Using the Topological Skeleton for Scalable Global Metrical Map-Building

Joseph Modayil, Patrick Beeson, and Benjamin Kuipers

Intelligent Robotics Lab
Department of Computer Sciences
University of Texas at Austin
{modayil, pbeeson, kuipers}@cs.utexas.edu

***Abstract*— Most simultaneous localization and mapping (SLAM) approaches focus on purely metrical approaches to map-building. We present a method for computing the global metrical map that builds on the structure provided by a topological map. This allows us to factor the uncertainty in the map into *local metrical uncertainty* (which is handled well by existing SLAM methods), *global topological uncertainty* (which is handled well by recently developed topological map-learning methods), and *global metrical uncertainty* (which can be handled effectively once the other types of uncertainty are factored out). We believe that this method for building the global metrical map will be scalable to very large environments.**

I. Introduction

Many simultaneous localization and mapping (SLAM) methods have the goal of building a global metrical map of the environment within a single frame of reference [1]–[4]. Although these methods have improved dramatically over the past few years, they are still afflicted with the problem of cumulative distortions in the map and robot position, along with the related problem of data association and the closing of large loops. If a mapping algorithm uses a single maximum-likelihood map, it can be very difficult to recover from premature commitment to an erroneous map. On the other hand, attempting to maintain a distribution over the space of maps is expensive in space and time, and is vulnerable to approximation failures such as particle depletion [5].

A topological map abstracts the environment to a discrete graph in which nodes represent places and edges represent path segments linking the places. The topological map representation provides a very concise representation for individual maps, which is particularly important when considering multiple maps and when exploration requires discriminating among qualitatively distinct structural hypotheses.

A natural hybrid solution is to combine topological maps of the global environment with metrical maps of local spaces. Local metrical maps in separate local frames of reference avoid most problems of global metrical maps. In relatively simple regions such as individual place neighborhoods and individual path segments, incremental localization is highly reliable and current metrical SLAM methods perform extremely well. Errors are restricted to the current local map and do not accumulate over travel. In [6], we describe our hybrid approach that is based on the Spatial Semantic Hierarchy [7]. The Atlas and CTS frameworks for hybrid mapping [8], [9] take a similar approach.

In this paper, we show how to use an accurate hybrid topological map as a skeleton for building an accurate global metrical map. Our method solves the loop-closing problem in the topological map, where structurally distinct alternative maps can be concisely represented, and where an efficient topological version of Markov localization, along with topological axioms, can be used to refute incorrect maps [6], [10]–[12]. Metrical uncertainty is factored into independent components rather than being allowed to accumulate along the robot's travel trajectory. This approach to managing metrical uncertainty means our method will scale more effectively than previous methods to very large environments.

II. Overview and Background

The major innovation in this paper is the ability to factor the uncertainty in the map into *local metrical uncertainty* (which is handled well by existing SLAM methods), *global topological uncertainty* (which is handled well by existing topological map-learning methods), and *global metrical uncertainty* (which can be handled effectively once the other types of uncertainty are factored out).

Our approach builds on many existing methods.

To *represent* metrical maps, we use occupancy grids [13] rather than landmark-based maps [14] in order to use all of the information in the range sensor signal, instead of abstracting the environment to a pre-specified set of landmark types.

To *build* local metrical maps online, we use existing SLAM methods based on Markov localization and implemented using particle filters [15] to represent the pose action model distribution $P(x_t|u_t, x_{t-1})$ (used in Figure 1). Local metrical maps avoid difficult localization problems such as closing large loops, so we can use efficient methods that maintain a single maximum likelihood map hypothesis. We apply this local metrical mapping technique to maintain a "scrolling map" around the robot as it explores. The scrolling map can be used as an observer providing more accurate local state estimates for local control laws and for analyzing the local topology of place neighborhoods.

To *identify and characterize place neighborhoods*, we build Voronoi graph descriptions [16] of the scrolling map, looking for nodes in the Voronoi graph whose edges lead off the scrolling map. This trims small terminal edges from the Voronoi graph, and allows us to describe the "local topology" (e.g., L, T, +, etc.) of each place neighborhood [6].

To *build the large-scale topological map*, we use "local topologies" to build a tree of all possible topological maps consistent with exploration experience [6], pruning the tree when the maps are inconsistent with topological axioms [10], [11]. The axioms are expressed in a non-monotonic logic, so prioritized circumscription provides a simplicity-based preference ordering over remaining consistent maps [10]. In future work, we plan to incorporate weak metrical constraints to define a probability ordering over consistent maps. Path planning can either use the most preferred consistent map or, if the goal is differential diagnosis of the set of map hypotheses, all the consistent maps at the leaves of the tree of maps.

To *solve the loop closing problem for global metrical maps* we factor the problem into (a) the selection of the correct topological map, and (b) the construction of the global metrical map given the correct topological map. If each place has a reliably perceived label, even with substantial ambiguity (i.e., perceptual aliasing), continued exploration can reliably determine the correct topological map [17]–[19], except in pathologically symmetrical environments. This can be expressed as a topological version of Markov localization [12]. In realistic environments, with rich sensors, the label set is larger, and localization becomes faster and more reliable, making it possible to learn accurate place recognition from sensory images [12].

Given the correct topological map, we *segment exploration experience* at points selected each time the robot's trajectory passes through an individual place neighborhood. Each segment of experience then describes motion from one place neighborhood to another along a particular path segment. Existing state-of-the-art SLAM methods applied to the scrolling map can build a local metrical map of that path segment, and the trajectory through it, in the frame of reference of the initial place neighborhood.

Given exploration along individual path segments, the *global metrical map* can be found by 1) estimating displacements between pairs of connected places, 2) finding a global layout of places from the local displacements, 3) calculating the global trajectories along each path segment anchored by the global place locations, 4) using the new global trajectory as a highly accurate proposal distribution for the posterior probability distribution over poses and maps.

The remainder of the paper is organized as follows. Section III describes the theory behind the Bayesian network leading up to the proposal distribution. Section IV describes several alternative implementation strategies. We then lay out the preliminary strategy we used to produce the results in Section V. Section VI discusses related work, and finally we present some concluding remarks.

III. A Theory for Building the Global Metrical Map from an Annotated Topological Map

A. Metrical Mapping in a Single Frame of Reference

The task of building metrical maps is often described as finding the posterior (or maximum likelihood) over m and x in $P(x, m|z, u)$ with the following symbol definitions.

- t: Where $0 \leq t \leq N$ represents the timesteps of the robot's experience.
- $x = x_{0:N}$: The sequence of robot poses x_t at each timestep t.
- m: The set of map elements, which may be landmarks or occupancy grid cells.
- $z = z_{0:N}$: The sequence of observations z_t.
- $u = u_{1:N}$: The sequence of robot actions u_t between timesteps.

Figure 1 shows the standard graphical dynamic Bayesian network (DBN) model for Markov localization.

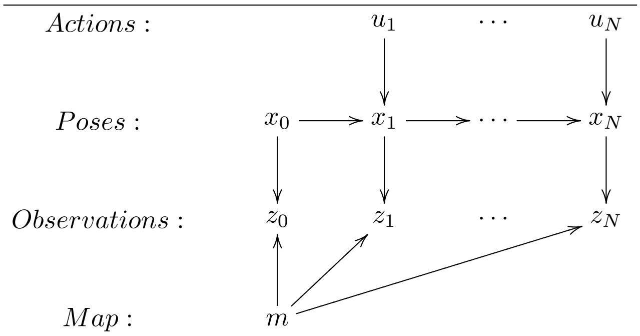

Fig. 1. Markov localization within a single frame of reference: $P(x_t|u_t, x_{t-1})$ and $P(z_t|x_t, m)$. Simultaneous localization and mapping (SLAM) combines localization with one of a number of mapping methods, for computing $P(m|x, z, u)$.

B. Global Topological Map and Local Metrical Maps

The topological map abstracts the environment by identifying a discrete set of *places*. We can define a local frame of reference at each place and build local metrical maps that model the place neighborhood. We divide the robot's experience into disjoint segments of travel from one place to another, by breaking the experience at distinguished time-points when the robot is in the neighborhood of a particular place.

Let τ represent the topological description of the environment created from observations. This description consists of the following entities.

- $\mathcal{P} = \{p_0, \ldots, p_l\}$: A set of symbols denoting places.
- t_i: Where $0 = t_0 < t_1 < \cdots < t_n = N$ is a subsequence of distinguished times when the robot is at a place. It is convenient to relabel the robot's state variables x, z, and u defining $x_{i,j} \equiv x_{t_i+j}$.
- $place(t_i) = p_j$: At time t_i the robot is at place p_j.
- $\tilde{m}_i$: The scrolling map that models the robot's surroundings between t_i and t_{i+1}. The map's

origin is defined at the robot's pose at time t_i.

$R(p)$: A frame of reference for place p.

O_p: The pose corresponding to (0,0,0) in $R(p)$.

$[y]_p$: The coordinates of the pose y in $R(p)$.

L_i: The robot's pose at time t_i in the coordinate frame $R(place(t_i))$.

Many of these concepts can be simply understood by examining Figure 2.

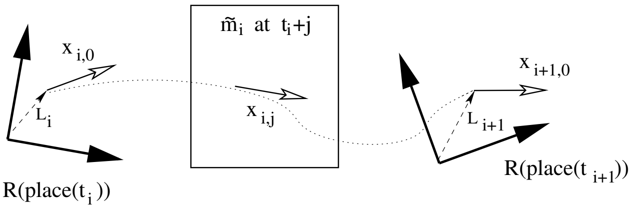

Fig. 2. The robot creates the local scrolling map $\tilde{m}_i$ when traveling between places. Each place has its own frame of reference.

Along with these observed quantities that comprise τ, we define some non-local metrical quantities that will be estimated.

λ_i: The location of $O_{place(t_i)}$ in the reference frame $R(place(t_{i-1}))$ using the experience from t_{i-1} to t_i.

$\chi_p \equiv [O_p]_m$: The pose of O_p in the global reference frame.

C. Building the Global Metrical Map

Our theory is based on the graphical model in Figure 3. Note that it contains instances of the simple graphical model (Figure 1) for the local metrical map associated with each travel between two adjacent places (at the top of the figure), plus one more for the global metrical map (at the bottom).

The joint probability of the pose history x and the map m can be decomposed as

$$P(x, m|z, u) = P(m|x, z, u) \cdot P(x|z, u) \tag{1}$$

which is just an application of the chain rule for probabilities. This decomposition is valuable since $P(m|x, z, u)$ (map-building given accurate localization) can be computed analytically and incrementally for popular map types [20]. This means that only $P(x|z, u)$ (localization) must be handled carefully.

We wish to find a *proposal* distribution over x which matches the posterior distribution well. First, to include the effect of possible environmental topologies on pose estimation, we integrate over the space of topologies (marginalization).

$$P(x|z, u) = \int P(x|\bar{\tau}, z, u) \cdot P(\bar{\tau}|z, u)\; d\bar{\tau} \tag{2}$$

Since the correct topology τ has been identified and is

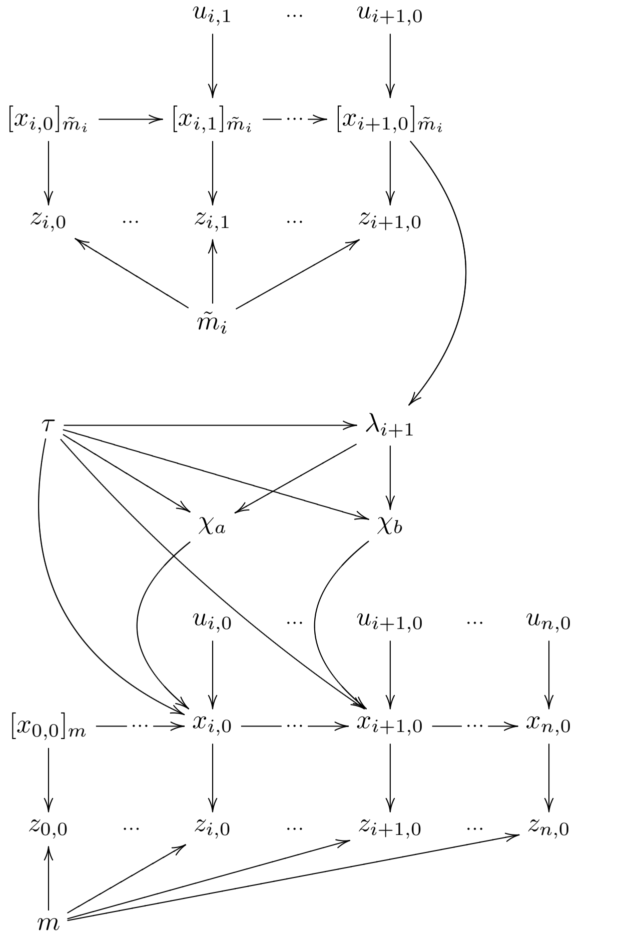

Fig. 3. Graphical model for building the global metrical map m using the topological map as a skeleton. This model contains instances of the model in Figure 1 for the local maps $\tilde{m}_i$ that scroll along path segments (top) and for the global map m (bottom). Each local metrical map $\tilde{m}_i$ is built from a segment of experience $(z_{i,0}, z_{i,1}, \ldots, z_{i+1,0}, u_{i,1}, \ldots, u_{i+1,0})$ that gives the probability distribution of the local displacements λ_{i+1} along the corresponding path segments. The set λ of path-segment displacements is used to estimate the place layout χ in the global reference frame. This provides a highly accurate proposal distribution for the final SLAM process that estimates the poses $x_{0:N}$ and the metrical map m in the global frame of reference.

given, only one topology τ has nonzero probability.[1]

$$P(x|z, u) = P(x|z, u, \tau) \tag{3}$$

Now we can marginalize over the poses of places χ and their estimated displacements λ.

$$P(x|z, u, \tau) = \iint P(x|\chi, \lambda, z, u, \tau) P(\chi|\lambda, z, u, \tau) \cdot P(\lambda|z, u, \tau)\; d\lambda d\chi \tag{4}$$

We can simplify the first term of the integral by noting $\lambda = \lambda(z, u)$.

$$P(x|\chi, \lambda, z, u, \tau) = P(x|\chi, z, u, \tau) \tag{5}$$

[1]This paper assumes that a unique topological map has been found before global metrical mapping begins. However, the two processes could be interleaved. During exploration, the number of plausible topologies would be finite (and probably quite small). Thus, the integral in equation (2) becomes a summation that could be feasible to evaluate even without a unique topological map.

Similarly in the second term, we can drop the dependence on z, u as these are incorporated in λ.

$$P(\chi|\lambda, z, u, \tau) = P(\chi|\lambda, \tau) \tag{6}$$

This means Equation 4 can now be rewritten as

$$P(x|z,u,\tau) = \int P(x|\chi, z, u, \tau) \int P(\chi|\lambda,\tau) P(\lambda|z,u,\tau)\, d\lambda d\chi \tag{7}$$

We divide equation (7) into simpler components represented by the following probability functions.

$$F(\lambda) = P(\lambda|z,u,\tau) \tag{8}$$

$$G(\chi) = \int P(\chi|\lambda,\tau) F(\lambda) d\lambda \tag{9}$$

$$H(x) = \int P(x|\chi,z,u,\tau) G(\chi) d\chi \tag{10}$$

Thus, we use the topological map τ to factor the localization term $P(x|z,u) = H(x)$ into three separate components: the displacements between places ($F(\lambda)$); the metrical layout of places in the global topological map ($G(\chi)$); and finally, the global metrical layout of the robot's pose trajectory ($H(x)$).

IV. Creating a Usable Mapping Algorithm

The decomposition above describes how topological information can be incorporated into the probability distribution for the global metrical map. We now discuss a variety of methods for implementing a practical algorithm, including our current implementation.

A. Building local metrical maps

Initially, our robot builds local metrical maps using incremental maximum likelihood occupancy grids [13], [15]. In contrast to the global maps that are commonly constructed using SLAM, we constrain the map size to prevent any problems with closing large loops. This does not constrain the robot's movement as the map scrolls with the robot in the center of the grid. Additionally, the local occupancy grid allows efficient local path planning to move around obstacles while traveling between places.

B. Computing τ

The local scrolling metrical map is used by the robot to determine when it is in a place. One requirement we impose on places is that the environment possesses sufficient structure to permit precise unambiguous localization in the local reference frame. Currently, we use Voronoi meet points to meet this criterion. The local map is also used to extract local topologies of the place neighborhood. This structure is used to build a unique, correct, global topological map [6].

In addition to the symbolic place associations in τ, there are reference frames for each place and pose estimates L_i. These are generated by saving images of $\tilde{m}_{i-1}$ at t_i, choosing an origin for the frame of reference, and relocalizing in the saved map on subsequent visits to the same place.

C. Estimating $F(\lambda)$

Given the topology τ, we can compute $F(\lambda)$. Note that each λ_i corresponds to a single experience of a path segment. Since closing large loops is not an issue when considering a single path segment, traditional SLAM methods may be employed to estimate $F(\lambda)$ by decoupling it into a set of independent probabilities.

$$F(\lambda) = \prod_{i=1}^{n} F_i(\lambda_i) \tag{11}$$

$$\text{where } F_i(\lambda_i) = P(\lambda_i | D_{i-1}, L_{i-1}, L_i) \tag{12}$$

$$\text{where } D_i = z_{i,0}, \ldots, z_{i,n_i}, u_{i,1}, \ldots, u_{i+1,0} \tag{13}$$

The distribution of λ_i is computed by composing three uncertain vectors: from $O_{place(t_{i-1})}$ to $x_{i-1,0}$; from $x_{i-1,0}$ to $x_{i,0}$; and finally from $x_{i,0}$ to $O_{place(t_i)}$.[2]

$$F_i(\lambda_i) = P(\lambda_i = \ (L_{i-1} \oplus [x_{i,0}]_{\tilde{m}_{i-1}} \oplus (\ominus L_i))) \tag{14}$$

A Rao-Blackwellized particle filter (RBPF) is an effective method for estimating $[x_{i,0}]_{\tilde{m}_{i-1}}$, the final robot pose at the end of an individual path segment, since particle depletion is not a major hazard along a single path segment.

Faster and simpler methods to estimate this distribution are to use just the action model (at the cost of larger variance) or incremental maximum likelihood mapping methods (higher bias) to estimate the final robot pose. All three methods result in a sample-based approximation of λ_i. For our current implementation, we use an incremental maximum likelihood method, and we model each F_i as a Gaussian.

D. Estimating $G(\chi)$

When $F(\lambda)$ is represented as a Gaussian, an Extended Kalman Filter is a simple way to approximate $G(\chi)$. The idea is to consider the χ_p as landmarks which are observed one at a time, by a robot taking actions λ_i between observing the landmarks p_{i-1} and p_i. This is essentially the classic approach of Smith, Self, and Cheeseman [21].

We can also evaluate $G(\chi)$ for an arbitrary distribution of $F(\lambda)$. Note that for a particular value of χ, $P(\chi|\lambda, \tau)$ will only be non-zero for a single value of λ, namely when each $\lambda_i = (\ominus\chi_{place(t_{i-1})}) \oplus \chi_{place(t_i)}$. Hence, $P(\chi|\lambda,\tau)$ is a Dirac delta function, and we get a simple expression for $G(\chi)$.

$$G(\chi) = \prod_{i=1}^{n} F_i((\ominus\chi_{place(t_{i-1})}) \oplus \chi_{place(t_i)}) \tag{15}$$

In our implementation, a greedy hill-climbing search quickly converges to a local maximum of $G(\chi)$. In practice, it appears that this local maximum is at or near the global maximum χ because the starting poses are provided by corrected odometry, given by the maximum likelihood scrolling map.

[2]We use the notation of the compounding operator [21]. Given two poses a and b, we write $[b]_a$ for the coordinates of b in the frame where a defines the origin. Then, $[c]_a = [b]_a \oplus [c]_b$. We also define an inverse operator, $[b]_a = \ominus[a]_b$.

In a more complicated environment, a sound estimation technique such as Metropolis-Hastings sampling [22] could be employed. Alternative Gaussian estimation techniques exist that make different tradeoffs among efficiency and soundness guarantees (CPE [23],SEIF [24], TJTF [25], and CTS [9]).

E. Estimating $H(x)$

Given $G(\chi)$, we can estimate $H(x)$ using an EKF. While this approach is sound, it requires more computation than necessary to produce a functional estimate of $H(x)$.

Given χ, we first estimate x_t, the robot's pose at each timestep. Each path experience D_i is handled independently. The robot's pose estimates x_t are generated in the following way:

1) Fix $x_{i-1,0} = \chi_{place(i-1)} \oplus L_{i-1}$ and $x_{i,0} = \chi_{place(i)} \oplus L_i$.
2) Find the rigid transformations from $[x_{i-1,0}]_{\tilde{m}_{i-1}}$ to $x_{i-1,0}$ and from $[x_{i,0}]_{\tilde{m}_{i-1}}$ to $x_{i,0}$
3) Interpolate between these endpoints in each dimension. This interpolation should be weighted by each pose's contribution to the total motion between the places.

This produces a new set of points $x_{i-1,k}$ along the path.

To create a distribution, we attach some uncertainty to each pose x_t. This is accomplished by an ad-hoc action model that creates an uncertainty that grows along a path segment starting at $x_{t-1,0}$ and then collapses at the next place $x_{t,0}$. The action model could be applied in both directions and combined in order to obtain maximum uncertainty midway through the path segment and less uncertainty at the endpoints.

F. Creating a map m

Given $H(x)$, we can create a detailed global metrical map m using the mode of $H(x)$. A common approach is to use a local optimization technique to align the pose positions with map estimates to converge upon a locally optimal map [26], [27].

A more principled approach is to run a Rao-Blackwellized particle-filtering algorithm, using $H(x)$ as the proposal: $p(x, m|z, u) = p(m|x, z, u) \cdot H(x)$. However, in Section V, we show that in practice the exact x_t values derived from the above interpolation adequately approximates the mode of the posterior.

V. Experiment

We tested our preliminary implementations of $F(\lambda)$, $G(\chi)$, and $H(x)$ in an office environment approximately 40 meters across (Figure 4). The environment contains 9 places and 6 paths (12 path segments) with 4 distinct local topologies. Considering the 14 travel actions that linked the 15 possible places, we actually close 4 loops of various sizes. The data for this experiment is available online [28].

The robot used in this experiment was an iRobot Magellan Pro with a SICK PLS laser range finder used for perception. Figure 5(a) shows the occupancy grid created when raw odometry is used. Along path segments, odometry is

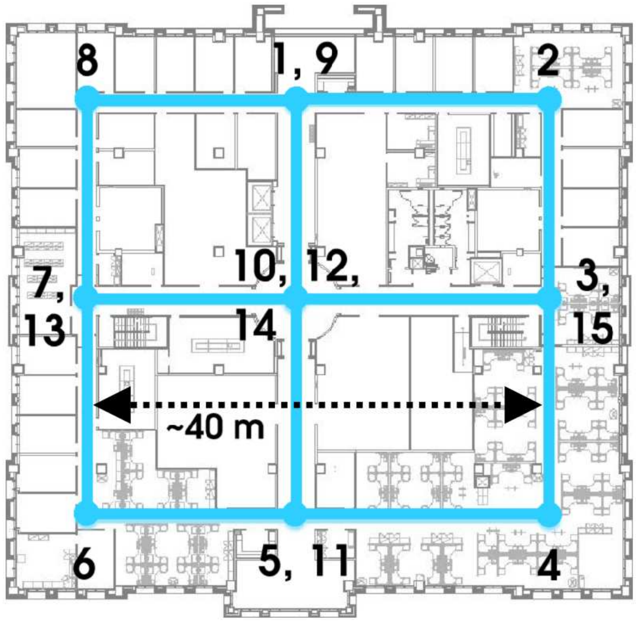

Fig. 4. CAD drawing of the test environment with multiple large-scale loops. The numbers represent the sequence of the robot's travel among the 9 places. This travel experience closes four large loops. Note that cubicles are not actually arranged exactly as shown here, and the hallways have glass walls into many offices.

reasonably accurate, but larger errors occur on turns. The corrected odometry, derived from the scrolling map, can be viewed in a single global frame of reference in Figure 5(b). Because the robot is localizing within a bounded map, a limited and fixed amount of computation is required at each step. The figure shows how localization in the local map reduces odometry error but does not prevent its accumulation over time and travel. In this moderate-sized and wheel-friendly environment, existing techniques are available to correct errors of the size shown here. However, our techniques are designed to apply to larger environments with weaker odometry and more accumulated error.

Local maps are saved on each occasion the robot enters a place, making for a total of 15 local maps shown in Figure 5(c). The topological mapper [6] uses the local topology at places to enumerate possible maps. After the 14 travel actions, the topological mapper finds 46 possible configurations of the environment that are consistent with the observed local topologies and the topological axioms [10], [11]. The circumscription policy finds a single minimal model, which is the correct topological map of the environment, shown in Figure 5(d). Our implementation can build the tree of models and determine the unique minimal map of this environment in $\sim$200 ms on the robot's Pentium III 450MHz processor. The topological mapper processes data incrementally so it can run concurrently with the exploration.

We use the corrected odometry from the scrolling map to determine $F(\lambda)$, which in turn leads to $G(\chi)$. Figure 6 shows the layout of the χ found by hill-climbing.

Using the χ we find through hill-climbing, we can generate a set of x_t that represent the robot's poses in

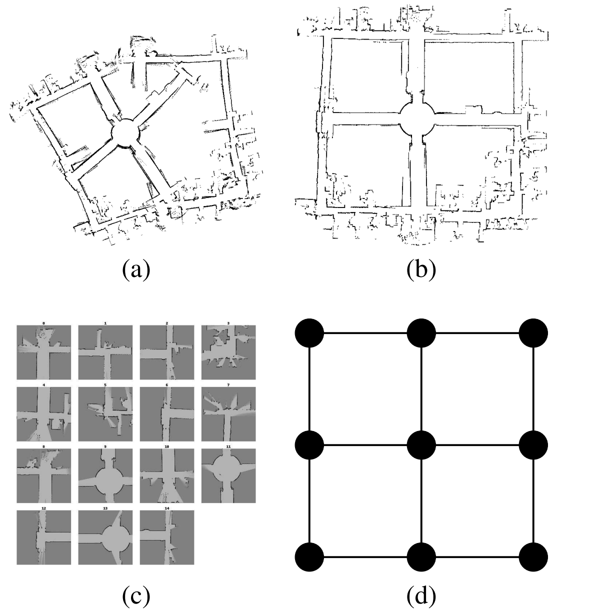

Fig. 5. (a) Occupancy grid created using the raw odometry sensor measurements to specify x_t. (b) Map built by the online, incremental maximum-likelihood method used by the scrolling map to resolve uncertainty locally. It is unable to resolve global uncertainty. (c) Local metrical maps are built of each place during exploration. (d) A search through topological maps consistent with the exploration finds the correct structure.

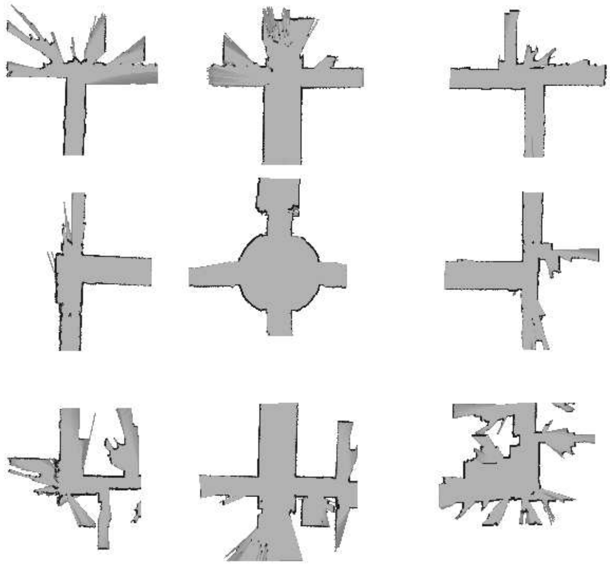

Fig. 6. Global layout χ of the topological map. Built after hill-climbing to find a good sample in $G(\chi)$. The local maps are the maps $\tilde{m}_i$ at topological places. Small orientation errors at the places are smoothed as shown in Figure 7 after the poses x_t are interpolated along each path segment between the place poses.

the global frame of reference. Building a map using laser readings, z_t, at each x_t (from $H(x)$), we get Figure 7. Clearly, using this as a proposal distribution for any state-of-the-art SLAM algorithm will yield better results than when using raw odometry or "corrected" odometry from maximum likelihood maps.

VI. Related Work

Most related work has been discussed previously, especially in Section II.

Atlas [8] and CTS [9] are the current mapping systems that are most similar to our work. They create local landmark-based maps with different local frames of reference and overlapping sets of landmarks. The topological map is implicit in the overlap between adjacent local maps. They essentially maintain a single topological map hypothesis, with a bias against false-positive place matches (and hence toward non-closing loops). By contrast, we exploit the conciseness of the topological map representation to maintain *all possible* topological map hypotheses, evaluating them for simplicity and (in future work) global metrical plausibility. The CTS algorithm for inferring the global relations among the local maps [9] is an option for computing the global place layout $G(\chi)$ in our framework.

Konolige [23] takes a hierarchical approach where only loop closures are considered to be places. This approach generates a map by maintaining Gaussian constraints between poses. Since constraints exist only between nearby poses, the graph of constraints is sparse. This sparsity is leveraged by collapsing sequences of constraints between places, estimating the place layout, and then stretching the intermediate poses between the places. This mirrors the construction of our functions F, G, and H quite closely. His work gains significant computational efficiency by restricting the error model to a Gaussian. The main limitation is that loop closures are made greedily and thus can fail.

Thrun, et al. [27] also take a hierarchical approach to building a global metrical map. They define a small set of indistinguishable "significant places" (by having a human operator press a button). Odometry and sensor data is first used to obtain an approximate layout of the coarse-grained map of "significant places" (essentially $G(\chi)$). This includes estimating the relative likelihood of different place-match hypotheses. After this, a fine-grained map is created using this skeleton as a starting point. However, their method implicitly computes a maximum-likelihood loop-closure hypothesis simultaneously with estimating $G(\chi)$, making the map vulnerable to the failure of that hypothesis. By contrast, we explicitly compute all possible topological maps, to be able to evaluate them for consistency with topological axioms before ordering the survivors according to preference criteria of simplicity and probability.

VII. Conclusions

Accurate global metrical maps can be created in a scalable way by factoring the robot's uncertainty into local metrical uncertainty (which can be handled by existing SLAM methods), global topological uncertainty (which can be handled by recently developed topological mapping methods), and global metrical uncertainty. Global metrical pose uncertainty can then be handled in three stages, first by using local metrical models to estimate the displacement between topological places, second by estimating the global layout of the places, and third by interpolating the poses along each path segment. This defines a very accurate

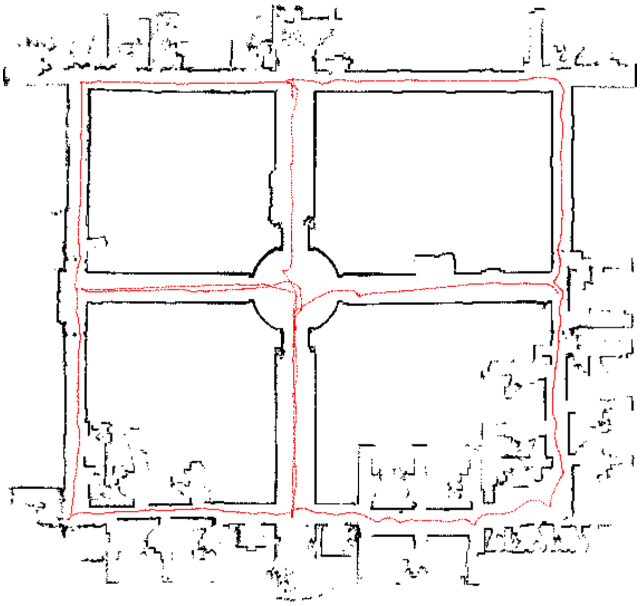

Fig. 7. Map built after hill-climbing to find a good sample in $G(\chi)$, then scaling corrected odometry (from Figure 5(b)) along each path segment. The estimated positions of the robot are shown as well.

proposal distribution $H(x)$ on the trajectories. Figure 7 shows that a map built on a trajectory drawn from this distribution is highly accurate.

While there are certainly many improvements to be made in the component metrical mapping technologies, we believe that the greatest research gains are to be found from improvements in topological mapping methods, including (a) better constraints and preference criteria for searching the space of all topological maps, and (b) the development of topological mapping methods appropriate to different types of environments.

ACKNOWLEDGEMENTS

This work has taken place in the Intelligent Robotics Lab at the Artificial Intelligence Laboratory, The University of Texas at Austin. Research of the Intelligent Robotics Lab is supported in part by the Texas Higher Education Coordinating Board, Advanced Technology Program (Grant 003658-0656-2001), and by an IBM Faculty Research Award.

REFERENCES

[1] J.-S. Gutmann and K. Konolige, "Incremental mapping of large cyclic environments," in *International Symposium on Computational Intelligence in Robotics and Automation (CIRA'99)*, 1999, pp. 318–325.

[2] M. Montemerlo, S. Thrun, D. Koller, and B. Wegbreit, "FastSLAM 2.0: an improved particle filtering algorithm for simultaneous localization and mapping that provably converges," in *Proc. 18th Int. Joint Conf. on Artificial Intelligence (IJCAI-03)*. Morgan-Kaufmann Publishers, 2003, pp. 1151–1156.

[3] A. Eliazar and R. Parr, "DP-SLAM: Fast, robust simultaneous localization and mapping without predetermined landmarks," in *Proc. 18th Int. Joint Conf. on Artificial Intelligence (IJCAI-03)*. Morgan Kaufmann, 2003, pp. 1135–1142.

[4] D. Hähnel, W. Burgard, D. Fox, and S. Thrun, "An efficient FastSLAM algorithm for generating maps of large-scale cyclic environments from raw laser range measurements," in *Proc. IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS-03)*, 2003.

[5] R. van der Merwe, A. Doucet, N. de Freitas, and E. Wan, "The unscented particle filter," Cambridge University Engineering Department, Tech. Rep. CUED/F-INFENG/TR 380, 2000.

[6] B. Kuipers, J. Modayil, P. Beeson, M. MacMahon, and F. Savelli, "Local metrical and global topological maps in the hybrid spatial semantic hierarchy," in *IEEE Int. Conf. on Robotics & Automation (ICRA-04)*, 2004, pp. 4845–4851.

[7] B. J. Kuipers, "The Spatial Semantic Hierarchy," *Artificial Intelligence*, vol. 119, pp. 191–233, 2000.

[8] M. Bosse, P. Newman, J. Leonard, M. Soika, W. Feiten, and S. Teller, "An Atlas framework for scalable mapping," in *IEEE International Conference on Robotics and Automation*, 2003.

[9] J. Leonard and P. Newman, "Consistent, convergent, and constant-time SLAM," in *Proc. 18th Int. Joint Conf. on Artificial Intelligence (IJCAI-03)*. Morgan Kaufmann, 2003, pp. 1143–1150.

[10] E. Remolina and B. Kuipers, "Towards a general theory of topological maps," *Artificial Intelligence*, vol. 152, pp. 47–104, 2004.

[11] F. Savelli and B. Kuipers, "Loop-closing and planarity in topological map-buiding," in *IEEE IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2004.

[12] B. Kuipers and P. Beeson, "Bootstrap learning for place recognition," in *Proc. 18th National Conf. on Artificial Intelligence (AAAI-2002)*. AAAI/MIT Press, 2002, pp. 174–180.

[13] H. P. Moravec, "Sensor fusion in certainty grids for mobile robots," *AI Magazine*, pp. 61–74, Summer 1988.

[14] J. J. Leonard and H. F. Durrant-Whyte, *Directed Sonar Sensing for Mobile Robot Navigation*. Boston: Kluwer Academic Publishers, 1992.

[15] S. Thrun, W. Burgard, and D. Fox, "A real-time algorithm for mobile robot mapping with applications to multi-robot and 3D mapping," in *IEEE International Conference on Robotics and Automation*. San Francisco, CA: IEEE, 2000.

[16] H. Choset and K. Nagatani, "Topological simultaneous localization and mapping (SLAM): toward exact localization without explicit localization," *IEEE Trans. on Robotics and Automation*, vol. 17, no. 2, pp. 125–137, April 2001.

[17] B. J. Kuipers and Y.-T. Byun, "A robot exploration and mapping strategy based on a semantic hierarchy of spatial representations," *Journal of Robotics and Autonomous Systems*, vol. 8, pp. 47–63, 1991.

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

[19] G. Dudek, P. Freedman, and S. Hadjres, "Using local information in a non-local way for mapping graph-like worlds," in *Proc. 13th Int. Joint Conf. on Artificial Intelligence (IJCAI-93)*, 1993, pp. 1639–1645.

[20] K. Murphy, "Bayesian map learning in dynamic environments," in *Neural Information Processing Systems (NIPS-99)*, 1999.

[21] R. Smith, M. Self, and P. Cheeseman, "Estimating uncertain spatial relationships in robotics," in *Autonomous robot vehicles*, I. J. Cox and G. T. Wilfong, Eds. New York: Kluwer Academic Publishers, 1990, pp. 167–193.

[22] D. J. C. MacKay, *Learning in graphical models*. MIT Press, 1998, ch. Introduction to Monte Carlo methods, pp. 175–204.

[23] K. Konolige, "Large-scale map-making," in *Proc. 19th National Conf. on Artificial Intelligence (AAAI-2004)*, 2004.

[24] S. Thrun, Y. Liu, D. Koller, A. Ng, Z. Ghahramani, and H. Durrant-Whyte, "Simultaneous localization and mapping with sparse extended information filters," April 2003, submitted for journal publication.

[25] M. A. Paskin, "Thin junction tree filters for simultaneous localization and mapping," in *Proc. 18th Int. Joint Conf. on Artificial Intelligence (IJCAI-03)*. Morgan Kaufmann, 2003, pp. 1157–1164.

[26] F. Lu and E. Milios, "Globally consistent range scan alignment for environment mapping," *Autonomous Robots*, vol. 4, pp. 333–349, 1997.

[27] S. Thrun, S. Gutmann, D. Fox, W. Burgard, and B. J. Kuipers, "Integrating topological and metric maps for mobile robot navigation: A statistical approach," in *Proc. 15th National Conf. on Artificial Intelligence (AAAI-98)*. AAAI/MIT Press, 1998, pp. 989–995.

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