We thank the reviewers for their feedback. We are glad that they found the problem of combining classical planning with Reinforcement Learning important (R3), our experimental results and ablations to be compelling (R1,R5), and our method to be sound (R1,R3). We are also pleased that the reviewers clearly identified the main contribution of our work – a general procedure for automatically pruning nodes in a graph over observations called Two-Way Consistency (TWC) (R1,R3,R5). We address the reviewers' feedback below and will incorporate all of it.

Shared Feedback (R1,R3,R5)

Comparison with SPTM (R3,R5) *"[is] original SPTM ... better ... because of the subsampled observations?"* (R3) *"SPTM originally uses human demonstrations"* (R5) The performance difference is not due to subsampling, but because the original SPTM paper uses human demonstrations. We follow the more general and more difficult problem statement of SoRB: long-horizon planning without demonstrations.

More test-time maps (R1,R3) *"evaluate ... on a single map of ViZDoom and SafetyGym."* (R1) *"In SPTM... there was a training ... validation ... and test environments."* (R3). We used SoRB's problem statement: an exploration phase followed by a deployment phase in the same maze. SPTM can only generalize to new mazes with an expert walkthrough of the new maze, whereas we study goal generalization in the same maze by sampling goals and starting points randomly with no demonstrations (exactly as in SoRB). Furthermore, we test in three diverse environments (PointEnv, VizDoom, SafetyGym), which is more than SoRB (two envs) and SPTM (one env).

Hyperparameters (R3,R5) *"Substantial number of hyperparameters [are] tuned for each environment..."* (R3) We used the exact architecture / hyperparameters of SoRB & SPTM. We only tune the thresholds τ_α (TWC) & τ_p (perceptual).

For Reviewer #1 (R1)

Complex graphs. *"It's impossible to use an optimal search method to plan on [more complex] graphs. The proposed algorithm may have its limitation when extended to more challenging tasks."* In PointEnv, we found that only 0.0038% of SGM's time to choose an action was taken up by graph search. We expect the benefits of SGM will only increase as graphs become more complex. SGM uses Dijkstra's algorithm, with $O(|\mathcal{E}| \log |\mathcal{V}|)$ complexity. With Line 12 of Alg 1 (k-NN edge filtering), $|\mathcal{E}|$ is a constant, and we can control the size of $|\mathcal{V}|$ by tuning τ_α.

For Reviewer #3 (R3)

Assumptions of theorem. *"The main theorem in the paper makes strong assumptions on the Q function."* Our main theorem makes no assumptions on the Q function. Instead, it bounds the additional error when using TWC to turn a dense graph into a sparse graph. The bound holds regardless of the original error in the dense graph.

Better optimality bound. *"[A] theorem showing optimality bounds of [TWC]... given an existing optimality bound on Q would be more useful."* Thank you for the excellent suggestion. We took your feedback and proved that, given an existing optimality bound on Q with error ϵ, TWC on plans of path length k has error at most $k\epsilon + 2k\tau_\alpha$.

Perceptual similarity. *"The importance of ... perceptual similarity is not addressed... it seems unlikely that its faster computationally to compute a perceptual embedding"* Perceptual consistency is substantially faster as it computes $|\mathcal{V}|$ embeddings and their pairwise L2 distances (a cheap vectorized computation) whereas TWC requires $|\mathcal{V}|^2$ queries to the neural distance function. In Table 3, we run an ablation with perceptual consistency. Perceptual consistency alone achieves 77.0% success whereas the full method achieves 92.9% success.

States vs observations. *"The paper makes no distinction between state and observations."* We'll be sure to clarify. We demonstrate SGM in environments with access to state as well as with access to visual observations only. *"The fact that high-level actions are also observations... limits the class of tasks that can be addressed."* SGM is no more limited than prior graphical memory work. An observation can precisely specify a waypoint or goal as long as it includes features important for the task. Even if the observation is more specific than desired, SGM can solve many tasks because the distance function $d(\cdot, \cdot)$ can be changed, e.g. to identify such features and ignore task-irrelevant details (L185-187).

Related work. *"previous works... [1, 2, 3]... are not addressed in the paper."* Thank you, we will add these to the related work! Ref. 1 assumes access to a 360°camera and a pose sensor whereas we do not. In Ref. 2, the nodes in the graph are manually specified by humans whereas we automatically abstract nodes from the data. Ref. 3 has no theoretical guarantees, requires trajectories rather than unordered observations, and uses human demonstrations.

Text suggestions. *"Not all results have confidence intervals included. Details of seeds for different runs are not described."* Thank you for the feedback. We will add the missing confidence intervals and seed info to the text.

For Reviewer #5 (R5)

Other methods. *"SGM is not compared with... MPC, policy optimization and Q-learning"*. We found that these methods achieved near 0% success rate (a finding known from SoRB). We'll clarify this in the text.