Author Response to Reviewers

Anonymous Author(s)

Abstract

We sincerely thank the reviewers for their insightful and valuable suggestions, many of which have sparked our thinking and will greatly contribute to the further improvement of our paper. Below, we address the reviewers' comments, providing clarifications for misunderstandings and outlining our proposed revision plan. The code can be found at https://github.com/xrj1014/test-DiskANN.

1 Response to Reviewer #1

R1.O1: The reason why we assume that the data are uniformly distributed is as follows: (1) Any distribution can be transformed into a uniform one via an optimal transport map, and vice versa, enabling us to extend the pruning probability bounds to diverse real-world distributions. (2) Spatial point distributions can typically be modeled by a point process. Given a fixed number of points in a defined region (e.g., DiskANN's setup), a common Poisson point process yields a uniform conditional distribution, as elaborated in "Graph-based Nearest Neighbor Search: From Practice to Theory" (ICML 2020): "It's worth noting that our proofs assume elements follow a Poisson point process on S^d with an expected n points. This simplifies proofs without altering results, as the distributions are asymptotically equivalent. Conditioning on the node count in the Poisson process gives uniformity." (3) The uniform distribution assumption is well-established, as in [11], where it underpins the theoretical analysis of nearest neighbor search. We will add these explanations in the revised version.

R1.O2: (a) We regret repeatedly referring to "Theorem 1" without a correct reference. In fact, "Theorem 1" is "Theorem 3.1". We will fix this issue in the revision. (b) Our "single round" claim refers to tuning R for a fixed α, not implying multiple construction iterations. In DiskANN, α governs pruning (e.g., 1 or 1.2 for short/long edges) and isn't iterative, while R—the termination condition—requires one-time optimization per α. We'll clarify this with added details in the revision. (c) In Section 5.1 of the updated manuscript, we'll add descriptions of the theorems and assumptions used: (1) Vamana's GreedySearch complexity in Section 5.1 depends on Theorem 3.4 (path length $O(\log n)$) and Theorem 3.1 (out-degree bound $O(n^{2/3})$) since its time complexity is $O(\text{path length} \times \text{out-degree})$. (2) In Algorithm 3 (Section 5.2), R's initial value is set to $n^{2/3}$ based on Theorem 3.1. (3) Section 5 still assumes uniformity. (d) We'll reword Theorem 3.1 for precision, listing conditions explicitly. We apologize for the lack of clarity. These conditions were expressed earlier and are generally satisfied for graph indices: (1) the difference equation is valid, and (2) the piecewise form of the pruning probability r_t is satisfied. With these, we can rewrite T_n's expression.

R1.O3: In Section 5, we tested Algorithm 3 on SIFT1M, GIST1M, and DEEP1M, computing optimal R^* for fixed α, evaluating recall and latency. Table 2 shows R^* (marked) alongside manually tuned control R values; their similar recall and latency suggest efficiency. However, we didn't compare R^* to exhaustive search optima, limiting proof of tuning avoidance. We'll update Section 5 with an end-to-end runtime, recall, and latency comparison of DiskANN using R^* versus binary search over R, clarifying results.

R1.O4: We will expand Section 1 in the revised manuscript with a dedicated discussion on our relationship with NSG, precisely outlining distinctions and highlighting how our approach broadens applicability beyond high-dimensional constraints. Both works analyze graph-based approximate nearest neighbor search. NSG [11] (Theorem 3, Page 17) proves a search path length for the Monotonic Relative Neighborhood Graph (MRNG), assuming uniform distribution in high-dimensional Euclidean space E^d, where bounds depend on dimensionality and the minimum distance difference Δr. Our work models the RobustPrune algorithm's graph construction and pruning iterations as a discrete-time stochastic process, deriving a degree bound (Theorem 3.1, Page 5) and a search path length bound (Theorem 3.4, Page 6). Regarding the high-dimensionality assumption in [11], we clarify that NSG's results hinge on uniform distribution in high-dimensional Euclidean space, where low-dimensional cases may invalidate its logarithmic bounds. Our work adopts the same uniform distribution assumption but eliminates the asymptotic dimensional dependency in Theorem 3.4, offering a more universally applicable $O(\log n)$ bound.

R1.O5: Unlike query-dependent tuning, our approach focuses on the algorithm itself, independent of a specific set of queries. Here, parameter tuning refers specifically to optimizing R (controls the complexity and termination condition of the RobustPrune algorithm), rather than involving the input or output of queries.

R1.O6: We appreciate the reviewer's suggestion. In the revision, we plan to restructure Section 5 into two distinct parts: (1) "Parameter Optimization Algorithm" (encompassing Sections 5.1 and 5.2), and (2) "Experimental Validation" (including Section 5.3 and a new comparison of runtime with a binary search). Sections 5.1 and 5.2 focus on complexity estimation and algorithm design, which are theoretical and methodological, while Section 5.3 emphasizes empirical validation.

R1.O7: We recognize that concepts like σ-algebras and F-measurable random variables, used without prior introduction, may affect readability. In the revised manuscript, we will include a concise explanation in Section 2, "Preliminaries," to introduce these probability fundamentals: A σ-algebra is a collection of subsets of a sample space, closed under complementation and countable unions, forming the basis for a probability measure. An F-measurable random variable is a function from the sample space to the real numbers such that the preimage of any Borel set lies in the σ-algebra F, ensuring it is quantifiable given F's information.

Conference'17, July 2017, Washington, DC, USA

ACM ISBN 978-x-xxxx-xxxx-x/YY/MM
https://doi.org/10.1145/nnnnnnn.nnnnnnn

2 Response to Reviewer #2

R2.O1: Below, we address the assumptions in Lemma 4 and Section 3 and their implications for our theoretical analysis. In the revised manuscript, we will clarify these assumptions and detail how overlap relates to the probability bounds. (1) Spatial point distributions can typically be modeled by a point process. Given a fixed number of points in a defined region (e.g., DiskANN's setup), a common Poisson point process yields a uniform conditional distribution, as noted in "Graph-based Nearest Neighbor Search" (ICML 2020): "A Poisson point process on S^d with n points simplifies proofs, with uniform conditioning." Uniformity persists across pruning steps, each a conditional restriction of the initial setup. (2) Any distribution can be transformed into a uniform one via an optimal transport map, and vice versa. Using optimal transport theory, we can sample any distribution and construct a mapping to a uniform distribution, allowing us to apply our probability bounds (e.g., pruning probabilities) derived under uniformity. (3) The uniform distribution assumption is well-established, as in [11], where it underpins the theoretical analysis of nearest neighbor search. Finally, addressing the 'overlapping area' assumption in the second paragraph of Section 3, our proof relies primarily on a lower bound for the one-step pruning probability. We define this probability as $p = V(\text{pruning region})/V(\text{point distribution region})$. To estimate it, we complete the point distribution region into a larger ring (adding volume v_1) and add the overlapping area into the pruning region to form a complete spherical cap (adding volume v_2). Since $v_1 > v_2$, by the inequality $(b+c)/(a+c) > b/a$, $p > (V(\text{pruning region}) + v_2)/(V(\text{point distribution region}) + v_1)$. In Lemma 4, we compute this completed volume ratio, yielding a lower bound of $1/3$, so the true p should be larger than $1/3$. Our subsequent derivations (using concentration inequalities and Wormald's theorem) depend on this lower bound, rendering the overlapping area's sparsity a minor factor.

3 Response to Reviewer #3

R3.O1.1&M7: Below, we compare our approach with the references—[1] ICML 2020, [2] arXiv:2303.06210v1, and [3] arXiv:2310.19126v1—explaining why our results differ and addressing the characterization of related works. We will refine Section 1 in the revision to discuss these works. Our paper assumes a uniform distribution over a ball, whereas [1] and [2] assume it over a sphere, and [3] makes no distributional assumption. Our paper and [3] focus on SNG-based deterministic pruning vs. random and threshold-based methods [1][2]. Both our work and [1] and [2] focus on average-case complexity, while [3] prioritizes worst-case guarantees over dataset scaling. Details: [1] uses a threshold-based method, unlike our SNG-based pruning, assumes a uniform distribution on a sphere, and analyzes general graph indices via monotonic search, while we model RobustPrune as a stochastic process, deriving $O(\log n)$ search complexity. [2] assumes uniform points on S^{d-1}, examines three ANN edge-construction strategies distinct from SNG, and proves Greedy Search converges with near-1 probability, while our SNG ensures guaranteed convergence. [3] analyzes DiskANN's strategy but focuses on worst-case complexity, avoiding distributional assumptions.

R3.O1.2: Our results primarily assume a uniform distribution of data points, which we justify and extend to broader cases. (1) Using optimal transport, any distribution can be mapped to a uniform one while preserving key structural properties, allowing our pruning probability bounds to apply to diverse real-world distributions. (2) For spatial point distributions, as in DiskANN's setup, a common Poisson point process yields a uniform conditional distribution. (3) The uniform distribution assumption is well-established, as in [11], where it underpins theoretical analysis of nearest neighbor search. While real-world datasets may not always be uniform, our framework adapts to non-uniform cases through these transformations, with further details to be elaborated in the revision.

R3.O1.3: (1) The current experiments demonstrate how recall varies with different R values (for each α) in Figures 6-8. While this is a known result for DiskANN, our intent was to compare the optimal R within each group against other R values to highlight its ability to balance recall and latency. We will revise the manuscript to include a direct comparison between the optimal R^* of analytical derivation (obtained via our expression) and the R from binary search, reporting runtimes to quantify efficiency gains. (2) Regarding the x-axes in Figures 6-8 not starting at zero, we apologize for this oversight. In the revision, we will update these figures to ensure the x-axis (latency) begins at zero and explicitly state the latency reduction in the text.

R3.O1.4: (1) The RobustPrune algorithm includes parameters, such as R, for the parameter R tuning part; we reduce the complexity from logarithmic iterations to a single iteration, but for the graph construction, there are multiple iterations, thus each node has multiple neighbors initially. (2) The whole space can be separated into three parts: $\alpha = 1$ (a hyperplane), $\alpha > 1$, and $\alpha < 1$ (curved surfaces). While DiskANN and RobustPrune typically specify $\alpha \geq 1$, our analysis in Section 4 extends to $\alpha \in (0, 1)$ to comprehensively cover all scenarios. (3) We'll merge these analyses into Section 3. Thank you for the suggestion.

R3.O2: We apologize for the writing issues. In the revision, we will: (1) Proofread the text to correct typos (e.g., "demenstrated" to "demonstrated" on PAGE 1) and grammatical errors, with professional editing assistance; (2) Define mathematical symbols upon first use, such as $I(t) = |P \setminus V|$, the number of checked points at iteration t of Algorithm 1; (3) Rewrite key sections (e.g., Section 3) using clearer, concise language to enhance readability.

R3.M1-M12: We apologize for various minor issues and appreciate the detailed feedback. We've corrected $N_{out}(p)$ as a set with Line 7 as $|N_{out}(p)| = R$, clarified that $d(p, p') > \alpha \cdot d(p^*, p')$ and $\alpha \cdot d(p^*, p') \leq d(p, p')$ are equivalent in high dimensions due to zero-measure equality, and extended $\alpha \in (0, 1)$ beyond DiskANN's $\alpha \geq 1$ for completeness. We'll define $I(t) = |P \setminus V|$ in Section 2.2 as the count of unchecked points, explain RobustPrune's multi-round pruning for $\alpha = 1$ (foundation for Lemma 4's bound) in merged Section 3, and fix the typo "demenstrated" to "demonstrated." Section 2 assumes uniform points in a ball of radius R_0 (distinct from R), and we'll add a notation table. We'll unify $d(p, p')$ as $dist(p, p')$ to avoid confusion with $\mathbb{R}^d$. For $\alpha = 1$, Figure 2's bisector has intersections A, B; for $\alpha < 1$, Figure 4's curved boundary shifts to A_1, B_1, with dark blue as pruned and light blue as unpruned—captions will be enhanced. We extend $\alpha > 1$ consistently with Lemma 4's bound and broaden RobustPrune's framework to $\alpha \in (0, 1)$ in Section 3.