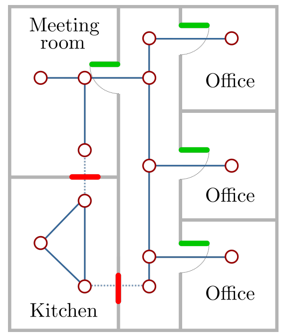
2020 IEEE International Conference on Robotics and Automation (ICRA)
31 May - 31 August, 2020. Paris, France

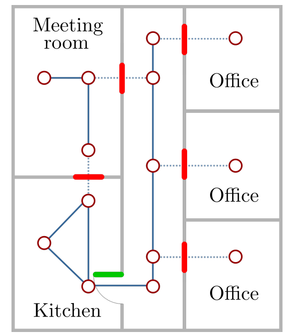
Long-Term Robot Navigation in Indoor Environments Estimating Patterns in Traversability Changes

Lorenzo Nardi[1,2] Cyrill Stachniss[1]

***Abstract*— Nowadays, mobile robots are deployed in many indoor environments such as offices or hospitals. These environments are subject to changes in the traversability that often happen following patterns. In this paper, we investigate the problem of navigating in such environments over extended periods of time by capturing and exploiting these patterns to make informed decisions for navigation. Our approach uses a probabilistic graphical model to incrementally estimate a model of the traversability changes from the robot's observations and to make predictions at currently unobserved locations. In the belief space defined by the predictions, we plan paths that trade off the risk to encounter obstacles and the information gain of visiting unknown locations. We implemented our approach and tested it in different indoor environments. The experiments suggest that, in the long run, our approach leads robots to navigate along shorter paths compared to following a greedy shortest path policy.**



(a) Working state.



(b) Coffee break.

Fig. 1: Patterns in traversability changes on the topological map of an office. The red circles are the nodes and the blue solid lines are the traversable connections among them.

I. Introduction

Over the last decade, many mobile robots have been deployed in indoor environments such as offices, hospitals, and shopping malls. Most robot navigation systems rely on static representations of the environment such as occupancy grid or topological maps for planning and navigating to targeted locations. In reality, robots are often employed in environments where the traversability changes continuously. Traditional navigation systems avoid obstacles by performing reactive strategies [6] or planning local deviations [24]. These approaches are effective to tackle unforeseen obstacles but have no memory about previous experiences. Thus, when encountering the same situation multiple times, robots may perform every time the same sub-optimal behavior.

In indoor environments, there are many changes in the traversability that are correlated or happen following certain patterns. For example, the doors in an environment could be open or closed at the same time according to certain patterns. In the environment illustrated in Fig. 1a, the offices' doors are typically open (green) while people are working. Whereas, if the kitchen is open, it is likely that people are enjoying a coffee and so that the offices' doors are closed (red), see Fig. 1b. When deploying a robot in such environments over a longer period of time, it can observe these patterns and exploit this knowledge to navigate along shorter paths, thus increasing the efficiency of its operations.

In this paper, we investigate the problem of (i) modeling and predicting the patterns of change in the environment traversability and (ii) planning paths that exploit the predictions to reduce the risk of encountering blocked passages. While existing approaches propose to make decisions according to periodic patterns of change [5], we focus on modeling spatial patterns that are independent of time information.

The main contribution of this paper is a novel system for robot navigation over extended periods of time in indoor environments where the traversability changes following spatial patterns. We consider a topological map of the environment like the one illustrated in Fig. 1, where the nodes are the locations of interest and the edges are the passages between these locations. We incrementally model how the traversability of the edges changes using the robot's observations during traversal. We use a probabilistic graphical model to represent this knowledge and to predict the traversability in the environment. We exploit the predictions to plan navigation strategies that account for the risk to encounter blocked passages and, at the same time, for the information gain of making observations to improve the model.

As a result of that, our approach is able to (i) learn incrementally a model of the patterns of change in the environment traversability from robot's observations; (ii) make predictions about the traversability at unobserved locations; (iii) plan paths that exploit the predictions to realize anticipatory strategies for navigation. Over time, our system leads the robot to encounter a reduced number of blocked passages and, thus, to navigate along paths that are on average shorter than following greedy-reactive strategies.

II. Related Work

In the literature, several approaches have been proposed to model changing environments for robot navigation. For

[1]Photogrammetry & Robotics Lab, University of Bonn, Germany.
[2]Autonomous Intelligent Driving GmbH, Munich, Germany.
This work has partially been funded by the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) under Germany's Excellence Strategy, EXC-2070 - 390732324 - PhenoRob.

978-1-7281-7395-5/20/$31.00 ©2020 IEEE

example, Stachniss and Burgard [25] map the typical configurations of low-dynamic areas of the environment for improving localization. Conditional transition maps [13] learn the motion patterns of objects on a grid-based representation. Dynamic Gaussian process occupancy maps [20] map long-term dynamics with a spatially-continuous representation that provides occupancy estimates. Fremen [10] enhances a topological map with a spectral model that allows for predicting the traversability of the edges as a function of the time of day. Fentanes *et al.* [5] use this model for planning paths that take into account the temporal periodicity of changes in the environment. In contrast to that, we use a probabilistic approach to learn a time-agnostic model of the traversability changes on the edges of a topological representation of the environment.

Representing the full joint probability over the traversability of the edges is intractable even for relatively small environments. In the context of SLAM, FAB-MAP [3] uses the Chow-Liu tree approximation for modeling the joint distribution of a set of visual features. For traffic prediction, Furtlehner *et al.* [8] use a factor graph representation to model and predict the road traffic from a probe vehicle. We consider a similar model based on factor graphs [12] that can capture the correlation between traversability changes and exploit this correlation to make probabilistic predictions.

Planning paths in the belief space defined by the predictions can be formulated as a partially observable Markov decision process (POMDP). However, POMDPs are in practice intractable for real-world environments [21]. The reactive planning problem [16] considers a set of possible configurations and plans policies that guarantee the robot to reach the goal. Murphy *et al.* [18] samples the edge costs from a probabilistic costmap, generates a list of paths using A^* and selects the most frequent path. RAG search [2] plans risk-aware paths on a graph where the edge costs are unknown by trading off exploration and exploitation. In the Canadian traveler problem (CTP) [22], an agent aims at traveling along the shortest path in a road network where some roads, unknown to the agent, are blocked. Lim *et al.* [15] introduce a variant of the CTP in which the roads' traversability may be correlated. The CTP is a PSPACE-complete problem [7], however different approximations have been proposed. Nikolova and Karger [19] approximate the CTP by considering graphs that consist only of disjoint paths. Whereas, CTP-UCT [4] is a Monte-Carlo search algorithm that computes policies by taking the uncertainty of the predictions into account. We extend this approach to plan paths that lead the robot to collect informative observations for improving the model of the environment traversability and its predictions.

Krause *et al.* [11] use a criterion based on mutual information for collecting information about the environment. Meliou *et al.* [17] plan informative paths by selecting the locations that maximize the mutual information. We use the mutual information for planning paths that trade off the exploration of informative locations and the exploitation of the traversability predictions to navigate along short paths.

III. Problem Definition and Assumptions

We consider a robot that navigates in indoor environments where the traversability changes according to certain patterns. Initially, these patterns are completely unknown to the robot. However, during navigation, it can observe which passages are frequently blocked at the same time.

Our robot navigation system relies on a topological map of the environment $\mathcal{G} = (\boldsymbol{V}, \boldsymbol{E})$, where $\boldsymbol{E}$ are the edges representing possible passages and $\boldsymbol{V}$ are the set of nodes representing their intersections. We refer to each navigation task performed by the robot as a run. At every run, the traversability of the environment may change. We represent the traversability of the i-th edge at a run t as the binary random variable e_i^t that is 0 if the edge is blocked or 1 if the edge is free. We refer to the state of all the edges of the topology during run t as the environment configuration $\boldsymbol{E}^t = \{\mathrm{e}_1^t, \ldots, \mathrm{e}_{|\boldsymbol{E}|}^t\}$.

In this work, we make the following key assumption:

1) when the robot starts a new run, it has no knowledge about the current environment configuration except for its previous observations;
2) *during* each run, the environment configuration does not change. This means that we account only for the low-frequency dynamic changes in the environment;
3) the environment configuration is independent on the temporal order of the runs, i.e., the configuration at run t has the same degree of dependence to the configuration at run $t+1$ than to the one at $t+k$;
4) when reaching one of the nodes $\mathrm{v} \in \boldsymbol{V}$, the robot can observe all the adjacent edges to v.

IV. Estimating Patterns in Traversability

To make informed decisions for navigating in changing environments, we aim at learning a model of the patterns of change to predict the traversability at unknown locations. We use the robot's observations during traversal to incrementally learn a probabilistic model that captures the correlation among the edges' traversability and that exploits this correlation to make predictions during navigation.

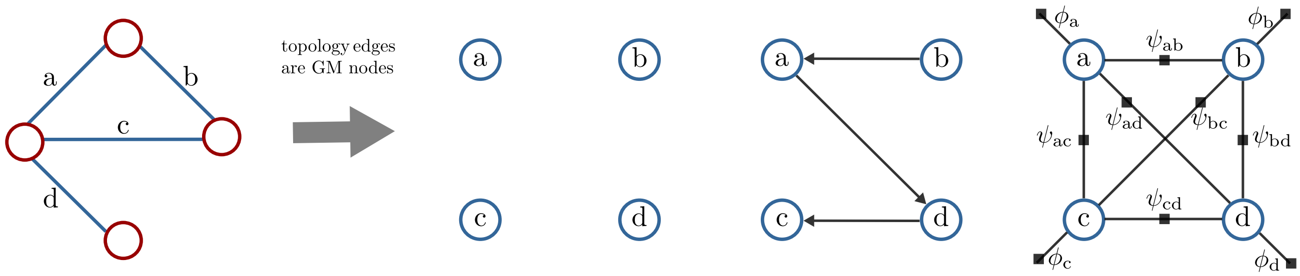
A. Modeling Environment Configurations

During navigation at run t, the environment presents a configuration $\boldsymbol{E}^t$ of which the robot typically observes only a subset of edges $\boldsymbol{Z}^t \subseteq \boldsymbol{E}^t$. To plan reliable routes, we aim at predicting the traversability of the currently unobserved edges $\boldsymbol{U}^t$, with $\boldsymbol{E}^t = \boldsymbol{Z}^t \cup \boldsymbol{U}^t$. We formulate this as the problem of estimating the probability of the unobserved edges $\boldsymbol{U}^t$ to be traversable conditioned on the partial observation of the environment configuration $\boldsymbol{Z}^t$:

$$p(\boldsymbol{U}^t \mid \boldsymbol{Z}^t) = \frac{p(\boldsymbol{U}^t, \boldsymbol{Z}^t)}{p(\boldsymbol{Z}^t)}, \quad \text{def. cond. prob.} \tag{1}$$

$$= \eta\, p(\boldsymbol{E}^t), \quad \boldsymbol{E}^t = \boldsymbol{Z}^t \cup \boldsymbol{U}^t \tag{2}$$

where η is a normalizer given the current observations $\boldsymbol{Z}^t$ and $p(\boldsymbol{E}^t)$ is the joint probability distribution over the traversability of the edges in the environment.



(a) Example topology. (b) Indep. variables approx. (c) Chow-Liu tree approx. (d) Our factor graph approx.

Fig. 2: Graphical model (GM) representations (b-d) of the joint probability over the edges in the example topology (a).

The distribution $p(\boldsymbol{E}^t)$ defines a probability function over the space of possible configurations. It captures the correlation among the traversability of edges and, thus, its knowledge is essential to make predictions about the environment configuration. In general, $p(\boldsymbol{E}^t)$ is a distribution without a special structure and the space required to represent it is exponential in the number of edges. Therefore, representing the joint distribution over the edges becomes quickly intractable.

There exist different approaches to compute a tractable approximation of the joint probability distribution. The simplest one is to consider each edge to be independent of all others as illustrated in Fig. 2b for the example topology in Fig. 2a. This representation is efficient to store and compute, but it is not able to capture the correlation among the edges. A more advanced approach is the Chow-Liu tree approximation [1] that represents the joint probability using a tree-structured Bayesian network as the one depicted in Fig. 2c. Chow-Liu representation requires quadratic space and is able to capture some correlation among edges. However, it requires training data and typically does not deal with incremental and partial data that characterize our problem. To deal with our requirements, we propose to approximate the joint probability distribution over the edge traversability by using a flexible but bounded factor graph representation.

B. Our Factor Graph Model

A factor graph is a probabilistic graphical model that allows for representing a general factorization of a function. It is structured as an undirected graph with two kinds of nodes: the variable nodes that correspond to the random variables and the factor nodes that represent local functions of the adjacent variable nodes.

We use a factor graph representation in which the variable nodes are the edges $\boldsymbol{E}$ of the topology. We model the correlation among the traversability of the edges by defining one unary factor node ϕ for each edge and one binary factor node ψ for each pair of topology edges, as in the factor graph illustrated in Fig. 2d. Considering this representation, we assume that we can approximate the joint distribution over the edges as:

$$p(\boldsymbol{E}^t) \approx p_{\phi\psi}(\boldsymbol{E}^t) = \eta \prod_i \phi_i \prod_j \psi_{ij}. \quad (3)$$

This factor graph representation allows for approximating the probability over the environment configurations by capturing some of the correlation between the edge traversability while storing only $|\boldsymbol{E}|(|\boldsymbol{E}|-1)/2 + |\boldsymbol{E}|$ low-dimensional factors. Therefore, our representation requires only quadratic space in the number of edges rather than exponential as in the case of the full joint probability distribution.

C. Computing Factors From Observations

Given our factor graph representation, we need to provide a definition of the factors ϕ and ψ such that the model corresponds to the robot's observations collected in the previous runs and we can efficiently update the model in an incremental manner as the robot acquires new observations.

The belief propagation algorithm [23] allows for performing inference on factor graphs. We use the BP algorithm to make predictions from current data (see Sec. IV-D) but also to estimate the model parameters from the robot's observations. Furtlehner *et al.* [8] introduce an approach to estimate the factor nodes from the marginal probabilities by using the fixed points of belief propagation. We use this approach to define the unary factors and binary factors of our factor graph as:

$$\phi_i = p(\mathrm{e}_i), \quad (4)$$

$$\psi_{ij} = \frac{p(\mathrm{e}_i,\, \mathrm{e}_j)}{p(\mathrm{e}_i)\, p(\mathrm{e}_j)}, \quad (5)$$

where $p(\mathrm{e}_i)$ and $p(\mathrm{e}_i, \mathrm{e}_j)$ are respectively the unary and binary joint probabilities of the edges to be traversable or blocked. Note that Eq. (5) has an analogy with the mutual information between e_i and e_j that is non-zero for $p(\mathrm{e}_i,\, \mathrm{e}_j) \neq p(\mathrm{e}_i)\, p(\mathrm{e}_j)$. This gives an intuition that such definition of the factors allows for modeling the correlation between edges.

This definition of factor nodes allows for computing the approximated joint probability distribution in Eq. (3) as:

$$p_{\phi\psi}(\boldsymbol{E}^t) = \eta \prod_i^{|\boldsymbol{E}|} p(\mathrm{e}_i) \prod_j^{|\boldsymbol{E}|} \frac{p(\mathrm{e}_i, \mathrm{e}_j)}{p(\mathrm{e}_i)\, p(\mathrm{e}_j)}. \quad (6)$$

We compute the unary and binary joint probabilities, $p(\mathrm{e}_i)$ and $p(\mathrm{e}_i,\, \mathrm{e}_j)$, from the robot's observations in the previous runs $\boldsymbol{Z}^{1:t\text{-}1}$. We achieve this by maintaining a counter of the number of observed occurrences of each unary and binary configurations. To prevent probabilities to take extreme values of 0 or 1 on a single observation, we initialize them with a uniform prior by assigning to each configuration an equal positive number of occurrences. After each run, we update the counters based on the robot's observations and recompute the probabilities. Dealing with unary and binary joint probabilities allows us to update only the probabilities corresponding to the observed edges. Using this procedure, we can incrementally and efficiently compute the model's parameters from the robot's observations. The other key advantage of this procedure is that it allows for easily incorporating partial observations of the environment. For instance, in the example illustrated in Fig. 2a, if the robot observes the edges a and b but not c and d, we update $p(\mathrm{a})$, $p(\mathrm{b})$, and $p(\mathrm{a},\,\mathrm{b})$, but not $p(\mathrm{a},\,\mathrm{c})$ and $p(\mathrm{b},\,\mathrm{d})$.

D. Predicting Traversability of Edges

Our factor graph model maintains a tractable approximation of the joint probability distribution over the traversability of the edges and provides us a tool for predicting the traversability of currently unknown edges. The belief propagation algorithm implements a message passing procedure in the graph to estimate the MAP environment configuration and the marginal probabilities of each edge to be traversable.

We predict the traversability of the unobserved edges $\boldsymbol{U}^t$ at run t by fixing the observed edges $\boldsymbol{Z}^t$ to the observed values in the factor graph and by performing belief propagation. This procedure allows for computing a belief about the environment configuration that estimates $p(\boldsymbol{U}^t \mid \boldsymbol{Z}^t)$.

V. Planning Exploiting Predictions

We aim at exploiting the traversability predictions provided by our factor graph model to plan anticipatory behaviors that lead the robot to encounter a reduced number of unforeseen obstacles during navigation in the long run. To achieve this, we explore the belief space of possible environment configurations and plan paths that trade off travel distance and information gain to improve the edge traversability model.

A. Minimizing Travel Distance From Predictions

We minimize the travel distance to reach the goal in a partially observed environment by exploring the belief space of the possible environment configurations defined by the predictions. We search the belief space by using an approach based on CTP-UCT [4]. CTP-UCT is a Monte-Carlo search algorithm based on the upper confidence bounds applied to trees [9] that allows for computing approximate solutions for the Canadian traveler problem.

Given a prediction of the environment configuration, we approximate the belief space of possible configurations by performing a sequence of rollouts. A rollout randomly samples a configuration according to the current belief and simulates robot navigation on this configuration. The robot has no initial knowledge about the sampled configuration but it can make observations during traversal. In CTP-UCT, the robot selects locations for navigation that led to the goal through short paths and have been selected less often in the previous rollouts. To this end, at each step of the rollouts, we consider a state s composed by the robot's current location, the set of known traversable and blocked edges, and the set of unknown edges. Let $\rho = \{s_0,\, s_1,\, \ldots,\, s\}$ be the current sequence of states at the k-th rollout, we select the next state s' that maximizes the UCT formula:

$$s' = \underset{s'}{\operatorname{argmax}}\, B \sqrt{\frac{\log R^{k\text{-}1}(\rho)}{R^{k\text{-}1}(\rho')}} - dist(s, s') - C^{k\text{-}1}(\rho'), \quad (7)$$

where $\rho' = \{\rho, s'\}$ is the new sequence of states, $B > 0$ is a parameter that biases the exploration in the belief space, $dist(s,\, s')$ is the travel distance to move from s to s', $R^{k\text{-}1}(\rho)$ is the number of previous rollouts that start with ρ, and $C^{k\text{-}1}(\rho)$ is the average travel distance to the goal in the previous rollouts that start with ρ.

After performing a number of rollouts, CTP-UCT selects the path $\mathcal{P}$ that minimizes:

$$cost_{\mathrm{CTP-UCT}}(\mathcal{P}) \;=\; \overline{length}(\mathcal{P}), \quad (8)$$

where $\overline{length}(\mathcal{P})$ is the average travel distance to the goal along $\mathcal{P}$ during the rollouts. We extend this cost function to plan paths that, in the initial runs, lead the robot to collect information about the traversability in the environment for improving the model and the predictions in the subsequent runs.

B. Collecting Informative Observations

A common approach to collect information about the environment is to make observations at locations that maximize the mutual information about the non-observed regions. The mutual information, also called information gain, between two discrete random variables a and b is defined as:

$$\begin{aligned} \mathcal{I}(\mathrm{a},\, \mathrm{b}) \;&=\; \mathcal{H}(\mathrm{a}) - \mathcal{H}(\mathrm{a} \mid \mathrm{b}), && (9)\\ &=\; \sum_{a \in \mathrm{a},\, b \in \mathrm{b}} p(a,\, b)\, \log \frac{p(a,\, b)}{p(a)\, p(b)}, && (10) \end{aligned}$$

where $\mathcal{H}(\cdot)$ and $\mathcal{H}(\cdot \mid \cdot)$ are respectively the entropy and the conditional entropy.

Given the current set of unobserved edges $\boldsymbol{U}^t$, we can bias the robot's behavior to collect informative observations by selecting paths that maximize the information gain:

$$\mathcal{I}(\boldsymbol{P}, \boldsymbol{U}^t) \;=\; \mathcal{H}(\boldsymbol{P}) - \mathcal{H}(\boldsymbol{P} \mid \boldsymbol{U}^t), \quad (11)$$

where $\boldsymbol{P}$ is the set of edges along the path $\mathcal{P}$.

Computing the entropy over $\boldsymbol{P}$ requires the knowledge of their joint probability. However, our factor graph model does not provide direct access to it. Therefore, we approximate $\mathcal{I}(\boldsymbol{P},\, \boldsymbol{U}^t)$ with the sum of the pairwise mutual information between the edges along the path $\boldsymbol{P}$ and the unobserved edges $\boldsymbol{U}^t$:

$$\mathcal{I}(\boldsymbol{P}, \boldsymbol{U}^t) \;\approx\; \hat{\mathcal{I}}(\boldsymbol{P}, \boldsymbol{U}^t) \;=\; \sum_{\mathrm{u}^t \in \boldsymbol{U}^t} \max_{\mathrm{p} \in \boldsymbol{P}} \mathcal{I}(\mathrm{p},\, \mathrm{u}^t), \quad (12)$$

where $\mathcal{I}(\mathrm{p},\, \mathrm{u}^t)$ is computed using Eq. (10). This approximation involves only unary and binary joint probabilities that are directly available from our factor graph model. Furthermore, we make sure that the mutual information for the same edge is not counted multiple times by considering the maximum mutual information for each unobserved edge u^t.

C. Exploration vs. Exploitation

We aim at computing paths that trade off exploration and exploitation to minimize the travel distance in the long run. To this end, we perform a sequence of rollouts as in the original CTP-UCT but we replace Eq. (8) by selecting paths that minimize:

$$cost(\mathcal{P}) \;=\; \overline{length}(\mathcal{P}) - \gamma^{\#\text{runs}} \left[\zeta\, \hat{\mathcal{I}}(\boldsymbol{P}, \boldsymbol{U}^t)\right], \qquad (13)$$

where $\overline{length}(\mathcal{P})$ is the estimated travel distance of $\mathcal{P}$ to the goal from the rollouts, $\hat{\mathcal{I}}(\boldsymbol{P}, \boldsymbol{U}^t)$ is an approximation of the information gain along $\mathcal{P}$, $\gamma \in [0, 1]$ is a parameter that controls the exploration term, and ζ is a constant that normalizes information gain and travel distance.

The exploratory behavior of the robot is determined by the parameter γ that decays exponentially with the number of runs performed by the robot. Initially, when few observations are available, γ leads the robot to favor exploratory behaviors for improving the model of the traversability of the edges. As the robot performs more and more runs and acquires several observations of the environment, the model and its ability to make predictions improve and the exploration becomes less prominent. When the learning process of the model converges, our problem becomes similar to a CTP in which the traversability of the edges is correlated. At this point, the exploratory term in Eq. (13) has a low weight and the robot can exploit the predictions similarly as in the original CTP-UCT to navigate along short paths.

VI. Experimental Evaluation

The main focus of this work is on robot navigation over extended periods of time in environments where the traversability changes by following patterns and on how the knowledge about such patterns can be obtained and exploited. Our experiments are designed to illustrate that our approach is able to (i) model and make predictions about the traversability changes in the environment from the robot's incremental observations; (ii) plan paths that exploit the predictions to make decisions for navigation; (iii) navigate along paths that are on average shorter than following greedy shortest path strategies in the long run.

In our experiments, we consider different topologies defined over real-world environments. Many approaches exist to build topological maps, for example, Kuipers *et al.* [14]. On these topologies, we simulate correlated traversability changes. To this end, we use a mixture of M template configurations computed by randomly sampling the traversability of the edges. At each run, we generate an environment configuration by sampling uniformly one of the M templates and applying random noise.

A. Predicting Environment Configurations

We designed the first experiment to show the capabilities of our approach to model the traversability changes and to make predictions about the environment configurations. In this experiment, we consider a relatively small topology composed by 9 nodes and 13 edges and assume that the robot observes at each run the whole environment. These assumptions allow for computing the full joint distribution over the traversability of the edges despite its exponential space complexity to use for comparison. We additionally compare our approach with a model that assumes each edge to be independent of all others, like the one illustrated in Fig. 2b. To investigate the performance of our approach, we consider two different cases: one in which the environment configurations are highly correlated (small M) and one in which the correlation is low (large M).

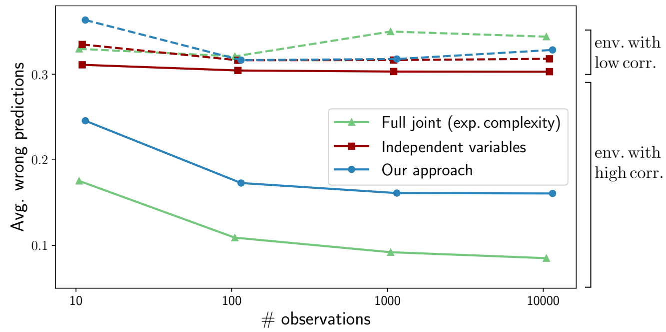


Fig. 3: Avg. ratio of wrong predictions for low and high correlated environment configurations using different models.

We compare the capabilities of each model to predict the edge traversability for 10000 partial configurations after the robot observed 10, 100, 1000, and 10000 configurations. In Fig. 3, we illustrate on average the ratio of wrong predictions for the three approaches. In cases in which the configurations are highly correlated (solid lines), our approach (blue) provides good predictions already after a few observations. The predictions improve incrementally as the robot makes more observations similarly as if using the full joint probability distribution (green). Instead, assuming the edges to be independent (red) cannot capture the correlation between edges and leads to worse predictions. In cases in which the configurations have low correlation (dotted line), the three approaches provide similar predictions. Therefore, also in situations with low correlation, our approach does not reveal worse performance than the model assuming independence among edges.

B. Navigation Exploiting Predictions

The second experiment is designed to show that our approach is able to exploit the predictions of the environment configurations to plan anticipatory strategies that lead the robot to navigate along shorter paths over time. In this experiment, we consider four different environments described in Tab. I. To evaluate the improvement over time, we repeated a fixed sequence of navigation of 25 tasks and configurations for a total of 500 runs in each environment. We compare our approach to the theoretical optimal path computed assuming to known the ground truth environment configuration (in practice unknown to the robot) and to an optimistic shortest path policy called SPO. This strategy plan paths using A* by assuming that every edge of the environment is traversable unless the robot observes the opposite and re-plans. SPO does not take into account the predictions of the environment

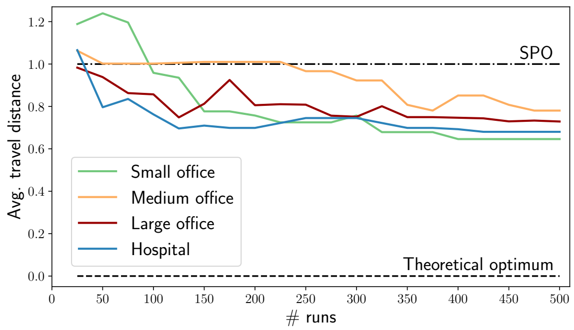


Fig. 4: Avg. distance traveled by the robot following our approach over the number of runs in the environments described in Tab. I normalized between the theoretical optimum (black dashed line) and the SPO solution (black dash-dotted line).

TABLE I: Environments considered in our experimental evaluation.

Environment	Dimensions	Nodes	Edges
Small office	25 × 20 m	16	18
Medium office	30 × 30 m	20	30
Large office	50 × 30 m	18	37
Hospital	125 × 35 m	40	55

configuration and, thus, independent of the number of runs, makes the same decisions as if navigating for the very first time. In our approach, we perform 50 rollouts per decision and set the parameter that regulates the exploratory term of the cost function to $\gamma = 0.95$.

The performance of our approach over the number of runs in the four environments are illustrated in Fig. 4. We evaluate the average difference in the travel distance to the theoretical optimum with ground truth knowledge available (0.0) normalized with respect to the SPO solution (1.0). Initially, when the robot collected little information about the environment, the predictions on the traversability are weak and the robot following our approach performs similarly as following SPO. After 100 runs, our approach starts discovering patterns in the traversability changes and plans paths leading the robot to the goal along shorter paths. Over time, when the robot collects more and more observations about the environment, the learning process of the traversability model converges and, after 500 runs, the robot following our approach navigates along paths that are on average 30% shorter than following an optimistic shortest path strategy.

C. Planning Performance Comparison

Besides the baselines discussed in the previous section, we compare the performance of our approach to other planners. For comparison, we consider the original CTP-UCT [4] that searches for the shortest path in the predicted belief as described in Sec. V-A. We compare our approach also to a strategy inspired by Lim *et al.* [15] called SPD. SPD makes a most likely assumption on the belief about the edge traversabilities and plans the shortest path using A* on the 'determinized' environment configuration. When the robot makes an observation incompatible with the current determinized configuration, SPD computes a new prediction and re-plans.

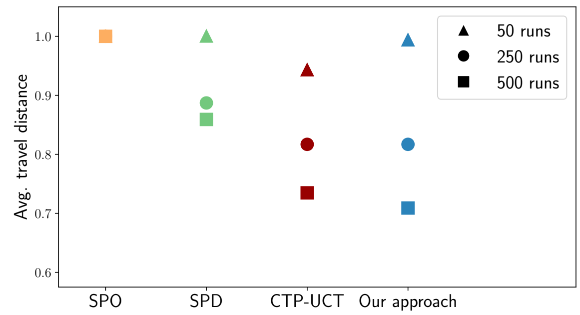


Fig. 5: Avg. distance traveled by the robot following different planning approaches. CTP-UCT and SPD use our factor graph model for computing the predictions.

It is important to note that CTP-UCT and SPD do not provide an approach to model and make predictions of the edge traversability in the environment. Therefore, we compute the predictions using our factor graph model also for these approaches.

The performance of the approaches after 50, 250, and 500 runs for navigating in the environments introduced in the previous section are illustrated in Fig. 5. SPO (orange) reveals a constant trend over time as it does not take the predictions into account. Taking into account the predictions, SPD (green) leads the robot to navigate along shorter paths over time. However, the determinization of the predicted configurations may cause the robot to follow paths that are distant from the optimal ones. CTP-UCT (red) consider a weaker approximation of the belief defined by the predictions by performing rollouts. Thus, it is able to make more informed decisions than SPD that lead the robot along shorter paths. Our approach (blue) extends CTP-UCT by considering an exploratory term that allows the robot to collect informative observations that explicitly improve the model and so the predictions about the traversability of the edges. The exploratory behavior leads initially to slightly longer travel distances than CTP-UCT but, in the long run, it allows the robot to navigate along shorter paths than following other approaches in our evaluation.

VII. Conclusion

In this paper, we investigate robot navigation over extended periods of time in environments where the traversability changes happen following patterns. We present an approach that learns a probabilistic model of the traversability changes from the robot's incremental observations during navigation. Our model exploits the estimated correlation between the traversability in the environment to predict the traversability at unknown locations. We exploit these predictions to make informed decisions that lead the robot to navigate encountering a reduced number of unforeseen obstacles in the long run.

Although our approach presents higher complexity compared to traditional planning systems, in environments where the traversability changes following patterns, it has the potential to automatically lead to navigate robots along shorter paths over time increasing the efficiency of their operations.

References

[1] C. Chow and C. Liu. Approximating Discrete Probability Distributions with Dependence Trees. *IEEE Trans. on Information Theory*, 14(3):462–467, 1968.

[2] J.J. Chungn, A. Smith, R. Skeele, and G. Hollinger. Risk-aware Graph Search with Dynamic Edge Cost Discovery. *Intl. Journal of Robotics Research (IJRR)*, 2018.

[3] M. Cummins and P. Newman. FAB-MAP: Probabilistic Localization and Mapping in the Space of Appearance. *Intl. Journal of Robotics Research (IJRR)*, 27(6):647–665, 2008.

[4] P. Eyerich, T. Keller, and M. Helmert. High-quality Policies for the Canadian Traveler's Problem. In *Proc. of the Conference on Advancements of Artificial Intelligence (AAAI)*, 2010.

[5] J. Fentanes, B. Lacerda, T. Krajník, N. Hawes, and M. Hanheide. Now or later? Predicting and Maximising Success of Navigation Actions from Long-Term Experience. In *Proc. of the IEEE Intl. Conf. on Robotics & Automation (ICRA)*, pages 1112–1117, 2015.

[6] D. Fox, W. Burgard, and S. Thrun. The Dynamic Window Approach to Collision Avoidance. *IEEE Journal of Robotics and Automation*, 4(1):23–33, 1997.

[7] D. Fried, S. Shimony, A. Benbassat, and C. Wenner. Complexity of Canadian Traveler Problem Variants. *Theoretical Computer Science*, 487:1–16, 2013.

[8] C. Furtlehner, J.M. Lasgouttes, and D. de La Fortelle. A Belief Propagation Approach to Traffic Prediction using Probe Vehicles. In *IEEE Trans. on Intelligent Transportation Systems (ITS)*, 2007.

[9] L. Kocsis and C. Szepesvári. Bandit based Monte-Carlo Planning. In *Proc. of the Europ. Conf. on Machine Learning*, pages 282–293, 2006.

[10] T. Krajník, J.P. Fentanes, J.M. Santos, and T. Duckett. Fremen: Frequency Map Enhancement for Long-Term Mobile Robot Autonomy in Changing Environments. *IEEE Trans. on Robotics (TRO)*, 33(4):964–977, 2017.

[11] A. Krause, C. Guestrin, A. Gupta, and J. Kleinberg. Near-optimal Sensor Placements: Maximizing Information while Minimizing Communication Cost. In *Proc. of the Intl. Conf. on Information Processing in Sensor Networks*, pages 2–10, 2006.

[12] F. Kschischang, B. Frey, and H.A. Loeliger. Factor Graphs and the Sum-Product Algorithm. *IEEE Trans. on Information Theory*, 47(2):498–519, 2001.

[13] T. Kucner, J. Saarinen, M. Magnusson, and A. Lilienthal. Conditional Transition Maps: Learning Motion Patterns in Dynamic Environments. In *Proc. of the IEEE/RSJ Intl. Conf. on Intelligent Robots and Systems (IROS)*, pages 1196–1201, 2013.

[14] B. Kuipers, J. Modayil, P. Beeson, M. MacMahon, and F. Savelli. Local Metrical and Global Topological Maps in the Hybrid Spatial Semantic Hierarchy. In *Proc. of the IEEE Intl. Conf. on Robotics & Automation (ICRA)*, volume 5, pages 4845–4851, 2004.

[15] Z.W. Lim, D. Hsu, W.S. Lee, and W. Sun. Shortest Path under Uncertainty: Exploration versus Exploitation. In *Proc. of the Conf. on Uncertainty in Artificial Intelligence (UAI)*, 2017.

[16] R. MacDonald and S. Smith. Active Sensing for Motion Planning in Uncertain Environments via Mutual Information Policies. *Intl. Journal of Robotics Research (IJRR)*, 38(2-3):146–161, 2019.

[17] A. Meliou, A. Krause, C. Guestrin, and J. Hellerstein. Nonmyopic Informative Path Planning in Spatio-temporal Models. In *Proc. of the Conference on Advancements of Artificial Intelligence (AAAI)*, volume 10, pages 16–7, 2007.

[18] L. Murphy and P. Newman. Risky Planning on Probabilistic Costmaps for Path Planning in Outdoor Environments. *IEEE Trans. on Robotics and Automation*, 29(2):445–457, 2013.

[19] E. Nikolova and D.R. Karger. Route Planning under Uncertainty: The Canadian Traveller Problem. In *Proc. of the Conference on Advancements of Artificial Intelligence (AAAI)*, pages 969–974, 2008.

[20] S. O'Callaghan and F. Ramos. Gaussian Process Occupancy Maps for Dynamic Environments. In *Proc. of the Intl. Sym. on Experimental Robotics (ISER)*, pages 791–805, 2016.

[21] C. Papadimitriou and J. Tsitsiklis. The Complexity of Markov Decision Processes. 12(3):441–450, 1987.

[22] C. Papadimitriou and M. Yannakakis. Shortest Paths Without a Map. *Theoretical Computer Science*, 84(1):127–150, 1991.

[23] J. Pearl. *Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference*. Elsevier, 2014.

[24] C. Stachniss and W. Burgard. An Integrated Approach to Goal-directed Obstacle Avoidance under Dynamic Constraints for Dynamic Environments. In *Proc. of the IEEE/RSJ Intl. Conf. on Intelligent Robots and Systems (IROS)*, pages 508–513, 2002.

[25] C. Stachniss and W. Burgard. Mobile Robot Mapping and Localization in Non-Static Environments. In *Proc. of the National Conference on Artificial Intelligence (AAAI)*, pages 1324–1329, 2005.