

“© 2018 IEEE. Personal use of this material is permitted. Permission from IEEE must be obtained for all other uses, in any current or future media, including reprinting/republishing this material for advertising or promotional purposes, creating new collective works, for resale or redistribution to servers or lists, or reuse of any copyrighted component of this work in other works.”

Planning-Aware Communication for Decentralised Multi-Robot Coordination

Graeme Best¹, Michael Forrai^{2,3}, Ramgopal R. Mettu⁴ and Robert Fitch^{5,1}

Abstract—We present an algorithm for selecting when to communicate during online planning phases of coordinated multi-robot missions. The key idea is that a robot decides to request communication from another robot by reasoning over the predicted information value of communication messages over a sliding time-horizon, where communication messages are probability distributions over action sequences. We formulate this problem in the context of the recently proposed decentralised Monte Carlo tree search (Dec-MCTS) algorithm for online, decentralised multi-robot coordination. We propose a particle filter for predicting the information value, and a polynomial-time belief-space planning algorithm for finding the optimal communication schedules in an online and decentralised manner. We evaluate the benefit of informative communication planning for a multi-robot information gathering scenario with 8 simulated robots. Our results show reductions in channel utilisation of up to four-fifths with surprisingly little impact on coordination performance.

I. INTRODUCTION

Agents of a decentralised multi-robot system must plan their actions cooperatively to maximise the performance of the team. Communication is fundamental to coordinated behaviour, as robots need to develop decision strategies that take into account the actions of others. Through the cooperative planning process, multi-robot systems become effective for complex real-world tasks, including cooperative localisation, target tracking, object recognition, exploration, surveillance, and environmental monitoring [1].

Typically, decentralised coordination algorithms assume over-simplified communication models, which lead to large communication overheads. This hinders the suitability for many of these algorithms for realistic scenarios where communication resources are limited [2], [3], [4], the network is unreliable [5], [6], [7], [8], or the signals are susceptible to interference from other robots [9]. These issues are compounded even further for larger teams [10]. Indeed there is growing interest in addressing “communication in the wild” [11]. Such scenarios motivate the development of

algorithms that enable mitigating these various communication issues, while also maintaining reasonable coordination performance.

In this paper, we study this issue in the context of decentralised Monte Carlo tree search (Dec-MCTS) [1], [12], a recently proposed algorithm for asynchronous multi-robot coordination. Dec-MCTS is applicable to a general class of problem formulations and has been shown to work particularly well at multi-robot information gathering scenarios. The algorithm jointly optimises sequences of actions by asynchronously cycling between: 1) searching the action sequence space using a new variant of Monte Carlo tree search [13] to obtain sets of promising action sequences, 2) optimising a probability distribution over action sequences in a distributed manner [14], and 3) communicating plans by broadcasting these distributions to other team members. Each robot incrementally updates its plan by cycling through these phases until it is ready to make a decision and execute an action. Theoretical analysis of Dec-MCTS provides guarantees for convergence rates to the optimal payoff sequence. A notable finding of [1] indicates that there is large amount of redundant communication being transmitted; this suggests that coordination performance could be maintained while reducing communication channel usage by reasoning over the information value of communication messages.

This observation motivates the problem of optimising *when* to communicate, and to *whom*, while the robots are performing decentralised planning. We aim to find a balance between minimising the use of limited communication resources, and satisfying the planning algorithm objectives. This problem is challenging because it is difficult to efficiently predict how communicating the current intention (plan) will impact the global coordination performance in the long term [15]. In particular, we focus on the setting in which communication is noisy, messages are dropped at random and these issues are compounded as bandwidth usage by the team increases. We note that existing work, such as in communication-aware planning, addresses problems in which communication performance is viewed as an objective for path planning [16], [17], [5], [18]. In contrast our focus is on “planning-aware communication”, in which task performance is viewed as an objective for communication planning.

Our main contribution is a novel planning algorithm that reasons over the value of communication messages to decide when and to whom each robot should communicate. We present this algorithm for the context of Dec-MCTS, where the aim is to minimise communication while maintaining

*This work was supported in part by the Australian Research Council’s Discovery Project funding scheme (No. DP140104203), the Australian Centre for Field Robotics, and the New South Wales Government.

¹G. Best and R. Fitch are with the Australian Centre for Field Robotics (ACFR), The University of Sydney, Sydney, Australia. {g.best, rfitch}@acfr.usyd.edu.au

²M. Forrai conducted this work at the School of Aerospace, Mechanical and Mechatronic Engineering, The University of Sydney, Sydney, Australia.

³M. Forrai is now with Accenture, Sydney, Australia. michael.forrai@accenture.com

⁴R. Mettu is with the Department of Computer Science, Tulane University, New Orleans, LA, USA. rmettu@tulane.edu

⁵R. Fitch is also with the Centre for Autonomous Systems (CAS), University of Technology Sydney, Sydney, Australia. rfitch@uts.edu.au

bounds on the uncertainty of the local reward probability distribution. Our approach first predicts the value of communication at future time steps, then uses this information to plan the optimal sequence of communication requests between all pairs of robots. The predictions are performed using a particle filter that simulates the probability distribution update step of Dec-MCTS. The optimal communication schedule is then planned using a dynamic programming algorithm that searches over a sliding time-horizon. Notably, the approach collapses the decision-tree into a smaller directed acyclic graph, enabling the schedule to be found in polynomial-time. Overall, this approach trades drastically reduced communication for a modest overhead in computation time.

We demonstrate the performance and behaviour of our communication planning algorithm in a multi-robot information gathering scenario [19]. We use the Dec-MCTS algorithm from [1] as a black-box for the decentralised coordination component. Our results show reductions in channel utilisation of up to four-fifths with surprisingly little impact on coordination performance. This demonstrates our approach is suitable for communication planning in real-world multi-robot scenarios.

II. RELATED WORK

Communication of the robots' intentions is critical to the coordinated behaviour of decentralised multi-robot teams, yet communication remains a bottleneck for deployed multi-robot systems. Many decentralised planning approaches have been proposed where messages are broadcast periodically without considering communication value or constraints, including Dec-MCTS [1], max-sum [20], decentralised task allocation [21], and STEAM [22]. Unfortunately these approaches may incur large communication overhead costs, although some have been demonstrated to be robust to unpredictable communication loss through redundancy [1], [20], [23], [8]. An alternative approach is communication-aware planning, which typically seeks to improve the available communication resources by repositioning the robots. This can be achieved by encoding communication reliability as a path planning objective [16], [17], [5], [18]. In this paper, instead, we explicitly plan when to communicate during planning-time by reasoning over the value of any given message.

In most cases, communication planning has been performed where the messages are observations. The value of these messages can be measured by considering their effect on data fusion accuracy [10], [24], [2]. However, for decentralised planning the information value should instead measure the expected effect on the planning reward [25], [15], [26]. Ideally, these reward predictions should be over a long time horizon [15]. While our problem is similar, we are instead interested in planning communication where the messages are the intentions (plans) of the robots.

Planning to communicate intentions is addressed for Dec-POMDP formulations in [27], [28] by defining communication value as the reduction in reward as a result

of not communicating. A key difference in our approach is that we maintain a probabilistic belief over the future plans using a particle filter, and then measure the information value as uncertainty of the reward distributions. This probabilistic formulation for predicting plans has major advantages over reasoning about deterministic or most likely actions alone [29]. Also, while our approach is intended to be general, we address these challenges in the context of Dec-MCTS.

Our approach is motivated by the belief-space planner for scheduling localisation hardware usage [30], which aims to conserve energy during a path-following scenario. Our belief graph is generated as a directed acyclic graph (DAG) in a similar way, except we use a particle filter relevant to our application rather than a linear Gaussian model. Additionally, we adopt a similar dynamic programming (DP) approach to find the optimal schedule. Related DAG generation and DP approaches for robotics applications appear in [18], [31].

III. BACKGROUND: DECENTRALISED PLANNING

In this paper, we address the problem of deciding when to communicate while a team of robots is performing decentralised planning. While the proposed formulation is intended to be general, we are particularly interested in solving this problem in the context of the Dec-MCTS algorithm [1]. In this section, we set the context for our communication planning algorithm by providing an overview of the general multi-robot planning scenario solved by Dec-MCTS, introducing relevant notation, and detailing the components of Dec-MCTS that are important for our proposed approach. A complete description of Dec-MCTS is presented in [1].

A. Multi-robot planning scenario

We consider a team of R robots $\{1, 2, \dots, R\}$, where each robot i plans its own sequence of actions $\mathbf{x}^i = (x_1^i, x_2^i, \dots)$. Each action x_n^i has an associated cost c_n^i and each robot has a cost budget B^i such that the sum of the costs must be less than the budget, i.e., $\sum_{x_n^i \in \mathbf{x}^i} c_n^i \leq B^i$. This budget may be an energy or time constraint, or a planning horizon. The feasible set of actions and associated costs at step n are a function of the previous actions $(x_1^i, x_2^i, \dots, x_{n-1}^i)$. Thus, there is a set of feasible action sequences for each robot $\mathbf{x}^i \in \mathcal{X}^i$. We denote \mathbf{x} as the set of action sequences selected by all robots $\mathbf{x} := \{\mathbf{x}^1, \mathbf{x}^2, \dots, \mathbf{x}^R\}$ and $\mathbf{x}^{(i)}$ as the set of action sequences for all robots except i .

The aim is to maximise a global objective function $g(\mathbf{x})$ which is a function of the action sequences of all robots. Each robot i knows the function $g(\mathbf{x})$, but needs to communicate to learn the plans $\mathbf{x}^{(i)}$ of other robots.

B. Probability distributions over action sequences

A key idea in Dec-MCTS and the communication planning approach in this paper is to describe a robot's current plan as a probability distribution over action sequences. In Dec-MCTS, these probability distributions are optimised by the planning algorithm and are communicated between robots. In our proposed communication planning, we directly

Algorithm 1 Dec-MCTS for robot i , as presented in [1].

input: global objective g , feasible action sequences
output: sequence of actions \mathbf{x}^i for robot i

- 1: $\mathcal{T} \leftarrow$ initialise MCTS tree
- 2: **while** computation budget not met **do**
- 3: $\hat{\mathcal{X}}_t^i \leftarrow$ SELECTSETOFSEQUENCES(\mathcal{T})
- 4: **for** fixed number of iterations **do**
- 5: $\mathcal{T} \leftarrow$ GROWTREE($\mathcal{T}, \hat{\mathcal{X}}_t^i, q_t^{(i)}, B^i$)
- 6: $q_{t+1}^i \leftarrow$ UPDATEDIST($\hat{\mathcal{X}}_t^i, q_t^i, \hat{\mathcal{X}}_t^i, q_t^{(i)}$)
- 7: COMMUNICATIONTRANSMIT($\hat{\mathcal{X}}_t^i, q_t^i$)
- 8: ($\hat{\mathcal{X}}_{t+1}^i, q_{t+1}^i$) \leftarrow COMMUNICATIONRECEIVE
- 9: **return** $\mathbf{x}^i \leftarrow \arg \max_{\mathbf{x}^i \in \hat{\mathcal{X}}_t^i} [q_t^i(\mathbf{x}^i)]$

reason over the information value of these probability distributions when deciding when to communicate.

More formally, we define a probability mass function q_t^i , such that $q_t^i(\mathbf{x}^i)$ defines the probability that robot i will select the action sequence \mathbf{x}^i , and t is the planning iteration. In general, the domain of the distribution q_t^i is the set of all possible action sequences \mathcal{X}_t^i . However, to enable realistic communication and tractable computation, the domain of q_t^i is restricted to a subset $\hat{\mathcal{X}}_t^i \subset \mathcal{X}_t^i$. During the decentralised planning of Dec-MCTS, both $\hat{\mathcal{X}}_t^i$ and q_t^i evolve over time t .

C. Overview of Dec-MCTS

The Dec-MCTS algorithm is a decentralised solution for finding the set of paths \mathbf{x} that maximises a global objective $g(\mathbf{x})$. The algorithm, shown in Alg. 1, runs simultaneously and asynchronously on all robots; here we present the algorithm from the perspective of robot i . The algorithm cycles between three phases: 1) grow a search tree for robot i 's plan using MCTS, while considering information about the other robots, 2) update the probability distribution over possible action sequences, and 3) communicate probability distributions with other robots. This cycle continues regardless of whether or not the communication was successful.

Rather than optimising directly for g , each robot i instead optimises with respect to a local utility function f^i :

$$f^i(\mathbf{x}) := g(\mathbf{x}^i \cup \mathbf{x}^{(i)}) - g(\mathbf{x}_\emptyset^i \cup \mathbf{x}^{(i)}), \quad (1)$$

where \mathbf{x}_\emptyset^i is a default ‘no reward’ sequence for robot i . Although the robots could use g directly, optimising with respect to f^i instead typically results in faster convergence since f^i is less affected by the unknown plans of robots that should not influence robot i 's plan. We also use f^i for predicting communication value in our proposed communication planning approach.

The tree search phase (Alg. 1 line 5) runs a new variant of MCTS specifically designed for decentralised planning. Then, periodically, breakpoints occur such that $\hat{\mathcal{X}}_t^i$ is updated to the most promising paths in the search tree (line 3). In the update phase (line 6), q_t^i is optimised using distributed gradient descent in a similar way to [14]. Specifically, the function \mathcal{Q} specifies the update for $q_t^i(\mathbf{x}^i)$ as a function of

the current distribution for all robots q_t , defined as

$$\begin{aligned} q_{t+1}^i(\mathbf{x}^i) &= \mathcal{Q}(q_t^i(\mathbf{x}^i), q_t) \\ &= q_t^i(\mathbf{x}^i) - \alpha q_t^i(\mathbf{x}^i) \left[\frac{\mathbb{E}_{q_t}[f^i] - \mathbb{E}_{q_t}[f^i|\mathbf{x}^i]}{\beta} \right. \\ &\quad \left. + \mathbb{H}(q_t^i) + \ln(q_t^i(\mathbf{x}^i)) \right], \quad (2) \end{aligned}$$

where \mathbb{E}_{q_t} is expectation with respect to the *joint* distribution q_t , \mathbb{H} is entropy, β decreases over time to slowly reduce entropy, and α is a fixed step size. The intuition behind (2) is to slowly increase the probability of selecting action sequences that would result in higher expected reward (evaluated in the expectation term) while also ensuring the entropy of the distribution does not converge too quickly (regulated by the last two terms).

Robot i periodically communicates $\hat{\mathcal{X}}_t^i$ and q_t^i to other robots (line 7). When communication is received (line 8), these new distributions are used in future planning iterations to improve coordination, and thus optimise g . A key component of our communication planning approach is to probabilistically simulate the update \mathcal{Q} for other robots. This allows predicting the future plans of other robots and thus estimate the value of communication messages.

IV. PROBLEM FORMULATION

Suppose that a team of robots is performing a decentralised algorithm to plan their sequences of actions, such as the Dec-MCTS algorithm outlined in Sec. III. Each robot alternates between performing local planning computations to incrementally update their plan, and communicating their current plan to other robots. We focus on the latter communication step only, and address the problem of, for each robot, deciding *when* to communicate and to *whom*, during the execution of the overall planning algorithm.

This problem setting is illustrated schematically in Fig. 1. Robot i must decide whether or not to request a communication message from each robot j . Robot i should make this decision by predicting the information value of communication for future planning iterations, and avoiding unnecessary communication. If requested, robot j communicates its plan to robot i , where its plan is represented as a probability distribution q_t^j over the space of action sequences $\hat{\mathcal{X}}_t^j$ (as defined in Sec. III-B).

A. Communication setup

We are particularly interested in settings where the robots are communicating with each other over a shared wireless communication medium. The communication channel is unreliable and unpredictable. This could be due to, for example, interference caused by multiple robots transmitting at the same time, or unpredictable attenuation through the environment. By selecting to communicate only valuable messages, we are reducing the communication channel usage to help mitigate these issues. We note that our approach does not require knowledge of any particular noise rate or bandwidth limitations, which may be difficult to model. Instead we utilize a parameter θ that determines whether a

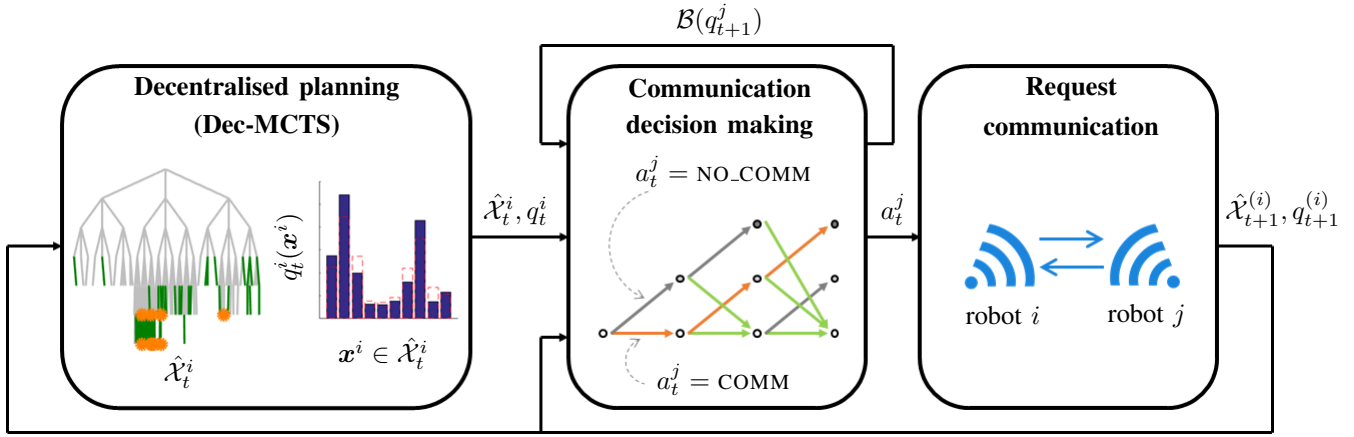


Fig. 1. The decentralised coordination approach from the perspective of robot i . The three phases are repeated asynchronously by all robots. (left) Decentralised planning optimises a probability distribution over action sequences. (middle) Schedule a sequence of communication requests using belief-space planning. (right) Request and receive distributions from selected robots. This paper focusses on the communication decision-making component of the approach, while considering the implications to the overall algorithm.

robot will request communication. In our experiments we assume θ is given as input, but it can be chosen empirically or adapted in an online manner.

B. Communication schedules

At iteration t of the planning algorithm, robot i may request a communication message from robot j . We denote these requests with boolean variables a_t^j , such that $a_t^j = \text{COMM}$ if robot i requests communication from robot j , and $a_t^j = \text{NO_COMM}$ if no such request occurs. The set of request variables from robot i to all other robots at time t is denoted a_t . The problem we consider is to optimise the communication schedule π for robot i , defined as a sequence of request variables over time: $\pi = \{a_t | t = 0, 1, \dots, T\}$.

C. Value of communication

Robot i maintains a probability distribution q_t^j over robot j 's action sequence. This distribution is a copy of the distribution stored locally by robot j , and robot i only updates its copy when it receives a communication message from j . In order for robot i to decide to request a new message from robot j , robot i also maintains a belief \mathcal{B}_t^j for the current and future distributions of robot j . This belief is defined as a probability distribution $\mathcal{B}_t^j(q_t^j)$ over probability distributions q_t^j . The definition and implementation for \mathcal{B} will depend on the decentralised planning algorithm being used by the team; in the following section we define a particle filter representation for \mathcal{B} designed for the context of Dec-MCTS.

We define σ_t^j as a measure of uncertainty of the expected local utility f^i for robot i , as defined in (1), that is caused by not knowing q_t^j . A threshold θ defines an upper bound on this uncertainty, such that robot i requests communication from robot j to satisfy the constraints $\sigma_t^j \leq \theta, \forall t$. The intuition behind this formulation is that if the uncertainty σ_t^j is large then robot i 's plan is likely to be influenced by knowing q_t^j , and therefore communication should be requested in order to help optimise the global objective $g(\mathbf{x})$. We define

σ_t^j specifically as the standard deviation of the expectation $\mathbb{E}_{q_t^j}$ of the local utility f^i , where the standard deviation is calculated with respect to the belief \mathcal{B}_t^j of the unknown q_t^j ; i.e.,

$$\sigma_t^j = \text{STDEV}_{\mathcal{B}_t^j(q_t^j)} \left(\mathbb{E}_{q_t^j \cup q_t^{(j)}} \left[f^i(\mathbf{x}^j \cup \mathbf{x}^{(j)}) \right] \right), \quad (3)$$

where $q_t^{(j)}$ is a single estimate of the distributions of the robots $\neq j$. By this definition, $\sigma_t^j = 0$ when communicating since robot i then knows q_t^j . Typically, σ_t^j then increases at each iteration until the next communication request.

D. Problem statement

We impose a unit cost for each occasion that robot i requests a communication message from robot j . The cost of a schedule π is defined as the sum of the costs of all communication requests made by robot i , i.e.,

$$C(\pi) = \sum_j \sum_t \mathbf{1}(a_t^j = \text{COMM}).$$

Thus, the optimisation problem to be solved by robot i is to find π that minimises the communication cost $C(\pi)$ while subject to the uncertainty constraints $\sigma_t^j \leq \theta, \forall j, t$.

We note that this problem is different to the general multi-robot planning problem defined in Sec. III-A; however, by optimising communication according to this problem statement, the intention is to enable the coordination algorithm to effectively optimise $g(\mathbf{x})$ while reducing communication.

V. COMMUNICATION PLANNING ALGORITHM

Our algorithm to solve the communication planning problem has two key components: 1) evolve and evaluate the belief over a finite time-horizon using a particle filter, then 2) find the optimal communication schedule over this time-horizon using dynamic programming. Pseudocode is provided in Alg. 2. We describe the approach from the perspective of robot i , who is deciding to request information from robot j at iteration 0. This process is repeated for all robots at every iteration.

Algorithm 2 Communication planning for robot i . Without loss of generality, the algorithm is presented for $t = 0$.

input: current beliefs $\mathcal{B}_0^1, \mathcal{B}_0^2, \dots$, local utility functions f^i , planning horizon T , uncertainty limit θ
output: communication request schedule π

- 1: **for** $t = 1, 2, \dots, T - 1$ **do**
- 2: **for** each robot j **do**
- 3: $q_t^j \leftarrow \mathcal{Q}(q_{t-1}^j, q_{t-1}^j)$
- 4: **for** each robot $j \neq i$ **do**
- 5: \triangleright Forward pass
- 6: **for** $t = 1, 2, \dots, T - 1$ **do**
- 7: \triangleright Update beliefs
- 8: $\mathcal{B}_{t,0}^j \leftarrow q_t^j$
- 9: $\sigma_{t,0}^j \leftarrow 0$ \triangleright Uncertainty
- 10: $\Omega_{t,0}^j \leftarrow \Omega_{t-1,0}^j + 1$ \triangleright Accumulated cost
- 11: **for** $\tau = 1, 2, \dots, t$ **do**
- 12: $\mathcal{B}_{t,\tau}^j \leftarrow \mathcal{Q}'(\mathcal{B}_{t-1,\tau-1}^j, \mathcal{B}_{t-1,\tau-1}^j \cup q_{t-1}^{(j)})$
- 13: $\psi_{t,\tau}^j = \tau - 1$ \triangleright Backpointer
- 14: $\sigma_{t,\tau}^j \leftarrow \text{Eqn. (3)}$ \triangleright Uncertainty
- 15: $\Omega_{t,\tau}^j \leftarrow \Omega_{t-1,\tau-1}^j$ \triangleright Accumulated cost
- 16: \triangleright Find best backpointer from $t, 0$
- 17: $\psi_{t,0}^j \leftarrow 0$ \triangleright Backpointer
- 18: **for** $\tau = 1, 2, \dots, t - 1$ **do**
- 19: **if** $\sigma_{t-1-k,\tau-k}^j \leq \theta, \forall k < \tau$ **then**
- 20: **if** $\Omega_{t-1,\tau}^j + 1 < \Omega_{t,0}^j$ **then**
- 21: $\psi_{t,0}^j = \tau$ \triangleright Update backpointer
- 22: $\Omega_{t,0}^j = \Omega_{t-1,\tau}^j + 1$
- 23: \triangleright Backward pass
- 24: $\tau^* \leftarrow \text{argmax}_{\tau} (\Omega_{T,\tau}^j)$
- 25: $\Psi^j = (\psi_{T,\tau^*}^j, \dots, \psi_0^j) \leftarrow$ backpointers from ψ_{T,τ^*}^j
- 26: $(a_0^j, \dots, a_{T-1}^j) \leftarrow \{a_t^j = \text{COMM iff } \psi_{t+1,0}^j \in \Psi^j\}$

A. Belief evolution

Robot i maintains a belief \mathcal{B}_t^j for robot j 's distribution q_t^j simulated forwards in time, as introduced in Sec. IV-C. We define $\mathcal{B}_{t,0}^j$ as the belief of q_t^j assuming they just communicated, and thus has uncertainty $\sigma_{t,0}^j = 0$. Further, $\mathcal{B}_{t,\tau}^j$ denotes the belief of q_t^j assuming the most recent communication occurred τ iterations earlier. This evolution manifests as a lookahead graph, shown in the second phase of Fig. 1. The edges of the graph represent communication decisions at future iterations. The nodes across the bottom represent $\mathcal{B}_{t,0}^j$, while $\mathcal{B}_{t,\tau}^j$ is represented by the nodes above. Notably, this is a directed acyclic graph rather than a full decision tree, due to our formulation of always returning to the same $\mathcal{B}_{t,0}^j$ nodes when communication occurs, regardless of the history of previous decisions. As discussed later in Sec. V-C, this ensures the algorithm has polynomial runtime.

We define \mathcal{B}_t^j for the context of Dec-MCTS as follows. We propose a particle filter representation such that \mathcal{B}_t^j is a finite discrete set of distributions that are possible realisations

of q_t^j . For predicting $\mathcal{B}_{t,0}^j$ (i.e., the $a_t^j = \text{COMM}$ cases), the update function \mathcal{Q} in (2) is applied recursively to a single particle (in Alg. 2 line 3). For $\mathcal{B}_{t,\tau}^j, \tau > 0$, we add noise to the belief by first replicating the noiseless particle $\mathcal{B}_{t-\tau,0}^j$, then recursively applying a modified update function \mathcal{Q}' to each particle (lines 8 and 12). The main source of uncertainty for predicting q_t^j is caused by not knowing the expectations in (2). Thus, \mathcal{Q}' is defined in the same way as \mathcal{Q} , but with Gaussian noise added to the expectations. When applying \mathcal{Q}' , we use the noiseless predicted distributions for $q^{(j)}$. Uncertainty $\sigma_{t,\tau}^j$ is calculated as (3).

B. Optimal belief-space planner

The communication actions are decided using a polynomial-time dynamic programming algorithm that plans over the above belief space over a sliding-horizon. The algorithm finds the optimal communication schedule π for the next T iterations that has the minimum communication cost and ensures σ does not exceed the threshold θ . At each iteration, communication is requested from the other robots according to the first planned action a_0 in π .

The dynamic programming algorithm is shown in Alg. 2. The line 1 loop predicts the distributions q_t^j for all robots and future iterations t , using the *noiseless* belief evolution (Sec. V-A). Then, each iteration of the line 4 loop plans the sequence of future communication requests to robot j . This sequence is determined by first performing a forward pass through the belief evolution graph, and then backtracking to find the optimal sequence. During the forward pass, the beliefs at the graph nodes are determined using the *noisy* belief evolution (Sec. V-A). The best feasible node that precedes each node is saved as the backpointer $\psi_{t,\tau}^j$ (line 21). A node is feasible only if $\sigma \leq \theta$ (line 19). The backpointer $\psi_{t,0}^j$ represents the optimal schedule for the sub-problem from iteration 0 to t where $a_t^j = \text{COMM}$. This schedule has cost $\Omega_{t,0}^j$ (line 22). When selecting $\psi_{t,\tau}^j$, ties are broken by minimising accumulated uncertainty. Finally, during the backward pass (lines 23-26), the schedule is found by choosing the node at iteration T with the lowest cost $\Omega_{T,\tau}^j$, and then following backpointers ψ^j .

C. Analysis

The schedule is optimal with respect to the belief and the time horizon and is guaranteed to satisfy the uncertainty constraints. The algorithm has polynomial runtime, with complexity $\mathcal{O}(BT^2RE)$, where B is the number of particles for each decision node, T is the number of steps in the planning horizon, R is the number of robots, and E is time taken to compute the expected utility. We note that due to our construction, the total number of particles generated by the algorithm is $\mathcal{O}(BT^2R)$, which, importantly, is polynomial in the time horizon (rather than exponential, as would occur in a typical decision tree). Typically, we expect that E is large for non-trivial problems; therefore, B or T should be selected to strike a balance between runtime and desired accuracy of the predictions.

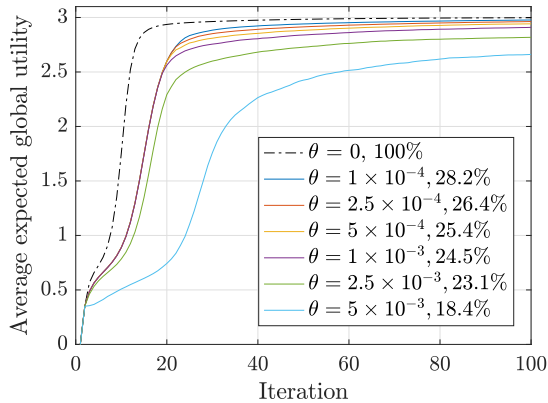


Fig. 2. Average expected global utility scores for different θ values, for each iteration of the program. The legend provides the corresponding communication rate relative to the full communication case ($\theta = 0$).

VI. EXPERIMENTS

We analyse the performance and behaviour of our proposed approach in two scenarios. Firstly, we use a simplified version of Dec-MCTS and a relatively simple objective function. Secondly, we use the full Dec-MCTS and an information gathering objective function that is a generalisation of the orienteering problem. Overall, the results show that the performance of Dec-MCTS can be maintained, even with significantly reduced communication rates.

The orienteering problem is a useful experimental scenario because it is a well-known abstraction that applies to a range of real tasks [19]. Although it is difficult to identify alternative algorithms that can be directly compared to ours, we do provide comparisons to a suite of communication reduction approaches. One of these is full (effectively all-to-all) communication. This scheme is not feasible in practice for our systems of interest, which are field robots with significant channel contention that arises from sources such as RTK GPS corrections, e-stop heartbeat messages, and telemetry. However, the full communication case provides a quality benchmark that allows us to measure relative coordination (task) performance.

A. Distributed gradient descent

1) *Experimental setup:* We demonstrate the behaviour of the algorithm for a simple planning scenario. Here, we use only the distributed gradient descent component of Dec-MCTS, which selects a single action per robot rather than a sequence of actions. A team of 4 robots $\{r^1, r^2, r^3, r^4\}$, selects one of 4 possible actions, i.e., $\mathbf{x}^i \in \{1, 2, 3, 4\}$. The global objective is defined such that each consecutive pair of robots $(i, i + 1)$ receives a reward of 1 if robot $i + 1$ picks an action that is one greater than robot i , and the pair receives a reward of 0 otherwise; i.e., $g(\mathbf{x}) = \sum_{i=1}^{R-1} \mathbf{1}(\mathbf{x}^i = a, \mathbf{x}^{i+1} = a + 1)$. The globally optimal plan is therefore to select $\mathbf{x}^i = i, \forall i$, which receives a reward of $g(\mathbf{x}) = 3$; however, there are many local optima that the robots must avoid by successfully coordinating their actions.

TABLE I

Average total communication between pairs of robots over 100 iterations, for a threshold $\theta = 1 \times 10^{-3}$.

		Replying robot			
		r^1	r^2	r^3	r^4
Requesting robot	r^1	-	11.3	0	0
	r^2	81.3	-	18.1	0
	r^3	0	17.7	-	85.4
	r^4	0	0	11.4	-

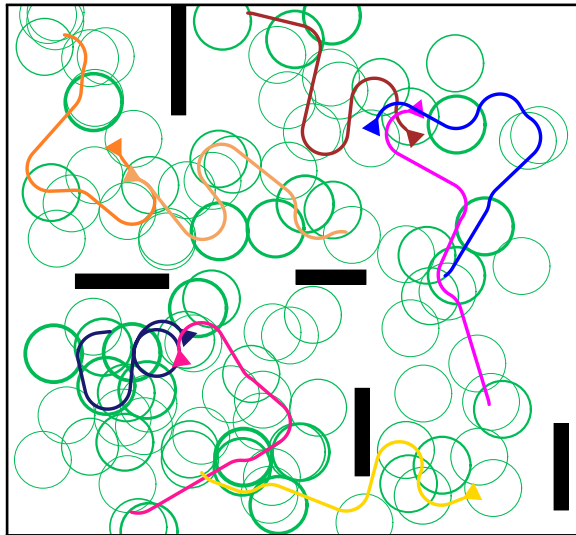
Robots evolve their lookahead graph over a 3-step horizon, with each ‘noisy’ node of the graph containing ten particles. The algorithm was tested for a range of θ values, with each being simulated for 100 iterations.

2) *Results:* The expected global utility for different values of θ , averaged over 100 trials, are given in Fig. 2. The results demonstrate a trade-off between communication rate and utility by varying θ , and show that reasonable final scores are achieved even when communication is reduced by 75%. The number of communications for each pair of robots is given in Table I, which shows communication is prioritised between robots that directly influence their expected local utility; e.g., r^2 requests updates from r^1 and r^3 , but not r^4 . Robots that converge slower (r^1 and r^4) receive more requests since higher entropy distributions contribute more to the uncertainty of the local utilities. Less communication is requested in later iterations as the entropy is reduced.

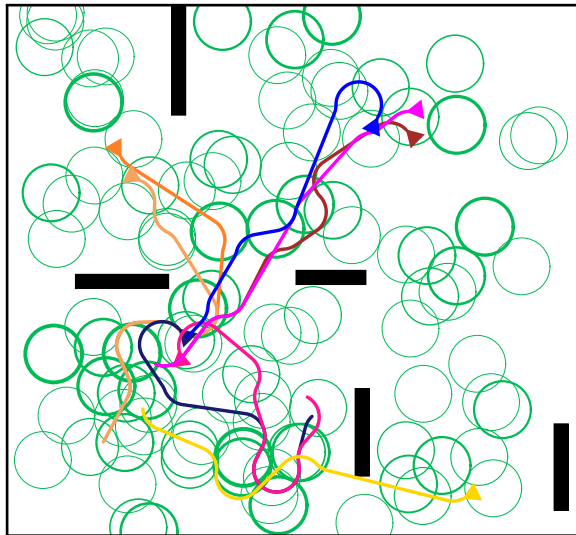
B. Coordinated information gathering

1) *Experimental setup:* We now investigate the performance in the context of Dec-MCTS. The scenario we consider here, illustrated in Fig. 3, is a multi-robot information gathering problem formulated as a generalisation of the orienteering problem (as in [1]). There are 8 Dubins robots that plan their paths over a PRM with 1000 nodes and several obstacles. The goal is to collectively maximise a weighted sum of disks visited by the team, where each disk can be visited from at least one vertex in the PRM. These disks may represent observation regions defined by a sensor model [19]. A reward is received for each visited disk, and the objective $g(\mathbf{x})$ is the sum of these rewards. Each robot’s path is constrained by a maximum path length, which may represent fuel constraints or a planning horizon.

In our experiments we compare 5 different scenarios. The *All-to-all* scenario makes the unrealistic assumption of perfect communication and the robots communicate their intentions at every iteration. *Random* represents a scenario where only 20% of the packets are successfully received due to uniform-random message loss (e.g., to model excessive contention on the communication channel). We compare two versions of our approach: *Horizon 4* plans with a planning horizon of $T = 4$, while *Greedy* only looks one time-step ahead. In the *Horizon 4* and *Greedy* scenarios, θ is selected such that they have a 20% average communication rate. As a baseline comparison, the *None* scenario assumes



(a) **Communication** (*All-to-all*) results in successful coordination.



(b) **No communication** (*None*) results in poor coordination.

Fig. 3. Information gathering problem instance for the experiments in Sec. VI-B, with example solution paths (coloured lines). Arrows show start location and orientation for 8 robots. Green disks are reward regions (weighted by reward).

all communication fails and no messages are successfully received.

2) *Results*: For a baseline comparison we observe that communication is important for coordination. In Fig. 3(a), the robots take advantage of the perfect setting of having full communication (*All-to-all*) to coordinate their plans effectively. In (b), there is no communication (*None*) and therefore there is no coordination, resulting in multiple visits to the same regions. Table II compares the planning performance for different communication scenarios. *All-to-all* naturally resulted in the highest reward, but the partial communication scenarios performed well despite having 80% less communication. As expected, *None* resulted in the poorest performance. Planning with a horizon of $T = 4$ achieved higher rewards than *Greedy*, showing the advantage

TABLE II

Reward collected (as a percentage) in the information gathering problem with the different communication scenarios. Iterations are the sum of planning iterations performed by the 8 robots. Rewards averaged over 50 trials in the environment illustrated in Fig. 3. Average standard deviation of reward is 7%.

Iterations	All-to-all	Random	Planned schedules		
			Horizon 4	Greedy	None
50	23.2	17.4	23.5	23.1	15.9
100	65.0	54.0	62.3	60.4	32.6
150	71.2	64.5	66.3	67.3	32.6
200	76.1	68.7	69.5	69.6	33.9
250	74.1	66.8	71.3	69.9	31.8
300	73.5	66.3	71.6	69.6	32.8
350	74.5	65.5	73.4	71.7	35.1
400	76.7	68.9	76.3	74.5	33.0

of planning over a time horizon. Both of these scenarios outperformed *Random*, which highlights the practical benefit of performing informative communication planning.

The planned communication scenarios achieved better results than *Random* since the proposed approach chose to communicate more frequently for pairs of robots that have a larger coupling between their local utilities. For the $T = 4$ scenarios, the highest communication rate (62%) is between the blue and pink robots in the bottom left of Fig. 3(a). We expect this pair to communicate more since their reachable regions significantly overlap. The yellow robot in the bottom right received the least requests (11%) since it is relatively isolated. The algorithm also selects *when* to communicate, which tended to be more during earlier iterations when successful coordination is most important.

VII. CONCLUSION AND FUTURE WORK

We have proposed a principled approach for making communication decisions during online decentralised coordination. We have demonstrated that our planning-aware communication approach achieves large reductions in channel utilisation with little impact on task performance.

Our results motivate several avenues for future work. We are interested in extending the approach for cases where we have models for predicting network integrity, e.g. [17], to more effectively use available resources. Our approach does not enforce hard constraints on bandwidth, however our method of measuring information value solves a necessary step towards enforcing hard bandwidth constraints while also considering the coordination performance. Our approach could be extended to the constrained bandwidth case by swapping the objectives and constraints; however, the interesting challenge here is to suitably generalise the communication decision making step without incurring the substantial time penalty of considering a full decision tree in this case. Other interesting lines of inquiry include extending the belief evolution model for Dec-MCTS to also predict the effect of the periodic domain changes caused by the tree search phase. Furthermore, prediction models could be developed for other decentralised planning algorithms,

such as max-sum [20]. It would also be interesting to trial planning-aware communication during other coordinated perception tasks, such as active object classification [32], [33]. Finally, different communication network models may be used, such as broadcasting rather than point-to-point, or sequential communication rather than concurrent.

REFERENCES

- [1] G. Best, O. Cliff, T. Patten, R. R. Mettu, and R. Fitch, "Dec-MCTS: Decentralized planning for multi-robot active perception," *Int. J. Robot. Res.*, 2018, doi:10.1177/0278364918755924.
- [2] A. Kassir, R. Fitch, and S. Sukkarieh, "Communication-aware information gathering with dynamic information flow," *Int. J. Robot. Res.*, vol. 34, no. 2, pp. 173–200, 2015.
- [3] M. Vinyals, J. A. Rodriguez-Aguilar, and J. Cerquides, "A survey on sensor networks from a multiagent perspective," *Comput. J.*, vol. 54, no. 3, pp. 455–470, 2010.
- [4] P. K. Penumarthi, A. Q. Li, J. Banfi, N. Basilico, F. Amigoni, J. O'Kane, I. Rekleitis, and S. Nelakuditi, "Multirobot exploration for building communication maps with prior from communication models," in *Proc. of MRS*, 2017, pp. 90–96.
- [5] G. A. Hollinger, S. Yerramalli, S. Singh, U. Mitra, and G. Sukhatme, "Distributed coordination and data fusion for underwater search," in *Proc. of IEEE ICRA*, 2011, pp. 349–355.
- [6] N. Yilmaz, C. Evangelinos, P. F. J. Lermusiaux, and N. Patrikalakis, "Path planning of autonomous underwater vehicles for adaptive sampling using mixed integer linear programming," *IEEE J. Oceanic Eng.*, vol. 33, no. 4, pp. 522–537, 2008.
- [7] S. Kemna, J. G. Rogers, C. Nieto-Granda, S. Young, and G. S. Sukhatme, "Multi-robot coordination through dynamic Voronoi partitioning for informative adaptive sampling in communication-constrained environments," in *Proc. of IEEE ICRA*, 2017, pp. 2124–2130.
- [8] M. Otte, M. Kuhlman, and D. Sofge, "Multi-robot task allocation with auctions in harsh communication environments," in *Proc. of MRS*, 2017, pp. 32–39.
- [9] V. Kuo and R. Fitch, "Scalable multi-radio communication in modular robots," *Robot. Auton. Syst.*, vol. 62, no. 7, pp. 1034–1046, 2014.
- [10] S. Williamson, E. Gerding, and N. Jennings, "A principled information valuation for communications during multi-agent coordination," in *Proc. of AAMAS Workshop on Multi-Agent Sequential Decision Making in Uncertain Domains*, 2008, pp. 137–151.
- [11] R. Fitch, D. Sofge, G. Hollinger, K. Dantu, M. Otte, and S. Omidshafiei, "Workshop on robot communication in the wild: Meeting the challenges of real-world systems," in *Robotics: Science and Systems*, 2017.
- [12] G. Best, O. M. Cliff, T. Patten, R. R. Mettu, and R. Fitch, "Decentralised Monte Carlo tree search for active perception," in *Proc. of WAFR*, 2016.
- [13] C. Browne, E. Powley, D. Whitehouse, S. Lucas, P. Cowling, P. Rohlfshagen, S. Tavener, D. Perez, S. Samothrakis, and S. Colton, "A survey of Monte Carlo tree search methods," *IEEE Trans. Comput. Int. AI Games*, vol. 4, no. 1, pp. 1–43, 2012.
- [14] D. H. Wolpert and S. Bieniawski, "Distributed control by Lagrangian steepest descent," in *Proc. of IEEE CDC*, 2004, pp. 1562–1567.
- [15] R. Becker, A. Carlin, V. Lesser, and S. Zilberstein, "Analyzing myopic approaches for multi-agent communication," *Comput. Intell.*, vol. 25, no. 1, pp. 31–50, 2009.
- [16] A. Ghaffarkhah and Y. Mostofi, "Communication-aware motion planning in mobile networks," *IEEE Trans. Autom. Control*, vol. 56, no. 10, pp. 2478–2485, 2011.
- [17] M. Lindh  and K. Johansson, "Exploiting multipath fading with a mobile robot," *Int. J. Robot. Res.*, vol. 32, no. 12, pp. 1363–1380, 2013.
- [18] G. Best, W. Martens, and R. Fitch, "Path planning with spatiotemporal optimal stopping for stochastic mission monitoring," *IEEE Trans. Robot.*, vol. 33, no. 3, 2017.
- [19] G. Best, J. Faigl, and R. Fitch, "Online planning for multi-robot active perception with self-organising maps," *Auton. Robots*, 2018, doi:10.1007/s10514-017-9691-4.
- [20] A. Farinelli, A. Rogers, A. Petcu, and N. R. Jennings, "Decentralised coordination of low-power embedded devices using the max-sum algorithm," in *Proc. of AAMAS*, 2008.
- [21] L. Liu, N. Michael, and D. A. Shell, "Communication constrained task allocation with optimized local task swaps," *Auton. Robots*, vol. 39, no. 3, pp. 429–444, 2015.
- [22] M. Tambe, "Towards flexible teamwork," *J. Artif. Intell. Res.*, vol. 7, pp. 83–124, 1997.
- [23] M. Otte and N. Correll, "Any-com multi-robot path-planning: Maximizing collaboration for variable bandwidth," in *Distributed Autonomous Robotic Systems: The 10th International Symposium*. Springer, 2013, pp. 161–173.
- [24] J. Renoux, A.-I. Mouaddib, and S. LeGloannec, "Distributed decision-theoretic active perception for multi-robot active information gathering," in *Modeling Decisions for Artificial Intelligence*. Springer, 2014, pp. 60–71.
- [25] M. Roth, R. Simmons, and M. Veloso, "Reasoning about joint beliefs for execution-time communication decisions," in *Proc. of AAMAS*, 2005, pp. 786–793.
- [26] A. Kassir, R. Fitch, and S. Sukkarieh, "Communication-efficient motion coordination and data fusion in information gathering teams," in *Proc. of IEEE/RSJ IROS*, 2016, pp. 5258–5265.
- [27] P. J. Gmytrasiewicz and E. H. Durfee, "Rational communication in multi-agent environments," *Auton. Agents Multi-Ag. Syst.*, vol. 4, no. 3, pp. 233–272, 2001.
- [28] V. V. Unhelkar and J. A. Shah, "ConTaCT: Deciding to communicate during time-critical collaborative tasks in unknown, deterministic domains," in *Proc. of AAAI*, 2016.
- [29] G. Best and R. Fitch, "Bayesian intention inference for trajectory prediction with an unknown goal destination," in *Proc. of IEEE/RSJ IROS*, 2015, pp. 5817–5823.
- [30] P. Ondruška, C. Gur u, L. Marchegiani, C. H. Tong, and I. Posner, "Scheduled perception for energy-efficient path following," in *Proc. of IEEE ICRA*, 2015, pp. 4799–4806.
- [31] S. Bopardikar, S. Smith, and F. Bullo, "On dynamic vehicle routing with time constraints," *IEEE Trans. Robot.*, vol. 30, no. 6, pp. 1524–1532, 2014.
- [32] T. Patten, W. Martens, and R. Fitch, "Monte Carlo planning for active object classification," *Auton. Robots*, vol. 42, no. 2, pp. 391–421, 2018.
- [33] T. Patten, M. Zillich, R. Fitch, M. Vincze, and S. Sukkarieh, "Viewpoint evaluation for online 3-D active object classification," *IEEE Robot. Autom. Lett.*, vol. 1, no. 1, pp. 73–81, 2016.