Nonmyopic Distilled Data Association Belief Space Planning Under Budget Constraints

Moshe Shienman(✉) and Vadim Indelman

Technion - Israel Institute of Technology, 32000 Haifa, Israel
smoshe@campus.technion.ac.il, vadim.indelman@technion.ac.il

Abstract. Autonomous agents operating in perceptually aliased environments should ideally be able to solve the data association problem. Yet, planning for future actions while considering this problem is not trivial. State of the art approaches therefore use multi-modal hypotheses to represent the states of the agent and of the environment. However, explicitly considering all possible data associations, the number of hypotheses grows exponentially with the planning horizon. As such, the corresponding Belief Space Planning problem quickly becomes unsolvable. Moreover, under hard computational budget constraints, some non-negligible hypotheses must eventually be pruned in both planning and inference. Nevertheless, the two processes are generally treated separately and the effect of budget constraints in one process over the other was barely studied. We present a computationally efficient method to solve the nonmyopic Belief Space Planning problem while reasoning about data association. Moreover, we rigorously analyze the effects of budget constraints in both inference and planning.

Keywords: Planning under uncertainty · Robust perception · SLAM

1 Introduction

Intelligent autonomous agents and robots are expected to operate reliably and efficiently under different sources of uncertainty. There are various possible reasons for such uncertainty, including noisy measurements; imprecise actions; and dynamic environments in which some events are unpredictable. In these settings, autonomous agents are required to reason over high-dimensional probabilistic states known as beliefs. A truly autonomous agent should be able to perform both inference, i.e. maintain a belief over the high-dimensional state space given available information, and decision making under uncertainty. The latter is also known as the Belief Space Planning (BSP) problem, where the agent should autonomously determine its next best actions while reasoning about future belief evolution. However, both inference and BSP are computationally expensive and practically infeasible in real-world autonomous systems where the agent is required to operate in real time using inexpensive hardware.

This research was partially supported by US NSF/US-Israel BSF, and by Israel PBC-VATAT and the Technion Artificial Intelligence Hub (Tech.AI).

© The Author(s), under exclusive license to Springer Nature Switzerland AG 2023
A. Billard et al. (Eds.): ISRR 2022, SPAR 27, pp. 102–118, 2023.
https://doi.org/10.1007/978-3-031-25555-7_8

In real-world scenarios, an autonomous agent should also be resilient to the problem of ambiguous measurements. These ambiguities occur when a certain observation has more than one possible interpretation. Some examples include the slip/grip behavior of odometry measurements; the loop closure problem in visual Simultaneous Localization and Mapping (SLAM); and unresolved data association. The latter is defined as the process of associating uncertain measurements to known tracks, e.g. determine if an observation corresponds to a specific landmark within a given map. Most existing inference and BSP algorithms assume data association to be given and perfect, i.e. assume a single hypothesis represented by a uni-modal state and map estimates. Yet, in perceptually aliased environments, this assumption is not reasonable and could lead to catastrophic results. Therefore, it is crucial to reason about data association, in both inference and planning, while also considering other sources of uncertainty.

Explicitly reasoning about data association, the number of hypotheses grows exponentially with time. As such, when considering real time operation using inexpensive hardware, hard computational constraints are often required, e.g. bounding the number of supported hypotheses. State of the art inference and planning approaches therefore use different heuristics, e.g. pruning and merging, to relax the computational complexity. However, this loss of information incurs loss in solution quality and there are usually no performance guarantees. Moreover, inference and planning are commonly treated separately and it is unclear how budget constraints in one process affect another.

In this work we extend our presented approach in [17] to a nonmyopic setting. Specifically, we handle the exponential growth of hypotheses in BSP by solving a *simplified* problem while providing performance guarantees. To that end, we analyze for the first time, the construction of a belief tree within planning given a mixture belief, e.g. Gaussian Mixture Models (GMM). We further show how to utilize the *skeleton* of such belief tree to reduce the computational complexity in BSP. Crucially, this paper thoroughly studies, for the first time, the impacts of hard budget constraints in either planning and/or inference.

2 Related Work

Several approaches were recently proposed to ensure efficient and reliable operation in ambiguous environments. Known as robust perception, these approaches typically maintain probabilistic data association and hypothesis tracking.

A good inference mechanism should handle false data association made by front-end algorithms and be computationally efficient. The authors of [16] recently suggested to re-use hypotheses' weights from previous steps to reduce computational complexity and improve current-time hypotheses pruning. Convex relaxation approaches over graphs were proposed in [2,13] to capture perceptual aliasing and find the maximal subset of internally coherent measurements, i.e. correct data association. The max-mixture model was presented in [14] to allow fast maximum-likelihood inference on factor graphs [12] that contain arbitrarily complex probability distributions such as the slip/grip multi modal problem. The authors of [7,8] used factor graphs with an expectation-maximization

approach to efficiently infer initial relative poses and solve a multi robot data association problem. In [21] the topological structure of a factor graph was modified during optimization to discard false positive loop closures. The Bayes tree algorithm [10] was extended in [5,9] to explicitly incorporate multi-modal measurements within the graph and generate multi-hypothesis outputs. These works, however, were only developed for the purpose of inference, i.e. without planning.

Ambiguous data association was also considered in planning. In [1] a GMM was used to model prior beliefs representing different data association hypotheses. However, the authors did not reason about ambiguous data association within future beliefs (owing to future observations), i.e. they assumed that it is solved and perfect in planning. In [15] the authors introduced DA-BSP where, for the first time, reasoning about future data association hypotheses was incorporated within a BSP framework. The ARAS framework proposed in [6] leveraged the graphical model presented in [5] to reason about ambiguous data association in future beliefs. All of these approaches handled the exponential growth in the number of hypotheses by either pruning or merging. The first work to also provide performance guarantees on the loss in solution quality was presented in [17]. Yet, the authors only considered a myopic setting.

The notion of simplification was introduced in [4], where, the authors formulated the loss in solution quality in BSP problems via bounds over the objective function. However, they only considered the Gaussian case and a maximum likelihood assumption. The authors of [23] used bounds as a function of *simplified* beliefs to reduce the computational complexity in nonmyopic BSP problems with general belief distributions. In [22] they incorporated this concept within a Monte Carlo Tree Search (MCTS) planning framework, i.e. without assuming that the belief tree is given, which is complimentary to our approach. Yet, they did not handle ambiguous data association nor budget constraints aspects.

3 Background and Notations

In this section we review some basic concepts from estimation theory and BSP which we will use in the following sections.

3.1 Inference

Consider an autonomous agent operating in a partially known or pre-mapped environment containing similar landmarks or scenes. The agent acquires observations and tries to infer random variables of interest that are application dependent while reasoning about data association.

We denote the agent's state at time instant k by x_k. Let $Z_k \triangleq \{z_{k,1}, ..., z_{k,n_k}\}$ denote the set of all n_k measurements and let u_k denote the agent's action. $Z_{1:k}$ and $u_{0:k-1}$ denote all observations and actions up to time k, respectively. The motion and observation models are given by

$$x_{k+1} = f\left(x_k, u_k, w_k\right), \quad z_k = h\left(x_k, x^l, v_k\right), \tag{1}$$

where x^l is a landmark pose and w_k and v_k are noise terms, sampled from known motion and measurement distributions, respectively.

Given n_k observations, the data association realization vector is denoted by $\beta_k \in \mathbb{N}^{n_k}$. Elements in β_k are associated according to the given observation model and each element, e.g. landmark, is given a unique label. A specific data association hypothesis is thus given by a specific set j of associations up to and including time k and is denoted as $\beta^j_{1:k}$.

At each time step the agent maintains a posterior belief over both continuous and discrete variables given by

$$b\left[x_k, \beta_{1:k}\right] \triangleq \mathbb{P}\left(x_k, \beta_{1:k} | z_{0:k}, u_{0:k-1}\right) = \mathbb{P}\left(x_k, \beta_{1:k} | H_k\right), \tag{2}$$

where $H_k \triangleq \{Z_{1:k}, u_{0:k-1}\}$ represents history. Using the chain rule, the belief becomes a mixture and can be written as a linear combination of $|M_k|$ hypotheses

$$b_k = \sum_{j \in M_k} \underbrace{\mathbb{P}\left(x_k | \beta^j_{1:k}, H_k\right)}_{b^j_k} \underbrace{\mathbb{P}\left(\beta^j_{1:k} | H_k\right)}_{w^j_k}, \tag{3}$$

where b^j_k is a conditional belief, with some general distribution, and w^j_k is the associated weight. Therefore, M_k is a set of maintained weighted conditional beliefs, representing different data association hypotheses. In this work, we interchangeably refer to each b^j_k as both a hypothesis and a component.

Each conditional belief hypothesis b^j_k in (3) can be efficiently calculated by maximum a posteriori inference, e.g. as presented in [10] for the Gaussian case. Nevertheless, our formulation and approach also applies to a non-parametric setting. Each component weight w^j_k is calculating by marginalizing over the state space and applying the Bayes rule (as developed in [15,17]).

Reasoning about data association, without any computational constraints, the number of considered hypotheses grows exponentially with time. In general, such belief is a function of $b_k = \psi_k\left(b_{k-1}, u_{k-1}, Z_k\right)$. However, under hard computational constraints, the number of hypotheses is bounded by $\mathcal{C} \in \mathbb{N}$. Therefore, the belief in each time step is a function of

$$b^\psi_k = \psi^{\mathcal{C}}_k\left(b_{k-1}, u_{k-1}, Z_k, \mathcal{C}\right), \tag{4}$$

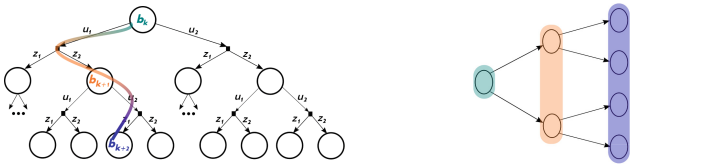
where $\psi^{\mathcal{C}}_k$ contains some heuristic function h^{inf} such that $|M^\psi_k| \leq \mathcal{C}$.

3.2 DA-BSP

Given a posterior belief (3) and a set of candidate action sequences $\mathcal{U}$ the goal of BSP is to find the optimal action sequence that would minimize/maximize a certain objective function. We note that while in this paper we consider, for simplicity, action sequences, our approach is applicable also to policies.

Reasoning about data association in planning, a user defined objective function J can be written as

$$J\left(b_k, u_{k:k+N-1}\right) = \underset{\beta_{(k+1)+}}{\mathbb{E}}\left[\underset{Z_{(k+1)+}|\beta_{(k+1)+}}{\mathbb{E}}\left[\sum_{n=1}^{N} c\left(b_{k+n}, u_{k+n-1}\right)\right]\right], \tag{5}$$



(a) belief tree (b) exponential growth of hypotheses

Fig. 1. (a) A belief tree constructed during planning. Each node represents a posterior belief (3); The number of belief components grows exponentially along the highlighted path as presented in (b).

where $\beta_{(k+1)+} \triangleq \beta_{k+1:k+N}$, $Z_{(k+1)+} \triangleq Z_{k+1:k+N}$ and $c(\cdot)$ denotes a cost function. The expectation is taken with respect to both future data association realizations and observations. The optimal action sequence $u^*_{k:k+N-1}$ is defined as

$$u^*_{k:k+N-1} = \underset{\mathcal{U}}{\operatorname{argmin}}\, J\left(b_k, u_{k:k+N-1}\right). \tag{6}$$

To solve (6) we need to consider all possible future realizations of Z_{k+n} for every $n \in [k+1, k+N]$ while marginalizing over all possible locations and data association realizations (see Section 5.2 in [15]). However, solving these integrals analytically is typically not feasible. In practice, the solution should be approximated by sampling future observations from the relevant distributions. Using these samples, the agent constructs and traverses a belief tree (as shown in Fig. 1a) which branches according to future actions and observations.

Nevertheless, the number of hypotheses grows exponentially with the planning horizon (see Fig. 1b). Specifically, given $|M_k|$ hypotheses and D data association realizations, i.e. different β_{k+i} at each look-ahead step, the number of belief components at the nth look-ahead step is $|M_{k+n}| = |M_k|\,|D|^n$. As such, considering every possible future hypothesis is not practical.

4 Methodology

In this section we first describe how to construct a belief tree *skeleton* during planning. We then present a general framework to reduce the computational complexity when solving a sampling based approximation of (5). Finally, we analyze the implications of using our proposed framework under different conditions.

4.1 Constructing the Belief Tree *skeleton*

Previous works addressed the exponential growth of the belief tree with the planning horizon without reasoning about data association. In this work we analyze and describe, for the first time, the structure of a belief tree given a mixture belief such as (3). In this setting there is an additional exponential growth in the number of belief components for every considered future observation realization

(see Fig. 1). These realizations are functions of future beliefs (3), data association realizations and actions

$$\mathbb{P}(Z_{k+1:k+n}|b_k, u_{k:k+n-1}, \beta_{k+1:k+n}). \tag{7}$$

To construct the belief tree in practice, we sample states from beliefs, sample data association given states and finally sample observations from (7).

Our key observation is that in order to construct a belief tree *skeleton*, i.e. without explicitly calculating or holding posterior beliefs at each node, we can sample future observations in two different ways. We describe these two options for a planning horizon of $n = 2$. Specifically, we can either rewrite (7) as

$$\mathbb{P}(Z_{k+2}|b_{k+1|\beta_{k+1}}, u_{k+1}, \beta_{k+2})\mathbb{P}(Z_{k+1}|b_k, u_k, \beta_{k+1}), \tag{8}$$

where $b_{k+1|\beta_{k+1}}$ is a posterior belief and each term is evaluated by integrating over $x_{k+1:k+2}$, or, by first integrating and then applying the chain rule as

$$\int_{x_{k+2}} \mathbb{P}(Z_{k+2}|x_{k+2}, \beta_{k+2}) \int_{x_{k+1}} \mathbb{P}(x_{k+2}|x_{k+1}, u_{k+1})\mathbb{P}(Z_{k+1}|x_{k+1}, \beta_{k+1})\mathbb{P}(x_{k+1}|b_k, u_k). \tag{9}$$

While these two expressions are analytically identical, they represent two different processes of sampling. In the former observations are sampled from posterior beliefs, while in the latter observations are sampled using the motion and observation models, similar to the MCTS particle trajectories techniques in [20,24].

Algorithm 1: Construct belief tree *skeleton*

Input: prior belief b_k, action sequence $u_{k:k+n-1}$
Output: sampled future observations $Z_{k+1:k+n}$

```
Z = ∅
x_k ~ b_k
for i ∈ [1, n] do
    x_{k+i} ~ P(x_{k+i} | x_{k+i-1}, u_{k+i-1})
    determine β_{k+i} based on x_{k+i}
    Z_{k+i} ~ P(Z_{k+i} | x_{k+i}, β_{k+i})
    Z = Z ∪ Z_{k+i}
return Z
```

To avoid the explicit representation of the exponential number of belief components, in this work we sample future observations using (9) and bypass the inference stage. We formulate this sampling method in Algorithm 1.

Yet, this is of little help if the posterior belief is required for calculating the cost function itself. We next describe our approach to avoid these calculations.

4.2 Nonmyopic Distilled Data Association BSP

Our goal is to reduce the computational complexity of nonmyopic BSP problems where ambiguous data association is explicitly considered, i.e. solving (6) efficiently. We start by writing (5) in a recursive form

$$J\left(b_k, u_{k:k+N-1}\right) = c\left(b_k, u_k\right) + \underset{\beta_{k+1}}{\mathbb{E}}\left[\underset{Z_{k+1}|\beta_{k+1}}{\mathbb{E}}\left[J\left(b_{k+1}, u_{k+1:k+N-1}\right)\right]\right]. \quad (10)$$

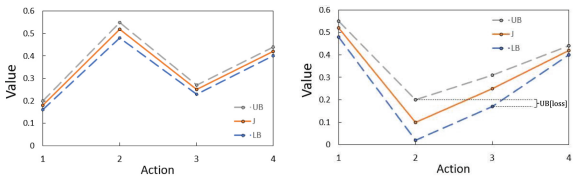
As in practice we approximate the solution via samples, we rewrite (10) as

$$\hat{J}\left(b_k, u_{k:k+N-1}\right) = c\left(b_k, u_k\right) + \underset{\beta_{k+1}}{\hat{\mathbb{E}}}\left[\underset{Z_{k+1}|\beta_{k+1}}{\hat{\mathbb{E}}}\left[\hat{J}\left(b_{k+1}, u_{k+1:k+N-1}\right)\right]\right]. \quad (11)$$

Using Bellman's principle of optimality, the optimal solution for (11) is

$$\hat{J}\left(b_k, \hat{u}^*_{k:k+N-1}\right) = \min_{u_k}\{c\left(b_k, u_k\right) + \underset{\beta_{k+1}}{\hat{\mathbb{E}}}\left[\underset{Z_{k+1}|\beta_{k+1}}{\hat{\mathbb{E}}}\left[\hat{J}\left(b_{k+1}, u^*_{k+1:k+N-1}\right)\right]\right]\}, \quad (12)$$

where $\hat{u}^*_{k:k+N-1} = \underset{\mathcal{U}}{\operatorname{argmin}}\, \hat{J}\left(b_k, u_{k:k+N-1}\right)$. To reduce the computational complexity in (12), we propose utilizing the belief tree *skeleton*, without having access to posterior beliefs, to solve an easier to compute version of the considered cost function.



(a) No overlap between bounds (b) Bounds overlap

Fig. 2. BSP using bounds over the objective function. In (a) choosing action #1 is guaranteed to be optimal as the corresponding upper bound is lower than all other lower bounds; In (b) choosing action #2 is not guaranteed to be optimal. The loss in solution quality, however, is upper bounded.

In general, the cost function over the original beliefs can be bounded using a *simplified* belief b^s_k as

$$\underline{c}\left(b^s_k, u_k\right) \leq c\left(b_k, u_k\right) \leq \bar{c}\left(b^s_k, u_k\right). \quad (13)$$

We note that this formulation also supports replacing the cost function itself with a computationally simpler function, as in [11,19].

Using the belief tree *skeleton* and some method to calculate the *simplified* beliefs, to be defined, we now traverse the belief tree from the leafs upwards. At each belief tree node the bounds over the objective function (5) are calculated recursively using the Bellman equation (12) and (13) for every $n \in [0, N-1]$

$$\underline{J}\left(b_{k+n}, u_{(k+n)+}\right) = \underline{c}\left(b^s_{k+n}, u_{k+n}\right) + \underset{\beta_{k+1}}{\hat{\mathbb{E}}}\left[\underset{Z_{k+n+1}|\beta_{k+1}}{\hat{\mathbb{E}}}\left[\underline{J}\left(b_{k+n+1}, u_{(k+n)+}\right)\right]\right],$$

$$\bar{J}\left(b_{k+n}, u_{(k+n)+}\right) = \bar{c}\left(b^s_{k+n}, u_{k+n}\right) + \underset{\beta_{k+1}}{\hat{\mathbb{E}}}\left[\underset{Z_{k+n+1}|\beta_{k+1}}{\hat{\mathbb{E}}}\left[\bar{J}\left(b_{k+n+1}, u_{(k+n)+}\right)\right]\right], \quad (14)$$

where $u_{(k+n)+} \triangleq u_{k+n:k+N-1}$. If these bounds do not overlap (see Fig. 2a), one can guarantee to select the optimal action sequence as in (12).

Our general Nonmyopic Distilled Data Association BSP (ND2A-BSP) approach is presented in Algorithm 2. The algorithm receives a belief tree *skeleton*; a heuristic function h used to select the subsets of hypotheses in each belief tree node, i.e. defines b^s_{k+n}; and a decision rule R which decides whether the considered subsets are enough, e.g. when no overlap between bounds is required or when calculations exceed a user defined time threshold, providing anytime performance guarantees. The algorithm returns the best action sequence, given the computational constraints, and an upper bound on the loss in solution quality.

It is worth mentioning that our approach can be adapted to a setting where the belief tree construction is coupled with Q function estimates, e.g. using MCTS and Upper Confidence Bound (UCB) techniques [20], following a similar approach to the one presented in [22]. However, we emphasize that as the belief tree *skeleton* approximates (10) via samples, our method provides performance guarantees with respect to that specific *skeleton*, i.e. with respect to (12). Not to be confused with the asymptotic guarantees of MCTS approaches, with respect to the theoretical problem (10), which is an entirely different aspect not related to the approach presented in this paper.

Algorithm 2: Generic Nonmyopic Distilled Data Association BSP

Input: belief tree *skeleton* T, simplification heuristic h, decision rule R
Output: action sequence u^*, loss

Function ND2A-BSP(T, h, R):
  $LB^*, UB^*, loss =$ PLAN$(T.root, h, R)$
  $u^* \leftarrow$ corresponding to LB^*, UB^*
  return $u^*, loss$
Function PLAN$(Node, h, R)$:
  $Node.b^s_{k+n} \leftarrow h(Node)$
  if $Node$ is a leaf **then**
    return $\underline{c}\left(Node.b^s_{k+n}\right), \bar{c}\left(Node.b^s_{k+n}\right), 0$ // $loss = 0$ at leaf
  $Node.bounds = \emptyset$
  foreach child C of $Node$ **do**
    $lb, ub, loss \leftarrow$ ND2A-BSP(C, h, R)
    $LB \leftarrow \underline{c}\left(Node.b^s_{k+n}\right) + lb$ // objective lower bound (14)
    $UB \leftarrow \bar{c}\left(Node.b^s_{k+n}\right) + ub$ // objective upper bound (14)
    $Node.bounds = Node.bounds \cup (LB, UB)$
  while $R(Node.bounds)$ is not satisified **do**
    ND2A-BSP$(Node, h, R)$ // further simplification is needed
  $LB^*, UB^*, loss \leftarrow Node.bounds$
  return $LB^*, UB^*, loss$

We now analyze different settings, within inference and planning, where the agent either has or does not have hard budget constraints. To the best of our knowledge, this is the first time that these aspects are addressed in works that attempt to reduce the computational complexity of the planning problem. The differences between the considered settings are summarized in Table 1.

4.3 No Budget Constraints in Inference

In this section we assume that there are no constraints in inference, i.e. each belief tree node can theoretically hold every possible hypothesis within the planning

Table 1. A summary of the considered scenarios, with respect to budget constraints on the number of supported hypotheses in each algorithm, for each considered case. Cases 1&2 are presented in Sect. 4.3 while cases 3&4 are presented in Sect. 4.4

	Budget constraints in inference	Budget constraints in planning
Case 1	✗	✗
Case 2	✗	✓
Case 3	✓	✗
Case 4	✓	✓

horizon. The objective of inference however is different than the main goal of BSP. In inference the agent tries to represent the considered state as accurately as possible while in planning the goal is to retrieve the optimal action sequence or policy. As such, in this setting, the problems are decoupled (see Fig. 3a).

We now further separate between two cases, when the planning algorithm either has budget constraints or not. In both cases, each belief tree node still has an exponential number of components, which we avoid calculating explicitly.

Case 1. With no budget constraints in planning we propose bounding the cost function as

$$\underline{c}\left(b_k, u_{k+}, Z_{(k+1)+}, b^s_{k+n}\right) \leq c\left(b_{k+n}, u_{k+n}\right) \leq \bar{c}\left(b_k, u_{k+}, Z_{(k+1)+}, b^s_{k+n}\right). \quad (15)$$

where $u_{k+} \triangleq u_{k:k+n-1}$ and $Z_{(k+1)+} \triangleq Z_{k+1:k+n}$. A key difference from the approach presented in [23] is that these bounds are not functions of b_{k+n}.

As the number of belief components grows exponentially we avoid calculating $c\left(b_{k+n}\right)$. Instead, we calculate a *simplified* belief b^s_{k+n}, using Bayesian updates via $u_{k:k+n-1}$ and $Z_{k+1:k+n}$, only for specific components from the prior belief b_k. This extends our proposed approach in [17] to the nonmyopic case. Each *simplified* belief is formally defined, using $M^s_{k+n} \subseteq M_{k+n}$ components, as

$$b^s_{k+n} \triangleq \sum_{r \in M^s_{k+n}} w^{s,r}_{k+n} b^r_{k+n}, \quad w^{s,r}_{k+n} \triangleq \frac{w^r_{k+n}}{w^{m,s}_{k+n}}, \quad (16)$$

where $w^{m,s}_{k+n} \triangleq \sum_{m \in M^s_{k+n}} w^m_{k+n}$ is used to re-normalize each corresponding weight. Most importantly, a *simplified* belief b^s_{k+n} is calculated using only a subset of hypotheses, i.e. without calculating the posterior belief b_{k+n}.

Using Algorithm 2 given a decision rule R, with no overlap between bounds (14), and a heuristic h, e.g. which chooses hypotheses greedily based on prior weights, we guarantee the selection of the optimal actions sequence, with respect to the specific belief tree, while reducing the computational complexity.

Case 2. Under budget constraints in planning, the algorithm can use up to $\mathcal{C}$ components, in each *simplified* belief b^s_{k+n}, to calculate the bounds in (15). Yet, each subset of components is chosen independently w.r.t. b_{k+n} which develops

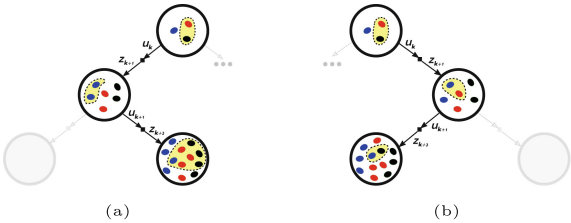


Fig. 3. No budget constraints in inference. Colors denote components generated from previous time steps. (a) Planning without budget constraints, the algorithm can choose any subset of components, highlighted in yellow, in each node to evaluate the bounds; (b) With budget constraints in planning, each subset selection is bounded in size by $\mathcal{C} = 2$.

exponentially, i.e. hypotheses chosen in time steps $k+n$ and $k+n+i$ are not necessarily related (see Fig. 3b).

In this setting, the number of possible distilled subsets for each b^s_{k+n} is $\binom{|M_{k+n}|}{\mathcal{C}}$ which can be very high. Moreover, there are no guarantees that the bounds between candidate actions would not overlap. However, using the bounds in (15), our proposed approach can yield the worst-case loss in solution quality, i.e. provide performance guarantees (see Fig. 2b).

4.4 Hard Budget Constraints in Inference

In the previous section we only considered that the belief at the root of the tree is provided from inference. As the posterior beliefs within the constructed belief tree were with an exponentially increasing number of components, i.e. without budget constraints, the key idea was to avoid making explicit inferences. Instead, we calculated bounds that utilized, under budget constraints in planning, a fixed number of components. In practice, however, real world autonomous systems do not work that way. Instead, they are often required to operate in real time using inexpensive hardware with hard computational budget constraints in both inference and planning.

Under hard budget constraints on the number of considered hypotheses in inference, the posterior belief in each belief tree node is determined by (4), i.e. $|M^{\psi}_{k+n}| \leq \mathcal{C}$ under some heuristic h^{inf}. Moreover, once a hypothesis is discarded in time step k it is no longer considered in future time steps. Yet, the decision regarding which components to choose, while calculating the bounds in planning, depends on either if the heuristic in (4) is given or determined within planning. To the best of our knowledge, the latter is a novel concept never considered.

Case 3. In this setting we consider the heuristic in (4) to be given within planning, i.e. posterior belief tree nodes exactly represent how the belief would evolve in inference under (4). In contrast to Sect. 4.3, as the number of components does not grow exponentially, we sample future observations according to (8) and construct the belief tree explicitly, i.e. perform inference in each node. Therefore, the planning algorithm can no longer choose any subset of components for each

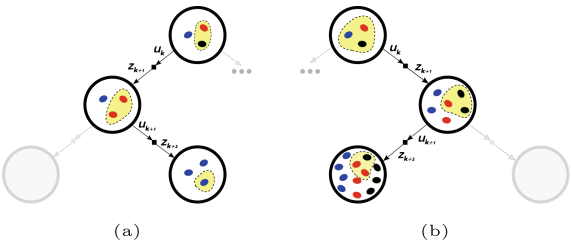


Fig. 4. Hard budget constraints in inference. Colors denote components generated from previous time steps. (a) Planning given the heuristic in inference, the algorithm can only evaluate the bounds using components that represent how the belief would evolve in inference; (b) The planning algorithm is free to choose components under any valid heuristics in inference given the budget $\mathcal{C}$. Each selected component in time step $k+n+1$ must originate from a selected component in time step $k+n$.

b^s_{k+n}, i.e. hypotheses discarded in time step $k+n$ cannot be considered in time step $k+n+1$ (see Fig. 4).

The bounds over the considered cost are now a function of the belief in the previous time step under (4). Specifically, we rewrite them as

$$\underline{c}\left(b^{\psi}_{k+n-1}, u_{k+n-1}, Z_{k+1}, b^s_{k+n}\right) \leq c\left(b^{\psi}_{k+n}, u_{k+n}\right) \leq \bar{c}\left(b^{\psi}_{k+n-1}, u_{k+n-1}, Z_{k+1}, b^s_{k+n}\right). \tag{17}$$

These bounds represent a recursive setting in contrast to the bounds in (15).

Using our approach iteratively in each time step, reduces the computational complexity of the considered cost function in planning while providing performance guarantees. As each posterior belief is determined by inference (Fig. 4a), performance guarantees are with respect to the given heuristic in inference (4).

Case 4. We now relax the assumption that the planning algorithm is confined to the specific heuristic in (4). Unlike in Case 2, where each subset of components can be used in each node to calculate the bounds, this setting has an additional constraint. We formulate this by representing the bounds from (15) in two consecutive time steps

$$\begin{aligned}
&\underline{c}\left(b_k, u_{k+}, Z_{(k+1)+}, b^s_{k+n}\right) \leq c\left(b_{k+n}\right) \leq \bar{c}\left(b_k, u_{k+}, Z_{(k+1)+}, b^s_{k+n}\right), \\
&\underline{c}\left(b_k, u_{k+}, Z_{(k+1)+}, b^s_{k+n+1}\right) \leq c\left(b_{k+n+1}\right) \leq \bar{c}\left(b_k, u_{k+}, Z_{(k+1)+}, b^s_{k+n+1}\right), \\
&\text{s.t.} \left|M^s_{k+n}\right|, \left|M^s_{k+n+1}\right| \leq \mathcal{C} \quad \text{and} \quad \forall b^{s,ij}_{k+n+1} \in b^s_{k+n+1} \Rightarrow b^{s,j}_{k+n} \in b^s_{k+n},
\end{aligned} \tag{18}$$

where $b^{s,j}_{k+n}$ denotes the jth hypothesis in the *simplified* subset b^s_{k+n} and $b^{s,ij}_{k+n+1}$ denotes the ith hypothesis in the *simplified* subset b^s_{k+n+1}, originated from $b^{s,j}_{k+n}$, i.e. as in Fig. 1b.

The components chosen in the sequence of bounds (18) which minimizes the loss, w.r.t. the original problem, define a heuristic $h^{p\star}$ (see Fig. 7c), which is valid in inference. The heuristic $h^{p\star}$ can be used with any BSP approach to solve (12) and to reduce computational complexity, using our approach, as described in

Case 3. To the best of our knowledge, leveraging $h^{p\star}$ is a novel concept. We note that while $h^{p\star}$ minimizes the loss in planning, it is generally different than h^{inf}. As such, the implications of utilizing such heuristic in inference are not straightforward. The study of such mechanism is left for future research.

4.5 The Cost Function

While the formulation thus far was for a general cost function, in this section we focus on active disambiguation of hypotheses. Specifically, we utilize the Shannon entropy, defined over posterior belief components weights. The cost for a belief b_{k+n} with M_{k+n} components is thus given by $\mathcal{H}_{k+n} \triangleq c\left(b_{k+n}\right) = -\sum_{r\in M_{k+n}} \frac{w^r_{k+n}}{\eta_{k+n}} log\left(\frac{w^r_{k+n}}{\eta_{k+n}}\right)$, where $\eta_{k+n} \triangleq \sum_{r\in M_{k+n}} w^r_{k+n}$. Similarly, for a simplified belief b^s_{k+n} with $M^s_{k+n} \subseteq M_{k+n}$ the cost is given by $\mathcal{H}^s_{k+n} \triangleq c\left(b^s_{k+n}\right) = -\sum_{r\in M^s_{k+n}} w^{s,r}_{k+n} log\left(w^{s,r}_{k+n}\right)$.

To allow fluid reading, proofs for all theorems and corollaries are given in the supplementary material [18].

Theorem 1. *For each belief tree node representing a belief b_{k+n} with M_{k+n} components and a subset $M^s_{k+n} \subseteq M_{k+n}$ the cost can be expressed by*

$$\mathcal{H}_{k+n} = \frac{w^{m,s}_{k+n}}{\eta_{k+n}}\left[\mathcal{H}^s_{k+n} + log\left(\frac{\eta_{k+n}}{w^{m,s}_{k+n}}\right)\right] - \sum_{r\in\neg M^s_{k+n}} \frac{w^r_{k+n}}{\eta_{k+n}} log\left(\frac{w^r_{k+n}}{\eta_{k+n}}\right), \quad (19)$$

where $\neg M^s_{k+n} \triangleq M_{k+n} \setminus M^s_{k+n}$.

Using Theorem 1, we derive bounds for $\mathcal{H}_{k+n}$ which are computationally more efficient to calculate as we only consider a subset of hypotheses. However, as evaluating η_{k+n} requires by definition evaluating all posterior components weights, which we do not have access to, we need to bound this term as well (denoted below as $\mathcal{LB}\left[\eta_{k+n}\right]$ and $\mathcal{UB}\left[\eta_{k+n}\right]$).

Theorem 2. *Given a subset of components $M^s_{k+n} \subseteq M_{k+n}$, the cost term in each belief tree node is bounded by*

$$\mathcal{LB}\left[\mathcal{H}_{k+n}\right] = \frac{w^{m,s}_{k+n}}{\mathcal{UB}\left[\eta_{k+n}\right]}\left[\mathcal{H}^s_{k+n} + log\left(\frac{\mathcal{LB}\left[\eta_{k+n}\right]}{w^{m,s}_{k+n}}\right)\right], \quad (20)$$

$$\mathcal{UB}\left[\mathcal{H}_{k+n}\right] = \frac{w^{m,s}_{k+n}}{\mathcal{LB}\left[\eta_{k+n}\right]}\left[\mathcal{H}^s_{k+n} + log\left(\frac{\mathcal{UB}\left[\eta_{k+n}\right]}{w^{m,s}_{k+n}}\right)\right] - \bar{\gamma} log\left(\frac{\bar{\gamma}}{|\neg M_{k+n}|}\right), \quad (21)$$

where $\bar{\gamma} = 1 - \sum_{r\in M^s_{k+n}} \frac{w^r_{k+n}}{\mathcal{UB}[\eta_{k+n}]}$ and $\left|\neg M^s_{k+n}\right| > 2$.

Furthermore, considering different levels of simplifications, i.e. adding belief components to M^s_{k+n}, these bounds converge.

Corollary 1. *The bounds in Theorem 2 converge to* $\mathcal{H}_{k+n}$ *when* $M^s_{k+n} \to M_{k+n}$

$$\lim_{M^s_{k+n} \to M_{k+n}} \mathcal{LB}\left[\mathcal{H}_{k+n}\right] = \mathcal{H}_{k+n} = \mathcal{UB}\left[\mathcal{H}_{k+n}\right]. \tag{22}$$

A recursive update rule is given in Section C of the supplementary material [18].

Theorem 3. *Given a subset of components* $M^s_{k+n} \subseteq M_{k+n}$*, the term* η_{k+n}*, in each belief tree node, is bounded by*

$$\mathcal{LB}\left[\eta_{k+n}\right] = w^{m,s}_{k+n} \leq \eta_{k+n} \leq w^{m,s}_{k+n} + \left(\frac{|M_{k+n}|}{|M_k|} - \sum_{r \in M^s_{k+n}} w^r_k\right) \prod_{i=1}^{n} \sigma^i = \mathcal{UB}\left[\eta_{k+n}\right], \tag{23}$$

where $\sigma^i \triangleq max(\mathbb{P}\left(Z_{k+i}|x_{k+i}\right))$ *and* w^r_k *is the prior weight at time* k *for every component in* M^s_{k+n} *at time* $k+n$*.*

As in Theorem 2, since we only consider a subset of hypotheses, these bounds are also computationally more efficient to calculate and converge. We also note that specifically for Case 3, the bounds in Theorem 2 and Theorem 3 are calculated iteratively in each time step $k+n$ given the belief b^{ψ}_{k+n-1} as presented in (17).

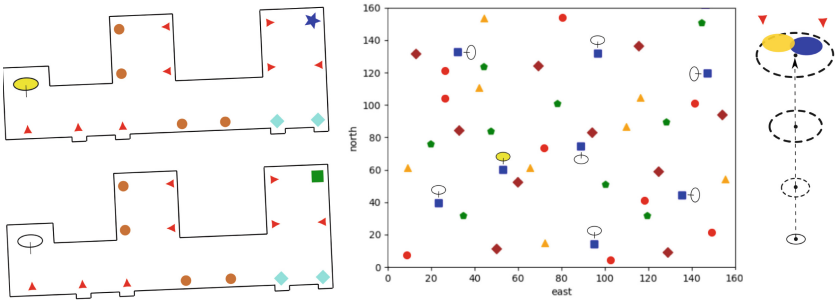
Corollary 2. *The bounds in Theorem 3 converge to* η_{k+n} *when* $M^s_{k+n} \to M_{k+n}$

$$\lim_{M^s_{k+n} \to M_{k+n}} \mathcal{LB}\left[\eta_{k+n}\right] = \eta_{k+n} = \mathcal{UB}\left[\eta_{k+n}\right]. \tag{24}$$

A recursive update rule is given in Section C of the supplementary material [18].

5 Results

We evaluate the performance of our approach for the different cases presented in Sect. 4. Our prototype implementation uses the GTSAM library [3]. Our considered scenarios represent highly ambiguous environments containing perceptually identical landmarks in different locations. In our first scenario, `floors`, the agent is initially located in one of F floors such that each floor contains a unique landmark, specific to that floor (Fig. 5a). In our second scenario, `2d_random`, the agent is initially placed in a random environment in front of a blue square (Fig. 5b). Both scenarios can be considered as versions of the kidnapped robot problem. With no other prior information, the initial belief, in both cases, is multi-modal containing $|M_0|$ hypotheses. The agent captures the environment using range measurements containing a class identifier, e.g. red triangle or green square. When the agent receives a measurement to some landmark which is ambiguous, i.e. it can theoretically be generated from more than one landmark, the number of hypotheses grows (see Fig. 5c). The number of identical landmarks can be adjusted to represent higher ambiguity, increasing the number of considered hypotheses. The agent's goal is to disambiguate between hypotheses by solving the corresponding BSP problem (6) at each planning session using entropy over posterior belief components weights as a cost function.



(a) (b) (c)

Fig. 5. (a) The `floors` environment where F identical floors represent different prior hypotheses. Each floor contains a unique landmark. The true location of the agent is highlighted in yellow; (b) The `2d_random` environment with many identical landmarks. The agent is initially placed in front of a blue square with no other prior information; (c) A planning session where ambiguous data association results in two hypotheses denoted by the yellow and blue ellipses.

In our first experiment we consider Case 1. We compare our approach with evaluating the cost function over the original belief, i.e. considering every possible future hypothesis. The heuristic in planning chooses the subset of components for each belief tree node greedily based on prior weights at time k. The decision rule R was set as no overlap, i.e. no loss with guaranteed optimal solution. The computational merits of our approach are presented in Fig. 6. Moreover, in Fig. 6c, f we can see that with a longer planning horizon the subset of hypotheses used for disambiguation becomes smaller. As more observations are utilized along the horizon, it is easier to discard wrong hypotheses in our considered cases.

In our second experiment we consider Case 2. In Fig. 7a we present the loss as a function of the budget size. As expected, with higher budget constraints the loss in solution quality becomes smaller. Moreover, as can be seen in Fig. 7b the loss is higher closer to the root of the belief tree, as bounds are accumulated in the non-myopic setting, increasing the overlap.

Considering Case 3, our experiments did not show any computational improvements between calculating the original cost function and using our approach. We indicate that this is because there is no exponential growth in the number of hypotheses within the horizon and our considered cost function is linear w.r.t. the number of components. However, as seen in [23], using a different cost, which is beyond the scope of this work, our approach can reduce the computational complexity while providing guarantees in Case 3 as well.

Finally, we consider Case 4. We first report that under this setting the computational complexity is high as every possible heuristic under the given budget is considered. In Fig. 7c preliminary results indicate that this process can improve the bounds over the loss in solution quality vs a given heuristic h^{inf}.

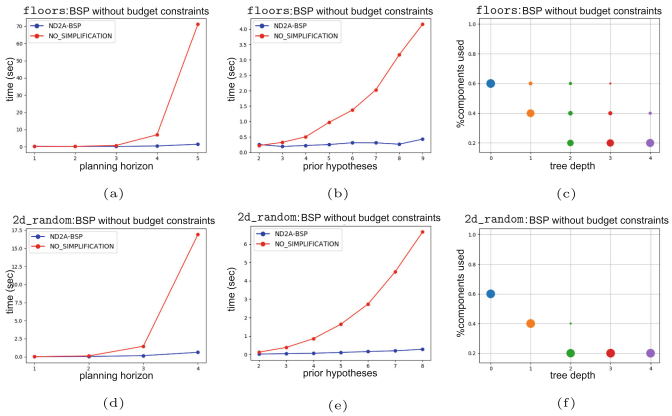


Fig. 6. Case 1 study for `floors` and `2d_random` environments. All scenarios presented carry zero loss. (a), (d) Planning time as a function of the planning horizon. In both environments, all settings the considered 4 prior hypotheses; (b), (e) Planning time as a function of the number of prior hypotheses. In both environments, all settings considered a planning horizon of 3; (c), (f) % components used to calculate bounds in each level of the belief tree. Circles scales are normalized as the number of nodes grows exponentially going down the tree.

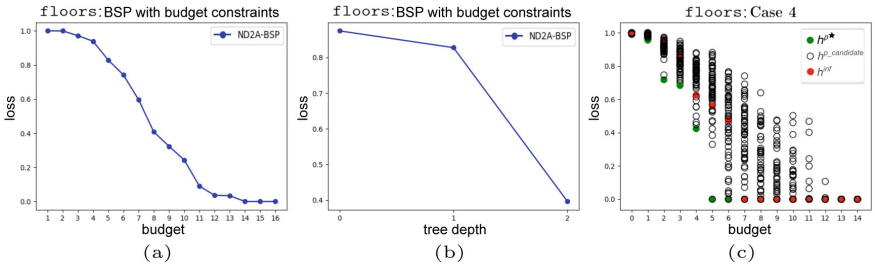


Fig. 7. (a) Normalized loss as a function of the size of the budget in Case 2. In all settings the number of floors, i.e. prior hypotheses was set to 12; (b) Normalized loss along the depth of a belief tree in Case 2 with $\mathcal{C} = 3$ components and a planning horizon of $n = 2$; (c) Normalized loss as a function of the size of the budget in Case 4, i.e. considering every valid heuristic in inference $h^{p_candidate}$. When $\mathcal{C} \leq 6$, the heuristic $h^{p\star}$ induces a smaller loss than h^{inf}. When $\mathcal{C} > 6$, both $h^{p\star}$ and h^{inf} induce zero loss, i.e. are optimal in this setting.

6 Conclusions

In this work we introduced ND2A-BSP, an approach to reduce the computational complexity in data association aware BSP with performance guarantees for the nonmyopic case. We rigorously analyzed our approach considering different settings under budget constraints in inference and/or planning.

Furthermore, future research will consider how to utilize information from planning in inference when the latter is subject to hard computational budget constraints, as in most real-world autonomous systems.

References

1. Agarwal, S., Tamjidi, A., Chakravorty, S.: Motion planning for active data association and localization in non-gaussian belief spaces. In: Algorithmic Foundations of Robotics XII. SPAR, vol. 13, pp. 288–303. Springer, Cham (2020). https://doi.org/10.1007/978-3-030-43089-4_19
2. Carlone, L., Censi, A., Dellaert, F.: Selecting good measurements via L1 relaxation: a convex approach for robust estimation over graphs. In: IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), pp. 2667–2674. IEEE (2014)
3. Dellaert, F.: Factor graphs and GTSAM: a hands-on introduction. Technical report, GT-RIM-CP&R-2012-002, Georgia Institute of Technology (2012)
4. Elimelech, K., Indelman, V.: Simplified decision making in the belief space using belief sparsification. Int. J. Robot. Res. (2021). arXiv: https://arxiv.org/abs/1909.00885
5. Hsiao, M., Kaess, M.: MH-iSAM2: multi-hypothesis iSAM using Bayes tree and hypo-tree. In: IEEE International Conference on Robotics and Automation (ICRA) (2019)
6. Hsiao, M., Mangelson, J.G., Suresh, S., Debrunner, C., Kaess, M.: Aras: ambiguity-aware robust active slam based on multi-hypothesis state and map estimations. In: IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), pp. 5037–5044. IEEE (2020)
7. Indelman, V., Nelson, E., Dong, J., Michael, N., Dellaert, F.: Incremental distributed inference from arbitrary poses and unknown data association: using collaborating robots to establish a common reference. IEEE Control Syst. Mag. (CSM) **36**(2), 41–74 (2016)
8. Indelman, V., Nelson, E., Michael, N., Dellaert, F.: Multi-robot pose graph localization and data association from unknown initial relative poses via expectation maximization. In: IEEE International Conference on Robotics and Automation (ICRA) (2014)
9. Jiang, F., Agrawal, V., Buchanan, R., Fallon, M., Dellaert, F.: iMHS: an incremental multi-hypothesis smoother. arXiv preprint arXiv:2103.13178 (2021)
10. Kaess, M., Johannsson, H., Roberts, R., Ila, V., Leonard, J., Dellaert, F.: iSAM2: incremental smoothing and mapping using the Bayes tree. Int. J. Robot. Res. **31**(2), 217–236 (2012)
11. Kitanov, A., Indelman, V.: Topological multi-robot belief space planning in unknown environments. In: IEEE International Conference on Robotics and Automation (ICRA), pp. 5726–5732 (2018)
12. Kschischang, F., Frey, B., Loeliger, H.A.: Factor graphs and the sum-product algorithm. IEEE Trans. Inform. Theory **47**(2), 498–519 (2001)
13. Lajoie, P.Y., Hu, S., Beltrame, G., Carlone, L.: Modeling perceptual aliasing in slam via discrete-continuous graphical models. IEEE Robot. Autom. Lett. (RA-L) **4**(2), 1232–1239 (2019)
14. Olson, E., Agarwal, P.: Inference on networks of mixtures for robust robot mapping. Int. J. Robot. Res. **32**(7), 826–840 (2013)

15. Pathak, S., Thomas, A., Indelman, V.: A unified framework for data association aware robust belief space planning and perception. Int. J. Robot. Res. **32**(2–3), 287–315 (2018)
16. Shelly, O., Indelman, V.: Hypotheses disambiguation in retrospective. IEEE Robot. Autom. Lett. (RA-L) **7**(2), 2321–2328 (2022)
17. Shienman, M., Indelman, V.: D2A-BSP: distilled data association belief space planning with performance guarantees under budget constraints. In: IEEE International Conference on Robotics and Automation (ICRA) (2022)
18. Shienman, M., Indelman, V.: Nonmyopic distilled data association belief space planning under budget constraints. Technical report, Technion - Israel Institute of Technology (2022). https://tinyurl.com/4h3b9mz8
19. Shienman, M., Kitanov, A., Indelman, V.: FT-BSP: focused topological belief space planning. IEEE Robot. Autom. Lett. (RA-L) **6**(3), 4744–4751 (2021)
20. Silver, D., Veness, J.: Monte-Carlo planning in large POMDPs. In: Advances in Neural Information Processing Systems (NIPS), pp. 2164–2172 (2010)
21. Sunderhauf, N., Protzel, P.: Towards a robust back-end for pose graph slam. In: IEEE International Conference on Robotics and Automation (ICRA), pp. 1254–1261. IEEE (2012)
22. Sztyglic, O., Zhitnikov, A., Indelman, V.: Simplified belief-dependent reward mcts planning with guaranteed tree consistency. arXiv preprint arXiv:2105.14239 (2021)
23. Sztyglic, O., Indelman, V.: Online pomdp planning via simplification. arXiv preprint arXiv:2105.05296 (2021)
24. Ye, N., Somani, A., Hsu, D., Lee, W.S.: DESPOT: online POMDP planning with regularization. JAIR **58**, 231–266 (2017)