Long-term robot motion planning for active sound source localization with Monte Carlo tree search

Quan Nguyen Van, Francis Colas, Emmanuel Vincent, François Charpillet

LONG-TERM ROBOT MOTION PLANNING FOR ACTIVE SOUND SOURCE LOCALIZATION WITH MONTE CARLO TREE SEARCH

Quan V. Nguyen[1,2,3], Francis Colas[1,2,3], Emmanuel Vincent[1,2,3] and François Charpillet[1,2,3]

[1]Inria, Villers-lès-Nancy, F-54600, France
[2]CNRS, LORIA, UMR 7503, Villers-lès-Nancy, F-54600, France
[3]Université de Lorraine, LORIA, UMR 7503, Villers-lès-Nancy, F-54600, France

ABSTRACT

We consider the problem of controlling a mobile robot in order to localize a sound source. A microphone array can provide the robot with information on source localization. By combining this information with the movements of the robot, the localization accuracy can be improved. However, random robot motion or short-term planning may not result in optimal localization. In this paper, we propose an optimal long-term robot motion planning algorithm for active source localization. We introduce a Monte Carlo tree search (MCTS) method to find a sequence of robot actions that minimize the entropy of the belief on the source location. A tree of possible robot movements which balances between exploration and exploitation is first constructed. Then, the movement that leads to minimum uncertainty is selected and executed. Experiments and statistical results show the effectiveness of our proposed method on improving sound source localization in the long term compared to other motion planning methods.

Index Terms— Source localization, long-term motion planning, mixture Kalman filter, Monte Carlo tree search.

1. INTRODUCTION

Robot audition plays an important role in enhancing human-robot interaction [1–3]. Equipped with microphones, a robot is capable of detecting sound sources and localizing their origin. Using that information, it can separate a mixture of sounds, process the data to extract other useful information, and better understand its surrounding environment.

One of the benefits that robots provide to source localization is active localization which fuses robot motion and the measurements taken at different poses of the robot [4–11]. As can be seen, the movements of the robot improve source localization [12–14]. For example, head rotation can resolve the ambiguity on the angle of arrival [15,16]. Moving towards a source can increase the signal-to-noise ratio [17, 18]. However, any robot motion is not optimal in all situations. Approaching a source is most useful when the robot is still far from it. Turning the head gives little additional information once the front-back ambiguity has been eliminated. Therefore, robot motion planning must be considered in order to locate an audio source as precisely as possible.

The localization accuracy can be improved by following a fixed patrol loop and covering a potential maneuver area [17]. More sophisticated motion strategies based on information-theoretic criteria, i.e., Shannon entropy, have been proposed [19–21]. The general idea is to drive the robot in the direction which leads to minimum uncertainty on the source location. In [21], a gradient ascent method was proposed to find the robot movement that minimizes the entropy of the belief on the position of the sound source one step ahead. This method yields a locally optimal robot motion but, in the long run, this sequence of local optima is generally not globally optimal. In [20], a long-term motion planning method was introduced by using a dynamic programming algorithm to find the optimal trajectory for localizing a static source. This method approximates the sum of entropies over a finite time horizon by assuming that the entropy at each future pose does not depend on the trajectory used to reach that pose. This assumption is required for dynamic programming, but the entropy actually depends on the followed trajectory in practice. This method is therefore also potentially suboptimal.

Monte Carlo tree search (MCTS) [22,23], which had great success with computer Go [24], is a stochastic algorithm for making optimal decisions in a given domain. The algorithm builds and uses an adaptive tree of possible future states. It evaluates each state in a search tree by the average outcome of simulations from that state. At each step, a new action is selected to create a new node and improve the evaluation of all parents of the tree. MCTS enables planning many time steps ahead and is often effective even with little or no prior domain knowledge. In addition, MCTS with the upper confidence bound for trees (UCT) algorithm [25] as the selection criterion can address the exploration-exploitation dilemma.

In this paper, we introduce a long-term motion planning algorithm for robot sound source localization based on MCTS. This algorithm is applicable to any moving robot and any source that is active for a sufficient amount of time, e.g., a human speaker. We provide two contributions. First, we build

an MCTS framework to find an optimal sequence of moves which will minimize the expected future entropy of the estimated source location. Each level of the tree corresponds to one future time step and each node in the tree represents one robot pose at one time. We estimate the belief corresponding to a sequence of poses by a nonlinear mixture Kalman filter [26]. Second, we conduct an experimental validation of our algorithm in simulation with several realistic scenarios and compare it with other motion planning algorithms.

The structure of the rest of this paper is as follows. The proposed method for long-term robot motion planning is introduced in Section 2. The implementation of the MCTS algorithm is presented in Section 3. Section 4 describes the experimental evaluation. Section 5 provides conclusions.

2. LONG-TERM ROBOT MOTION PLANNING

We consider long-term planning to find an optimal robot motion that reduces the uncertainty on source location. In this section, we describe the state vector and the derivation of an appropriate cost function for robot motion control.

2.1. State and observation vectors

In addition to diffuse background noise, we assume that there is a single directional sound source in the room, which is possibly not always active and may be moving. The state vector at time step k is defined by [26]

$$\begin{bmatrix} X_k & a_k \end{bmatrix}^T = \begin{bmatrix} X_{rk} & X_{sk} & a_k \end{bmatrix}^T =$$
$$\begin{bmatrix} x_{rk} & y_{rk} & \theta_{rk} & x_{sk} & y_{sk} & \theta_{sk} & v_{sk} & w_{sk} & a_k \end{bmatrix}^T \quad (1)$$

where X_{rk} is the pose of the robot, i.e., its absolute position $[x_{rk}, y_{rk}]$ and its orientation θ_{rk} w.r.t. the x-axis; X_{sk} is the state of the source, i.e., its absolute position $[x_{sk}, y_{sk}]$, its orientation θ_{sk} w.r.t. the x-axis, and its linear and angular velocities $[v_{sk}, w_{sk}]$; a_k is the source activity, where $a_k = 1$ indicates that the source is active, otherwise $a_k = 0$. Including the source activity in the estimation enables the robot to track an intermittent source and deal with false measurements.

The observation vector Z_k consists of one angle of arrival measurement Z_k^l obtained via a far-field instantaneous audio localization technique and one source activity measurement Z_k^a obtained via a source activity detection (SAD) technique:

$$Z_k = \begin{bmatrix} Z_k^l & Z_k^a \end{bmatrix}^T. \quad (2)$$

2.2. Mixture Kalman filter

Let us assume that the robot has taken measurements up to a certain time step k. All the knowledge about the source location, source activity and robot pose at time k is represented by the belief $P(X_k, a_k|Z_{1:k})$. We model this belief

via a mixture of Gaussians and estimate it using the mixture Kalman filter (MKF) implemented in [26]. In [26], we also define $P(a_{t+1}|a_t)$ as the source activity transition probability table, $P(X_{t+1}|X_t)$ as the state transition model and $P(Z_{t+1}|X_{t+1}, a_{t+1})$ as the observation model. The latter is built by acoustic simulation, taking reverberation and noise into account. We assume the robot does not move while taking measurements. If it were moving, the observation probability could be changed to account for ego noise.

2.3. Cost function

Now, we want to move the robot to a new pose at time $k + 1$. For every possible motion sequence $u_{k+1:k+T}$ up to horizon $k + T$, we quantify the uncertainty in the estimated source location by the entropy of the belief at each future time step $k + i$, $1 \leq i \leq T$. We sum these entropies with a forgetting factor, in order to tune the tradeoff between short vs long term. These entropies depend on future observations which are unknown at current time k. By considering their expectation, the cost function can be expressed as

$$J_T = \sum_{i=1}^{T} \lambda^{i-1} \mathbb{E}_{Z_{k+1:k+i}}[H(P(X_{k+i}, a_{k+i}|Z_{1:k+i}))] \quad (3)$$

where λ, $\mathbb{E}$ and H stand for the forgetting factor, the expectation and the entropy, respectively. In practice, we approximate the expectation over $Z_{k+1:k+T}$ by drawing N_s random samples Z^s from the distribution $P(Z_{k+1:k+T}|Z_{1:k})$:

$$J_T \approx \sum_{i=1}^{T} \lambda^{i-1} \frac{1}{N_s} \sum_{Z^s} H(P(X_{k+i}, a_{k+i}|Z_{1:k}, Z^s)). \quad (4)$$

One sample from this distribution is obtained by first drawing a sample Z_{k+1}^s from $P(Z_{k+1}|Z_{1:k})$, followed by a sample Z_{k+2}^s from $P(Z_{k+2}|Z_{1:k}, Z_{k+1}^s)$, and so on. Then, we have:

$$P(Z_{k+1:k+T}|Z_{1:k}) = \prod_{i=1}^{T} P(Z_{k+i}|Z_{1:k+i-1}) \quad (5)$$

with

$$P(Z_{k+i}|Z_{1:k+i-1}) = \sum_{a_{k+i}} \int P(Z_{k+i}|X_{k+i}, a_{k+i})$$
$$P(X_{k+i}, a_{k+i}|Z_{1:k+i-1})dX_{k+i}. \quad (6)$$

The entropies of the estimated beliefs cannot be computed in closed form for mixtures of Gaussians. An approximation based on a second-order Taylor series was proposed in [27] which we adopt hereafter.

Ultimately, the aim is to find the optimal motion sequence $u_{k+1:k+T}$ minimizing the entropy. The optimal pose at time $k + 1$ is achieved by performing the first action u_{k+1} in that sequence and the optimization is done iteratively after each new observation. In the following, we use the terms "motion" and "action" interchangeably.
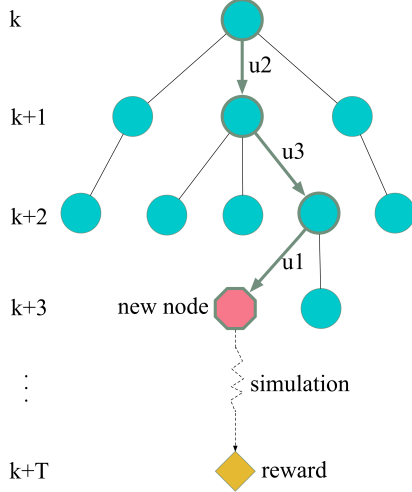
Fig. 1. An iteration of the MCTS algorithm.

3. MONTE CARLO TREE SEARCH

In practice, considering all possible motion sequences $u_{k+1:k+T}$ is intractable. In this section, we briefly describe the MCTS algorithm [22] and explain how it can be used to efficiently search the tree of possible sequences.

Fig. 1 shows an iteration of the MCTS algorithm at time k. Each level of the tree corresponds to one future time step. Each node n contains the information about: the pose of the robot, the estimated belief $b(n)$ on the source location, the untried actions among a finite set of possible actions, the accumulated reward $Q(n)$ (see below), and the visit count $N(n)$. The root node n_0 represents the pose of the robot at time k and carries the estimated belief $P(X_k, a_k|Z_{1:k})$. The links between a node and its child nodes represent different actions.

Starting from the root, a tree is built iteratively by selecting a node, adding a child corresponding to an untried action from this node, and following a random robot trajectory from this new node up to time $k + T$. The entropy corresponding to this trajectory is propagated upwards the tree to update the reward Q and the visit count N.

An efficient method for selecting a node is the UCT criterion [25]:

$$ UCT = \underset{n' \in \text{ children of } n}{\arg\max} \frac{Q(n')}{N(n')} + C_p \sqrt{\frac{2\ln N(n)}{N(n')}} \quad (7) $$

where C_p can be tuned to achieve a suitable tradeoff between exploration (low value) and exploitation (high value). This criterion derives from the Chernoff-Hoeffding inequality which is valid for a bounded reward function. In our case, entropy is bounded upwards by the entropy of the uniform distribution on the state vector, and downwards by the entropy of the probability distribution in the case when there is no front-back ambiguity and the sound source is as close as possible to the robot (0.18 m due to the size of the robot).

In the example depicted in Fig. 1, the UCT criterion is iteratively used at each level to select a node until depth $t = k + 2$ where an untried action is chosen to create a new node. The predicted belief distribution $P(X_{t+1}, a_{t+1}|Z_{1:t})$ is first obtained:

$$ P(X_{t+1}, a_{t+1}|Z_{1:t}) = $$
$$ \sum_{a_t} \int P(X_{t+1}, a_{t+1}|X_t, a_t)P(X_t, a_t|Z_{1:t})d(X_t) = \quad (8) $$
$$ \sum_{a_t} \int P(a_{t+1}|a_t)P(X_{t+1}|X_t)P(X_t, a_t|Z_{1:t})d(X_t). \quad (9) $$

Given one observation Z_{t+1} sampled as in (6), the updated belief at time step $t + 1$ is expressed as:

$$ P(X_{t+1}, a_{t+1}|Z_{1:t+1}) = $$
$$ \eta P(Z_{t+1}|X_{t+1}, a_{t+1})P(X_{t+1}, a_{t+1}|Z_{1:t}) \quad (10) $$

where η is a normalizing constant.

From the new expanded node, we perform a simulation until time step $k + T$ by selecting an action at random at each time step. The procedure to update the estimated belief after selecting one action is similar to above. At the end of the simulation, we evaluate the reward by summing the negative entropy of the expected future belief from time step $k+1$ until time step $k + T$ with a corresponding forgetting factor λ:

$$ Q = -\sum_{i=1}^{T} \lambda^{i-1} H(P(X_{k+i}, a_{k+i}|Z_{1:k+i})). \quad (11) $$

After finishing the simulation, the number of times a node has been visited and the accumulated reward value are updated upwards, up to the root node.

The iterations of building the tree terminate when the computational budget has been used up. The optimal pose at time $k + 1$ is then chosen based on the average reward evaluated from that pose. The robot performs the corresponding action in order to move to this pose, takes a new real measurement Z_{k+1}, builds a new tree to find the next optimal pose, and so on.

4. EXPERIMENTAL EVALUATION

In order to obtain statistically meaningful results, a large number of experiments is needed that can hardly be conducted with a real robot. Therefore, we conducted simulated experiments mimicking the *smart room* at Inria Nancy, where the robot is a Turtlebot equipped with a 4-microphone Kinect sensor forming a linear array. We assumed the sound source to be static and always active. We simulated the robot movements and the resulting location and activity measurements employing state-of-the-art techniques, whose parameters closely match the real reverberation and acoustic noise conditions in that room [20]. We generated 200 experiments for different random initial robot locations and source locations.
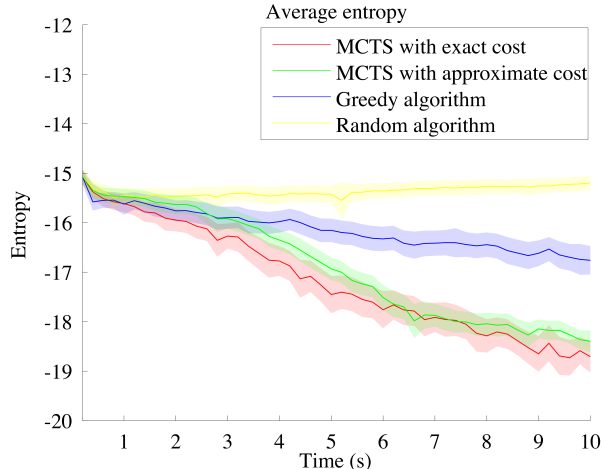
Fig. 2. Average entropy and 95% confidence interval of the 4 algorithms in all 200 experiments over time.

Fig. 3. Average estimation error and 95% confidence interval of the 4 algorithms in all 200 experiments over time.

4.1. Algorithm settings

To start from an informative belief, the robot first follows a fixed trajectory while updating the belief using MKF [26] for 3 s. After this, it follows the proposed MCTS algorithm with $\lambda = 0.5$, $T = 20$, and 700 tree nodes. The action set contains 13 discrete actions. They are basically moving forward, moving backward, turning left or right while moving forward or backward, turning with different speed and radius. The optimal selected action is applied 5 times in a row, with 0.2 s time step. For comparison, we applied the same procedure to three other motion planning algorithms: a greedy algorithm inspired from [21], MCTS with an approximate cost function inspired from [20], and random motion. The greedy algorithm finds the optimal action that minimizes the expected entropy of the belief one step ahead. The MCTS algorithm with approximate cost function computes the expected entropy for each future pose in (11) by recursively predicting the belief via (9) at all time steps, but updating it via (10) only for the last step. The random method simply choses the next pose of the robot at random.

4.2. Results

Fig. 2 shows the entropy of the estimated belief over time for all algorithms, on average over all experiments. After only 1 s of motion planning, i.e., at time $k = 4$ s, the two flavors of MCTS already yield significantly lower entropy compared to the greedy algorithm and the random algorithm. The entropy decreases drastically at further time steps. This is because MCTS optimizes the entropy in the long run. In contrast to that, random motion barely reduces it and the greedy algorithm only optimizes it one time step ahead.

This result is expected as our method is actually optimizing the entropy. However, the objective of this work is to use robots to better localize sound sources. Therefore the evalua-
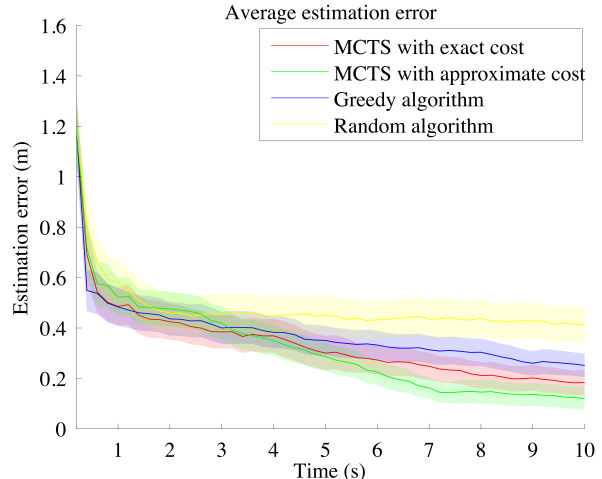
tion of the algorithms should be done on the estimation error, that is the distance between the estimated source position and the true position, which is available only in simulation. The average estimation error over time is presented for all algorithms in Fig. 3. Again, the average estimation error of both MCTS methods is smaller compared to greedy and random motion in the long run.

With a Wilcoxon signed-rank test, we can show that both MCTS variants yield a significantly smaller entropy and estimation error than the greedy and random algorithms ($p < 0.01$). The MCTS variant with approximate cost has significantly smaller estimation error than the MCTS variant with exact cost ($p < 0.01$). However, the two MCTS methods are not significantly different in terms of entropy value.

5. CONCLUSIONS

We presented a long-term robot motion planning algorithm for sound source localization. Our main theoretical contributions concern an objective function and a practical MCTS algorithm for finding an optimal sequence of robot movements that will minimize the estimation uncertainty in the long run. The experiments showed that our long-term planning algorithm has better performance compared to greedy or random motion. Future work will focus on improving the selection and simulation step in MCTS by providing prior knowledge when building the tree.

6. ACKNOWLEDGMENTS

7. REFERENCES

[1] K. Nakadai, T. Lourens, H. G. Okuno, and H. Kitano, "Active audition for humanoid," in *Proc. AAAI*, 2000, pp. 832–839.

[2] K. Nakadai, T. Takahashi, H. G. Okuno, H. Nakajima, Y. Hasegawa, and H. Tsujino, "Design and implementation of robot audition system 'HARK' — open source software for listening to three simultaneous speakers," *Advanced Robotics*, vol. 24, no. 5–6, pp. 739–761, 2010.

[3] H. Okuno and K. Nakadai, "Robot audition: Its rise and perspectives," in *Proc. ICASSP*, 2015, pp. 5610–5614.

[4] E. Martinson and A. Schultz, "Auditory evidence grids," in *Proc. IROS*, 2006, pp. 1139–1144.

[5] Y. Sasaki, S. Kagami, and H. Mizoguchi, "Multiple sound source mapping for a mobile robot by self-motion triangulation," in *Proc. IROS*, 2006, pp. 380–385.

[6] B. P. DeJong, "Auditory occupancy grids with a mobile robot," *J. Autom., Mobile Robot., Intell. Syst.*, vol. 6, no. 3, 2012.

[7] I. Marković, A. Portello, P. Danès, I. Petrović, and S. Argentieri, "Active speaker localization with circular likelihoods and bootstrap filtering," in *Proc. IROS*, 2013, pp. 2914–2920.

[8] A. Portello, G. Bustamante, P. Danès, J. Piat, and J. Manhès, "Active localization of an intermittent sound source from a moving binaural sensor," in *Proc. Forum Acustium*, 2014.

[9] H. Barfuss and W. Kellermann, "An adaptive microphone array topology for target signal extraction with humanoid robots," in *Proc. IWAENC*, 2014.

[10] A. Magassouba, N. Bertin, and F. Chaumette, "Sound-based control with two microphones," in *Proc. IROS*, 2015, pp. 5568–5573.

[11] V. Tourbabin, H. Barfuss, B. Rafaely, and W. Kellermann, "Enhanced robot audition by dynamic acoustic sensing in moving humanoids," in *Proc. ICASSP*, 2015, pp. 5625–5629.

[12] M. Cooke, Y.-C. Lu, Y. Lu, and R. Horaud, "Active hearing, active speaking," in *Proc. ISAAR*, 2007, pp. 33–46.

[13] Y.-C. Lu and M. Cooke, "Motion strategies for binaural localisation of speech sources in azimuth and distance by artificial listeners," *Speech Communication*, vol. 53, no. 5, pp. 622–642, 2011.

[14] C. Evers, A. Moore, and P. Naylor, "Towards informative path planning for acoustic SLAM," in *Proc. DAGA*, 2016.

[15] K. Nakadai, H. G. Okuno, and H. Kitano, "Robot recognizes three simultaneous speech by active audition," in *Proc. ICRA*, 2003, pp. 398–405.

[16] E. Berglund and J. Sitte, "Sound source localisation through active audition," in *Proc. IROS*, 2005, pp. 509–514.

[17] E. Martinson and A. Schultz, "Discovery of sound sources by an autonomous mobile robot," *Autonomous Robots*, vol. 27, pp. 221–237, 2009.

[18] K. Song, Q. Liu, and Q. Wang, "Olfaction and hearing based mobile robot navigation for odor/sound source search," *Sensors*, vol. 11, pp. 2129–2154, 2011.

[19] E. Sommerlade and I. Reid, "Information theoretic active scene exploration," in *Proc. CVPR*, May 2008.

[20] E. Vincent, A. Sini, and F. Charpillet, "Audio source localization by optimal control of a mobile robot," in *Proc. ICASSP*, 2015, pp. 5630–5634.

[21] G. Bustamante, P. Danès, T. Forgue, and A. Podlubne, "Towards information-based feedback control for binaural active localization," in *Proc. ICASSP*, 2016, pp. 6325–6329.

[22] C. B. Browne, E. Powley, D. Whitehouse, S. M. Lucas, P. I. Cowling, et al., "A survey of Monte Carlo tree search methods," *IEEE Transactions on Computational Intelligence and AI in Games*, vol. 4, no. 1, pp. 1–43, Mar. 2012.

[23] G. Chaslot, S. Bakkes, I. Szita, and P. Spronck, "Monte-Carlo tree search: A new framework for game AI," in *Proc. AIIDE*, 2008.

[24] D. Silver, A. Huang, C. J. Maddison, A. Guez, L. Sifre, et al., "Mastering the game of Go with deep neural networks and tree search," *Nature*, vol. 529, no. 7587, pp. 484–489, Jan. 2016.

[25] L. Kocsis, C. Szepesvári, and J. Willemson, "Improved Monte-Carlo search," Tech. Rep. 1, University of Tartu, Estonia, 2006.

[26] Q. V. Nguyen, F. Colas, E. Vincent, and F. Charpillet, "Localizing an intermittent and moving sound source using a mobile robot," in *Proc. IROS*, 2016.

[27] M. F. Huber, T. Bailey, H. Durrant-Whyte, and U. D. Hanebeck, "On entropy approximation for Gaussian mixture random vectors," in *Proc. MFI*, 2008, pp. 181–188.