. More recently, lightweight uncertainty surrogates have been proposed to scale active SLAM to larger environments [21]. These works show a progression from sparsification to planning to multi-robot coordination, with spectral and information-theoretic measures at the core. However, they all assume full or revisitable access to data and do not address the one-pass, memory-constrained regime.

Contributions:

- We formalize streaming loop-closure selection under memory constraints using a log-determinant (D-optimality) objective defined on the reduced weighted Laplacian;
- We propose a one-pass preemptive greedy algorithm with exactly k memory slots and prove a uniform $1/4$ approximation guarantee on the log-determinant improvement beyond a fixed odometry-only baseline, relative to the hindsight-optimal size-k solution;
- We show on a standard benchmark that our method nearly matches offline greedy under the same budget and outperforms streaming baselines, achieving strong empirical performance under strict memory limits.

To illustrate the streaming setting, Fig. 1 shows a toy pose-graph during exploration in early stages. Three snapshots correspond to the arrival of the first, second, and third loop-closure candidates. The starting vertex is orange, the current pose is blue, odometry edges are black, selected closures are green, and discarded ones are red. The robot may store at most $k = 2$ closures. As odometry accumulates, the backbone grows and the log-determinant objective shifts. In this example, the first two closures are kept, but when the third arrives the policy discards the first and accepts the third, whose marginal gain is higher.

II. Preliminaries and Problem Definition

A. Notation

Finite sets are written in sans-serif, e.g., A, with cardinality $|\mathsf{A}|$. Graphs are denoted by calligraphic letters, e.g., $\mathcal{G}_t$ at loop-closure arrival index t. We use $\mathbb{Z}$ and $\mathbb{R}$ for integers and reals, and $[n] = \{1, \ldots, n\}$ for $n \in \mathbb{Z}_{>0}$. Vectors and matrices are bold, e.g., $\mathbf{v}$ and $\mathbf{A}$, with entries $\mathbf{v}(i)$ and $\mathbf{A}(i, j)$. The transpose is $(\cdot)^\top$, the identity is $\mathbf{I}$, and the all-zero vector or matrix is $\mathbf{0}$. For symmetric matrices, $\mathbf{A} \succeq \mathbf{0}$ and $\mathbf{A} \succ \mathbf{0}$ denote positive semidefinite (PSD) and positive definite (PD), respectively. We write $\det(\cdot)$ for determinant and $\operatorname{logdet}(\cdot)$ for its logarithm. A one-pass streaming algorithm processes elements in arrival order without revisiting past elements. A value-oracle call evaluates the objective on a feasible set. We measure streaming complexity by the number of passes, memory usage, oracle calls per element, and the approximation factor relative to the offline optimum [11]. For a set function g and $e \notin \mathsf{S}$, the marginal gain is $\Delta_g(e \mid \mathsf{S}) \triangleq g(\mathsf{S} \cup \{e\}) - g(\mathsf{S})$.

B. Graph-Theoretic Representation in a Streaming Setting

In pose-graph SLAM [4], [5], [22], the robot maintains a growing undirected graph whose vertices represent poses and whose edges represent relative measurements. Odometry edges connect consecutive poses and are always retained to preserve connectivity. Loop-closure edges connect nonconsecutive poses and arrive intermittently as the robot revisits previously seen places. We adopt a streaming viewpoint in which the index $t \in \mathbb{Z}_{\geq 1}$ counts loop-closure arrivals. At arrival t, the current graph is $\mathcal{G}_t = (\mathsf{V}_t, \mathsf{E}_t)$ with $\mathsf{V}_t = \{v_0, \ldots, v_{n_t}\}$ and $|\mathsf{V}_t| = n_t + 1$. The edge set E_t contains the odometry backbone accumulated up to that time together with the loop-closure candidates observed so far, and the stream ends at T.

Graph connectivity is captured by the incidence matrix $\widetilde{\mathbf{Q}}_t$ and the (unreduced) Laplacian $\widetilde{\mathbf{L}}_t = \widetilde{\mathbf{Q}}_t \widetilde{\mathbf{Q}}_t^\top$. Since $\widetilde{\mathbf{L}}_t$ is singular, we anchor v_0 and remove its row to obtain the reduced incidence $\mathbf{Q}_t$ and reduced Laplacian $\mathbf{L}_t = \mathbf{Q}_t \mathbf{Q}_t^\top$, which is positive definite when the odometry backbone is connected. Measurement reliability is incorporated through positive weights $w(e_j) > 0$, leading to the weighted Laplacian

$$\mathbf{L}_{w,t} = \sum_{e_j \in \mathsf{E}_t} w(e_j)\, \mathbf{E}_t^{(j)}, \qquad \mathbf{E}_t^{(j)} = \mathbf{q}_t^{(j)} \mathbf{q}_t^{(j)\top}, \tag{1}$$

where $\mathbf{q}_t^{(j)}$ is the column of $\mathbf{Q}_t$ associated with edge e_j.

The loop-closure index t advances only when a loop-closure arrives, while odometry can add new vertices between two successive arrivals. To compare edge contributions across different graph sizes, we embed previously formed incidence vectors into the current dimension using zero padding. Specifically, for any $t' < t$ and any edge e_j already defined at time t', we use the convention

$$\mathbf{q}_t^{(j)} = \begin{bmatrix} \mathbf{q}_{t'}^{(j)} \\ \mathbf{0}_{n_t - n_{t'}} \end{bmatrix}, \qquad \mathbf{E}_t^{(j)} = \begin{bmatrix} \mathbf{E}_{t'}^{(j)} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} \end{bmatrix},$$

which preserves each past edge contribution while keeping all spectral quantities well defined at time t.

C. Graph-Based SLAM

Graph-based SLAM estimates the robot trajectory using a *pose graph*, where vertices represent robot poses and edges represent relative measurements between poses [5]. (If landmarks are present, they can be marginalized, yielding a pose-only estimation problem while retaining the geometric constraints needed for accurate localization.) At loop-closure index t, the pose graph is $\mathcal{G}_t = (\mathsf{V}_t, \mathsf{E}_t)$ with $m_t = |\mathsf{E}_t|$. Each edge $e_j = \{v_i, v_k\} \in \mathsf{E}_t$ encodes a noisy relative measurement relating poses $\mathbf{x}_i$ and $\mathbf{x}_k$.

Assuming Gaussian measurement noise, the maximum a posteriori (MAP) trajectory estimate is obtained by solving the nonlinear least-squares problem [2], [23]

$$\mathbf{x}_t^* = \operatorname*{argmin}_{\mathbf{x}_t} J(\mathbf{x}_t), \tag{2}$$

where $\mathbf{x}_t \in \mathbb{R}^{d(n_t+1)}$ stacks the n_t+1 pose variables (with pose dimension d). The objective is

$$J(\mathbf{x}_t) = \frac{1}{2}\sum_{j=1}^{m_t} \mathbf{e}_t^{(j)}(\mathbf{x}_t)^\top \big(\mathbf{\Sigma}_t^{(j)}\big)^{-1}\mathbf{e}_t^{(j)}(\mathbf{x}_t),$$

with residual $\mathbf{e}_t^{(j)}(\mathbf{x}_t)$ on edge e_j and covariance $\mathbf{\Sigma}_t^{(j)}$.

Solving (2) is typically done in one of two ways. In *batch* optimization, the robot postpones estimation until the end of the mission and solves once on the final graph $\mathcal{G}_T$. In *incremental* optimization, the estimate is updated online as new measurements arrive. While batch methods avoid the bookkeeping required by incremental solvers, both approaches face scalability challenges because computation grows with the number and density of edges. Batch systems also run into storage and communication limits, since it is often impractical to retain and exchange all measurements over long trajectories. Incremental systems, on the other hand, must process every accepted loop-closure, which can trigger costly updates and waste computation when many closures are redundant. Because odometry constraints are required to preserve connectivity, pruning is effectively limited to loop-closure edges.

Reducing loop-closures is therefore essential to control memory in batch systems, limit solver and energy overhead in incremental systems, and improve numerical conditioning in both. This motivates the problem of *streaming loop-closure selection*, in which candidates arrive sequentially and must be accepted or discarded irrevocably under a fixed memory budget.

D. Spectral Surrogates for Estimation Uncertainty in Streaming Pose-Graphs

At loop-closure arrival t, each measurement edge $e_j \in \mathsf{E}_t$ contributes a local Fisher information matrix (FIM) $\mathbf{\Phi}_t^{(j)} \triangleq (\mathbf{\Sigma}_t^{(j)})^{-1}$. Aggregating over all measurements yields the global FIM

$$\mathbf{Y}_t \;=\; \sum_{j=1}^{m_t} \widetilde{\mathbf{E}}_t^{(j)} \otimes \mathbf{\Phi}_t^{(j)},$$

where $\widetilde{\mathbf{E}}_t^{(j)} = \widetilde{\mathbf{q}}_t^{(j)}\widetilde{\mathbf{q}}_t^{(j)\top}$ is the (unreduced) Laplacian edge generator and $\otimes$ denotes the Kronecker product [5].

To quantify estimation confidence, we adopt the D-optimality criterion $\rho_D(\mathbf{M}) \triangleq \operatorname{logdet}(\mathbf{M})$, defined for positive definite matrices. This score captures the volume of the Gaussian uncertainty ellipsoid [24] and, for sums of rank-one updates, induces a monotone submodular set function [9], [25]. However, directly evaluating $\rho_D(\mathbf{Y}_t)$ is impractical in streaming SLAM because $\mathbf{Y}_t$ scales with the full state dimension.

Instead, we work with the reduced weighted Laplacian (1), which provides a tractable surrogate for the global information matrix [4], [5]. We encode measurement reliability with a scalar edge weight derived from the local Fisher information,

$$w(e_j) \;=\; \exp\Big(\tfrac{1}{d}\rho_D(\mathbf{\Phi}_t^{(j)})\Big) \;=\; \det(\mathbf{\Phi}_t^{(j)})^{1/d} \;>\; 0. \tag{3}$$

This construction guarantees strictly positive weights and preserves the ranking induced by the local D-optimality score.

Prior work establishes that $\rho_D(\mathbf{L}_{w,t})$ closely tracks $\rho_D(\mathbf{Y}_t)$ [5], and that in 2D pose-graph SLAM the D-optimality of the estimation covariance is tightly bounded by that of the reduced weighted Laplacian [4]. These results motivate using $\rho_D(\mathbf{L}_{w,t})$ as our streaming utility. With the zero-padding convention from Section II, each arriving loop-closure can be scored by its marginal log-determinant gain in a common ambient dimension, enabling one-pass selection without forming the full FIM.

E. Problem Statement: Streaming Loop-Closure Selection under Memory Constraints

We study *streaming loop-closure selection* for pose-graph SLAM under strict resource limits. Loop-closure candidates arrive sequentially. At arrival t, a new candidate edge e_t must be either stored in a finite buffer M_t with $|\mathsf{M}_t| \le M$ or discarded permanently. From this buffer, the SLAM back-end uses an *active set* $\mathsf{S}_t \subseteq \mathsf{M}_t$ of loop-closures, constrained by $|\mathsf{S}_t| \le k \le M$. Odometry edges E_t^o are always retained to preserve connectivity.

Each loop-closure edge e_j is assigned a reliability weight $w(e_j)$ as in (3). Its contribution to the reduced Laplacian is the rank-one generator $\mathbf{E}_t^{(j)} = \mathbf{q}_t^{(j)}\mathbf{q}_t^{(j)\top}$. When the pose set grows, previously accepted edges are embedded into the current dimension using the zero-padding rule in Section II. The resulting reduced weighted Laplacian at time t is

$$\mathbf{L}_{w,t} = \mathbf{L}_{w,t}^o + \sum_{e_j \in \mathsf{S}_t} w(e_j)\mathbf{E}_t^{(j)},$$

where $\mathbf{L}_{w,t}^o$ is the odometry-only reduced Laplacian. Since $\mathbf{L}_{w,t}^o$ is positive definite and each update $w(e_j)\mathbf{E}_t^{(j)}$ is positive semidefinite, $\mathbf{L}_{w,t}$ remains positive definite. We measure estimation confidence using the D-optimality surrogate

$$f_t(\mathsf{S}_t) = \rho_D\Big(\mathbf{L}_{w,t}^o + \sum_{e_j \in \mathsf{S}_t} w(e_j)\mathbf{E}_t^{(j)}\Big). \tag{4}$$

For each fixed t, the set function $f_t(\cdot)$ is monotone and submodular over loop-closure subsets [25]. Across time, however, the objective changes because the odometry backbone grows between loop-closure arrivals. This growth induces two structured temporal effects. First, for any fixed set S, the value $f_t(\mathsf{S})$ is nondecreasing in t (base growth). Second, the marginal benefit of adding a loop-closure typically decreases as odometry strengthens the graph (marginal decay). Appendix A formalizes these properties. This time variation distinguishes our setting from classical streaming submodular maximization, which assumes a static objective [10]–[13], [26].

Formal Problem Statement. Given T loop-closure arrivals and candidate set $\mathsf{E}_T^c = \{e_1, \ldots, e_T\}$, design a one-pass policy that processes each e_t once, maintains feasibility $|\mathsf{M}_t| \leq M$ and $|\mathsf{S}_t| \leq k$, and returns a final set S_T that is competitive with the hindsight-optimal size-k solution

$$\mathsf{OPT} \in \underset{\mathsf{S} \subseteq \mathsf{E}_T^c,\ |\mathsf{S}| \leq k}{\operatorname{argmax}} \ f_T(\mathsf{S}), \tag{5}$$

in the sense that $f_T(\mathsf{S}_T) \geq \alpha f_T(\mathsf{OPT})$ for some $\alpha \in (0, 1]$.

Remark 1: We focus on the strict k-slot regime with $M = k$, but we keep M in the notation to align with prior work that allows larger buffers.

III. A One-Pass Preemptive Streaming Algorithm for Loop-Closure Selection

We now present a streaming policy for loop-closure selection that enforces a strict one-pass constraint and uses exactly k memory slots for loop-closures. For a fixed loop-closure arrival index t, the objective f_t in (4) is monotone and submodular, so the natural offline reference is the classical greedy algorithm. At the terminal time T, offline greedy builds a size-k set by repeatedly adding the candidate with the largest marginal gain, achieving a $(1 - 1/e)$ approximation to the hindsight optimum [27]. This benchmark, however, assumes access to all loop-closures and the full pose graph.

In the streaming SLAM setting, each loop-closure candidate must be processed once and either stored or discarded irrevocably. Decisions must be made using the instantaneous objective f_t, which changes over time because odometry continues to add edges and strengthen the backbone between loop-closure arrivals. As a result, offline greedy is not implementable, and streaming variants that rely on larger buffers, multiple passes, or repeated access to past elements under a static objective do not match our setting [10].

To address these constraints, we build on the preemptive streaming framework in [13] and adapt it to the time-varying Laplacian objective induced by odometry growth. The policy maintains a feasible set S_t of at most k loop-closures. It accepts the first k arrivals to fill the memory. For each subsequent candidate, it evaluates the k possible single-edge swaps with the current retained set and identifies the replacement that yields the largest improvement in f_t. The swap is performed only if this best improvement exceeds a threshold, otherwise the candidate is discarded. This rule uses $O(k)$ value-oracle calls per arrival, operates in one pass under arbitrary stream order, and respects the strict k-slot memory budget.

A. Preemptive Greedy with Fixed-Baseline Threshold

Let f_t denote the instantaneous utility in (4). We work with the baseline-shifted objective $h_t(\mathsf{S}) \triangleq f_t(\mathsf{S}) - b$, where $b \triangleq f_{t_\circ}(\varnothing)$ and $t_\circ$ is the first loop-closure arrival. By base growth, this shift ensures $h_t(\varnothing) \geq 0$ for all t. It also leaves marginal gains unchanged, so it does not affect any accept or reject decision.

Algorithm 1 maintains an active set S_t of at most k loop-closures. It first fills the k slots by accepting the first k arrivals. For each subsequent candidate e_t, it evaluates all k single-edge swaps with the current set and identifies the replacement that yields the largest improvement in f_t.

Algorithm 1 Preemptive Greedy (one-pass, k-memory)

Require: Stream $(e_1, \ldots, e_T)$; budget k; threshold $c > 0$.
Ensure: Final set S_T with $|\mathsf{S}_T| \leq k$.

```
Initialize S_0 ← ∅, b ← f_1(∅), and h_t(S) ≜ f_t(S) − b.
for each arriving e_t do
    if t ≤ k then
        Accept: S_t ← S_{t−1} ∪ {e_t}.
    else
        Let e* ∈ argmax_{e∈S_{t−1}} f_t((S_{t−1} \ {e}) ∪ {e_t}).
        if f_t((S_{t−1} \ {e*}) ∪ {e_t}) − f_t(S_{t−1}) ≥ (c/k) h_t(S_{t−1}) then
            Swap: S_t ← (S_{t−1} \ {e*}) ∪ {e_t}.
        else
            Reject: S_t ← S_{t−1}.
        end if
    end if
end for
```

The candidate is accepted only if this best improvement exceeds the threshold $(c/k)\, h_t(\mathsf{S}_{t-1})$, where $c > 0$ is a user-chosen threshold parameter that controls swap aggressiveness. Smaller c yields more frequent swaps, while larger c is more conservative. This rule is permissive early on when the baseline-shifted utility is small, and it becomes more selective as the odometry backbone strengthens and $h_t(\mathsf{S}_{t-1})$ grows.

The next result (Theorem 1) quantifies the performance guarantee of Algorithm 1. The guarantee is stated under the following standing conditions.

Assumption 1 (Conditions for Theorem 1): **(C1)** For every loop-closure arrival t, the odometry edges E_t^o keep the pose graph connected, so the odometry reduced Laplacian $\mathbf{L}_{w,t}^o$ is positive definite for all $t = 1, \ldots, T$. **(C2)** Loop-closures arrive in a single pass in an arbitrary order (reflecting the exploration order). At each arrival, the policy maintains $|\mathsf{S}_t| \leq k$ and uses only value-oracle evaluations of $f_t(\cdot)$ on sets of size at most k.

Recall. The objective $f_t(\cdot)$ is defined in (4) with strictly positive weights $w(e) > 0$ and the zero-padding convention from Section II. For each fixed t, $f_t(\cdot)$ is monotone and submodular over loop-closure sets; see, e.g., [9]. Moreover, odometry growth satisfies the base-growth and marginal-decay properties in Lemma 1.

Theorem 1 (Approximation guarantee): Under Assumption 1 and the recalled properties of $f_t(\cdot)$ stated above, for any $c > 0$, Algorithm 1 returns S_T such that

$$h_T(\mathsf{S}_T) \ \geq \ \frac{c}{(c+1)^2}\, h_T(\mathsf{OPT}),$$

where $h_t(\mathsf{S}) = f_t(\mathsf{S}) - b$ and $b = f_{t_\circ}(\varnothing)$ is the odometry-only baseline at the first loop-closure arrival $t_\circ$. Equivalently,

$$f_T(\mathsf{S}_T) \ \geq \ \frac{c}{(c+1)^2}\, f_T(\mathsf{OPT}) + \Big(1 - \frac{c}{(c+1)^2}\Big) b.$$

In particular, with $c = 1$ the guarantee specializes to $f_T(\mathsf{S}_T) \geq \frac{1}{4} f_T(\mathsf{OPT}) + \frac{3}{4} b$.

The proof is deferred to Appendix B. Theorem 1 establishes a uniform constant-factor guarantee for Algorithm 1 under a strict one-pass stream, arbitrary arrival order, and exactly k memory slots. The guarantee is stated for the proxy objective used for selection, so it does not directly certify the trajectory error produced by a particular nonlinear back-end.

The guarantee should be interpreted in the context of classical streaming limits. With one pass and strict k-memory,

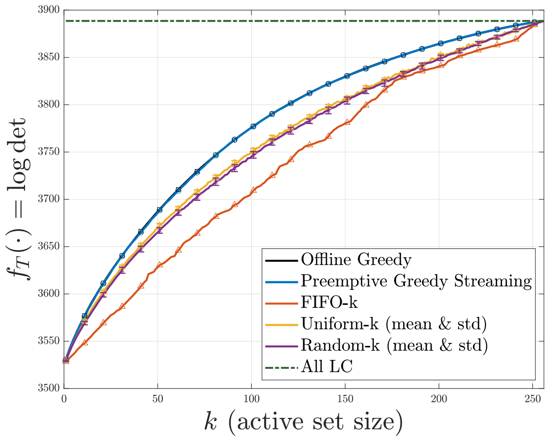

Fig. 2. Objective value $f_T(\cdot)$ versus memory budget k on the Intel dataset. Error bars indicate the standard deviation across 30 trials for the randomized baselines (Random-k and Uniform-k). The proposed streaming algorithm consistently tracks offline greedy, with negligible gap, while outperforming FIFO and randomized baselines across all budgets.

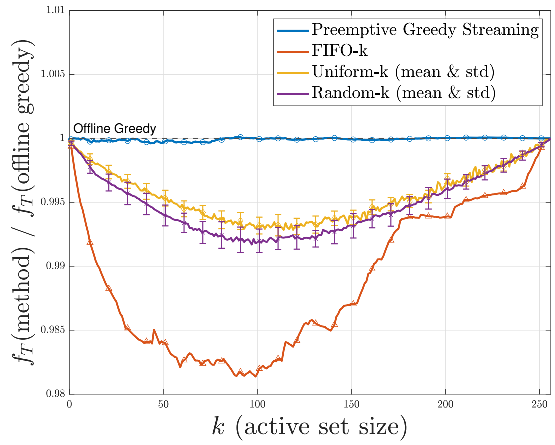

Fig. 3. Normalized performance $f_T(\text{method})/f_T(\text{offline greedy})$ versus k. The streaming preemptive greedy remains essentially indistinguishable from offline greedy across all budgets. In contrast, FIFO and randomized baselines exhibit a U-shaped trend—close to greedy when k is very small or very large, but farthest at moderate k—reflecting redundancy in uninformed selections.

no algorithm can beat a $1/2$ approximation under adversarial order in the standard static objective setting [12, Thm. 1.1]. Sieve-based methods such as SIEVE-STREAMING [11] and SIEVESTREAMING++ [26] approach the $1/2$ frontier by using superlinear memory and oracle calls per element, which conflicts with a strict k-slot budget. Stronger guarantees are known under relaxed stream models, for example random-order or i.i.d. assumptions [12], [14], but these conditions are not natural for loop-closures, which are correlated and environment dependent.

Our setting adds an additional difficulty because the objective drifts as odometry accumulates, violating the static objective assumption used in most streaming analyses. The proposed rule retains the preemptive, exactly k-memory footprint of constant-competitive online policies for static submodular objectives [13], while extending the analysis to the time-varying Laplacian structure induced by SLAM. In deployment, the algorithm outputs a loop-closure subset, and a back-end such as iSAM2 [28] or g2o [29] computes the estimate from that fixed measurement set. Theorem 1 is therefore a selection-level guarantee for the surrogate (4), independent of solver details such as variable ordering or incremental elimination. Its practical relevance is supported by prior connections between pose-graph information measures and the reduced weighted Laplacian log-determinant surrogate [4], [5]. Experiments in Section IV further show that the bound is conservative on structured SLAM streams, with performance closely tracking the offline greedy benchmark.

IV. EXPERIMENTAL RESULTS

We evaluate the proposed streaming loop-closure selection algorithm (Algorithm 1) on the Intel Research Lab dataset [30]. This benchmark contains 1228 poses and 1483 edges, of which 256 are loop-closure candidates and the remaining 1227 are odometry constraints. To emulate an online stream on Intel, we fix a single exploration order and then process loop-closure candidates once in that order. Specifically, we keep the odometry backbone throughout and present the loop-closure edges sequentially by their identifiers. We compare the proposed method against the following baselines.

- *Offline Greedy* [4]: the standard hindsight greedy algorithm applied at the terminal time T, selecting the best size-k loop-closure set using full access to all candidates. This serves as a strong offline reference under the same budget.
- *FIFO-k*: accept the first k loop-closures in the stream and discard all subsequent candidates.
- *Uniform-k*: split the stream into k equal-length segments and sample one loop-closure uniformly at random from each segment.
- *Random-k*: sample k loop-closures uniformly at random from the entire candidate set.
- *All LC*: retain all loop-closures, ignoring the memory constraint (included only as an upper-bound reference).

Except for FIFO-k, these baselines are *offline* in the sense that they are formed using access to the full set of loop-closure candidates (hindsight) and are therefore not constrained to make irrevocable decisions online. For the randomized baselines (Uniform-k and Random-k), we report the mean and standard deviation over 30 independent trials.

The objective is the log-determinant of the reduced weighted Laplacian in (4). We fix the preemptive threshold at $c = 0.05$, where smaller c lowers the acceptance threshold and induces more frequent swaps (more aggressive), while larger c reduces swap frequency and yields a more conservative policy that is less sensitive to small marginal gains. We vary the memory budget k from 1 to 256. Fig. 2 reports the terminal objective value $f_T(\cdot)$ versus k, and Fig. 3 shows performance normalized by offline greedy. As expected, offline greedy achieves the highest utility among the k-budgeted methods, consistent with the $(1-1/e) \approx 0.63$ approximation bound for NP-hard monotone submodular maximization [27]. Across all budgets, our streaming preemptive greedy closely tracks the offline curve. This indicates that the one-pass policy retains near-optimal utility despite irrevocable decisions, and that preemptive thresholding is effective in practice even though Theorem 1 provides only a conservative worst-case guarantee.

We include two randomized baselines to separate the effects of chance from informed selection. Random-k selects k loop-closures uniformly at random from the stream, providing a naïve lower bound on performance. Uniform-k instead enforces temporal coverage by choosing one closure uniformly within each of k equal-length segments of the stream, reducing variance but still ignoring measurement informativeness.

To highlight relative performance, Fig. 3 normalizes each method against offline greedy. The results show that streaming decisions made without hindsight incur virtually no loss in long-term performance. By contrast, the simpler baselines reveal clear limitations. FIFO-k performs worst because early loop-closures are often redundant or geometrically weak, creating a persistent gap from greedy. The randomized strategies achieve higher average utility than FIFO but their reliance on chance introduces trial-to-trial variance, reflecting sensitivity to which closures happen to be chosen.

An interesting pattern in Fig. 3 is the U-shaped performance of the FIFO and randomized baselines. Their results approach offline greedy when k is very small or very large but diverge at moderate budgets. This is expected because for small k all methods keep only a few closures, so the total gain beyond odometry is limited and the gap to greedy is small. At moderate k, greedy and streaming focus on the most informative closures, while FIFO and random methods waste slots on redundant or weak ones, creating the largest gap. As k nears the full set of loop-closures, all methods converge to All-LC and the gap narrows again. This U-shaped trend shows that uninformed policies are competitive only when selection pressure is minimal, while principled strategies are crucial under realistic memory limits.

Beyond these relative differences, it is worth noting why all methods saturate quickly on the Intel dataset. The odometry backbone already provides a strong baseline, and adding loop-closures yields a more modest incremental gain in the log-determinant objective. This helps explain why uninformed strategies such as random or uniform selection can appear competitive on this dataset, since many different subsets can realize similar improvements beyond odometry. This observation also motivates our baseline-shifted analysis in Theorem 1, which quantifies performance on the improvement beyond an odometry-only baseline.

This behavior should not be overinterpreted, as it reflects the specific characteristics of the Intel dataset, a dense indoor environment with frequent, redundant loop-closures. In sparser pose graphs, where informative loop-closures are rare, the incremental value of each selected edge is expected to be more significant and uninformed selection policies would typically perform worse. We focus on Intel because it is a widely used benchmark that ensures reproducibility, while evaluations on additional datasets are omitted due to space limitations.

Finally, we comment on the threshold choice. In all experiments we fix $c = 0.05$ across all budgets without tuning, and the method remains reliable. Theorem 1 holds for any $c > 0$, with approximation factor $c/(c+1)^2$, which is maximized at $c = 1$ and yields the canonical $1/4$ guarantee. Empirically, smaller thresholds promote more frequent swaps, which suits Intel where many loop-closures provide modest but non-negligible gains, whereas larger c can lead to overly conservative acceptance.

These results indicate that, under strict one-pass and k-slot constraints, the proposed policy consistently attains near-offline D-optimal performance on Intel, while substantially outperforming simple streaming baselines. This supports the practicality of principled streaming selection as a drop-in replacement for hindsight selection when memory is limited.

V. Conclusion and Future Directions

We studied streaming loop-closure selection in pose-graph SLAM under strict memory constraints. Unlike prior formulations that optimize a static objective with full access to all measurements, our setting is inherently online: loop-closures arrive sequentially, only k can be stored, and the objective changes over time as the odometry backbone grows. To capture this effect, we formulated a time-varying D-optimality surrogate based on the log-determinant of the reduced weighted Laplacian.

We then proposed a one-pass preemptive greedy policy that operates with exactly k loop-closure slots. We proved a uniform constant-factor guarantee for the baseline-shifted objective, establishing a $1/4$-approximation on the improvement beyond an odometry-only baseline relative to the hindsight-optimal size-k set. On the Intel benchmark, the proposed method achieves objective values that are nearly indistinguishable from offline greedy across the tested budgets, despite the conservative worst-case bound.

Promising directions for future work include: (i) developing sharper guarantees that explicitly account for odometry growth and marginal decay; (ii) evaluating the algorithm on large, sparse outdoor trajectories where informative loop-closures are rare; (iii) coupling selection with incremental solvers to reduce update costs; and (iv) designing adaptive mechanisms to tune the threshold parameter c, for example using a reinforcement-learning policy that selects c based on a reward signal. Pursuing these extensions will further bridge the gap between theoretical guarantees and robust lifelong autonomy in real-world SLAM systems.

Appendix

A. Properties of the Instantaneous Objective

As noted in Section II-E, odometry growth induces two temporal effects: *base growth* and *marginal decay*. We now formalize these effects and state a proof sketch; the full proofs are omitted due to space limitations.

Lemma 1 (Base growth and marginal decay): Fix $t \geq 1$. Let S_{t-1} be any feasible loop-closure set, embedded into the state dimension at loop-closure arrival index t (via the zero-padding rule), so that $f_t(\mathsf{S}_{t-1})$ is well-defined. Then:

(i) *Base growth:* $f_t(\mathsf{S}_{t-1}) \geq f_{t-1}(\mathsf{S}_{t-1})$.

(ii) *Marginal decay:* for any edge $e_j \notin \mathsf{S}_{t-1}$ whose endpoints are already present at time $t-1$,

$$\Delta_{f_t}(e_j \mid \mathsf{S}_{t-1}) \;\leq\; \Delta_{f_{t-1}}(e_j \mid \mathsf{S}_{t-1}),$$

where $\Delta_{f_t}(e \mid \mathsf{S}) \triangleq f_t(\mathsf{S} \cup \{e\}) - f_t(\mathsf{S})$.

Moreover, both statements remain true if f_t is replaced by the baseline-shifted objective $h_t(\cdot) \triangleq f_t(\cdot) - b$ for any constant $b \in \mathbb{R}$.

Proof sketch. For base growth, the weighted matrix-tree theorem implies that $\det(\mathbf{L}_{w,t})$ equals the total weight of spanning trees of the (anchored) pose graph. Odometry growth only adds positive-weight edges and vertices while preserving connectivity, so the total weighted spanning-tree count, and hence $\log\det(\cdot)$, cannot decrease. For marginal decay, the matrix determinant lemma expresses each loop-closure marginal as a logarithm of a weighted effective-resistance term; adding odometry edges decreases effective resistances (Rayleigh monotonicity), so these marginals are nonincreasing in t. The same statements hold for h_t since subtracting a constant baseline does not change marginals.

B. Proof of Theorem 1

This section proves Theorem 1 for Algorithm 1. To build the proof, we first develop and prove two auxiliary lemmas used in the main argument. For $0 \le t \le T$, define the *all-time accepted set* $\mathsf{A}_t \triangleq \bigcup_{j=1}^{t} \mathsf{S}_j$, the union of all loop-closures ever accepted up to time t, including those later preempted. Let $t_\circ$ be the first loop-closure arrival and set the baseline $b = f_{t_\circ}(\varnothing)$. We analyze the baseline-shifted objective $h_t(\mathsf{S}) = f_t(\mathsf{S}) - b$, which satisfies $\Delta_{h_t}(e \mid \mathsf{S}) = \Delta_{f_t}(e \mid \mathsf{S})$.

The acceptance rule of Algorithm 1 is

$$f_t\big((\mathsf{S}_{t-1} \setminus \{e^*\}) \cup \{e_t\}\big) - f_t(\mathsf{S}_{t-1}) \ge \tfrac{c}{k}\, h_t(\mathsf{S}_{t-1}),$$

where e^* is the edge swapped out (chosen as in Algorithm 1) and $c > 0$ is the threshold parameter.

Lemma 2 (Marginal bound): In the streaming loop-closure setting, consider a time step $t > k$ where the memory budget is already full. Let $e^* \in \mathsf{S}_{t-1}$ be the loop-closure selected for removal by Algorithm 1 when evaluating the incoming candidate e_t. Then the marginal gain from performing this swap, evaluated at time t, satisfies

$$\begin{aligned} &f_t\big((\mathsf{S}_{t-1} \setminus \{e^*\}) \cup \{e_t\}\big) - f_t(\mathsf{S}_{t-1}) \\ &\ge \Delta_{f_t}(e_t \mid \mathsf{A}_{t-1}) - \frac{1}{k}\, h_t(\mathsf{S}_{t-1}). \end{aligned}$$

Proof: By choice of e^*,

$$\begin{aligned} &h_t\big((\mathsf{S}_{t-1}\setminus\{e^*\}) \cup \{e_t\}\big) - h_t(\mathsf{S}_{t-1}) \\ &\ge \frac{1}{k} \sum_{e \in \mathsf{S}_{t-1}} \Big(h_t\big((\mathsf{S}_{t-1}\setminus\{e\}) \cup \{e_t\}\big) - h_t(\mathsf{S}_{t-1})\Big). \end{aligned}$$

For each $e \in \mathsf{S}_{t-1}$, submodularity gives

$$\begin{aligned} &h_t\big((\mathsf{S}_{t-1}\setminus\{e\}) \cup \{e_t\}\big) - h_t(\mathsf{S}_{t-1}) \\ &\ge \big[h_t(\mathsf{S}_{t-1} \cup \{e_t\}) - h_t(\mathsf{S}_{t-1})\big] + \big[h_t(\mathsf{S}_{t-1}\setminus\{e\}) - h_t(\mathsf{S}_{t-1})\big]. \end{aligned}$$

Moreover, since $\mathsf{S}_{t-1} \subseteq \mathsf{A}_{t-1}$ and h_t is submodular, $\Delta_{h_t}(e_t \mid \mathsf{S}_{t-1}) \ge \Delta_{h_t}(e_t \mid \mathsf{A}_{t-1})$. Averaging over $e \in \mathsf{S}_{t-1}$ yields

$$\begin{aligned} &\frac{1}{k} \sum_{e \in \mathsf{S}_{t-1}} \Big(h_t\big((\mathsf{S}_{t-1}\setminus\{e\}) \cup \{e_t\}\big) - h_t(\mathsf{S}_{t-1})\Big) \\ &\ge \Delta_{h_t}(e_t \mid \mathsf{A}_{t-1}) + \frac{1}{k} \sum_{e \in \mathsf{S}_{t-1}} \big(h_t(\mathsf{S}_{t-1}\setminus\{e\}) - h_t(\mathsf{S}_{t-1})\big). \end{aligned}$$

Finally, by the telescoping removal bound for submodular functions,

$$\begin{aligned} \frac{1}{k} \sum_{e \in \mathsf{S}_{t-1}} \big(h_t(\mathsf{S}_{t-1}\setminus\{e\}) - h_t(\mathsf{S}_{t-1})\big) &\ge \frac{h_t(\varnothing) - h_t(\mathsf{S}_{t-1})}{k} \\ &\ge -\frac{1}{k} h_t(\mathsf{S}_{t-1}), \end{aligned}$$

where the last inequality uses $h_t(\varnothing) \ge 0$. Combining the displays proves the claim in h_t-units, and the statement in f_t-units follows since $h_t = f_t - b$. ■

Lemma 3: In the streaming loop-closure selection process with time-varying objectives $f_t(\cdot)$ defined in (4), let S_t be the set of currently retained loop-closures at time t. Under the fixed-baseline normalization $h_t(\mathsf{S}) = f_t(\mathsf{S}) - b$ with $b = f_{t_\circ}(\varnothing)$, we have for every $0 \le t \le T$, $h_t(\mathsf{S}_t) \ge \frac{c}{c+1}\, h_t(\mathsf{A}_t)$. Equivalently, in the original units,

$$f_t(\mathsf{S}_t) \ge \frac{c}{c+1}\, f_t(\mathsf{A}_t) + \frac{1}{c+1}\, b.$$

Proof: Define the potential $\Phi_t \triangleq h_t(\mathsf{S}_t) - \frac{c}{c+1}\, h_t(\mathsf{A}_t)$. We prove $\Phi_t \ge 0$ for all t by showing $\Phi_t - \Phi_{t-1} \ge 0$ and noting $\Phi_0 = 0$.

Write

$$\begin{aligned} \Phi_t - \Phi_{t-1} = &\underbrace{\big(h_t(\mathsf{S}_t) - h_t(\mathsf{S}_{t-1})\big) - \tfrac{c}{c+1}\big(h_t(\mathsf{A}_t) - h_t(\mathsf{A}_{t-1})\big)}_{\text{(I) same-time increment}} \\ &+ \underbrace{\big(h_t(\mathsf{S}_{t-1}) - h_{t-1}(\mathsf{S}_{t-1})\big) - \tfrac{c}{c+1}\big(h_t(\mathsf{A}_{t-1}) - h_{t-1}(\mathsf{A}_{t-1})\big)}_{\text{(II) cross-time drift}}. \end{aligned}$$

(I) *Same-time increment.*

- If e_t is rejected, $\mathsf{S}_t = \mathsf{S}_{t-1}$ and $\mathsf{A}_t = \mathsf{A}_{t-1}$, so (I) $= 0$.
- If $t \le k$ and e_t is accepted, then by submodularity and $\mathsf{S}_{t-1} \subseteq \mathsf{A}_{t-1}$, $h_t(\mathsf{S}_t) - h_t(\mathsf{S}_{t-1}) = \Delta_{h_t}(e_t \mid \mathsf{S}_{t-1}) \ge \Delta_{h_t}(e_t \mid \mathsf{A}_{t-1}) = h_t(\mathsf{A}_t) - h_t(\mathsf{A}_{t-1})$, so (I) $\ge (1 - \frac{c}{c+1})(h_t(\mathsf{A}_t) - h_t(\mathsf{A}_{t-1})) \ge 0$.
- If $t > k$ and e_t is accepted via swap, two bounds hold: $h_t(\mathsf{S}_t) - h_t(\mathsf{S}_{t-1}) \ge \Delta_{h_t}(e_t \mid \mathsf{A}_{t-1}) - \frac{1}{k} h_t(\mathsf{S}_{t-1})$, according to Lemma 2, and $h_t(\mathsf{S}_t) - h_t(\mathsf{S}_{t-1}) \ge \frac{c}{k}\, h_t(\mathsf{S}_{t-1})$ (swap acceptance rule).
  Taking the convex combination with weights $\frac{c}{c+1}$ and $\frac{1}{c+1}$ cancels the $h_t(\mathsf{S}_{t-1})$ term and yields
  $$\begin{aligned} h_t(\mathsf{S}_t) - h_t(\mathsf{S}_{t-1}) &\ge \tfrac{c}{c+1}\, \Delta_{h_t}(e_t \mid \mathsf{A}_{t-1}) \\ &= \tfrac{c}{c+1}\, \big(h_t(\mathsf{A}_t) - h_t(\mathsf{A}_{t-1})\big). \end{aligned}$$
  So (I) ≥ 0.

(II) *Cross-time drift.* Let $\mathbf{P}_t \succeq 0$ denote the odometry PSD update $\mathbf{P}_t \triangleq \mathbf{L}^o_{w,t} - \mathbf{L}^o_{w,t-1}$. For any set S, write $\mathbf{L}_{w,t-1}(\mathsf{S}) \triangleq \mathbf{L}^o_{w,t-1} + \sum_{e_j \in \mathsf{S}} w(e_j)\, \mathbf{E}^{(j)}_{t-1}$, so that

$$\begin{aligned} &f_t(\mathsf{S}) - f_{t-1}(\mathsf{S}) \\ &= \operatorname{logdet}\big(\mathbf{L}_{w,t-1}(\mathsf{S}) + \mathbf{P}_t\big) - \operatorname{logdet}\big(\mathbf{L}_{w,t-1}(\mathsf{S})\big) \\ &= \operatorname{logdet}\Big(\mathbf{I} + \mathbf{L}^{-1/2}_{w,t-1}(\mathsf{S})\, \mathbf{P}_t\, \mathbf{L}^{-1/2}_{w,t-1}(\mathsf{S})\Big). \end{aligned}$$

If $\mathsf{S} \subseteq \mathsf{R}$ then $\mathbf{L}_{w,t-1}(\mathsf{S}) \preceq \mathbf{L}_{w,t-1}(\mathsf{R})$ and hence

$$\begin{aligned} &\operatorname{logdet}\Big(\mathbf{I} + \mathbf{L}^{-1/2}_{w,t-1}(\mathsf{S})\, \mathbf{P}_t\, \mathbf{L}^{-1/2}_{w,t-1}(\mathsf{S})\Big) \\ &\ge \operatorname{logdet}\Big(\mathbf{I} + \mathbf{L}^{-1/2}_{w,t-1}(\mathsf{R})\, \mathbf{P}_t\, \mathbf{L}^{-1/2}_{w,t-1}(\mathsf{R})\Big), \end{aligned}$$

since the map $\mathbf{A} \mapsto \operatorname{logdet}(\mathbf{I} + \mathbf{A}^{-1/2}\mathbf{P}\mathbf{A}^{-1/2})$ is monotone *decreasing* in the Loewner order (operator concavity of logdet). Therefore, with $\mathsf{S}_{t-1} \subseteq \mathsf{A}_{t-1}$, $f_t(\mathsf{S}_{t-1}) - f_{t-1}(\mathsf{S}_{t-1}) \ge f_t(\mathsf{A}_{t-1}) - f_{t-1}(\mathsf{A}_{t-1}) \ge 0$.

Because $h_t(\cdot) = f_t(\cdot) - b$, the same identity holds for h_t, i.e., $h_t(\mathsf{S}) - h_{t-1}(\mathsf{S}) = f_t(\mathsf{S}) - f_{t-1}(\mathsf{S})$, hence

$$0 \le h_t(\mathsf{S}_{t-1}) - h_{t-1}(\mathsf{S}_{t-1}) \ge h_t(\mathsf{A}_{t-1}) - h_{t-1}(\mathsf{A}_{t-1}),$$

and thus (II) ≥ 0.

Combining (I) ≥ 0 and (II) ≥ 0 gives $\Phi_t - \Phi_{t-1} \ge 0$. Since $\Phi_0 = 0$, we conclude $\Phi_t \ge 0$ for all t, i.e., $h_t(\mathsf{S}_t) \ge \frac{c}{c+1}\, h_t(\mathsf{A}_t)$, which is equivalent to the stated inequality in f_t-units. ■

Proof of Theorem 1: Fix $e \in \mathsf{OPT} \setminus \mathsf{A}_T$ and let $\tau(e)$ be its arrival time. Since e is rejected at $\tau(e)$, the threshold rule implies

$$\frac{c}{k}\,h_{\tau(e)}\big(\mathsf{S}_{\tau(e)-1}\big) > f_{\tau(e)}\big((\mathsf{S}_{\tau(e)-1}\setminus\{e^*\})\cup\{e\}\big) - f_{\tau(e)}\big(\mathsf{S}_{\tau(e)-1}\big), \tag{6}$$

where $e^* \in \mathsf{S}_{\tau(e)-1}$ is the element that would be removed if a swap occurred.

By Lemma 2,

$$\begin{aligned} &f_{\tau(e)}\big((\mathsf{S}_{\tau(e)-1}\setminus\{e^*\})\cup\{e\}\big) - f_{\tau(e)}\big(\mathsf{S}_{\tau(e)-1}\big) \\ &\geq \Delta_{f_{\tau(e)}}\big(e \mid \mathsf{A}_{\tau(e)-1}\big) - \frac{1}{k}\,h_{\tau(e)}\big(\mathsf{S}_{\tau(e)-1}\big). \end{aligned} \tag{7}$$

Combining (6) and (7) yields

$$\Delta_{f_{\tau(e)}}\big(e \mid \mathsf{A}_{\tau(e)-1}\big) < \frac{c+1}{k}\,h_{\tau(e)}\big(\mathsf{S}_{\tau(e)-1}\big). \tag{8}$$

Lifting to terminal time T. We use two standard, time-wise facts for f_t (see Lemma 1): (i) *base growth*: for any fixed set S, $f_t(\mathsf{S})$ is nondecreasing in t; (ii) *marginal decay*: for any fixed S and e, $\Delta_{f_t}(e \mid \mathsf{S})$ is nonincreasing in t. Since $h_t(\cdot) = f_t(\cdot) - b$ with fixed b, base growth for f_t implies the same for h_t, and by the acceptance rule $h_t(\mathsf{S}_t)$ is nondecreasing in t. From (i) and algorithm monotonicity,

$$h_{\tau(e)}\big(\mathsf{S}_{\tau(e)-1}\big) \leq h_T\big(\mathsf{S}_{\tau(e)-1}\big) \leq h_T(\mathsf{S}_T).$$

From (ii) and submodularity in the set argument,

$$\Delta_{f_T}(e \mid \mathsf{A}_T) \leq \Delta_{f_T}\big(e \mid \mathsf{A}_{\tau(e)-1}\big) \leq \Delta_{f_{\tau(e)}}\big(e \mid \mathsf{A}_{\tau(e)-1}\big).$$

Combining with (8) gives the uniform terminal bound

$$\Delta_{f_T}(e \mid \mathsf{A}_T) < \frac{c+1}{k}\,h_T(\mathsf{S}_T), \qquad \forall\, e \in \mathsf{OPT}\setminus\mathsf{A}_T. \tag{9}$$

By submodularity and monotonicity of f_T,

$$f_T(\mathsf{OPT}) \leq f_T(\mathsf{A}_T) + \sum_{e\in\mathsf{OPT}\setminus\mathsf{A}_T} \Delta_{f_T}(e \mid \mathsf{A}_T).$$

Applying (9) and $|\mathsf{OPT}\setminus\mathsf{A}_T| \leq k$ yields

$$f_T(\mathsf{OPT}) \leq f_T(\mathsf{A}_T) + (c+1)\,h_T(\mathsf{S}_T). \tag{10}$$

From Lemma 3 at time T, $h_T(\mathsf{A}_T) \leq \frac{c+1}{c}\,h_T(\mathsf{S}_T)$, hence $f_T(\mathsf{A}_T) = h_T(\mathsf{A}_T) + b \leq \frac{c+1}{c}\,h_T(\mathsf{S}_T) + b$. Substituting into (10) yields $f_T(\mathsf{OPT}) \leq \frac{(c+1)^2}{c}\,h_T(\mathsf{S}_T) + b$, i.e., $h_T(\mathsf{S}_T) \geq \frac{c}{(c+1)^2}\big(f_T(\mathsf{OPT}) - b\big)$. Equivalently,

$$f_T(\mathsf{S}_T) \geq \frac{c}{(c+1)^2}\,f_T(\mathsf{OPT}) + \Big(1 - \frac{c}{(c+1)^2}\Big)\,b.$$

References

[1] C. Cadena, L. Carlone, H. Carrillo, Y. Latif, D. Scaramuzza, J. Neira, I. Reid, and J. J. Leonard, "Past, present, and future of simultaneous localization and mapping: Toward the robust-perception age," *IEEE Transactions on robotics*, vol. 32, no. 6, pp. 1309–1332, 2017.

[2] G. Grisetti, R. Kümmerle, C. Stachniss, and W. Burgard, "A tutorial on graph-based SLAM," *IEEE Intelligent Transportation Systems Magazine*, vol. 2, no. 4, pp. 31–43, 2011.

[3] K. J. Doherty, D. M. Rosen, and J. J. Leonard, "Spectral measurement sparsification for pose-graph SLAM," in *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems*. IEEE, 2022, pp. 01–08.

[4] K. Khosoussi, M. Giamou, G. S. Sukhatme, S. Huang, G. Dissanayake, and J. P. How, "Reliable graphs for SLAM," *The International Journal of Robotics Research*, vol. 38, no. 2-3, pp. 260–298, 2019.

[5] J. A. Placed and J. A. Castellanos, "A general relationship between optimality criteria and connectivity indices for active graph-SLAM," *IEEE Robotics and Automation Letters*, vol. 8, no. 2, pp. 816–823, 2022.

[6] Y. Chen, S. Huang, L. Zhao, and G. Dissanayake, "Cramér–Rao bounds and optimal design metrics for pose-graph SLAM," *IEEE Transactions on Robotics*, vol. 37, no. 2, pp. 627–641, 2021.

[7] D. M. Rosen, L. Carlone, A. S. Bandeira, and J. J. Leonard, "Se-sync: A certifiably correct algorithm for synchronization over the special euclidean group," *The International Journal of Robotics Research*, vol. 38, no. 2-3, pp. 95–125, 2019.

[8] J. Nam, S. Hyeon, Y. Joo, D. Noh, and H. Shim, "Spectral trade-off for measurement sparsification of pose-graph SLAM," *IEEE Robotics and Automation Letters*, vol. 9, no. 1, pp. 723–730, 2023.

[9] M. Shamaiah, S. Banerjee, and H. Vikalo, "Greedy sensor selection: Leveraging submodularity," in *49th IEEE conference on decision and control (CDC)*. IEEE, 2010, pp. 2572–2577.

[10] R. Gomes and A. Krause, "Budgeted nonparametric learning from data streams." in *ICML*, vol. 1, 2010, p. 3.

[11] A. Badanidiyuru, B. Mirzasoleiman, A. Karbasi, and A. Krause, "Streaming submodular maximization: Massive data summarization on the fly," in *Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining*, 2014, pp. 671–680.

[12] A. Norouzi-Fard, J. Tarnawski, S. Mitrovic, A. Zandieh, A. Mousavifar, and O. Svensson, "Beyond 1/2-approximation for submodular maximization on massive data streams," in *International Conference on Machine Learning*. PMLR, 2018, pp. 3829–3838.

[13] N. Buchbinder, M. Feldman, and R. Schwartz, "Online submodular maximization with preemption," *ACM Transactions on Algorithms (TALG)*, vol. 15, no. 3, pp. 1–31, 2019.

[14] S. Buschjäger, P.-J. Honysz, L. Pfahler, and K. Morik, "Very fast streaming submodular function maximization," in *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*. Springer, 2021, pp. 151–166.

[15] P.-Y. Lajoie and G. Beltrame, "Swarm-SLAM: Sparse decentralized collaborative simultaneous localization and mapping framework for multi-robot systems," *IEEE Robotics and Automation Letters*, vol. 9, no. 1, pp. 475–482, 2023.

[16] W. Gao, Z. Sun, M. Zhao, C.-Z. Xu, and H. Kong, "Active loop closure for osm-guided robotic mapping in large-scale urban environments," in *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems*. IEEE, 2024, pp. 12 302–12 309.

[17] R. Zhang, H. M. Bong, and G. Beltrame, "Active semantic mapping and pose graph spectral analysis for robot exploration," in *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems*. IEEE, 2024, pp. 13 787–13 794.

[18] R. Bai, H. Guo, W.-Y. Yau, and L. Xie, "Graph-based SLAM-aware exploration with prior topo-metric information," *IEEE Robotics and Automation Letters*, vol. 9, no. 9, pp. 7597–7604, 2024.

[19] R. Bai, S. Yuan, H. Guo, P. Yin, W.-Y. Yau, and L. Xie, "Multi-robot active graph exploration with reduced pose-SLAM uncertainty via submodular optimization," in *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems*. IEEE, 2024, pp. 10 229–10 236.

[20] Y. Tian, K. Khosoussi, and J. P. How, "A resource-aware approach to collaborative loop-closure detection with provable performance guarantees," *The International Journal of Robotics Research*, vol. 40, no. 10-11, pp. 1212–1233, 2021.

[21] H. Yin, J. J. Park, M. Almeida, M. Labrie, J. Zamiska, and R. Kim, "Probabilistic active loop closure for autonomous exploration," in *Proceedings of the IEEE International Conference on Robotics and Automation*. IEEE, 2024, pp. 18 048–18 054.

[22] L. Carlone, "A convergence analysis for pose graph optimization via gauss-newton methods," in *Proceedings of the IEEE International Conference on Robotics and Automation*. IEEE, 2013, pp. 965–972.

[23] T. D. Barfoot, *State estimation for robotics*. Cambridge University Press, 2024.

[24] S. Joshi and S. Boyd, "Sensor selection via convex optimization," *IEEE Transactions on Signal Processing*, vol. 57, no. 2, pp. 451–462, 2008.

[25] R. Vafaee and M. Siami, "Learning-based sparse sensing with performance guarantees," *IEEE Transactions on Automatic Control*, vol. 70, no. 1, pp. 387–402, 2024.

[26] E. Kazemi, M. Mitrovic, M. Zadimoghaddam, S. Lattanzi, and A. Karbasi, "Submodular streaming in all its glory: Tight approximation, minimum memory and low adaptive complexity," in *International Conference on Machine Learning*. PMLR, 2019, pp. 3311–3320.

[27] G. L. Nemhauser, L. A. Wolsey, and M. L. Fisher, "An analysis of approximations for maximizing submodular set functions–I," *Mathematical programming*, vol. 14, pp. 265–294, 1978.

[28] M. Kaess, H. Johannsson, R. Roberts, V. Ila, J. J. Leonard, and F. Dellaert, "iSAM2: Incremental smoothing and mapping using the Bayes tree," *The International Journal of Robotics Research*, vol. 31, no. 2, pp. 216–235, 2012.

[29] R. Kümmerle, G. Grisetti, H. Strasdat, K. Konolige, and W. Burgard, "g2o: A general framework for graph optimization," in *Proceedings of the IEEE International Conference on Robotics and Automation*. IEEE, 2011, pp. 3607–3613.

[30] A. Howard and N. Roy, "The robotics data set repository (radish)," 2003. [Online]. Available: http://radish.sourceforge.net/