Werner, Felix and Gretton, Charles and Maire, Frederic D. and Sitte, Joaquin (2008) Induction of Topological Environment Maps from Sequences of Visited Places. In *Proceedings International Conference on Intelligent Robots and Systems (IROS)*, Nice, France.

Induction of Topological Environment Maps from Sequences of Visited Places

Felix Werner[†,‡], Charles Gretton[‡], Frederic Maire[†,‡] and Joaquin Sitte[†,‡]
[†]Faculty of Information Technology
Queensland University of Technology
Brisbane, QLD 4001, Australia
[‡]NICTA
Queensland Lab
St Lucia, QLD, 4072, Australia
{felix.werner, charles.gretton, frederic.maire, joaquin.sitte}@nicta.com.au

***Abstract*— In this paper we address the problem of topologically mapping environments which contain inherent perceptual aliasing caused by repeated environment structures. We propose an approach that does not use motion or odometric information but only a sequence of deterministic measurements observed by traversing an environment. Our algorithm implements a stochastic local search to build a small map which is consistent with local adjacency information extracted from a sequence of observations. Moreover, local adjacency information is incorporated to disambiguate places which are physically different but appear identical to the robots senses. Experiments show that the proposed method is capable of mapping environments with a high degree of perceptual aliasing, and that it infers a small map quickly.**

I. Introduction

Competent navigation in an environment is a major requirement for an autonomous mobile robot to accomplish its mission. High level operations that we expect of a truly autonomous robot – such as exploration, path planning, and collision avoidance – require enhanced navigation strategies which incorporate the robots ability to infer a *map* as an internal representation of the environment. Autonomously building maps as spatial representations of the environment from sensor data is considered one of the most important problems in the quest to build truly autonomous robots [1].

There are two approaches to computing an internal representation of a robots environment: *Metric* and *topological* mapping [1]–[3]. We shall focus on the topological case in this paper. Topological maps are an abstract and compact representation of the environment that captures *key places* and their *connectivity* for localisation and navigation. The map is represented by a *labelled graph* where *vertices* represent places and *edges* reflect the connectivity between places [2]. The *labels* of vertices refer to *fingerprints* which characterise the place. Constructing topological maps from sensor data is liable to the *perceptual aliasing* problem, that is, a robot needs to decide when it is visiting a new place or revisiting a memorised place (loop closing) [1], [2].

A particular class of aliasing that has not been addressed fully by existing approaches in topological mapping is the problem of *inherent perceptual aliasing* which is caused by repeated structures in the environment. This occurs, for example, if an agent is equipped with a *radio-frequency identification (RFID)* reader and asked to map the locations of RFID tagged product groups in a store or warehouse [4]. An RFID reader deterministically senses whether an RFID tag is in its sensing range or not. Due to stocking strategies it occurs that identical products are stored at different locations. For topological mapping of locations of the product groups, trolleys can be equipped with RFID readers to continuously measure information about the location of the stock. In this setting, sequences of observations of products are acquired, however, knowledge of metric information or motion actions is not available. Here, the task of topologically mapping the product locations from the sequence of deterministic observations is complicated by identical products being stored at different locations.

A. Related Work

Research in topological mapping has for the most part been concerned with a particular aspect of the perceptual aliasing problem which occurs due to limited sensor capabilities. In particular, noisy data sampled at low frequency can cause the robot to perceive fingerprints from distinct places as being non-distinct. Various probabilistic approaches to solve this *correspondence* problem have been proposed [5]–[10]. These approaches do not properly address situations where places are indistinguishable even with perfect sensing. Moreover, in our setting sensor readings are deterministic, thus, general systems that resolve perceptual ambiguities with probabilistic reasoning are not suitable.

The robot's perceptual abilities can be supported using metric information gained from odometry measurements [11]–[13]. However, odometry information is known to be liable to cumulative errors, especially on non-solid and slippery surfaces such as gravel or uneven terrains. Moreover, mapping environments independent of the robots odometry or motion information allows us to collect the data from several sessions of sensing [14] or by using multiple agents.

Other approaches incorporate a sequence of observations and actions to generate an automaton which corresponds to a topological map of the environment [9], [10], [15]. Also, a strategy has been proposed to build a collection of candidate topologies and prune this collection to find the map which is most feasible according to a sequence of actions and observations [16]. These methods exploit the knowledge of the actions of the agent to disambiguate otherwise perceptually identical places. For example, two hotel rooms at each end of a corridor which are identical can be distinguished using the agents actions, as it needs to travel forward or backward towards one of the rooms.

B. Contribution

In this work, we address the problem of inferring a topological map from sequences of perceptions only. We suppose that actions (e.g. turn left) and odometry cannot be sensed directly. In particular, we are concerned with the problem of perceptual aliasing which occurs in this setting due to inherent environment ambiguities and repeating structures. More formally, given an unknown environment whose inherent topology is modelled as a labelled graph, we propose an approach to infer a topological map from sequences of labels obtained from the traversal of the environment. This is an abstraction of the problem of topological mapping from a sequence of deterministic perceptions which is independent of metric and motion information. The labels of the graph represent deterministic sensor readings and vertices represent places. If the set of labels is smaller than the set of vertices, we have to resolve perceptual aliasing.

The method we propose for inferring a topological map from sequences of labels makes use of the neighbourhood structure of a vertex for disambiguation. If a vertex label is not distinctive, the neighbourhood of the vertex can be considered in order to disambiguate otherwise identical places. Local neighbourhood structures are obtained from the sequence of labels from a given traversal. A particular problem is that the number of vertices is unknown in advance. Even if we know the number of vertices in advance, disambiguation is still difficult unless every vertex has a distinct label. We propose to solve this ambiguity according to Occam's razor principle[1] by constructing a small map in terms of vertices that best explains the sequence of labels. Our method is scalable in the amount of information sensed from the environment. Increasing the information obtained from traversing the environment can enhance the quality of the map.

The remainder of this paper is organised as follows. First, we give background information and introduce terms and notations which are required to describe our method for topological mapping. Second, the approach is described in detail. We then present results from experiments and conclusions.

II. Background

In this section we introduce background and notations for describing the algorithm.

The inherent topology of the environment is modelled as a labelled graph.

Definition 1: A *labelled graph* is a triple $G = (V, E, L)$ of sets where $E \subseteq V^2$. The elements of V are the *vertices* of graph G and the elements of E are its *edges* [17]. Each vertex is mapped to a label in L.

If the environment contains inherent ambiguities due to repeated structures, several vertices share the same label.

Definition 2: Vertices of a graph are *aliases* if they map to the same label.

We denote the *environment graph* G^{env} and the *map graph*, which we want to infer, G^{map}. The environment graph is unknown and the only available information about it is a finite history $h \in L^*$ of labels of visited vertices obtained from the traversal of the environment graph (Here, * the a Kleene star). Note, it is not assumed that the history is exhaustive in the sense that the entire environment graph has been explored completely.

A. Local Adjacency Information: n-Grams

Our method exploits the neighbouring structure of a vertex to disambiguate aliases.

Definition 3: The *k-neighbourhood* of a vertex $v \in V$ in a labelled graph $G = (V, E, L)$ is the subgraph of G which contains all vertices that can be reached by traversing G starting at v using $0, \ldots, k$ edges. The parameter k is called the *depth* of the neighbourhood.

The neighbourhood structures of the environment graph are not accessible directly as it is unknown. Making the usual Markov assumption – that the current vertex the robot occupies is dependent only on the previous visited vertex and the last action executed – we have that the history includes neighbouring information about the environment. Local neighbouring information which is contained in the history can be accessed through sub-sequences of length n.

Definition 4: A sub-sequence of length n is called *n-gram*. Consecutively visited vertices are represented by consecutive labels in the history and, in turn, consecutive labels in the history must originate from adjacent vertices in the environment graph. Hence, the set of n-grams, which can be obtained from a history, corresponds to a feature space on the history.

Definition 5: The set *Grams(h,n)* contains all n-grams which can be extracted from the n-tuples in history h.

The number of unique n-grams which can be extracted from a history of length m is $m - n + 1$. Thus, the set of n-grams which is used as input data for topological inference grows linearly with the length of the history. The length of the history can be arbitrary and covers theoretically all possible n-grams which can be observed while traversing an

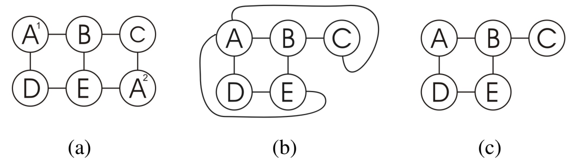


Fig. 1. (a) Example environment graph. Note the two aliases, labelled $\mathtt{A}^1$ and $\mathtt{A}^2$. The set of 2-grams and the set of 3-grams is given in Table I. (b) Map graph of the environment graph shown in (a) using the set of 2-grams from Table I. (c) Possible partial map.

[1]"Entia non sunt multiplicanda praeter necessitatem" or "Entities should not be multiplied unnecessarily". William Occam (1285-1349).

environment graph. The maximal number of unique n-grams which can be represented in a history is dependent on the size of the graph and the length of the n-grams. Given a strongly connected graph, the maximal number of n-grams which can be extracted from an history obtained by traversing the graph is $O(|V|^{n-1})$.

An example environment graph in shown in Figure 1(a). A possible history obtained from traversing this environment graph and, the extracted 2-grams and 3-grams from this history are shown in Table I.

B. Mapping Constraints

For mapping the environment graph, we would like to have that the k-neighbourhood of each vertex of the environment graph corresponds to a k-neighbourhood in the map graph. However, in our case it is not possible to compare k-neighbourhoods of the map graph directly with k-neighbourhoods of environment graph as the latter is unknown. Consequently we propose to measure the consistency of graphs in their feature spaces; i.e., the sets of n-grams of two graphs.

Definition 6: Two labelled graphs $G_1 = (V_1, E_1, L_1)$ and $G_2 = (V_2, E_2, L_2)$ are *n-consistent*, where $n > 1$, iff for $h_1 \in L_1^*$ generated by G_1 and for $h_2 \in L_2^*$ generated by G_2 we have $Grams(h_1, n) \equiv Grams(h_2, n)$.

According to Definition 6, the map graph we infer must exclusively explain the history so the consistency requirement formulates a *hard constraint* the map graph must satisfy.

Our method aims to find a small map, minimising the number of vertices while maintaining consistency with a given history. A map graph which consists of one component for each n-gram in $Grams(h,n)$ is n-consistent according to Definition 6 but is inappropriate for navigation, containing too many vertices. The aim for minimising the number of required vertices can be formulated as a *soft constraint* we wish the map to satisfy.

The n-consistency concept realises the idea of k-neighbourhoods of vertices (see Definition 3) on histories if $n = 2k+1$. For example, when traversing the 1-neighbourhood of a vertex, we can visit at most three different vertices. Thus, we need to consider 3-grams from the histories and hence perform 3-consistency mapping.

history: $h =$< A,B,C,A,E,D,A,B,E,A,C,B,E,D,A,B,C >
$Grams(h,2)$ = {< A,B>,<D,A>,<B,C>,<E,B>, <C,A>,<E,D>,<E,A>}
$Grams(h,3)$ = {<A,B,A>,<A,B,C>,<E,B,A>, <A,D,A>,<E,D,A>,<B,A,B>, <D,A,B>,<B,C,B>,<A,C,B>, <B,E,B>,<D,E,B>,<B,E,A>, <C,B,C>,<C,B,E>,<C,A,C>, <C,A,E>,<D,A,D>,<D,E,D>, <A,E,D>,<E,B,E>,<E,D,E>, <E,A,E>,<A,C,A>,<A,E,A>}

TABLE I

POSSIBLE HISTORY h AS IT COULD BE OBTAINED FROM THE ENVIRONMENT GRAPH IN FIGURE 1 AND SET OF 2- AND 3-GRAMS EXTRACTED FROM THIS HISTORY. NOTE, WE ALLOW THE ROBOT TO PERFORM U-TURNS.

Algorithm 1 Algorithm for inducing a map graph G^{map} given a set of n–grams $Grams(h,n)$ from a environment graph. $V(G)$ denotes the set V of vertices of a graph G.

Require: $Grams(h,n)$
1: $\Gamma \leftarrow Grams(h,n)$
2: $G^{map} \leftarrow \emptyset$
3: **repeat**
4: pick arbitrary $\gamma \in \Gamma$
5: determine set M of possibilities to merge γ to G^{map}
6: sort M w.r.t number of required vertices
7: **for** $i \leftarrow 1 \ldots |M|$ **do**
8: **if** merge γ with G^{map} according to m_i does not violate a mapping constraint **then**
9: delete all merge possibilities from M which require more vertices
10: **else**
11: remove m_i from M
12: **end if**
13: **end for**
14: **if** $M \neq \emptyset$ **then**
15: select arbitrary $m' \in M$
16: merge γ with G^{map} according to m'
17: $\Gamma \leftarrow \Gamma \setminus LocalGrams(G^{map}_{\gamma,m'}, n))$
18: **end if**
19: **until** $\Gamma = \emptyset$ **or** all $\gamma \in \Gamma$ have been tried to be merged with G^{map}
20: **if** $\Gamma \neq \emptyset$ **then**
21: **return failure**
22: **else**
23: **return** G^{map}
24: **end if**

The effect of the parameter n is demonstrated using the example environment graph in Figure 1(a). The map graph shown in Figure 1(b) is 2-consistent to the environment graph as it generates the equivalent set of 2-grams (see Table I). However, it is obvious that this map graph does not truly represent the environment graph. To achieve a map graph which represents the environment graph well, 3-consistency is required.

III. TOPOLOGICAL MAP INDUCTION

In this section we describe our algorithm to induce a topological map given the set of n-grams extracted from a history. We propose to infer the map by merging the n-grams using a stochastic local search with respect to the mapping constraints. The merging relates the local adjacency information contained in the n-grams and the k-neighbourhoods of the vertices in the map graph. Algorithm 1 lists the pseudocode for the proposed algorithm.

The mapping process starts with an empty map graph $G^{map} = \emptyset$ and the set $\Gamma = Grams(h,n)$ initially contains n-grams extracted from the history of a traversal of the environment graph. In the main loop Algorithm 1 selects an n-gram $\gamma \in \Gamma$ and tries to merge it with the map graph G^{map} (lines 3-19). If a merge is successful, γ is removed from Γ (lines 14-18). A merge is unsuccessful if it violates the mapping constraints. In either case, successful or unsuccessful merge, the algorithm proceeds by trying to merge another $\gamma \in \Gamma$ with the map graph until Γ is empty or, otherwise the possibilities for adding n-grams have been exhausted and the map is aborted (line 21). The order in which the γs are merged is arbitrary.

Usually there are several possibilities for merging an n-gram to a map graph resulting in different inferred map graphs. For example, there are two possibilities to merge the 3-gram `C-A-E` with the map graph in Figure 1(c). These merges result in the graphs shown in Figure 1(a) and Figure 1(b). However, the merge possibility that results in the map graph shown in Figure 1(b) violates the mapping constraints thus must be abandoned. To test whether a merge possibility is appropriate, the set M of all merge possibilities to merge γ with G^{map} is calculated and sorted in ascending order according to the number of vertices the resulting map graph requires (see Algorithm 1, lines 5 and 6). Then, beginning with a merge possibility that requires the fewest vertices, every $m \in M$ is tested to see whether it satisfies the mapping constraints (line 8). If a merge is successful, all merge possibilities which require more vertices are immediately removed from M due to the minimisation constraint (line 9). Merges which violate the mapping constraints are removed from M (line 11).

To test whether a merge satisfies the constraints (line 8), a temporary map graph $G^{map}_{\gamma,m}$ is built combining γ with G^{map} according to a merge possibility m. In order to perform a consistency check according to Definition 6 we extract a set of local n-grams.

Definition 7: The set $LocalGrams(G^{map}_{\gamma,m}, n)$ contains exactly those n-grams which can be obtained by traversing the local k-neighbourhood of a vertex which is used by combining γ with G^{map} to $G^{map}_{\gamma,m}$ according to a merge possibility m.

An n-gram contained in $LocalGrams(G^{map}_{\gamma,m}, n)$ but not in $Grams(h,n)$ indicates a violation of the consistency constraint and the merge possibility m must be removed from M (line 11).

If there is more than one merge possibility for merging γ with G^{map} which satisfies the mapping constraints, we arbitrarily select one m' and merge γ with the map graph according m' to (lines 15-16). From set Γ of n-grams which have yet to be merged with G^{map} all n-grams are removed which are implicitly produced by the merge (line 17).

Algorithm 1 infers a map graph which is consistent with a given set of n-grams generated from a history. However, an unfortunate order of choices when selecting an arbitrary n-gram from Γ (line 4) can cause Algorithm 1 to produce large graphs or, get stuck so that n-grams from Γ cannot be merged with G^{map}. We have no way of deciding in advance how to do this crucial selection more successfully. Our current strategy to cope with this problem is to run the entire mapping process several times while arbitrarily selecting n-grams and eventually keep a map graph which requires the smaller number of vertices.

In summary, the proposed mapping method simultaneously determines the number of vertices, assigns appropriate labels to the vertices, and infers the connectivity of the map graph. Hence, a small automaton is inferred which is consistent with a history.

In general, the problem of inferring the minimum automaton given set of input/output pairs is NP-complete [18], [19]. Moreover, finding an automaton close to the smallest one in polynomial time is intractable, assuming $P \neq NP$ [20]. For our approach, we do not perceive actions, however, assuming a "non"-action between the perceptions makes the inference of topological maps in our setting analogous with inferring the minimal automaton and hence their complexity. The complexity motivates our approach as stochastic local search has been successfully applied to NP-problems such as satisfiability and constraint satisfaction [21].

IV. Experimental Results

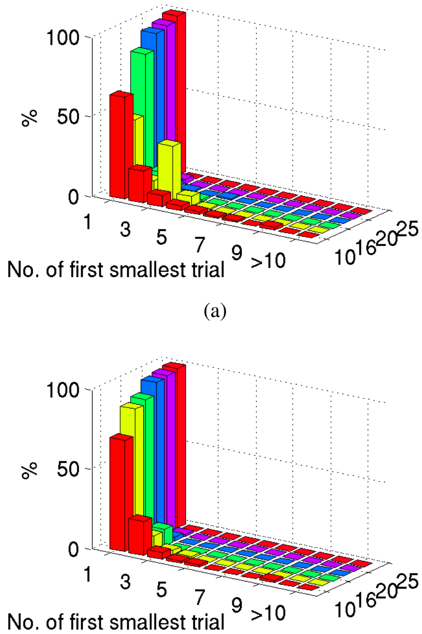
In this section we empirically evaluate our approach for inducing topological maps from a history. We demonstrate the general functionality of the method and evaluate the approach on artificial, randomly generated environment graphs. All experiments were conducted on a 64 bit Intel® Itanium® 2 CPU running at 1.6 GHz.

For the following evaluations, environment graphs $G = (V,E,L)$ are created with 25, 36, 49, 64, 81 and 100 vertices arranged and (possibly) connected in a rectangular grid. We have chosen a grid layout to ensure planarity; most of the environments our robot traverses for its operations are planar. On this grid, we simulate different edge densities, from $|E|^{min} = |V| - 1$ where each vertex has 2 or less adjacencies to a maximum of $|E|^{max} = (|V|^2 - 4 \times \text{grid\_size})/2$ what refers to a fully connected grid. The vertices of each graph are arbitrarily labelled with elements from a set whose cardinality corresponds to 40%, 50%, 60%, 70%, 80%, 90% and 100% of the cardinality of the set of vertices. We assume, that every environment graph has been explored exhaustively so all unique n-grams can be extracted from the exploration history.

A. Mapping Performance

Our evaluations demonstrate, that the smallest map graph out of mapping 100 trials is usually found very quickly. Figure 2 shows this for the mapping of 6000 environment graphs with 25 vertices which were generated using the scheme described above. It is demonstrated that for environment graphs which do not contain inherent ambiguities and so have a set of 25 different labels, inferring a map is trivial so our algorithm finds the obvious map after only one trial. In environments without ambiguities, there is only one possible map. Environment graphs with an increasing number of aliases challenge the mapping process more however, our algorithm finds most of the smallest map graphs with the first trial.

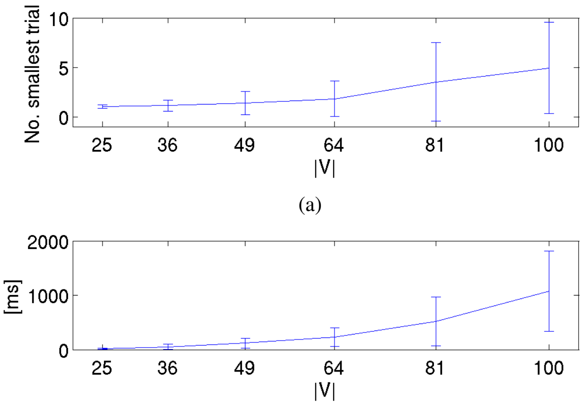
Mapping with respect to 5-consistency 2(b) usually finds the smallest map graph after less trials than mapping with 3-consistency 2(a). This occurs because 5-grams contain more



(a)

(b)

Fig. 2. Histograms of the first smallest map found out of 100 trials for 6000 environment graphs with 25 vertices created according to the described schemata. The cardinality of the set of labels is denoted $|L|$.



(a)

(b)

Fig. 3. (a) Average number of smallest trials for environment graphs with increasing number of vertices ($|V|$) for 3-consistency. The error bars denote the variance. (b) Time consumption for finding the smallest mapping out of 100 trials for environment graphs with increasing sets of vertices for 3-consistency.

information for disambiguation than 3-grams. However, a trial in 5-consistency mapping requires more time than in mapping with respect to 3-consistency. Using our computing environment, in average a mapping trial (25 vertices) takes about 17ms with respect to 3-consistency whereas 5-consistency mapping requires about 120ms per trial. The significant increase of time consumption with an increasing degree of n-consistency is due to the exponential growth depending on n of the set of n-grams which can be extracted in maximum from a graph.

The difficulty of mapping increases with the size of the graph. This may occur as large graphs are more likely to contain repeating structures so that the mapping process may get stuck more often. To evaluate this, we have created 1000 different environment graphs according to our generation scheme where we have we have fixed the size of the set of labels to 70% of the set of vertices. Figure 3(a) shows that for mapping larger graphs, the number of trials required to find the smallest map out of 100 trials grows quickly.

In terms of time consumption, the time required for mapping grows with the size of the graph. As Figure 3(b) shows, that even graphs with 100 vertices which were labelled with 70 different labels take not much more than one second in average to be mapped.

B. Quality of the Map

The overall goal in robotic mapping is to build an internal representation which is isomorphic to the environment. Here, we investigate whether the environment graph is isomorphic to the inferred map graph to measure the quality of the proposed strategy to infer topological maps from a set of n-grams. By quality we mean how likely our method yields a map which is isomorphic to the the environment graph.

Definition 8: Two graphs $G^1 = (V^1, E^1)$ and $G^2 = (V^2, E^2)$ are *isomorphic* if there exists a bijection $f : V^1 \to V^2$ with $xy \in E^1 \Leftrightarrow f(x)f(y) \in E^2 \ \forall x, y \in V^1$ [17].

No polynomial-time algorithm is known for graph isomorphism, neither is it known to be *NP*-complete or to be tractable [22]. An approach to solve *NP*-complete problems is to translate the problem into a *satisfiability (SAT)* problem. In particular, we pose the problem of deciding if a mapping from G^1 to G^2 exists as a decision problem expressed as a *conjunctive normal form (CNF)* propositional formula and present it to the *RSat* [23] solver[2].

Before coding the problem of graph isomorphism into SAT, preliminary tests ensure, that G^1 and G^2 have the same number of vertices of a particular label and degree. If this test fails, we can immediately determine non-isomorphism. Then, we translate the graph isomorphism problem into CNF. For each vertex $v_i \in G^1$ and $v_j \in G^2$, so that their degrees and labels are equal, we have a Boolean proposition ψ_{ij} that says that there exists a bijection f that yields isomorphism where $f(v_i) = v_j$. There are two axiom schemata to encode the graph isomorphism constraints as follows:

I For all $v_k \in G^2$ such that $k \neq j$, we have $\psi_{ij} \to \neg\psi_{ik}$.

II For each ψ_{ij}, letting $N(v_i)$ denote the neighbours of v_i, we have a clause $\psi_{ij} \to \forall k \in N(v_i)\ \exists l \in N(v_j)\ \psi_{kl}$.

Given that the preliminary tests passed, we have that clauses I and II are satisfiable iff G^1 and G^2 are isomorphic. Every environment graph and the corresponding map graph are translated into a CNF formula and isomorphism is then determined by the SAT solver RSat.

We have mapped 6000 environment graphs with 25 vertices generated according to the previously described method to examine the mapping quality of our approach. Figure 4

[2] http://reasoning.cs.ucla.edu/rsat/index.html

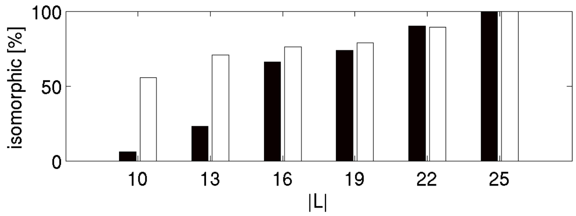


Fig. 4. The quality of the mappings for 6000 environment graphs with 25 vertices with different degree of aliasing. Mapping to 3-consistency is shown in black and mapping with respect to 5-consistency in white. The cardinality of the set of labels is denoted $|L|$.

compares the quality for 3-consistency and 5-consistency mappings. It is to see, that where topology inference exploits deeper neighbourhood information according to n-consistency, we find that it is more likely that we obtain a topology isomorphic to that of the environment.

V. Discussion

We developed an approach for inferring topological maps that is designed to address the perceptual aliasing problem caused by periodical structures in the inherent topology of the environment. The method does not rely on any motion model or metric information, rather exploits the history of deterministic measurements.

The algorithm aims to find a small topological map in terms of vertices which is consistent to the observed history. We have introduced the concept of n-consistency to describe the degree to which the inferred map must be consistent to the information from the history. The particular problem of disambiguating aliases for inducing consistent topological maps is solved by exploring local neighbourhood information of vertices.

Due to the stochastic nature of our algorithm, the mapping process needs to be run several times to ensure that a small map is inferred. Our experiments demonstrate that our method usually finds a small n-consistent topological map after only a few trials. In general, increasing the consistency requirement decreases the number of trials, however, rapidely increases the time a trial requires.

We found that if the degree of aliasing in an environment does not exceed a certain limit, the proposed method often finds a topology that is isomorphic to that of the underlying environment.

The set of mapping constraints does not have to be not limited to the two used in this paper, so any user-defined constraints can be added, such as planarity or maximal degree of vertices. Hence, the proposed method is scalable in terms of prior knowledge about the environment expressed though the mapping constraints. Future work should include extending the information and mapping constraints to increase the quality of the map and accelerate the mapping procedure.

VI. Acknowledgments

NICTA is funded by the Australian Government as represented by the Department of Broadband, Communications and the Digital Economy and the Australian Research Council through the ICT Centre of Excellence program.

Computational resources and services used in this work were provided by the HPC and Research Support Group, Queensland University of Technology, Brisbane, Australia.

References

[1] S. Thrun, "Robotic mapping: A survey," in *Exploring Artificial Intelligence in the New Millenium*, 1st ed., G. Lakemeyer and B. Nebel, Eds. San Francisco, CA, USA: Morgan Kaufmann, July 2002, pp. 1–35.

[2] D. Filliat and J.-A. Meyer, "Map-based navigation in mobile robots: I. a review of localization strategies," *Cog. Sys. Res.*, vol. 4, no. 4, pp. 243–282, December 2003.

[3] J.-A. Meyer and D. Filliat, "Map-based navigation in mobile robots: II. a review of map-learning and path-planning strategies," *Cog. Sys. Res.*, vol. 4, no. 4, pp. 283–317, December 2003.

[4] I. Ehrenberg, C. Floerkemeier, and S. Sarma, "Inventory management with an rfid-equipped mobile robot," in *ICASE*. IEEE, September 2007, pp. 1020–1026.

[5] A. Tapus and R. Siegwart, "Incremental robot mapping with fingerprints of places," in *IROS*. Edmonton, AB, Canada: IEEE, August 2005, pp. 2429–2434.

[6] T. N. Tapus, A. and R. Siegwart, "Topological global localization and mapping with fingerprint and uncertainty," in *IROS*. Singapore: IEEE, June 2004, pp. 2429 – 2434.

[7] C. Valgren, T. Duckett, and A. Lilienthal, "Incremental spectral clustering and its application to topological mapping," in *ICRA*. Roma, Italy: IEEE, April 2007, pp. 4283–4288.

[8] H. Andreasson and T. Duckett, "Topological localization for mobile robots using omni-directional vision and local features," in *IAV*. Lisabon, Portugal: IEEE, July 2004, pp. 3348 – 3353.

[9] K. Basye, T. Dean, and L. P. Kaelbling, "Learning dynamics: system identification for perceptually challenged agents," *Artif. Intell.*, vol. 72, no. 1, pp. 139–171, January 1995.

[10] T. Dean, K. Basye, and L. Kaelbling, "Uncertainty in graph-based map learning," *Robot Learning*, pp. 171–192, 1993.

[11] P. Lamon, A. Tapus, E. Glauser, N. Tomatis, and R. Siegwart, "Environmental modeling with fingerprint sequences for topological global localization," in *IROS*, vol. 4. Las Vegas, NV, USA: IEEE, October 2003, pp. 3781–3786.

[12] A. Ranganathan and F. Dellaert, "Inference in the space of topological maps: An mcmc-based approach," in *IROS*, vol. 2. Sendai, Japan: IEEE, September - October 2004, pp. 1518–1523.

[13] A. Ranganathan and F. Dellaert, "Data driven mcmc for appearance-based topological mapping," in *RSS*. Cambridge, USA: MIT Press, June 2005.

[14] F. Amigoni, S. Gasparini, and M. Gini, "Map building without odometry information," in *ICRA*, vol. 4. New Orleans, LA, USA: IEEE, April 2004, pp. 3753–3758.

[15] R. L. Rivest and R. E. Schapire, "A new approach to unsupervised learning in deterministic environments," in *ICML*, 1987.

[16] E. Remolina and B. Kuipers, "Towards a general theory of topological maps," *Artif. Intell.*, vol. 152, no. 1, pp. 47–104, January 2004.

[17] R. Diestel, *Graph Theory*, 3rd ed., ser. Graduate Texts in Mathematics. Springer-Verlag, Heidelberg, 2005, vol. 173.

[18] D. Angluin, "On the complexity of minimum inference of regular sets," *Inform. Contr.*, vol. 39, no. 3, pp. 337–350, 1978.

[19] E. M. Gold, "Complexity of automaton identification from given data," *Inform. Contr.*, vol. 37, pp. 302–320, 1978.

[20] L. Pitt and M. K. Warmuth, "The minimum consistent dfa problem cannot be approximated within any polynomial," *J. ACM*, vol. 40, no. 1, pp. 95–142, January 1993.

[21] H. Hoos and T. Stützle, *Stochastic Local Search: Foundations & Applications*. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., September 2004.

[22] J. Köbler, U. Schöning, and J. Torán, *The graph isomorphism problem: its structural complexity*, 1st ed. Basel, Switzerland: Birkhauser Verlag, July 1993.

[23] K. Pipatsrisawat and A. Darwiche, "Rsat 2.0: Sat solver description," Automated Reasoning Group, Computer Science Department, UCLA, Tech. Rep. D-153, 2007.