Efficient Search for Correct and Useful Topological Maps

Collin Johnson and Benjamin Kuipers

***Abstract*— We present an algorithm for probabilistic topological mapping that heuristically searches a tree of map hypotheses to provide a usable topological map hypothesis online, while still guaranteeing the correct map can always be found. Our algorithm annotates each leaf of the tree with a posterior probability. When a new place is encountered, we expand hypotheses based on their posterior probability, which means only the most probable hypotheses are expanded. By focusing on the most probable hypotheses, we dramatically reduce the number of hypotheses evaluated allowing real-time operation. Additionally, our approach never prunes consistent hypotheses from the tree, which means the correct hypothesis always exists within the tree.**

I. Introduction and Related Work

Topological mapping is the process of discovering the connectivity of places in an environment, of finding the underlying decision structure. Topological mapping abstracts the continuous experience of the robot into a discrete sequence of place events. Each place event corresponds to the robot's arrival at or departure from a distinctive state, which might be a hallway intersection or visually unique landmark. A topological map represents the world as places, where qualitatively distinct decisions are presented to the robot, and paths, which are simple connections between places. The topological map abstraction provides a useful representation for planning by factoring the problem into a simple graph search through large-scale space and metric motion planning in small-scale space. Topological maps also scale well, as large portions of the metric environment – the paths between places – can be ignored or represented as simply the displacement between the places on either end of the path.

In laser-based maps, the location of decision points, typically hallway intersections in indoor environments, often appear very similar, making it difficult to assert that two places are definitively the same or different. Therefore, each place encountered as a robot explores an environment could generate a number of new topological map hypotheses. The number of hypotheses can grow exponentially and thus become intractably large. Analytic constraints, like planarity [1], can reduce the number of map hypotheses, but still leave too many viable hypotheses to yield a feasible algorithm for large environments.

This work has taken place in the Intelligent Robotics Lab in the Computer Science and Engineering Division of the University of Michigan. Research of the Intelligent Robotics lab is supported in part by grants from the National Science Foundation (CPS-0931474 and IIS-1111494), and from the TEMA-Toyota Technical Center.

C. Johnson is with the Computer Science and Engineering, University of Michigan, Ann Arbor, MI 48105, USA collinej@umich.edu

B. Kuipers is with Faculty of Computer Science and Engineering, University of Michigan, Ann Arbor, MI 48105, USA kuipers@umich.edu

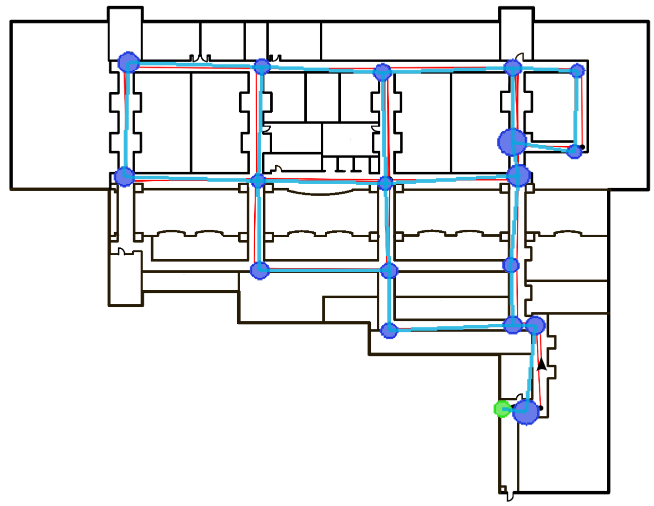

Fig. 1. Topological map created by our algorithm of the third floor of the EECS building at the University of Michigan.

Topological mapping algorithms build graph-like maps that identify the connectivity of places or landmarks within a robot's environment. The dramatic reduction in data achieved by the topological map abstraction comes at the cost of increased perceptual aliasing, different places appearing the same, and image variability, the same place appearing different at different times [2]. The explicit modeling of loop closures in topological maps makes false positives or false negatives especially problematic. A data association error in a metrical map will usually lead to small errors but leave the map consistent with the environment. Falsely asserting two different places to be the same in a topological map, however, introduces an unrecoverable inconsistency into the map. Most topological mapping algorithms use a combination of two techniques to handle perceptual aliasing: an exploration strategy to confirm that a particular topological map is correct, and incorporating metric information like odometry and vision-based appearance models to increase distinctiveness of places and paths.

Early work by Kuipers and Byun [3] decided among a set of possible maps by attempting to navigate a route planned within the map. If the robot was unable to follow the route, the map hypothesis was discarded. Choset and Nagatani [4] use a similar strategy to determine whether a new place was previously visited. They create a set of possible places at which the robot could have arrived, select a path to follow, and rule out inconsistent matches based upon the next visited place. They continue this process until a single match remains. Dudek et al. [5] describe an exploration strategy to find the correct topological map by maintaining a set of distinct markers. The robot leaves these markers at places with unexplored paths. When the robot traverses a new path, it can then determine if the place at the end was previously visited or is new. In later work, Marinakis and Dudek [6] discuss the tradeoff an exploration strategy must make between full exploration of the environment and keeping the number of map hypotheses tractable. These exploration-based approaches to finding the correct map are ultimately unsatisfactory because they depend on complete control of the robot's actions, which robots that serve humans (such as a robotic wheelchair) or that interact with other agents likely will not have, and they may require a large number of traversals through the environment, which is again impractical on a robot operating in the human environment. In this paper, we assume the robot has no control over the order in which places in the environment are explored.

In recent years, probabilistic algorithms for topological mapping have emerged to focus the search for the correct map on the most probable hypotheses. Ranganathan and Dellaert [7] use a Rao-Blackwellized particle filter where each sample represents a topological map hypothesis. The algorithm maintains the N most probable map hypotheses. If N is small enough, the algorithm is able to run in real-time and, based on the presented results, discover the correct map. However, the particle filtering approach inherently contains the risk of discarding or never generating the correct map. Specifically, if an erroneous measurement causes the correct hypothesis to be removed from the sample set, the correct map can never be found because the history of map hypotheses is not maintained.

An alternative probabilistic approach is presented by Tully et al. in [8] based on the tree of maps described by Dudek et. al in [9]. The tree of maps is a data structure where each node represents a consistent topological map hypothesis, and the leaves of the tree are the set of map hypotheses consistent with all place events. When a new place event occurs, a new set of hypotheses are generated for each leaf by asserting different possible loop closures. Thus, the depth of the tree is equal to the number of place events that have occurred. In [8], the nodes of the tree are annotated with a posterior probability using a recursive Bayes formulation. To avoid the exponential growth of the tree, hypotheses with a low measurement likelihood or posterior probability are pruned after each update. This pruning step still leaves a substantial number of hypotheses in the tree, making real-time operation unlikely. Furthermore, pruning the tree means no guarantee can be made that the correct map will be found.

We propose a new algorithm for topological mapping that combines a probabilistic tree of maps similar to that used in [8] with a heuristic search that focuses tree expansion on the most likely hypotheses. We never prune consistent hypotheses from the tree, ensuring the correct map can always be found.

Our algorithm performs a heuristic search through the tree of maps. We annotate each leaf in the tree with a posterior probability. When a new place event occurs, map hypotheses are expanded in order of decreasing posterior probability. For each update, we only expand a subset of all leaf hypotheses because only the most probable hypotheses are expanded when a new place is visited. Therefore, as the robot explores the environment, the leaves of the tree come to represent different points in the sequence of events because less probable hypotheses will have not been expanded. To approximate the posterior of these hypotheses from previous events, we use a heuristic based on the most probable map hypothesis at the current depth of the tree. Measurements and actions from each event are saved, so a hypothesis from a previous event can be expanded when its estimated likelihood is the maximum among all hypotheses.

Our topological map is a layer in the Hybrid Spatial Semantic Hierarchy (HSSH) [10], which provides us with a detailed local perceptual map (LPM), a metrical map, for each place. The LPM defines the physical extent of each topological place. We combine the physical extent of a place with an optimized global layout of the environment as part of our calculation of a map's likelihood.

Lazy evaluation of a hypothesis tree using map likelihoods has been previously applied in the context of robotic mapping. Hähnel et al. [11] proposed a lazy evaluation approach to determining data associations in a feature-based metric map. Similar to our heuristic search for loop closures, they maintain a tree of data association decisions for all time steps. Each node is labeled with the log-likelihood of the measurements given the data association. The node with the highest log-likelihood is considered until the maximum log-likelihood node is a leaf in the tree. Our approach differs by using a heuristic to estimate the log-likelihood of a node at a future time. By doing so, our tree search potentially considers fewer hypotheses and contains a smaller set of leaf nodes.

By performing lazy evaluation of the map hypotheses in the tree of maps, our algorithm avoids the need to prune hypotheses to maintain computational feasibility, unlike [8]. Furthermore, our heuristic search focuses expansion of the tree on the most likely hypotheses, allowing a small number of hypotheses to be evaluated for each event to achieve online updates. Thus, our algorithm operates online, while ensuring the correct map remains in the search space and can be found, which, to the best of our knowledge, no existing topological mapping algorithm achieves.

II. PROBABILISTIC TREE OF MAPS

The space of possible topological maps grows super-exponentially [7] with the number of places visited by the robot. In order to handle the exponential growth, we formulate a probabilistic tree of maps where the measurement likelihood is calculated for each leaf in the tree. Each depth of the tree corresponds to a new place event p_k. Using the likelihoods for the leaves, we perform a heuristic search through the probabilistic tree of maps to focus computation on the most likely portions of the hypothesis space.

The probabilistic tree of maps, H_k, is a tree of height k where the nodes represent topological map hypotheses. The height of the tree is equal to the number of place events that have occurred. We describe a map hypothesis symbolically as M_i^n, where the subscript i indicates the last place event included in the hypothesis, and thus its depth in H_k. A posterior probability, $l_{M_i^n} = p(Z^i|M_i^n, U^i)$, is calculated for each map hypothesis. When generating new map hypotheses, we consider only the set of leaf hypotheses, $L_k \subseteq H_k$. For those $M_i^n \in L_k$ with $i < k$, we use a heuristic to approximate the posterior, $\tilde{l}_{M_k^n}$, at k.

A. Symbol Definitions

$U^k = u_{1:k}$:
: The sequence of actions.

$Z^k = z_{0:k}$:
: The sequence of observations.

$z_k = \langle m_k, s_k, \lambda_{k-1,k} \rangle$:
: The observations at event k.

m_k: The LPM of the observed place at event k.

s_k: The local topology of the place observed at event k.

$M^n = \langle P^n, T^n, \chi^n \rangle$:
: A map hypothesis in the tree of maps.

$\langle p, m_p, s_p \rangle : p \in P^n$:
: The set of places in a map.

T^n: The topological map for M^n.

χ^n: The planar embedding of T^n.

Λ_{obs}^n: The observed displacements between places in M^n.

$l_{M_i^n}$: The posterior for a map hypothesis M^n that includes events up to i.

B. Assumptions

We make the following assumptions in our calculations:

1) The places in the environment are static.
2) Place detection is deterministic. There are no false positives or false negatives in our place detector. While place detection is deterministic, place matching is imperfect.
3) The correct path is followed when leaving a place.

C. Map Hypothesis Likelihood

1) Local Topology, s: Given a series of observed local topologies, $s_{0:k}$, we calculate $p(s_{0:k}|P^n, U^k)$, the likelihood of observing the series of local topologies given the topology of the map. The local topology is assumed to be estimated correctly, so the likelihood for a single local topology is (1), where $s_{p_k}^n$ is the local topology of the place visited at event k in M^n. The likelihood of all local topology measurements (2) can be evaluated recursively because the local topology for a place does not change over time.

$$p(s_k|s_{p_k}^n, u_k) = \begin{cases} 1 & \text{if } s_k = s_{p_k}^n \\ 0 & \text{if } s_k \neq s_{p_k}^n \end{cases} \tag{1}$$

$$p(s_{0:k}|P^n, U^k) = \prod_{i=0}^{k} p(s_i|s_{p_i}^n, u_i) \tag{2}$$

2) Place Layout, χ: Given a topology T^n and a set of metric path transformations, Λ_{obs}^n, we calculate the maximum posterior place layout, χ_{MAP}^n. Each $\lambda_{ab} \in \Lambda_{obs}^n$ is a Gaussian distribution representing the transformation from the origin, $(0,0,0)_b$, of place b to the origin, $[(0,0,0)_b]_a$, of place a.

$$\mu_{\lambda_{ab}} = (\Delta x, \Delta y, \Delta\theta) \tag{3}$$

$$\Sigma_{\lambda_{ab}} = \begin{bmatrix} \sigma_{\Delta x}^2 & 0 & 0 \\ 0 & \sigma_{\Delta y}^2 & 0 \\ 0 & 0 & \sigma_{\Delta\theta}^2 \end{bmatrix} \tag{4}$$

The place layout χ minimizes the objective function in (5)

$$\begin{aligned} E_\chi &= (\Lambda_\chi - \Lambda_{obs}^n)^T \Sigma_{\Lambda_{obs}^n}^{-1} (\Lambda_\chi - \Lambda_{obs}^n) \\ &\propto -\log p(\chi|\Lambda_{obs}^n) \end{aligned} \tag{5}$$

Thus, we minimize the log-likelihood $-\log p(\chi|\Lambda_{obs}^n)$ to obtain (6).

$$\chi_{MAP}^n = \arg\max_\chi p(\chi|\Lambda_{obs}^n) \tag{6}$$

We then use this value $p(\chi_{MAP}|\Lambda_{obs}^n)$ as the likelihood of a given place layout.

3) Place Compatibility, p: Each place $p \in P^n$ is assumed to represent a distinct portion of the environment. Therefore, a map hypothesis in which places overlap is less likely than a map hypothesis with no overlap. Using χ^n, the area occupied by each LPM can be transformed into a single global reference frame, which allows for straightforward calculation of the overlap between two places.

For each $p \in P^n$, χ^n specifies the pose of the place center, χ_p. The associated LPM, m_p, can be transformed to be centered at χ_p, giving m_p^χ. The compatibility between two places is defined in (7) and is based on the ratio of the overlapping area between the places to the minimum of the place areas.

$$c(m_i, m_j) = \frac{\text{area}(\text{intersection}(m_i^\chi, m_j^\chi))}{\text{min}(\text{area}(m_i), \text{area}(m_j))} \tag{7}$$

Using (7), the overall place compatibility, $p(P^n|\chi^n)$, of a map with N_p distinct places is the product of the compatibility between each pair of places in M^n.

$$p(P^n|\chi^n) = \prod_{i=0}^{N_p} \prod_{j=i+1}^{N_p} exp(-\kappa c(m_i, m_j)) \tag{8}$$

D. Map Posterior

We use the same prior as [8], shown in (9), where N_{P^n} is the number of places in a map and k is the total number of place events, which favors map hypotheses with fewer places. The overall likelihood of a map hypothesis, $p(Z^k|M^n, U^k)$, factors into the three quantities described in Section II-C. We assume each measurement is independent. Therefore, the overall likelihood of a map hypothesis, (10), is the product of the individual measurement likelihoods. The full map posterior combines the likelihood terms from above with the map prior to yield (11).

$$p(M^n|U^k) \propto \exp(N_{P^n} \log k) \tag{9}$$

$$p(Z^k|M^n, U^k) = p(s_{0:k}|P^n, U^k)p(\chi_{MAP}|\Lambda^n_{obs}) \quad p(P^n|\chi^n_{MAP}) \tag{10}$$

$$p(M^n|Z^k, U^k) = \eta p(Z^k|M^n, U^k)p(M^n|U^k) \tag{11}$$

E. Posterior Heuristic

Our likelihood heuristic, hl_d, where d is the depth of the tree at which the heuristic applies, uses the minimum change in measurement likelihood from a parent hypothesis to its children. This quantity is equivalent to the maximum ratio of child likelihood to parent likelihood as shown in (12). The heuristic gives an estimate of how we expect the likelihood of a map hypothesis to change when incorporating the next place event.

$$hl_d = max(\frac{l_{M^n_d}}{l_{parent(M^n_d)}}) : M^n_d \in H_k) \tag{12}$$

Our prior heuristic, hp_d, where d is the depth of the tree at which the heuristic applies, uses the maximum values of the posterior among the hypotheses at that depth of the tree.

$$hp_d = max(p(M^n_d|U^k)) \tag{13}$$

Combining (12) and (13), we calculate the estimated posterior $\tilde{l}_{M^n}$ for a map hypothesis in (14).

$$\tilde{l}_{M^n_k} = l_{M^n_i} hp_d \prod_{d=i+1}^{k} hl_d \tag{14}$$

For those hypotheses with $i = k$, $\tilde{l}_{M^n_i} = l_{M^n_i}$.

III. Lazy Evaluation of Map Hypotheses

Our algorithm for searching through the probabilistic tree of maps, described in Section II, proceeds as follows.

When place event $k + 1$ occurs, map hypotheses are evaluated in descending order of $\tilde{l}_{M^n_i}$ using a priority queue. When expanding a hypothesis, M^n_i, place event $i + 1$ is used. Each child hypothesis is added to the priority queue and might be evaluated during the current update. Hypotheses are expanded until the queue is empty or at least N_{expand} hypotheses have been expanded and $max(l_{M^n_{k+1}}) > max(\tilde{l}_{M^n_k})$.

The key step in Algorithm 1 is $children \leftarrow Expand(h_{k-1}, e_k)$ described in Algorithm 2. When expanding a map hypothesis with a new event, all frontier places – those places with at least one unexplored path segment – are compared against the topology of the new place. If the topology of the places match and the entry path fragment of the new place is a frontier path fragment in the matched place, then a new map hypothesis is generated. The new hypothesis asserts a new loop closure between the robot's

Algorithm 1 LazyEvaluation(M)

Require: M is a valid hypothesis tree
  $L \leftarrow Leaves(M)$
  $Q \leftarrow PriorityQueue$
  for all $l \in L$ **do**
    $Push(l, Q)$
  end for
  $n \leftarrow 0$
  $maxP \leftarrow 0$
  while $HasNext(Q)$ **and** $n < N_{expand}$ **and** $maxP < Posterior(Top(Q))$ **do**
    $h = Top(q)$
    $children \leftarrow Expand(h)$
    for all $c \in children$ **do**
      if $Depth(c) < Height(M)$ **then**
        $Push(c, Q)$
      else if $Posterior(c) > maxP$ **then**
        $maxP \leftarrow Posterior(c)$
      end if
    end for
    $n \leftarrow n + 1$
  end while

Algorithm 2 Expand(h, e)

Require: h is a valid topological map hypothesis up to event $k - 1$.
Require: e is the k event in the event sequence.
  $Children \leftarrow List$
  $p \leftarrow h.LastPlaceVisited()$
  {If on a known path in this map hypothesis, a new place will not be added}
  {for this event. Only the position of the robot in the map changes.}
  if $h.OnKnownPath()$ **then**
    $child \leftarrow h.UpdatePosition(e)$
    $Children.Add(child)$
  else
    $Frontiers \leftarrow h.FrontierPlaces()$
    {If position was moving toward a frontier, then find all frontier places with compatible local topologies, S_f, and create a child hypothesis for each one.}
    for all $f \in Frontiers$ **do**
      if $S_f = S_e$ **and** $S_f.IsFrontier(S_e.Entry())$ **then**
        $child \leftarrow h.ConnectPlaces(p, f)$
        $Children.Add(c)$
      end if
    end for
  end if
  return $Children$

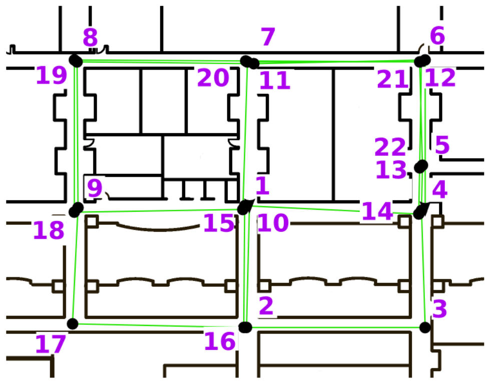

Fig. 2. Sequences of places visited by the robot in our first dataset.

previous location in the parent map hypothesis and the frontier place.

IV. Evaluation

We evaluate our lazy evaluation algorithm in indoor environments at the University of Michigan. We used our robotic wheelchair, Vulcan, equipped with a forward-facing laser rangefinder and an inertial measurement unit for our experiments. Vulcan is not equipped with wheel encoders. The robot was manually driven through the environment. Our algorithm was implemented as the global topological layer of the HSSH. Places are located at decision points in the environment. Decision points correspond to regions in the robot's local perceptual map containing two or more unaligned gateways, as detailed in [10].

Our first dataset, Figure 2 consists of an exploration through an environment with only T and plus intersections. We used this dataset to compare the growth of the tree of maps using our lazy evaluation approach with the fully expanded tree of maps. We ran our lazy evaluation algorithm with different settings for N_{expand}, the minimum number of hypotheses to expand for each event to further explore the possibilities for reducing the tree size. In all cases, the correct map had the highest posterior at the end of the dataset.

Figure 3 shows the growth in the number of map hypotheses versus the number of events. The behavior matches that of [7], [8], where a small number of map hypotheses eventually dominate the posterior distribution. While the overall behavior of each algorithm is similar, the performance differs in significant ways.

Our topological place representation uses the decision structure extracted from a metric representation of the place. This representation describes exactly the number of paths incident to a place, which limits the space of possible map hypotheses because a place with all paths connected to places will no longer be a candidate for a loop closure. Figure 4 shows how this representation limits the growth. When $N_{expand} = 0$, after event 17, only a single hypothesis is expanded for each event because the robot is moving only between known places.

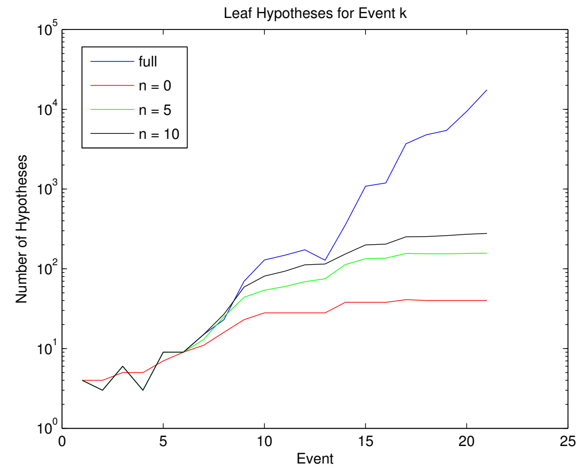

Fig. 3. The growth in the number of hypotheses as a function of N_{expand}. The full search grows exponentially, as expected. The growth of the lazily evaluated trees begins to slow as more events occur because the distribution of the map hypotheses becomes more peaked.

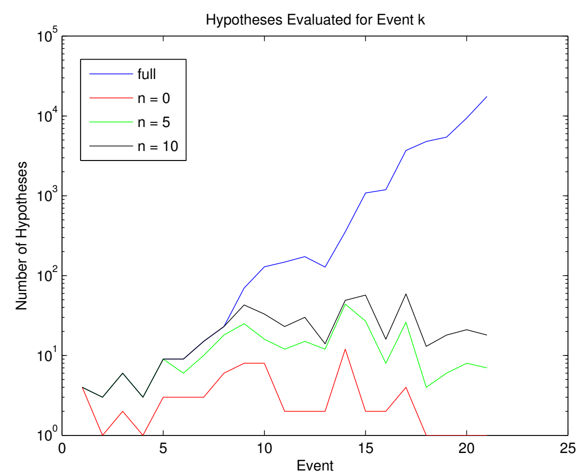

Fig. 4. The number of hypotheses evaluated for at each event. The full search expands every hypothesis for each event, so the number of evaluated hypotheses grows exponentially. With lazy evaluation, only a subset of the leaf nodes are expanded for each event, limiting the overall growth of the tree. $N_{expand} = 0$ corresponds to best-first greedy search and after event 17 expands only the most probable hypothesis, which corresponds to the correct map. Only one hypothesis is evaluated because the robot is moving within known portions of the map.

In contrast, the place representation used in [7] treats each place as a simple landmark with no information about the number of incident paths. As a result, each event generates a potentially larger number of hypotheses when compared with our representation. The dataset gathered in Figure 5 contains 36 place events. The average update time per event was 0.26s and the maximum update time was 1.4s. In comparison, the simulated dataset from [7], which also used no laser or visual appearance models and consists of 33 place events, took their algorithm an average of 4.2s per update.

Tully et al. [8] use a similar place representation to our own that is based on junction points in the Voronoi diagram.

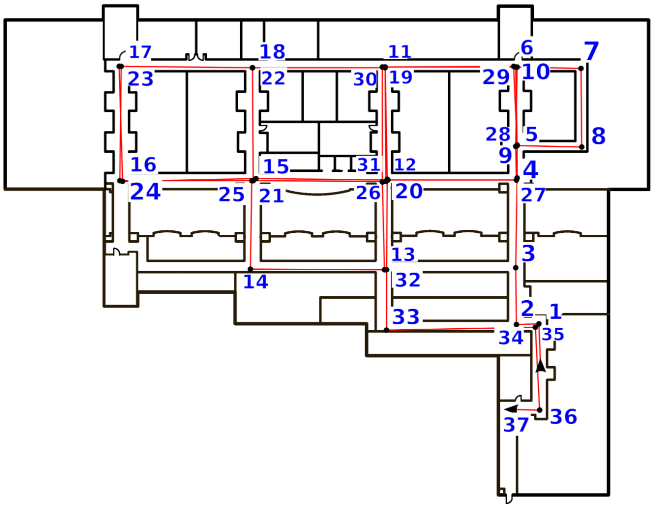

Fig. 5. Sequences of places visited by the robot in our second dataset.

They also perform their search on a tree of maps annotated with posterior probabilities. However, their algorithm actively prunes improbable hypotheses which means the correct hypotheses could be pruned from the tree, making the correct map unrecoverable.

We believe our approach will scale better than [8] for the following reasons. First, though hypotheses are pruned from the tree, their algorithm expands every leaf hypothesis for each update. As the map grows larger, the number of expansions will grow considerably. In comparison, we focus our search on a smaller number of hypotheses and expand the tree only as much as necessary. Second, we use an optimized place layout to evaluate the likelihood of a map hypothesis. When an incorrect match is made between two places, the place layout becomes globally inconsistent. Their approach uses only local path information between places. Therefore, an incorrect match between two places has less effect on the posterior of a map, which keeps the posterior higher for more map hypotheses. Evaluation of these assumptions on a large dataset is left for future work.

V. Conclusion

We have a demonstrated a probabilistic topological mapping algorithm that effectively reduces the search space of potential topological map hypotheses to allow topological mapping in real-time while never eliminating the correct map from the search space. Our algorithm takes a lazy evaluation approach by only expanding the most probable map hypotheses. By focusing our search on only the most probably hypotheses, we have shown the number of maps considered is reduced by at least an order of magnitude for our testing environment. The tree of maps is never pruned. Instead, our algorithm assumes leaves exist at different depths of the tree and handles this case explicitly. Never pruning the tree guarantees that the correct map can always be found, though extremely poor measurements could make the correct map very improbable. This problem is general to all mapping algorithms, though.

The performance of our algorithm depends on N_{expand}. For future work, we plan to investigate how we can leverage the tradeoffs presented by this parameter. N_{expand} effectively controls how greedy the search is. At early stages, more exploration through the tree is desired, but as time goes on, focusing the search on only the most probable hypotheses seems wise. Can we correlate a value of N_{expand} with the posterior distribution of map hypotheses?

Finally, the algorithm presented here makes no use of laser or visual appearance models. The use of appearance models should yield a much higher peak in the map posterior around the correct hypothesis. We plan to incorporate a visual appearance-based model like FAB-MAP [12] into our lazy evaluation algorithm.

References

[1] F. Savelli and B. Kuipers, “Loop-closing and planarity in topological map-building,” in *Intelligent Robots and Systems, 2004. (IROS 2004). Proceedings. 2004 IEEE/RSJ International Conference on*, vol. 2, sept.-2 oct. 2004, pp. 1511 – 1517 vol.2.

[2] B. Kuipers and P. Beeson, “Bootstrap learning for place recognition,” in *AAAI/IAAI*, 2002, pp. 174–180.

[3] B. Kuipers and Y.-T. Byun, “A robust, qualitative method for robot spatial learning,” in *AAAI*, 1988, pp. 774–779.

[4] H. Choset and K. Nagatani, “Topological simultaneous localization and mapping (slam): toward exact localization without explicit localization,” *IEEE Transactions on Robotics*, vol. 17, no. 2, pp. 125–137, 2001.

[5] G. Dudek, M. Jenkin, E. Milios, and D. Wilkes, “Robotic exploration as graph construction,” *Robotics and Automation, IEEE Transactions on*, vol. 7, no. 6, pp. 859 –865, dec 1991.

[6] D. Marinakis and G. Dudek, “Pure topological mapping in mobile robotics,” *IEEE Transactions on Robotics*, vol. 26, no. 6, pp. 1051–1064, 2010.

[7] A. Ranganathan and F. Dellaert, “Online probabilistic topological mapping,” *The International Journal of Robotics Research*, vol. 30, no. 6, pp. 755–771, 2011. [Online]. Available: http://ijr.sagepub.com/content/30/6/755.abstract

[8] S. Tully, G. Kantor, H. Choset, and F. Werner, “A multi-hypothesis topological slam approach for loop closing on edge-ordered graphs,” in *Intelligent Robots and Systems, 2009. IROS 2009. IEEE/RSJ International Conference on*, oct. 2009, pp. 4943 –4948.

[9] G. Dudek, P. Freedman, and S. Hadjres, “Using local information in a non-local way for mapping graph-like worlds,” in *IJCAI*, 1993, pp. 1639–1647.

[10] P. Beeson, J. Modayil, and B. Kuipers, “Factoring the mapping problem: Mobile robot map-building in the hybrid spatial semantic hierarchy,” *The International Journal of Robotics Research*, vol. 29, no. 4, pp. 428–459, 2010. [Online]. Available: http://ijr.sagepub.com/content/29/4/428.abstract

[11] D. Hahnel, S. Thrun, B. Wegbreit, and W. Burgard, “Towards lazy data association in slam.” in *ISRR’03*, 2003, pp. 421–431.

[12] M. Cummins and P. Newman, “Appearance-only slam at large scale with fab-map 2.0,” *The International Journal of Robotics Research*, vol. 30, no. 9, pp. 1100–1123, 2011. [Online]. Available: http://ijr.sagepub.com/content/30/9/1100.abstract