# Local Metrical and Global Topological Maps in the Hybrid Spatial Semantic Hierarchy

Benjamin Kuipers, Joseph Modayil, Patrick Beeson, Matt MacMahon, and Francesco Savelli†

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

†ALCOR Group
Dipartimento di Informatica e Sistemistica
Università di Roma "La Sapienza"
Francesco.Savelli@dis.uniroma1.it

***Abstract*— Topological and metrical methods for representing spatial knowledge have complementary strengths. We present a hybrid extension to the Spatial Semantic Hierarchy that combines their strengths and avoids their weaknesses. Metrical SLAM methods are used to build local maps of *small-scale space* within the sensory horizon of the agent, while topological methods are used to represent the structure of *large-scale space*. We describe how a local perceptual map is analyzed to identify a local topology description and is abstracted to a topological place. The map-building method creates a set of topological map hypotheses that are consistent with travel experience. The set of maps is guaranteed under reasonable assumptions to include the correct map. We demonstrate the method on a real environment with multiple nested large-scale loops.**

***Index Terms*— map-building, topological maps, metrical maps, closing loops, structural ambiguity.**

## I. Introduction

Large-scale space is space that extends beyond the sensory horizon of an agent. When exploring and mapping large-scale space, a mobile robot must often decide whether it has returned to a previously visited location. For simple environments without large loops, this problem is easily solved. However, if a large loop may (or may not) have been closed, or if there are several different ways the loop may have closed, then there is a *structural ambiguity* between alternate map hypotheses (Figure 1).

We present a hybrid mapping method that extends the Spatial Semantic Hierarchy (SSH) [2] to combine the strengths of metrical mapping methods in local regions with the ability of topological mapping methods to resolve global structural ambiguity. We describe the results of a real-world mapping experiment in an environment with multiple nested large-scale loops.

### A. Metrical Mapping

Powerful probabilistic methods have been developed recently for simultaneous localization and mapping (SLAM) within a single frame of reference [4]. Many of these methods are accurate and reliable when doing online incremental localization within local neighborhoods. In local regions, many correspondence problems, such as the closing of large loops, can be excluded. The absence of large loops eliminates the problem of large-scale structural ambiguity in the metrical map. Sensing with sufficiently high frequency relative to local motion guarantees large overlap between successive sensory images and means that data association only has to be handled locally.

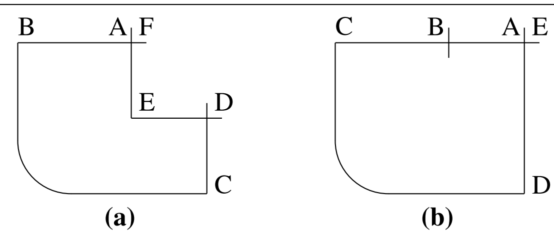



Fig. 1. **Structural Ambiguity Examples.** In these street-and-intersection environments, places are explored in alphabetical order and at least one street is long and/or convoluted enough to prevent accurate localization in a single frame of reference. In **(a)**, place D looks just like place A, so there is a structural ambiguity between a map in which $D = A$ and one in which $D \neq A$. After one more exploration step, $D = A$ is refuted by observing $E \neq B$. The structural ambiguity between $F = A$ and $F \neq A$ is resolved because $F = A$ implies the simplest consistent map [3]. In **(b)**, the structural ambiguity is between three map hypotheses: $E = A$, $E = B$ and $E \neq A \wedge E \neq B$. More exploration will resolve these alternatives similarly.

In recent work, metrical SLAM methods have been applied to larger and more complex environments, with impressive results. However, on closer examination, none of these have solved the problem of large-scale structural ambiguity. Some assume correct data association in order to simplify the mapping problem [5], [6]. Others accept false negative place matches, mislabeling a previously seen place as a new place [7]. This can eliminate the possibility of finding the correct map.

To avoid premature commitment to a possibly-incorrect maximum likelihood map hypothesis, some particle-filtering methods explicitly represent a distribution of belief over the space of possible maps [8]–[10]. Intractably large numbers

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.

M. MacMahon is supported by the Naval Research Laboratory.

Thanks to Philip Hendrix for work on visualization software.

Portions of this work have been presented in [1].

of particles may be required to avoid particle depletion when exploring large loops, resulting in the failure to have a particle in the distribution that adequately approximates the correct map.

### B. Topological Mapping

A topological map is a concise description of the large-scale structure of the environment [3], [11]. It compactly describes the environment as a collection of places linked by paths. When large loops in the environment result in structural ambiguity, a topological representation can concisely represent the loop attachment hypotheses using only a single map for each qualitatively distinct alternative.

An important strength of the topological representation is that the process of generating possible topological maps from experience and testing them for consistency can provide formal guarantees that the correct map is generated and never discarded [12], [13]. A logic-based theory of topological maps makes explicit the assumptions upon which those guarantees depend.

### C. A Hybrid Approach

Metrical and topological representations for space are very different in character, or more precisely, in ontology. The topological map describes the structure of large-scale space. It abstracts away the specific nature of sensory input and the specific methods used for matching sensory images when the topological map is created. The local perceptual map, on the other hand, exists precisely to capture the structure within the sensory horizon: small-scale space.

We use the term *local perceptual map* (LPM) for the metrical map resulting from applying an online SLAM method to a simple local region. During travel, a temporary LPM can be used as a *scrolling map* of the robot's immediate surround, for local motion planning and obstacle avoidance. Our current implementation of the LPM is an occupancy grid created using a particle filter method [14], but any accurate and reliable local metrical mapping algorithm will suffice.

To create a hybrid mapping method, using a metrical representation for small-scale space and a topological representation for large-scale space, we must show how processes operating on the LPM can create the objects necessary for building maps of large-scale space.

The hybrid SSH has several advantages that aid online map-building. First, local metrical motion planning and obstacle avoidance can take place within the LPM. Second, metrical localization can be done quickly after entering a place neighborhood, rather than requiring physical hill-climbing to a distinctive pose. If resources needed for metrical localization become unavailable, navigation can fall back on hill-climbing as in the basic SSH. Third, structural ambiguity due to large loops or false positive place matches can be represented compactly by a set of alternative topological maps, some of which may be discarded by future observations.

We demonstrate our hybrid SSH implementation exploring an office environment with multiple nested large loops and perceptual aliasing among place neighborhoods. During map-building, we deliberately refrain from using odometry to link frames of reference of adjacent place neighborhoods, in order to show that these methods will scale up to more demanding environments where odometry is unreliable. We describe the creation of LPMs, models of their local topology, the tree of alternative global topological structures, and methods for discriminating among alternate topological maps.

## II. The Spatial Semantic Hierarchy

The Spatial Semantic Hierarchy (SSH) [2] represents knowledge of large-scale space with four distinct representations: 1) control laws for reliable motion among *distinctive states* (dstates) $x_i$; 2) causal state-action-state *schemas* $\langle x,a,x' \rangle$ and relations $view(x,v)$ between a state and its observable view, abstracting the continuous world to a deterministic finite automaton; 3) a topological model consisting of places, paths, and regions explaining how the distinctive states are linked by turn and travel actions; 4) local metrical information about the magnitudes of actions, the lengths of path segments, and the directions of paths at place neighborhoods.

The Spatial Semantic Hierarchy factors spatial uncertainty into distinct components, controlled in distinct ways. *Movement uncertainty* is controlled by the behavior of feedback-driven motion control laws. *Pose uncertainty* is controlled in the basic SSH by hill-climbing to dstates and in the hybrid SSH by incremental localization within the local perceptual map. *Structural ambiguity* about the large-scale topology of the environment is controlled by the abduction process that builds the topological map. *Global metrical uncertainty* is controlled by relaxing metrical information from separate local frames of reference into a single global frame of reference, guided by the topological map.

The basic SSH explores the environment by selecting an alternating sequence of trajectory-following and hill-climbing control laws, moving between and localizing at distinctive states. In the hybrid SSH, localization by hill-climbing is replaced by localization in an LPM.

Remolina and Kuipers [13], [15] present a formalization of the SSH framework as a non-monotonic logical theory. The theory contains axioms describing the properties and relationships of actions, views, distinctive states, causal schemas, places, paths, and regions. One set of axioms describes the robot's sensorimotor experience by asserting causal schemas. Another set of axioms enforces the SSH topological properties. A third set of axioms (not used in the research reported in this paper) incorporates local metrical information, such as bounds on the path distances between places and the local radial angles between paths at a place.

This theory provides a precise specification of the possible logical models (topological maps) that are consistent with the axioms and the sequence of actions and views observed while exploring. A prioritized circumscription policy (expressed as a nested abnormality theory [16]) specifies how these logical models are ordered by simplicity.

Suppose the robot performs an action $a$, resulting in a view $v$. For each $\langle M,x\rangle$ on the fringe of the tree:

1) If $M$ includes $\langle x,a,x'\rangle$ and $view(x',v')$,
   - if $v' = v$, then $\langle M,x'\rangle$ is the successor to $\langle M,x\rangle$;
   - if $v' \neq v$, then mark $\langle M,x\rangle$ as inconsistent.
2) Otherwise, $M$ does not include $\langle x,a,x'\rangle$. Let $M'$ be $M$ extended with a new distinctive state symbol $x'$ and the assertions $view(x',v)$ and $\langle x,a,x'\rangle$. Consider the $k \geq 0$ dstates $x_j$ in $M$ such that $view(x_j,v)$. Then $\langle M,x\rangle$ has $k+1$ successors:
   - $\langle M' \cup \{x' = x_j\}, x'\rangle$ for $1 \leq j \leq k$, plus
   - $\langle M' \cup \{\forall j\, x' \neq x_j\}, x'\rangle$.
3) If any of the new successor maps violates the topological axioms, mark it inconsistent.
4) If the prioritized circumscription policy finds more than one "simplest" map, perform active exploration to discriminate among the alternatives.

Fig. 2. **Building the tree of topological maps in the basic SSH.**

Conceptually, the topological map-builder maintains a tree whose nodes are pairs $\langle M,x\rangle$, where $M$ is a topological map and $x$ is a dstate within $M$ representing the robot's current position. The leaves of the tree represent all possible topological maps consistent with current experience. Figure 2 gives the algorithm for extending the tree after an action and resulting observation. So far, it has been feasible to do exhaustive expansion of the tree of maps, terminating upon identifying a unique minimal topological map that explains the observations.

## III. EXTENDING THE SSH

Our new hybrid SSH extends the basic SSH by using metrical mapping methods to create and store a *local perceptual map* (LPM) of each place neighborhood. Exploration at the control level identifies *gateways* where control shifts from motion between place neighborhoods to localization within a neighborhood. The set of gateways in a local perceptual map can be analyzed to describe the local topology of the place neighborhood, simplifying the construction of the topological map.

### A. Local Perceptual Maps

The SSH topological map describes a place according to its role in large-scale space: a place is a point location on one or more one-dimensional paths. Each path has two directions denoting downstream (+) or upstream (−) in the order on its set of places. Each place has a dstate facing each direction along each of its paths. After arriving at one dstate at a place, a turn action takes the robot to another dstate, ready to travel along a path to a different place.

This simplified large-scale-space description of a place is an abstraction of a richer small-scale-space description of the place as a region within the sensory scope of the agent. The local perceptual map describes this region in terms of small-scale space.[1] The LPM serves as a virtual sensor (or "observer"), supporting reliable localization within the LPM's frame of reference, local path planning, and obstacle avoidance.

When creating an LPM of a place, the robot must explore enough to eliminate uncertainty about which positions are free or obstructed within the bounded scope of the local place neighborhood. In return, the robot obtains two substantial benefits in terms of its spatial knowledge. First, it can localize unambiguously at any pose within the LPM rather than relying on the basic SSH strategy of hill-climbing to an unambiguous pose. Second, it constructs a *complete* representation of the paths at the place, and hence of the dstates and possible turn actions. This complete local topology description is valuable when constructing the global topological map.

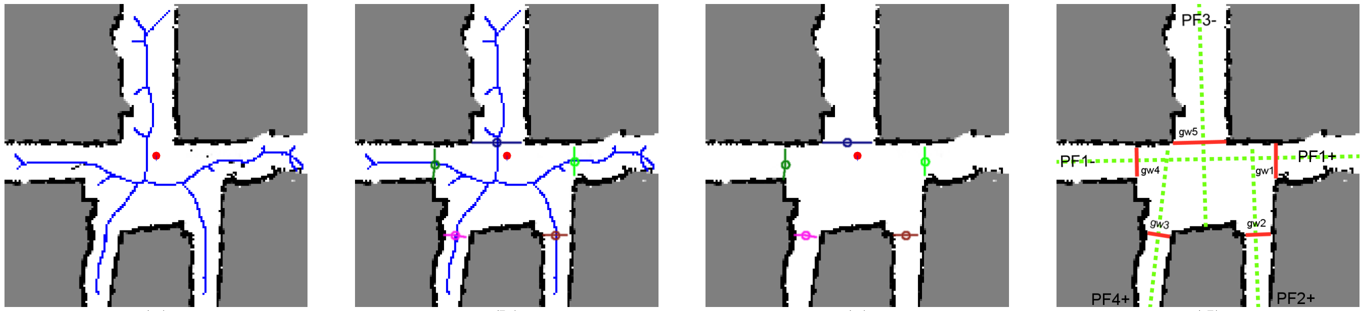
### B. Identifying Gateways and Path Fragments

A *gateway* is a boundary between qualitatively different regions of the environment: in the basic SSH, the boundary between trajectory-following and hill-climbing applicability. Each gateway has two directions, *inward* and *outward*. Gateways can be identified using several different criteria that appear to be closely related. We plan to investigate these alternatives in more detail in future work.

- Positions within an LPM can be tagged with the control law applicability conditions they satisfy, allowing gateways to be defined as boundaries between regions of different control law applicability.
- Chown *et al.* [17] define gateways as the locations of major changes in visibility. *"In buildings, these [gateways] are typically doorways.... Therefore, a gateway occurs where there is at least a partial visual separation between two neighboring areas and the gateway itself is a visual opening to a previously obscured area. At such a place, one has the option of entering the new area or staying in the previous area"* (page 32). We believe that these visibility changes can be quantified using the concept of *isovists* [18].
- A geometric criterion for identifying gateways *in corridors* is the medial axis of free space in the LPM. A gateway corresponds to a "constriction" (or "critical line" [19]) along a medial axis edge, where the distance between the edge and obstacles is a local minimum near a larger maximum. This is the criterion used in the examples in Figures 3 and 5.

The local perceptual map of a place neighborhood also includes *path fragments*: portions of large-scale paths that are grounded in small-scale space. Each gateway is associated with exactly one path fragment, while each path fragment is associated with either one or two gateways. If a path fragment terminates at a place, it will have only one gateway. If it passes through, it will have two. (See Figure 3(a-d).)

[1] Arriving at a dstate at a place corresponds to arriving at a gateway associated with a path fragment in the LPM, facing inward. A turn action between dstates corresponds to local motion within the LPM from an inbound gateway to an outbound gateway.




(a) (b) (c) (d)

**Small-scale star description from PF1+**

| | | |
|---|---|---|
| ((PF1+, | (gw1,out) & (gw4,in), | MIDLINE), |
| (PF2+, | (gw2,out), | MIDLINE), |
| (PF3+, | (gw5,in), | DEADEND), |
| (PF4+, | (gw3,out), | MIDLINE), |
| (PF1-, | (gw4,out) & (gw1,in), | MIDLINE), |
| (PF4-, | (gw3,in), | DEADEND), |
| (PF3-, | (gw5,out), | MIDLINE), |
| (PF2-, | (gw2,in), | DEADEND)) |

**(e)**

**An example large-scale star abstraction**

| | |
|---|---|
| ((ds1, | Pa1+), |
| (ds2, | Pa2+), |
| (ds3, | (Pa3+)), |
| (ds4, | Pa4+), |
| (ds5, | Pa1-), |
| (ds6, | (Pa4-)), |
| (ds7, | Pa3-), |
| (ds8, | (Pa2-))) |

**(f)**

Fig. 3. **Identifying Gateways and Local Topology in an LPM.** Our current implementation of a local perceptual map (LPM) is a bounded occupancy grid. The robot is shown as a circle in the center of the LPM. **(a)** To find gateways in corridor environments, the algorithm computes the medial axis of the occupancy grid free space. **(b)** The maximum of the medial axis graph is found (where the distance of obstacles from the graph is maximal) and each edge is traversed, looking for "constrictions" (where the distance between the graph edge and obstacles is a local minimum). **(c)** The final gateways are drawn as lines connecting the graph edge minima (circle) with the closest obstacles. **(d)** Given the gateways, the path fragments are identified. **(e)** The local topology is initially described by the small-scale star for the place. The star enumerates all the path fragments encountered in clockwise order. **(f)** Once a local topology is extracted, the map builder enumerates dstates and paths for the place (a large-scale star). This environment has five gateways, four paths, and eight dstates. Directed paths in parentheses denote termination at the place.

The robot is required to have a procedure for analyzing the LPM, identifying all path fragments and gateways within it, and determining whether a path fragment passes through or terminates at the place. The same procedure constructs the circular order on gateways at the place (Section III-C and Figure 3(e)). In our implementation, the criterion for path continuity is that for each of two gateways, after arrival at one in the inbound direction, the clear unique default travel continuation is the other in the outbound direction. However, any other criterion that gives consistent decisions for path continuity and circular order will work.

### C. Describing the Local Topology

The *local topology* of a place is the circular order of directed paths at that place. In the small-scale-space ontology of the LPM, the *small-scale star* describes the local topology in terms of directed gateways, directed path fragments, and travel control laws (Figure 3(e)). This is then mapped into the large-scale-space ontology of the cognitive map, where the *large-scale star* describes the local topology in terms of distinctive states and directed paths (Figure 3(f)).

Whether a robot is reasoning about large-scale or small-scale space, when the robot arrives at a place on a path, it must select a path and direction on which to depart. From the complete set of path fragments and gateways identified in the LPM, the local topological structure of the place neighborhood defines the options for this choice.

The small-scale star is constructed as follows:

1) Create a set of tuples $\langle PF, GW, CL \rangle$, where:
   - $PF$ is a path fragment and a direction ($+$ or $-$) along it;
   - $GW$ is the directed gateway (or pair of gateways) on $PF$ facing the same direction;
   - $CL$ is the control law for a travel action away from the place along the path fragment in that direction. For an office environment, these are: MIDLINE, LEFTWALL, RIGHTWALL, DEADEND, and NONE. (For terminating path fragments, DEADEND means that further travel is blocked, while NONE means that no control law is applicable.)
2) Initialize the circular order with those tuples containing outward-facing gateways, in their clockwise sequence around the place. For path fragments that both enter and leave the place neighborhood, both directed path fragments will now appear in the order.
3) Each remaining tuple contains an inward-facing gateway and a path fragment that terminates at the place. It is inserted into the order between outward-facing gateways, as determined by the procedure that decides whether a path fragment continues through the neighborhood.

When an LPM is abstracted to a place in the topological map, each of its path fragments is bound to a corresponding path, and a distinctive state is defined for each (path, direction) combination. The small-scale star description can then be transformed into the large-scale star, entirely within the large-scale-space ontology.

The local topology description provides a purely qualitative account of "left" and "right". Starting from the robot's current path fragment and direction, the subset of path fragments (or dstates in the large-scale star) encountered by moving down the circular order until observing the same path fragment in the opposite direction yields the possible destinations of a "Turn right" action. This avoids the need to define "right" and "left" in terms of a threshold on some angular variable.

Since the set of path fragments and gateways in the LPM is complete, the description of the dstates and directed paths at the place in the circular order of the large-scale star is also complete. Causal schemas for turn actions $\langle x_i, turn, x_j \rangle$ are implicitly defined between every pair of dstates $x_i$ and $x_j$ at the place. This simplifies construction of the global topological map.

### D. Building the Topological Map

Within the hybrid SSH, the structure of the algorithm for building the topological map remains the same as in Figure 2, with one important difference. Just as before, the topological map-builder maintains a tree whose nodes are pairs $\langle M, x \rangle$, where $M$ is a topological map and $x$ is a dstate within $M$ representing the robot's current position. However, view matching is done by matching local topologies from the perspective of a particular gateway, so the unit of matching in the algorithm becomes the place rather than the distinctive state, and ambiguity can only arise after travel actions.

A view is represented by the structure $v = \langle LPM, \tau, gw \rangle$, where *LPM* is the local perceptual map of the current place neighborhood, $\tau$ is the small-scale star description of its local topology, and *gw* is the gateway at which the last action terminated. Matching two views consists of (a) testing whether the two local topologies $\tau$ are isomorphic, and (b) testing whether the two LPMs match, with alignment given qualitatively by the corresponding gateways.

Suppose the robot performs an action $a$, resulting in experiencing a view $v = \langle LPM, \tau, gw \rangle$. With the revised definition of views and view-matching, the algorithm in Figure 2 becomes substantially more efficient. Since the local topology of a place completely describes all possible turn actions, every turn action is handled in Step 1 in Figure 2, so the branching of the map-tree in Step 2 takes place only for travel actions.

There is an exponential decrease in complexity from the dstate-matching version of the topological map-building algorithm (Figure 2) to this place-matching version of the same algorithm. Both algorithms are worst-case exponential, $O(b^d)$, where $b-1$ is the number of known matches to the observations and $d$ is the number of actions that can cause branching. The size of $b$ is reduced because matching places, using complete local topologies and LPMs, is more restrictive than matching views at dstates. And $d$ is reduced by at least a factor of two because only travel actions, rather than both travels and turns, can lead to branching.

---

After performing action $a$ and observing the resulting view $v = \langle LPM, \tau, gw \rangle$, for each $\langle M, x \rangle$ on the fringe of the tree:

1) If $M$ includes $\langle x, a, x' \rangle$ and $view(x', v')$,
   - if $\langle \tau', gw' \rangle = \langle \tau, gw \rangle$, then $\langle M, x' \rangle$ is the successor to $\langle M, x \rangle$;
   - if $\langle \tau', gw' \rangle \neq \langle \tau, gw \rangle$, then mark $\langle M, x \rangle$ as inconsistent.
2) Otherwise, $a$ is a travel action, and $M$ does not include $\langle x, a, x' \rangle$. Let $M'$ be $M$ extended with a new distinctive state symbol $x'$ and the assertions $view(x', v)$ and $\langle x, a, x' \rangle$. Consider the $k \geq 0$ dstates $x_j$ in $M$ such that the local topology of $view(x_j)$ matches the local topology of $view(x')$. Then $\langle M, x \rangle$ has $k+1$ successors:
   - $\langle M' \cup \{x' = x_j\}, x' \rangle$ for $1 \leq j \leq k$, plus
   - $\langle M' \cup \{\forall j\, x' \neq x_j\}, x' \rangle$.
3) If any of the new successor maps violates the topological axioms, mark it inconsistent.
4) If the prioritized circumscription policy finds more than one "simplest" map, discriminate among the alternatives by (a) apply more expensive global topological constraints to test model consistency; (b) testing dstate equalities $x' = x_j$ by matching their LPMs, not just their local topologies; (c) apply more expensive global metrical constraints by relaxing the local places into a single global metrical frame of reference; and (d) actively explore the environment in search of discriminating observations (cf. Figure 1).

Fig. 4. **Building the tree of topological maps in the hybrid SSH.**

---

We can make the algorithm of Figure 2 into the even more efficient algorithm in Figure 4 by dividing view-matching into the simple, reliable matching of local topologies, and the more expensive matching of LPMs. Creating new map hypotheses based on matching local topologies is more liberal than matching LPMs as well, but it preserves the guarantee that all consistent maps are generated and leaves the problem of filtering out the inconsistent maps to subsequent stages of the process.

## IV. Experimental Results

We applied the hybrid SSH map-builder to an exploration of an office environment with multiple nested large loops. Figure 5 shows (a) the pattern of exploration of the environment, (b) the sequence of LPMs observed at successive place neighborhoods, and (c) the unique simplest topological map that resulted from the mapping algorithm, with LPMs overlaid at corresponding places in the correct topological map. No attempt was made to build a global metrical map of this environment within a single frame of reference (though this specific environment is within the capabilities of an offline metrical mapping implementation).

The environment contains 6 paths and 9 places with 4 distinct local topologies. After an exploration consisting of

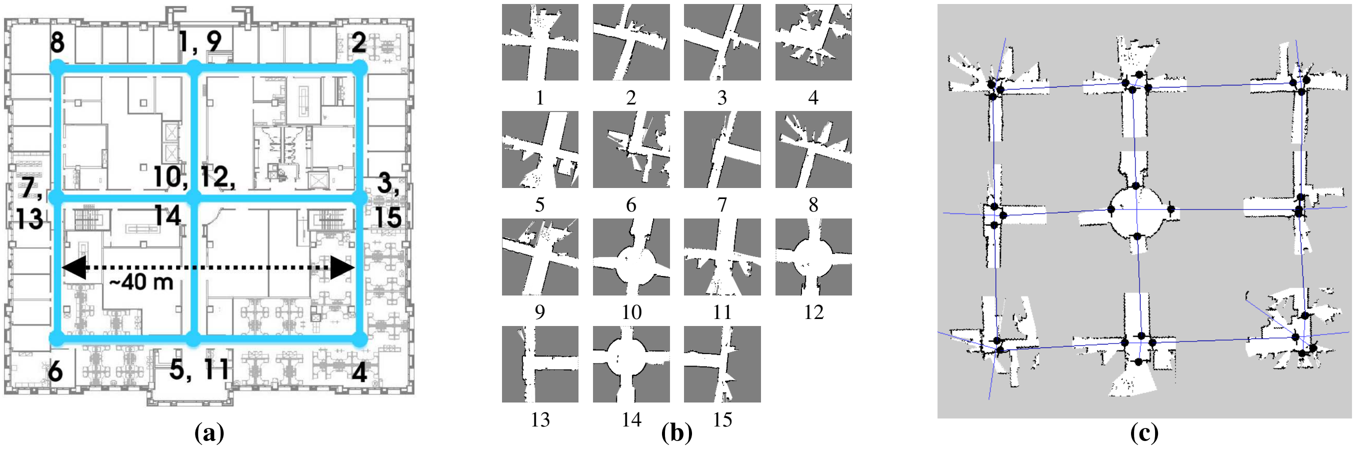



Fig. 5. **An environment with multiple nested loops.** In the CAD drawing **(a)**, we show the path traveled between places in the environment. We enumerate the order of places in the exploration taken by the robot. In **(b)**, we show the LPMs created at the places during the travel. The stars generated from these LPMs are used to search through the space of consistent topological maps. In **(c)**, we show the unique topological map generated after matching local stars and LPMs. The map is overlaid with the LPMs generated at the places, with the gateways, and with the connections between gateways which lie on the same path.

14 travel actions, the topological mapper finds 83 possible configurations of the environment that are consistent with the observed local topologies and the topological axioms. The prioritized circumscription [13], [16] on this set of maps produces 4 minimal models. All but one of these can be eliminated with further exploration or by simply matching LPMs using the alignments specified by the four minimal maps. This final map model is the correct topological representation of the environment.

If we assume planarity of the environment, we can use a more sophisticated version of the topological map-building algorithm [20] that rules out many more models as inconsistent. Here, there are only 46 consistent configurations of the exploration experience, and the circumscription policy produces a single minimal model, which is the correct topological map of the environment (Figure 5(c)). Currently, 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 hybrid SSH mapping method depends on two capabilities: (1) the ability to construct local bounded LPMs within separate frames of reference and (2) the ability of travel actions to move reliably from one place neighborhood to the next. Because these methods are not dependent on the overall size of the environment, the mapping method will be scalable. The results presented on this office environment would be unchanged if the path segments were longer or convoluted so that odometry failed, as long as these two conditions are satisfied.

## V. Related Work in Hybrid Mapping

We review related work on two important themes in hybrid metrical-topological mapping. First, the "patchwork metrical map" consists of local metrical maps with separate frames of reference, linked by topological relations. Second, ambiguity in the topological map means multiple maps are consistent with the travel experience. Other approaches combine topological and metrical maps, for example creating a global metrical map first and then parsing it into topological regions [19], but these fall outside the scope of this paper.

In the "patchwork metrical map," places in the global topological map are annotated with local metrical maps with limited extent and separate frames of reference. In early work on the SSH, Kuipers and Byun [3] associated topological places with local landmark maps and path segments with generalized cylinder models. The local frames of reference could be relaxed into a single global frame of reference by spreading metrical errors evenly across the topological map.

Yeap and Jefferies [21] build maps consisting of adjacent metrical maps of rooms, which are directly connected by gateway-like entities. Bosse *et al.* [7] link together perceptual maps of a fixed number of landmark features. Similarly, Duckett and Saffiotti [22] connect overlapping local occupancy grid maps to form a dense topological network. Lankenau *et al.* [23] create a topological graph of travel paths annotated with metrical maps at the "corners" where paths intersect.

Most mapping implementations only maintain a single map hypothesis, selected through greedy or maximum likelihood methods. Perceptual aliasing may lead to ambiguity about whether the robot has closed a loop or just reached a similar, nearby place. A single map hypothesis cannot model this ambiguity. However, a few hybrid mapping techniques do reason about structural ambiguity.

Kuipers and Byun [3] detect perceptual aliasing and check for possible loop closures by performing physical motion to obtain more evidence. Tomatis *et al.* [24] use a probabilistic framework to localize and detect loops. When the estimate of the robot's location has two probable hypotheses, the framework assumes it is recreating a previously known portion of the topological map. The robot will then physically backtrack until the location estimate converges to a single hypothesis, producing a simpler topological map. Assuming the environment does not contain large topologically identical substructures, both of these methods handle the two cases of whether a loop is present or not; however Tomatis *et al.* cannot handle environments with multiple possible locations for the loop closure.

As in our work, Dudek *et al.* [12] construct an "exploration tree" of all possible consistent world models. To extend the exploration tree, they use the degree of each node (the number of graph edges) in the same way that we use local topology to match places. By using local topology (and even LPMs when necessary), our method has a smaller branching factor. To prune the exploration tree, they use exhaustive search to check for global topological inconsistencies whereas we find axiomatically inconsistent maps. They forecast the use of values to rank map hypotheses in the tree, while we actually are able to rank map hypotheses using the circumscription operator.

## VI. Conclusion

We have presented a hybrid mapping algorithm that combines the detail of metrical maps in small-scale space with the conciseness of topological maps in large-scale space. These two spatial ontologies are connected by extracting the topology of a local place, which is then used for efficient global topological inference. Finally, we have demonstrated the capabilities of this algorithm with an implementation that finds all possible topological maps before selecting the minimal, correct map. This is done entirely without the use of a global frame of reference.

## References


[1] P. Beeson, M. MacMahon, J. Modayil, J. Provost, F. Savelli, and B. Kuipers, "Exploiting local perceptual models for topological map-building," in *IJCAI-2003 Workshop on Reasoning with Uncertainty in Robotics (RUR-03)*, Acapulco, Mexico, August 2003, pp. 15–22.

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

[3] 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.

[4] S. Thrun, "Probabilistic algorithms in robotics," *AI Magazine*, vol. 21, no. 4, pp. 93–109, Winter 2000.

[5] 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.

[6] 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.

[7] 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.

[8] D. Hähnel, D. Fox, W. Burgard, and S. Thrun, "A highly efficient fast-slam algorithm for generating cyclic maps of large-scale environments from raw laser range measurements," in *Proc. of the Conf. on Intelligent Robots and Systems (IROS)*, 2003.

[9] 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.

[10] 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.

[11] 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.

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

[13] E. Remolina and B. Kuipers, "Towards a general theory of topological maps," *Artificial Intelligence*, vol. 152, no. 1, pp. 47–104, 2004.

[14] S. Thrun, D. Fox, and W. Burgard, "Monte Carlo localization with mixture proposal distribution," in *Proc. of the 17th National Conf. on Artificial Intelligence (AAAI-2000)*. AAAI Press/The MIT Press, 2000, pp. 859–865.

[15] E. Remolina and B. Kuipers, "A logical account of causal and topological maps," in *Proc. of the 17th Int. Joint Conf. on Artificial Intelligence (IJCAI-01)*. San Mateo, CA: Morgan Kaufmann, 2001, pp. 5–11.

[16] V. Lifschitz, "Nested abnormality theories," *Artificial Intelligence*, vol. 74, pp. 351–365, 1995.

[17] E. Chown, S. Kaplan, and D. Kortenkamp, "Prototypes, location, and associative networks (PLAN): Towards a unified theory of cognitive mapping," *Cognitive Science*, vol. 19, no. 1, pp. 1–51, 1995.

[18] M. L. Benedikt, "To take hold of space: isovist fields," *Environment and Planning B: Planning and Design*, vol. 6, pp. 47–65, 1979.

[19] S. Thrun, "Learning metric-topological maps for indoor mobile robot navigation," *Artificial Intelligence*, vol. 99, no. 1, pp. 21–71, 1998.

[20] F. Savelli and B. Kuipers, "Loop-closing and planarity in topological map-building," in *Proc. of the Conf. on Intelligent Robots and Systems (IROS)*, 2004.

[21] W. K. Yeap and M. E. Jefferies, "Computing a representation of the local environment," *Artificial Intelligence*, vol. 107, no. 2, pp. 265–301, 1999.

[22] T. Duckett and A. Saffiotti, "Building globally consistent gridmaps from topologies," in *Proc. of the 6th Int. IFAC Symp. on Robot Control (SYRORO-00)*. Vienna, Austria: Elsevier, September 2000.

[23] A. Lankenau, T. Röfer, and B. Krieg-Brückner, "Self-localization in large-scale environments for the bremen autonomous wheelchair," in *Spatial Cognition III*, ser. Lecture Notes in Artificial Intelligence, vol. 2685. Berlin, Germany: Springer-Verlag, 2002, pp. 34–61.

[24] N. Tomatis, I. Nourbakhsh, and R. Siegwart, "Hybrid simultaneous localization and map building: Closing the loop with multi-hypotheses tracking," in *IEEE International Conference on Robotics and Automation*, Washington, DC, May 2002, pp. 2749–2754.