On the Bayes Filter for Shared Autonomy

Lukas Luft, Federico Boniardi, Alexander Schaefer, Daniel Büscher, and Wolfram Burgard

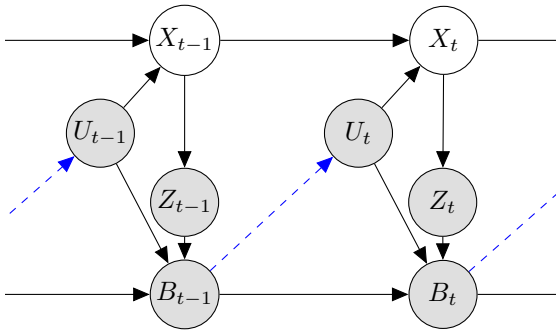
***Abstract*—The Bayes filter is the basis for many state-of-the-art robot localization algorithms. In the literature, its derivation typically requires the robot controls to be chosen independently of all other variables. However, this assumption is not valid for a robotic system that is to act purposefully. The contribution of this paper is twofold: We prove that the Bayes filter is also exact for an autonomous system that chooses the controls depending on any subset of all observable variables. We further show how to augment the filter if a human agent chooses controls on the basis of parts of the state space that are not directly accessible to the robot. In this case, modeling the agent's purpose improves the pose estimate as the control selection provides additional information about the hidden state space. A careful derivation of the Bayes filter then leads to an additional pseudo measurement update step. Simulation and real-world experiments with a teleoperated mobile robot as well as evaluations on the KITTI dataset show that the localization accuracy significantly improves if we augment a particle filter with the proposed pseudo measurement update. Finally, we present an analytical example for an augmented Kalman filter, which leads to a more accurate estimate than the standard Kalman filter.**

***Index Terms*—Localization; Probability and Statistical Methods**

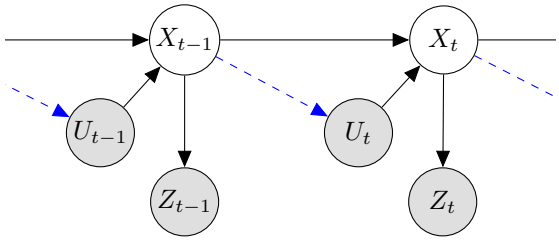
I. Introduction

ONE of the fundamental challenges for mobile robots is self-localization. Besides graph-based approaches, the Bayes filter is the basis for most state-of-the-art robot localization algorithms, including Kalman filters and particle filters, see Chapters 7 and 8 of Thrun *et al.* [1]. The Bayes filter is a recursive update of the pose belief given the latest incoming controls and measurements. It is considered to deliver the optimal estimate if the motion model and the measurement model are given. To prove the optimality one typically assumes that the controls are chosen randomly – or more precisely – independently of all other variables. The problem is that for an autonomous robot, the assumption of randomly chosen controls is not valid: An informed choice of the next action must include a specific goal and the current state. If the goal is, for example, to move to a particular position, the optimal control strongly depends on the current pose of the robot. The first contribution of this paper is to prove that the Bayes filter is exact not only for randomly chosen controls, but also in the case of an autonomous system that chooses the controls based on any subset of all observed information.

However, the situation is different if the choice of the controls depends on further variables that are not accessible to the robot. Consider, for example, a human agent who steers the robot along a desired trajectory. The human's decisions can depend on parts of the state space that are observable by the human but not by the robot. In this case, the standard Bayes filter is not the optimal estimator anymore. However, if we have a model of the human behavior given the robot state, the chosen control provides additional information on the hidden state which improves the estimate. If we know, for example, that a human driver wants to keep his car in the middle of the lane, a steering command to the right on a straight road indicates that the car is on the left side of the lane. Another common example is a navigation system that loses the GPS signal when the car enters a tunnel. As soon as the car turns, it is likely that it has reached a junction and the system can recover if we take this information into account. Note that processing odometry measurements like from inertial measurement units



(a) The robot chooses the control u_t autonomously on the basis of the current belief state $B_{t-1} := bel(x_{t-1})$. Here, we assume that the belief can be computed recursively.



(b) A human chooses the control u_t based on the most recent robot pose x_{t-1}. The purpose model (dashed edges) encodes complex processes including the human vision and decision making.

Fig. 1: The graphs of the causal models for different purpose modalities. Observable variables are gray. Variables that are hidden to the robot are white. The dashed blue edges correspond to the purpose models. If we remove these edges, we recover the graph for randomly chosen controls, i.e. we switch from feedback control to open loop control.

Manuscript received: February, 24, 2019; Revised May, 16, 2019; Accepted June, 14, 2019.

This paper was recommended for publication by Editor Cyrill Stachniss upon evaluation of the Associate Editor and Reviewers' comments.

All authors are with the Department of Computer Science, University of Freiburg, Germany luft@cs.uni-freiburg.de

Digital Object Identifier (DOI): see top of this page.

or wheel encoders in a traditional way can propagate the belief but always increases the pose uncertainty, whereas our method can use this proprioceptive information to decrease the uncertainty without further exteroceptive measurements. There are many examples where the driver's behavior provides information on the location of the vehicle, including reducing velocity in regions with speed limits, holding at stop signs, and many more.

In the following, we lay out the framework to incorporate this additional information. We do so by carefully deriving the motion update of the Bayes filter and taking into account on which variables the control depends in different situations. This results in a filtering scheme similar to the standard Bayes filter augmented with an additional update which we call the pseudo measurement update. In addition to the theoretical framework, we present an experiment with a remotely controlled mobile robot, evaluations on the KITTI dataset [2], and a simulation experiment where a mobile robot and an external controller cooperate in a navigation task.

The pseudo measurement only takes effect if the control choice is influenced by properties of the state space that are not observable by the system. Reversely, our proof that the standard Bayes filter is exact for completely autonomous agents implies that heuristically leveraging a purpose model is invalid for these systems. Intuitively, this is because the action choice can only depend on the current pose estimate and the information of which action is executed cannot, in turn, improve this estimate. Otherwise, information is taken into account repeatedly, at the risk of over-confidence.

II. Related work

This section highlights our contributions beyond the related approaches of Min Oh *et al.* [3] and Winterhalter *et al.* [4], before it references works that cover different aspects of shared autonomy, modeling of user behavior, or general augmentations of the Bayes filter.

Classical Markov localization approaches relate observations with a map of the environment, while the motion of the robot is typically defined independently of this map [5]. However, Min Oh *et al.* [3] augment the motion model with a prior probability to be in a certain region of the map. Winterhalter *et al.* [4] also implicitly take the map into account for the motion model by penalizing particles that move through walls when applying the motion model. Among the references in this section, these two publications have the closest connection to our present paper in the sense that they augment the filter with additional terms that improve the localization. The proposed approach advances these ideas by allowing the integration of priors not only on the positions [3] and on the motions between positions [4] but also on the control selection. Moreover, it integrates these approaches into a general framework: Priors on poses and movements can always be implemented within the standard Bayes filter framework by integrating map information into the motion model. The prior on the control, however, can only be implemented within the generalization of the filter that we derive in the present paper. It is straightforward to combine the mentioned approaches with our framework, as we do in IV-C. Kaelbling *et al.* [6] also give an example where the information of the control choice can reduce the pose uncertainty by including map information into the motion model, see Section 3.2 of their paper. Concretely, repeatedly moving towards a wall increases the probability to end up close to the wall. This behavior results from using an appropriate nonlinear motion model in the standard Bayes filter framework. There are two main differences that set the present approach apart from the methods in [3] and [4]. The latter two methods augment the motion model in a heuristic way to account for the physical information in the map. Our approach, in contrast, lays out a probabilistic framework that gives rise to an entirely new term that models the user's intention. We also derive that using this information is only valid for external inputs while it is invalid for autonomous systems. In Monte Carlo localization applications, our approach can punish particles before they are propagated and can, therefore, provide a better proposal distribution, while the other approaches can only punish particles that have already been propagated – due to their final position [3], or due to unfeasible motions e.g. through walls [4]. While the proposed filtering scheme is probabilistically exact, providing a heuristic-free method to design the purpose model is out of the scope of this paper.

However, estimating human behavior for navigation purposes is an active field of research. An example is the work of Kretschmar *et al.* [7], which predicts human trajectories to facilitate collaborative navigation. A second example is the work of Demeester *et al.* [8], which learns the intention of wheelchair drivers to provide robotic assistance. The behavior of humans is also of high interest for autonomous driving: on the one hand for save navigation among pedestrians – see the work of Bai *et al.* [9] – and on the other hand for human-like motion planning – see the work of Gu *et al.* [10].

Besides the modeling of human intentions, another aspect of the present paper is that it augments the Bayes filter with an additional term. The following two references augment the classical localization (respectively SLAM) frameworks to additionally estimate parameters of the system. Martinelli and Siegwart [11] augment the state space in a Kalman filter context to simultaneously estimate odometric parameters while simultaneously localizing the robot. Kümmerle *et al.* [12] present a framework that performs simultaneous SLAM and parameter estimation by optimizing a hypergraph that includes the odometry parameters as nodes.

One aspect of shared autonomy not considered in the present paper is estimating the user's intention on the fly and select actions to assist. Formulating this problem as a partially observable Markov decision process can yield a distribution over intentions and therefore enable directed assistance [13]. In a deep reinforcement learning context, one can learn a mapping from observations to purposeful robot actions [14].

III. The Bayes filter for autonomous systems and shared autonomy

A. A system model for mobile robots

We model the state of a mobile robot as a time-discrete dynamic system with value x_t at time t. The robot moves

within an environment and can execute controls u_t and gather measurements z_t. Therefore, the system consists of the jointly distributed random variables

$$\mathcal{V} = \{X_{0:t}, U_{1:t}, Z_{0:t}\}. \tag{1}$$

The relevant information about the environment is encoded in a map, which we assume to be known and treat as given background knowledge. Without loss of generality, we can exclude the map variable from $\mathcal{V}$ to ease notation. However, we keep in mind that the functions that model our system can strongly depend on the map. We assume

$$x_t = g\left(x_{t-1}, u_t, \epsilon_t\right) \tag{2}$$
$$z_t = h\left(x_t, \nu_t\right), \tag{3}$$

where the motion model g and the measurement model h are deterministic functions and ϵ_t and ν_t are noise terms that are distributed independently of all other variables. Further, we specify a function π that encodes how the controls are generated. We call π the *purpose model* as it captures the purpose of the action selection. We distinguish the purpose models for different situations: For randomly chosen controls, π is an algorithm that does not depend on any variable in $\mathcal{V}$. For an autonomous system, it can depend on all the observable variables $\{U_{1:t-1}, Z_{0:t}\}$, which are known to the robot at the time t of its decision. A human agent might have access to parts of the state space which is hidden to the robot. Thus, she can in principle consider the entire set $\mathcal{V} \setminus \{U_t\}$. For now, we assume that she only accounts for the most recent pose of the robot. Thus, for the three different situations, we get

$$u_t = \begin{cases} \pi\left(w_t\right) & \text{random controls} \quad (4) \\ \pi\left(u_{1:t-1}, z_{0:t-1}, w_t\right) & \text{autonomous system} \quad (5) \\ \pi\left(x_{t-1}, w_t\right) & \text{human agent} \quad (6) \end{cases}$$

where the noise terms w_t are independent of all other variables in $\mathcal{V}$.

As we will recap below, the derivation of the Bayes filter relies on the three conditional independence assumptions, or Markov conditions

$$p(x_t \mid x_{t-1}, z_{0:t-1}, u_{1:t}) = p(x_t \mid x_{t-1}, u_t) \tag{7}$$
$$p(z_t \mid x_t, z_{0:t-1}, u_{1:t}) = p(z_t \mid x_t) \tag{8}$$
$$p(x_{t-1} \mid z_{0:t-1}, u_{1:t}) = p(x_{t-1} \mid z_{0:t-1}, u_{1:t-1}). \tag{9}$$

Conditions (7) and (8) directly follow from the dependencies in the motion model (2) and the measurement model (3), respectively. For the third Markov condition (9), it is more subtle: It obviously holds for randomly chosen controls (4). For autonomous systems (5), it is also valid, as we show in the Appendix. However, for general purpose models – and in particular for human agents (6) – it does not hold. Fig. 1 and Fig. 2 show the directed acyclic graphs of the causal models defined above for different purpose models. For a recap on causal models, see Chapter 1 of Pearl [15].

B. The Bayes filter for autonomous systems

To pinpoint the crucial assumption, in the following, we recap the well-known derivation of the Bayes filter for robot localization as presented by Thrun *et al.* [1]. As the state space is not directly accessible, we define the belief as the posterior distribution over the latest state x_t given the readings of all observable variables thus far:

$$bel(x_t) := p(x_t \mid z_{0:t}, u_{1:t}) \tag{10}$$
$$\overline{bel}(x_t) := p(x_t \mid z_{0:t-1}, u_{1:t}). \tag{11}$$

The Bayes Filter is a recursive update of the belief $bel(x_t)$ with the last belief $bel(x_{t-1})$, the most recent control u_t, and the latest obtained measurement z_t:

$$bel(x_t) = \eta\, p(z_t \mid x_t)\, \overline{bel}(x_t) \tag{12}$$
$$\overline{bel}(x_t) = \int\limits_{x_{t-1}} p(x_t \mid x_{t-1}, u_t)\, bel(x_{t-1}) dx_{t-1}. \tag{13}$$

We omit the derivation of the correction step (12), which makes use of the Markov condition (8).

The following derivation proves the prediction step (13) for the two cases of randomly chosen controls (4) and autonomous systems (5):

$$\overline{bel}(x_t) \overset{(11)}{=} p(x_t \mid z_{0:t-1}, u_{1:t}) \tag{14}$$
$$\overset{(7)}{=} \int\limits_{x_{t-1}} p(x_t \mid x_{t-1}, u_t)\, \underbrace{p(x_{t-1} \mid z_{0:t-1}, u_{1:t})}_{=:bel_{u_t}(x_{t-1})}\, dx_{t-1} \tag{15}$$
$$\overset{(9)}{=} \int\limits_{x_{t-1}} p(x_t \mid x_{t-1}, u_t)\, p(x_{t-1} \mid z_{0:t-1}, u_{1:t-1})\, dx_{t-1} \tag{16}$$
$$\overset{(10)}{=} \int\limits_{x_{t-1}} p(x_t \mid x_{t-1}, u_t)\, bel(x_{t-1})\, dx_{t-1}. \tag{17}$$

C. The augmented Bayes filter for shared autonomy

We now show how to augment the Bayes filter for the case that a human agent chooses the controls on the basis of direct observations of the state space that are not accessible to the robot. Therefor, we apply Bayes rule to the term $bel_{u_t}(x_{t-1})$:

$$bel_{u_t}(x_{t-1}) \tag{18}$$
$$:= p(x_{t-1} \mid z_{0:t-1}, u_{1:t}) \tag{19}$$
$$= \frac{p(u_t \mid x_{t-1}, z_{0:t-1}, u_{1:t-1}) p(x_{t-1} \mid z_{0:t-1}, u_{1:t-1})}{p(u_t \mid z_{0:t-1}, u_{1:t-1})} \tag{20}$$
$$= \eta' p(u_t \mid x_{t-1}, z_{0:t-1}, u_{1:t-1}) bel(x_{t-1}). \tag{21}$$

The denominator does neither depend on the integration variable x_{t-1} nor on the variable x_t for which we calculate the distribution $bel(x_t)$. Thus we treat it as a normalization constant η'. We see that (21) has the same form as the measurement update (12), with the purpose model $p(u_t \mid x_{t-1}, z_{0:t-1}, u_{1:t-1})$ instead of the measurement model. Therefore, we call (21) the pseudo measurement update.

For the case of a human controller, we get the augmented prediction step

$$\overline{bel}(x_t)$$
$$= \eta' \int\limits_{x_{t-1}} p(x_t \mid x_{t-1}, u_t)\, p(u_t \mid x_{t-1}) bel(x_{t-1})\, dx_{t-1}, \tag{22}$$

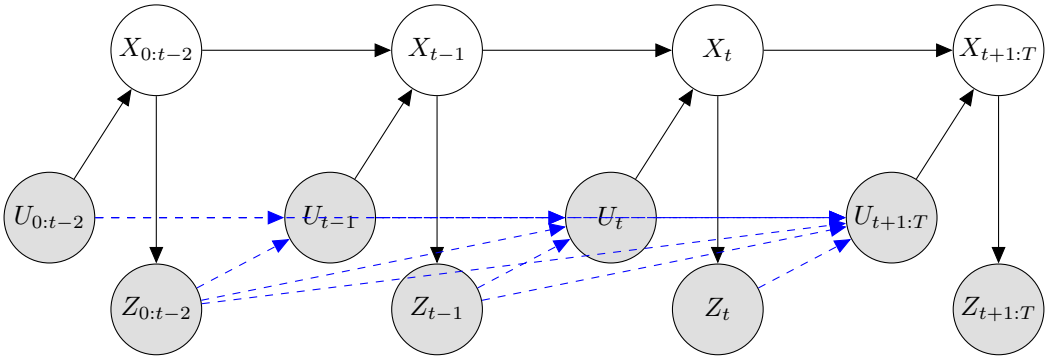


Fig. 2: The graph of the causal model for the case that robot chooses the control u_t autonomously on the basis of all variables $\{U_{1:t-1}, Z_{0:t-1}\}$ that it has observed so far. Observable variables are gray. Variables that are hidden to the robot are white. The dashed blue edges correspond to the purpose model. If we remove these edges, we recover the graph for randomly chosen controls, i.e. we switch from feedback control to open loop control.

by following the same derivation as in III-B. We can recover the traditional prediction step (13) from the augmented prediction step (22) if we assume that the purpose model does not depend on x_{t-1} – as it is the case for randomly or autonomously chosen controls. In the latter case, the independence only holds conditioned on $\{Z_{0:t-1}, U_{1:t-1}\}$.

Algorithm 1 contains the pseudo code for updating the belief with incoming control information.

Algorithm 1 The motion update for the augmented Bayes filter.

Input: $bel(x_{t-1})$, u_t
Output: $\overline{bel}(x_t)$
if u_t is selected by an external controller **then**
    $bel_{u_t}(x_{t-1}) = p(u_t \mid x_{t-1}) bel(x_{t-1})$
else if u_t is selected autonomously by the robot **then**
    $bel_{u_t}(x_{t-1}) = bel(x_{t-1})$
end if
$\overline{bel}(x_t) = \int_{x_{t-1}} p(x_t \mid x_{t-1}, u_t) bel_{u_t}(x_{t-1})\, dx_{t-1}$
normalize $\overline{bel}(x_t)$

D. An analytical example

We introduce a simple model to illustrate how the true belief changes for different purpose models. Then, we analytically compute the results of the standard Bayes filter and of the augmented Bayes filter. Consider a vehicle that is supposed to be kept in the middle of the lane. The one-dimensional state space is the displacement of the vehicle with respect to the center of the lane. We define the controls u_t as lateral translations of the car with respect to the lane. In the autonomous case, the controller chooses the action u_t based on the most recent estimate $\hat{x}_{t-1}$:

$$u_t = \pi\left(u_{1:t-1}, z_{0:t-1}, w_t\right) = -\hat{x}_{t-1}, \tag{23}$$

where $\hat{x}_{t-1}$ can depend on all variables observed thus far and independent noise. If a human driver has direct access to the state, she chooses the control deterministically as

$$u_t = \pi\left(x_{t-1}, w_t\right) = -x_{t-1}. \tag{24}$$

We assume a Gaussian motion model and a Gaussian prior

$$p(x_t \mid x_{t-1}, u_t) = \mathcal{N}\left(x_t; x_{t-1} + u_t, \sigma_t^\epsilon\right) \tag{25}$$

$$p\left(x_0\right) = \mathcal{N}\left(x_0; \hat{x}_0, \sigma_0\right). \tag{26}$$

Then, the standard Bayes filter is identical to the Kalman filter. Thus, for T iterations of the motion update, we get

$$\hat{x}_T = \hat{x}_0 + \sum_{t=1}^{T} u_t \quad \text{and} \quad \sigma_T = \sigma_0 + \sum_{t=1}^{T} \sigma_t^\epsilon. \tag{27}$$

Considering the purpose model (23) of the autonomous system, we get $\hat{x}_T = 0$. Note that in both cases, the covariance increases linearly with t as in Brownian motion. However, if we consider the purpose model of the human driver (24), estimate improves. The augmented motion update for our model reads

$$\begin{aligned}
& bel(x_t) \\
\overset{(22)}{=} & \eta' \int_{x_{t-1}} p(x_t \mid x_{t-1}, u_t) p(u_t \mid x_{t-1}) bel(x_{t-1}) dx_{t-1} \\
\overset{(24)(25)}{=} & \eta' \int_{x_{t-1}} \mathcal{N}\left(x_t; x_{t-1} + u_t, \sigma_t^\epsilon\right) \delta_{u_t + x_{t-1}} bel(x_{t-1}) dx_{t-1} \\
= & \eta' \mathcal{N}\left(x_t; 0, \sigma_t^\epsilon\right) bel(x_{t-1} = -u_t) \\
= & \mathcal{N}\left(x_t; 0, \sigma_t^\epsilon\right)
\end{aligned}$$

Thus, after T iterations, we still have

$$\hat{x}_T = 0 \quad \text{and} \quad \sigma_T = \sigma_T^\epsilon. \tag{28}$$

IV. Experiments

A. Indoor localization with a mobile robot

As a first example of how the use of a purpose model improves the estimate, we analyze the localization accuracy of a teleoperated Kuka OmniRob in an industrial indoor scenario. A human controller steers the robot along arbitrary trajectories within the hall. He is requested to execute turning maneuvers preferably only in a safety zone $\mathcal{Z} \subset \mathbb{R}^2$, as depicted in Fig. 3. The instructions for the human controller were kept deliberately vague to reflect the characteristically high uncertainty in

human navigation behavior. A deterministic instruction like a specific maneuver at a specific pose would automatically result in nearly perfect pose knowledge, as in III-D. The robot is equipped with wheel encoders and a laser scanner. We define the motion command as $u_t = [v_t^x, v_t^y, \omega_t]$, with the linear velocities v_t^x and v_t^y and the angular velocity ω_t and model the noise $\epsilon_t = [\mathcal{N}(0, \sigma_t^x), \mathcal{N}(0, \sigma_t^y), \mathcal{N}(0, \sigma_t^\omega)]$ with

$$(\sigma_t^x, \sigma_t^y, \sigma_t^\omega)^T = \begin{pmatrix} 0.15 & 0.05 & 0.05 \\ 0.05 & 0.15 & 0.05 \\ 0.05 & 0.05 & 0.15 \end{pmatrix} (|\, v_t^x \,|, |\, v_t^y \,|, |\, \omega_t \,|)^T \tag{29}$$

A 2-D occupancy grid map is constructed with the ROS package *gmapping*, which is an implementation of laser-based SLAM with a Rao-Blackwellized particle filter as introduced by Grisetti*et al.* [16].The resolution of the grid is $5\,\mathrm{cm}$. For the duration of the mapping phase, the safety zone is marked with small walls to include its boundaries in the map, see Fig. 3. For the localization experiments, these walls are removed so that the robot does not see the safety zone. We perform Monte Carlo localization, where the pseudo measurement update is a weighing of the particles. We define the purpose model as

$$p\left(\omega_t \mid x_{t-1}\right) \propto \begin{cases} 0.15 & \text{for } \omega_t > 0.5\,\mathrm{rad/s} \wedge x_{t-1} \notin \mathcal{Z} \\ 1 & \text{else.} \end{cases}$$

On four different trajectories of two minutes each we evaluate the Monte Carlo localization for the standard filter and the augmented filter with the additional purpose model. We repeat each run 100 times with different random seeds for three modalities: without using the laser scanner (dead reckoning), using a single beam, and using the full scans. In Table I, we present the mean localization errors, their standard deviations over the runs and the p-values resulting from a one-tailed, two-sample t-test.

In Fig. 4, we present the position error in meters for one of the four trajectories. Note that the wave-like behavior in the upper part of Fig. 4 is a side effect for systems with orientation uncertainty that multiply revisit certain locations, see for example the work of Winterhalter *et al.* [4]. It can be explained as follows. For both approaches, the augmented and the standard filter, the orientation error increases when the robot turns, as no lidar information is used. A subsequent linear motion then leads to an increase in the position error. When coming back to the region of the initial rotation, the position error decreases as the estimated trajectory intersects with the ground-truth trajectory. In our case, the minima in the position error correspond to the robot being positioned near the safety zone. While our approach substantially reduces the position error near the safety zone, it only slightly reduces it remote from the safety zone. This is because in this particular example the purpose model does not provide rotational information such that remote from the safety zone only particles without orientation offset profit from the purpose model. As an important result of our approach, it can be observed that the position error near the safety zone of our augmented filter is bound to approximately half a meter, as opposed to the standard approach, where the error increases. The lower part of Fig. 4 represents a selected run with the

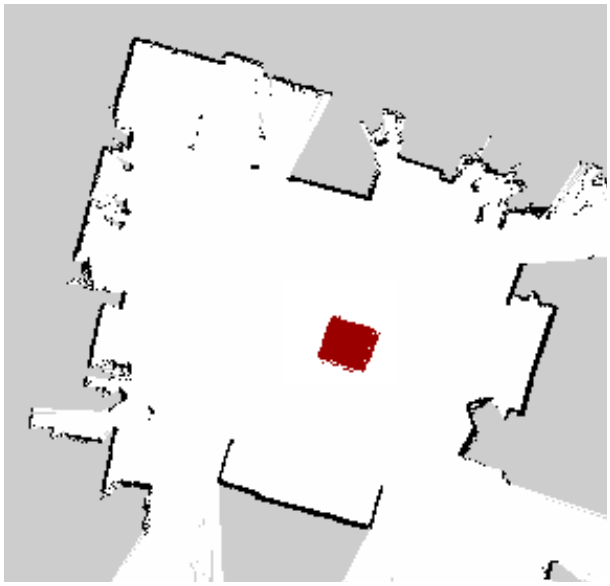


Fig. 3: The map of the indoor localization experiment. The red square in the middle of the room defines the safety zone $\mathcal{Z}$ of approximately $1\,\mathrm{m}^2$.

one-dimensional lidar. The standard filter diverges while the augmented filter remains a bound localization error. With the two-dimensional laser scans, the localization accuracy is always below $10\,\mathrm{cm}$ and thereby of a lower magnitude than the size of the safety zone. Therefore, using the purpose model does not significantly improve the localization accuracy. The effect of the purpose model decreases with the quality of the exteroceptive sensor information, which makes it particularly relevant for featureless, or ambiguous environments. Another important use case is examined in the next section.

Note that in the presented experiment, the robot navigates on a structureless ground, where no position priors can be assigned and no particle violates a physical constraint. Thus the approaches of Min Oh *et al.* [3] and Winterhalter *et al.* [4] are equivalent to the standard approach.

B. Localization on the KITTI dataset

As a second example of how the use of a purpose model improves the estimate, we track a car on a trajectory of the KITTI dataset by Geiger *et al.* [2]. For the ground truth, we use the GPS data. In our experiment, we simulate a failure of the GPS signal so that the localization has to rely on the inertial measurement unit (IMU). This corresponds to the tunnel scenario described in the introduction of this paper. We handcraft the purpose model as

$$p\left(a_t \mid x_{t-1}\right) \propto \begin{cases} 0.15 & \text{for } a_t > 1\,\mathrm{m/s^2} \wedge x_{t-1} \notin \mathcal{Z} \\ 1 & \text{else} \end{cases}$$

with the lateral acceleration a_t from the IMU and $\mathcal{Z} \subset \mathbb{R}^2$ being regions of possible turning points as in the junction depicted in Fig. 5. We use a standard particle filter as in our previous work [17] and apply the purpose model as an update of the particle weights as in the indoor experiments above. We

TABLE I: Mean position error and standard deviation over 100 runs in meters for four different trajectories of two minutes each. The p-values from a one-tailed, two-sample t-test are the probabilities that our method is outperformed by the standard approach.

		No Lidar		
Standard filter	2.824 ± 0.030	2.994 ± 0.030	2.966 ± 0.031	2.870 ± 0.028
Purpose model	1.406 ± 0.026	1.598 ± 0.031	1.663 ± 0.033	1.549 ± 0.032
p-value	0.000	0.000	0.000	0.000
		1D Lidar		
Standard filter	0.476 ± 0.215	0.898 ± 0.521	0.586 ± 0.461	0.685 ± 0.263
Purpose model	0.442 ± 0.029	0.582 ± 0.121	0.419 ± 0.021	0.66 ± 0.259
p-value	0.056	0.000	0.000	0.256
		2D Lidar		
Standard filter	0.078 ± 0.003	0.085 ± 0.003	0.079 ± 0.003	0.075 ± 0.003
Purpose model	0.078 ± 0.004	0.085 ± 0.003	0.079 ± 0.003	0.075 ± 0.003
p-value	0.001	0.215	0.218	0.769

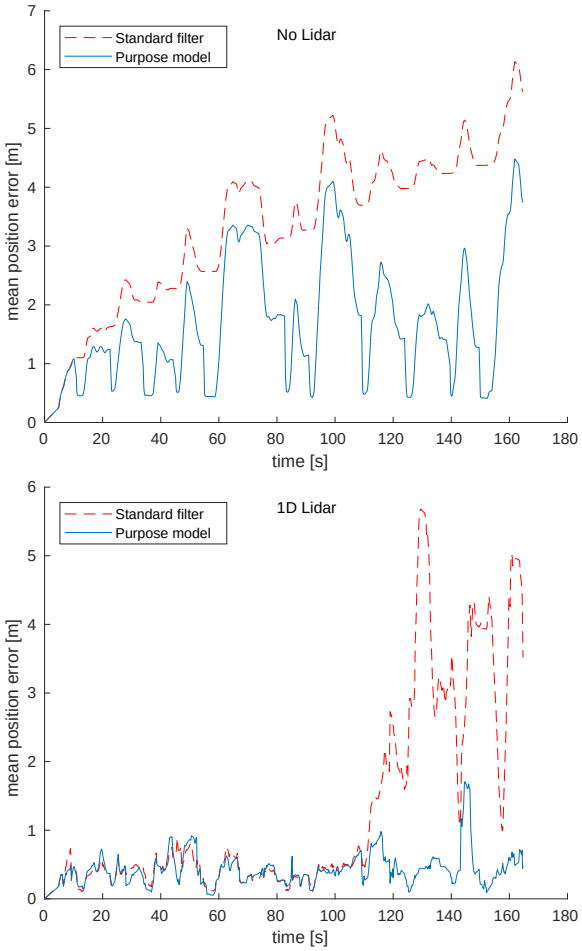


Fig. 4: Localization of our OmniRob with dead reckoning (upper part) and lidar (lower part). While standard lidar localization fails in 11.3 % of the 400 runs (as in this manually selected example), the failure rate of the proposed method is 3.0 %. We define a run to be a failure when the mean position error averaged over the run exceeds 1 m. The results over all trajectories are presented in Table I.

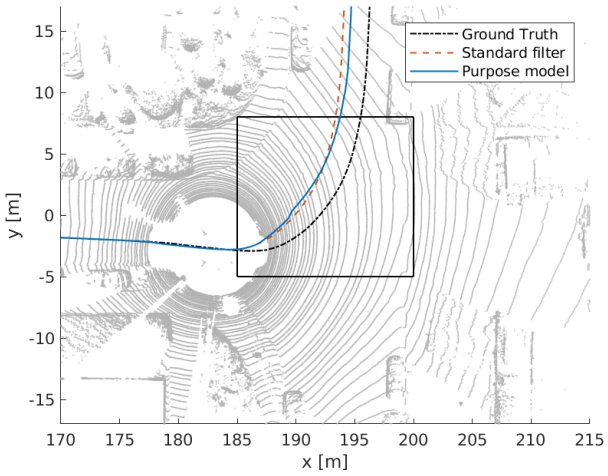


Fig. 5: A laser scan of one of the junctions from the KITTI dataset. We assign low probabilities to lateral accelerations outside the zone $\mathcal{Z}$ marked with the black square.

additionally combine the proposed approach (Purpose model) with the approach of Min Oh *et al.* [3] (Map prior). For the latter, we use a prior proportional to 0.15 for locations not on the road. We present the localization results for a trajectory with two junctions in Fig. 6. The time intervals that the car spends at the junctions are marked with the gray boxes. Here, the car turns at the first junction and goes straight at the second junction. The results show that our approach does not only improve the accuracy when the car turns but also that it does not impair the estimate if the car goes straight.

C. Autonomous navigation with intervention

The following simulation experiment demonstrates how the augmented filter improves the localization in an autonomous navigation task when an external controller intervenes when necessary. It further demonstrates that applying the purpose model to controls chosen autonomously on the basis of the current belief can impair the estimate.

The OmniRob is instructed to follow back and forth a straight line in the hall along the y-axis of the map with

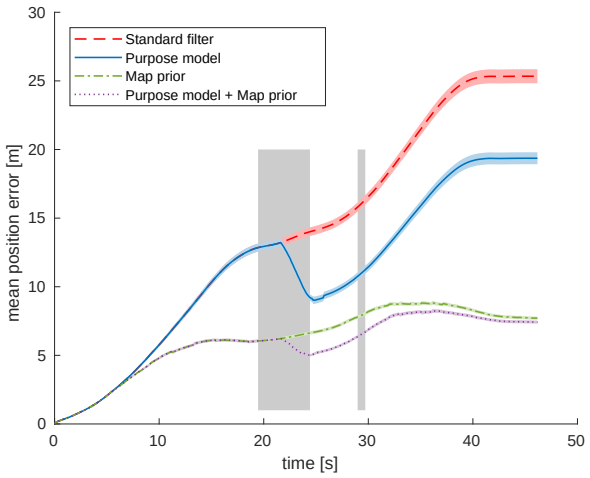


Fig. 6: Mean position errors and their standard errors (bands) on the KITTI dataset for a trajectory with two junctions. The gray areas indicate the time when the ground-truth position is inside a junction zone.

a line follower that takes the Monte Carlo pose estimate as input. We execute runs with the one-dimensional lidar and runs with the full scan. As soon as the ground truth pose is outside a $0.5\,\mathrm{m}$ wide corridor around the desired trajectory, an external proportional controller intervenes with a velocity command directed to the closest point on the trajectory and steering in the direction of the path. When the robot is back on the desired path with a $5\,\mathrm{cm}$ tolerance, the autonomous line following controller takes over again. We repeat the experiment with a $0.25\,\mathrm{m}$ wide corridor. Note that there is an obvious choice of the purpose model that allows localization accuracy of below $5\,\mathrm{cm}$ in x-direction after each intervention. However, we confine to a more uninformed purpose model to demonstrate that we still benefit in the case the exact workings of the external controller are unknown. We define the purpose model of the external controller as $p(u_t = \{v_x, v_y, \omega\} \mid x_{t-1} = \{x, y, \theta\}) = p\,(v_x \mid x)\,p\,(\omega \mid \theta)$ with

$$p\,(v_x \mid x) \propto \begin{cases} 0.15 & \text{if } v \cdot x > 0 \\ 1 & \text{else} \end{cases}$$

where v_x is the x-component of the velocity command and x is the x-component of the position and

$$p\,(\omega \mid \theta) \propto \begin{cases} 0.15 & \text{if } \omega \cdot \theta > 0 \\ 1 & \text{else} \end{cases}$$

where ω is the angular velocity and θ the robot's heading.

We evaluate five different methods: (1) standard Monte Carlo localization (standard filter), (2) the approach of Min Oh *et al.* [3] with a prior proportional to 0.15 outside the tolerance corridor (map priors), (3) a naive application of the purpose model to all controls (purpose model naive), (4) correct application of the purpose model only to the commands chosen by the external controller as in Algorithm 1 (purpose model correct), and (5) a combination of (4) and (2). In Table II we present the results for the one-dimensional lidar with three different metrics: (1) the localization error in meters averaged over all particles (mean error), (2) the distance of the ground truth to the weighted average position over all particles (error of mean), (3) and the distance of the ground truth position to the closest point on the desired trajectory (path deviation). All metrics are averaged over all time steps of twenty runs. The standard deviations are taken over the twenty runs. Comparing the naive and the correct application of the purpose model, we observe that the difference in the error of the means is more significant than the difference between the mean errors, which is consistent with our expectation: A naive application of the purpose model reduces the particle spread (and therefore the mean error if the estimation is not completely off) to a higher degree than it improves the error of the mean.

With the full scan, the robot never leaves the tolerance corridor. Thus, the external controller never takes over and the standard approach and the proposed approach are equivalent. As expected, a naive application of the purpose model impairs the localization: The mean error increases from $4.4\,\mathrm{cm}$ (purpose model correct) to $6.1\,\mathrm{cm}$ (purpose model naive), the error of mean from $3.1\,\mathrm{cm}$ to $5.2\,\mathrm{cm}$, and the path deviation from $1.6\,\mathrm{cm}$ to $3.8\,\mathrm{cm}$, with the variances over the runs being below $0.4\,\mathrm{cm}$ for all metrics.

V. Conclusion

In this paper, we presented a rigorous probabilistic framework to augment the Bayes filter for self-localization with an additional correction step for the case when a model of the action selection – a purpose model – is given. We proved that our augmented filter reduces to the well-known Bayes filter in the case that the actions are chosen (1) autonomously on the basis of a subset of all observable variables or (2) independently of all other variables. In that sense, our filter is a generalization of the Bayes filter. Reverse, we also show that using the purpose model for autonomous systems is invalid and can impair the estimate. Our approach outperforms baseline methods in three different localization scenarios. It is general and serves as a framework for a wide range of applications including navigation and driving with shared autonomy. A general framework to design purpose models for arbitrary situations is subject to future research.

Appendix
The Markov condition for autonomous systems

To prove the Markov condition (9) for autonomously chosen controls, we define a *causal model* for our system.

Definition 1 (Causal model): Assume a set $\mathcal{V} = \{V_j\}_{1 \leq j \leq |\mathcal{V}|}$ of $|\mathcal{V}|$ jointly distributed random variables and a directed acyclic graph $\mathcal{G}$ with nodes $\mathcal{V}$ and edges $\mathcal{E} \subset \{e_{ij}\}_{1 \leq i,j \leq |\mathcal{V}|}$. Assume further that every variable can be written as a deterministic function

$$v_j = f_j\left(\{v_i\}_{V_i \in \mathrm{parents}(V_j)}, w_j\right) \tag{30}$$

where the W_j are mutually independent noise terms and $\mathrm{parents}(V_j)$ is the set of all V_i such that $e_{ij} \in \mathcal{E}$. Then, the set

$$\left\{\mathcal{G}, \{f_j\}_{1 \leq j \leq |\mathcal{V}|}\right\} \tag{31}$$

TABLE II: The localization errors in meters averaged over all particles (mean error), the distance of the ground truth to the weighted average position over all particles (error of mean), and the distance of the ground truth position to the closest point on the desired trajectory (path deviation).

	Wide corridor		
	mean error	error of mean	path deviation
(1) Standard filter	0.676 ± 0.114	0.519 ± 0.106	0.178 ± 0.014
(2) Map prior	0.313 ± 0.004	0.213 ± 0.005	0.167 ± 0.003
(3) Purpose model naive	0.253 ± 0.024	0.245 ± 0.025	0.149 ± 0.011
(4) Purpose model correct	0.237 ± 0.019	$\mathbf{0.143} \pm 0.014$	$\mathbf{0.106} \pm 0.011$
(5) Purpose model correct + Map prior	$\mathbf{0.177} \pm 0.003$	0.152 ± 0.006	0.130 ± 0.008
	Narrow corridor		
	mean error	error of mean	path deviation
(1) Standard filter	0.615 ± 0.061	0.463 ± 0.067	0.088 ± 0.010
(2) Map prior	0.135 ± 0.010	0.121 ± 0.011	0.084 ± 0.007
(3) Purpose model naive	0.168 ± 0.014	0.137 ± 0.017	0.063 ± 0.004
(4) Purpose model correct	0.182 ± 0.016	0.098 ± 0.008	$\mathbf{0.043} \pm 0.005$
(5) Purpose model correct + Map prior	$\mathbf{0.114} \pm 0.002$	$\mathbf{0.098} \pm 0.002$	0.056 ± 0.001

forms a *causal model*. We define $\text{nondescendants}(V_j)$ as the set of variables V_i such that that no directed path that starts in V_j ends in V_i.

To get from the causal model to the desired conditional independences, we use the following Lemma.

Lemma 1 (Causal Markov Condition): In a causal model with the directed acyclic graph $\mathcal{G}$, every variable is independent of its non-descendants given its parents:

$$V_i \perp \text{nondescendants}(V_i) \mid \text{parents}(V_i).$$

Proof: Theorem 1.4.1 in *Causality: Models, Reasoning, and Inference* [15]. ∎

By decomposition of conditional independences, Lemma 1 directly implies

$$V_i \perp V_j \mid \text{parents}(V_i), \tag{32}$$

for all $V_j \in \text{nondescendants}(V_i)$ and all $V_i \in \mathcal{V}$.

The generative, physical model for autonomous systems (5) is deterministic. Therefore, we can find a causal model with Graph $\mathcal{G}$ such that

$$\text{parents}(U_t) = \{Z_{0:t-1}, U_{1:t-1}\}, \tag{33}$$

as in Fig. 2. There are no edges pointing backwards in time and thus no directed path from variables of the time stamp t to variables with the time stamp $< t$ exists. Thus, we have

$$X_{t-1} \in \text{nondescendants}(U_t). \tag{34}$$

Combining (32)-(34) directly yields the independence assumption

$$U_t \perp X_{t-1} \mid \{Z_{0:t-1}, U_{1:t-1}\} \tag{35}$$

and thus the desired Markov condition

$$p(x_{t-1} \mid z_{0:t-1}, u_{1:t}) = p(x_{t-1} \mid z_{0:t-1}, u_{1:t-1}).$$

Equivalently, we can prove the Markov condition (9) by interpreting the graph $\mathcal{G}$ of the causal model for autonomously chosen controls in Fig. 2 as a Bayesian network and apply the *D-separation criterion* [15].

ACKNOWLEDGEMENT

We thank Ronja Neumann for the inspiring discussions.

REFERENCES

[1] S. Thrun, W. Burgard, and D. Fox, *Probabilistic Robotics (Intelligent Robotics and Autonomous Agents)*. The MIT Press, 2005.
[2] A. Geiger, P. Lenz, C. Stiller, and R. Urtasun, "Vision meets robotics: The kitti dataset," *International Journal of Robotics Research*, 2013.
[3] S. Min Oh, S. Tariq, B. Walker, and F. Dellaert, "Map-based priors for localization," in *Intelligent Robots and Systems*, 2004.
[4] W. Winterhalter, F. Fleckenstein, B. Steder, L. Spinello, and W. Burgard, "Accurate indoor localization for RGB-D smartphones and tablets given 2D floor plans," in *Intelligent Robots and Systems*, 2015.
[5] D. Fox, W. Burgard, F. Dellaert, and S. Thrun, "Monte Carlo localization: Efficient position estimation for mobile robots," *Proceedings of the Sixteenth National Conference on Artificial Intelligence*, 1999.
[6] L. P. Kaelbling, M. L. Littman, and A. R. Cassandra, "Planning and acting in partially observable stochastic domains," *Artificial Intelligence*, 1998.
[7] H. Kretzschmar, M. Kuderer, and W. Burgard, "Learning to predict trajectories of cooperatively navigating agents," in *Robotics and Automation*, 2014.
[8] E. Demeester, A. Huntemann, D. Vanhooydonck, G. Vanacker, A. Degeest, H. V. Brussel, and M. Nuttin, "Bayesian estimation of wheelchair driver intents: Modeling intents as geometric paths tracked by the driver," in *Intelligent Robots and Systems*, 2006.
[9] H. Bai, S. Cai, N. Ye, D. Hsu, and W. S. Lee, "Intention-aware online pomdp planning for autonomous driving in a crowd," in *Robotics and Automation*, 2015.
[10] Y. Gu, Y. Hashimoto, L.-T. Hsu, M. Iryo-Asano, and S. Kamijo, "Human-like motion planning model for driving in signalized intersections," *IATSS Research*, 2017.
[11] A. Martinelli and R. Siegwart, "Estimating the odometry error of a mobile robot during navigation," 2003.
[12] R. Kümmerle, G. Grisetti, and W. Burgard, "Simultaneous parameter calibration, localization, and mapping," *Advanced Robotics*, 2012.
[13] S. Javdani, J. A. Bagnell, and S. S. Srinivasa, "Shared autonomy via hindsight optimization," *Robotics science and systems*, 2015.
[14] S. Reddy, A. D. Dragan, and S. Levine, "Shared autonomy via deep reinforcement learning," *CoRR*, 2018.
[15] J. Pearl, *Causality: Models, Reasoning, and Inference 2nd edn*. Cambridge: Cambridge University Press, 2009.
[16] G. Grisetti, C. Stachniss, and W. Burgard, "Improving grid-based SLAM with rao-blackwellized particle filters by adaptive proposals and selective resampling," in *Robotics & Automation*, 2005.
[17] L. Luft, A. Schaefer, T. Schubert, and W. Burgard, "Closed-form full map posteriors for robot localization with lidar sensors," in *Intelligent Robots and Systems*, 2017.