Probabilistic Planning for AUV Data Harvesting from Smart Underwater Sensor Networks

Matthew Budd[1], Georgios Salavasidis[2], Izzat Kamarudzaman[2], Catherine A. Harris[2],
Alexander B. Phillips[2], Paul Duckworth[1], Nick Hawes[1] and Bruno Lacerda[1]

Abstract— **Harvesting valuable ocean data, ranging from climate and marine life analysis to industrial equipment monitoring, is an extremely challenging real-world problem. Sparse underwater sensor networks are a promising approach to scale to larger and deeper environments, but these have difficulty offloading their data without external assistance. Traditionally, offloading data has been achieved by costly, fixed communication infrastructure. In this paper, we propose a planning under uncertainty method that enables an autonomous underwater vehicle (AUV) to adaptively collect data from smart sensor networks in underwater environments. Our novel solution exploits the ability of sensor nodes to provide the AUV with time-of-flight acoustic localisation, and is able to prioritise nodes with the most valuable data. In both simulated experiments and a real-world field trial, we demonstrate that our method outperforms the type of hand-designed behaviours that has previously been used in the context of underwater data harvesting.**

I. INTRODUCTION

Underwater acoustic sensor networks (UWASNs) are collections of spatially distributed underwater sensor nodes that measure environmental phenomena. These networks can bring substantial value, since it is difficult to collect ocean data manually, and applications have included monitoring of climate, wildlife and infrastructure. However, underwater acoustic communications have limited range and low throughput; acoustic modems with cost and power requirements suited to UWASNs are generally limited to $\sim$2km range and in the order of 100s of bits per second bandwidth [1]. Thus, unlike their land-based counterparts, UWASNs do not have the luxury of a reliable, high bandwidth, low latency connection to the internet.

Traditionally, UWASNs would transmit their data to a *gateway* node situated on a surface buoy or ship [1], [2], which then relays the data onwards. Across larger distances, multi-hop networks transfer data between nodes to the gateway. However, to ensure all data can reach the gateway, such network architectures rely on relatively dense node distributions or many surface gateways. Gateway buoys far out to sea must rely on expensive and low throughput satellite communications, and can be completely impractical for sparse or deep sensor networks. Furthermore, communication power usage significantly increases on nodes close to the gateway when they must relay other nodes' messages. This shortens their lifespan and therefore that of the network. As a motivating example, consider the remote large-scale RAPID array [3] in the north Atlantic ocean. Climate data must be harvested from this network by a research ship every 18 months at a significant cost.

Recent advances in mobile robotics allow for more intelligent data harvesting solutions. In particular, the use of autonomous underwater vehicles (AUVs) to retrieve data from sensor networks is becoming a popular research area [4], [5]. A key advantage is the AUVs' ability to travel close to sensor nodes to benefit from increased acoustic throughput or make use of short-range high-bandwidth optical or radio communication [6], [7]. However, existing AUV data retrieval methods do not adequately consider the high uncertainty inherent to the real-world data harvesting task. In particular, cost-effective small AUVs have limited computing power and poor dead-reckoning underwater localisation. Although depth information is straightforwardly known via sensed pressure, 2D latitude/longitude position uncertainty for these vehicles can be between $\sim 10\%$ to $\sim 30\%$ of distance traveled without external position feedback, depending on the environment [8]. This localisation error is caused by the unobserved effects of water currents and vehicle dynamics uncertainty and noise. AUV navigation can therefore vary significantly in the end location and time taken. Furthermore, sensor node data contents are not typically known a priori. Thus, decisions on which sensors should be harvested first must be made during the mission, when the information regarding their contents is communicated to the AUV. Such decisions are typically rule based. In contrast, our approach considers a prior distribution over the data content of each sensor node at planning time. This uncertainty is then resolved when the information is received during the mission, allowing the AUV to optimally select which nodes to harvest given the amount of time still left in the mission.

The main contribution of this paper is a novel Markov decision process (MDP) formulation of an AUV mission planning problem that aims to maximise the expected total utility of data retrieved from a UWASN under a time bound. Unlike existing data harvesting methods, our approach accounts for uncertainty both in underwater environmental dynamics and the UWASN data contents. To ensure our approach can be deployed in low-cost AUVs, which typically have few on-board computation capabilities, the MDP is solved offline and the synthesised policy is loaded onto the AUV. Policy execution is then achieved by a simple lookup table. Furthermore, the MDP state is based on the sensor nodes the AUV is able to communicate with, significantly reducing

[1] Oxford Robotics Institute, University of Oxford; {mbudd, pduckworth, nickh, bruno}@robots.ox.ac.uk
[2] Marine Autonomous and Robotic Systems, National Oceanography Centre, Southampton; {geosal, izzat.kamarudzaman, cathh, abp}@noc.ac.uk

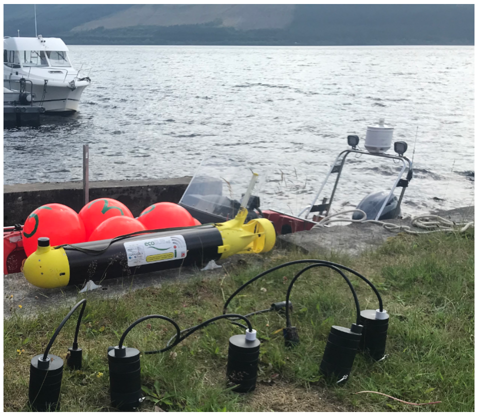

Fig. 1. ecoSUB AUV alongside 5 smart sensor nodes (foreground) at the trial site, Loch Ness, Scotland.

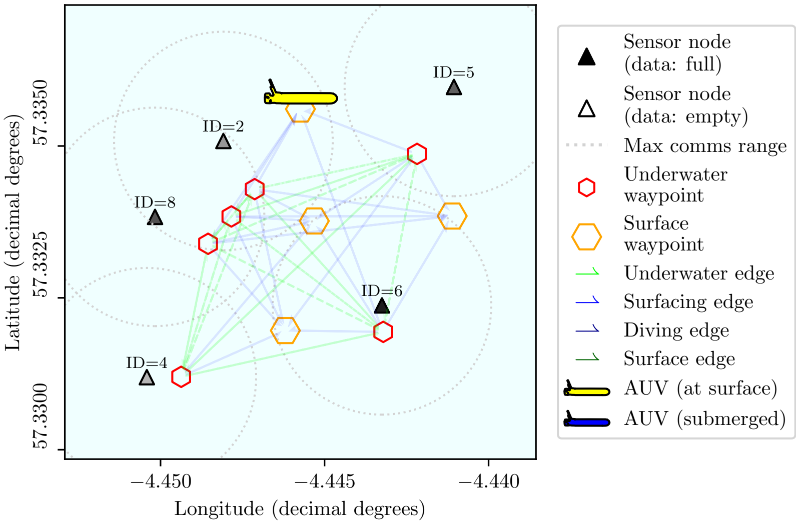

Fig. 2. Planner representation of the Loch Ness field trial data gathering scenario, showing 4 surface and 6 underwater waypoints.

the computational burden of state estimation too. Empirical evaluation is carried out in simulated AUV experiments, and a fully integrated system is demonstrated as proof-of-concept in a real-world field trail at Loch Ness in Scotland using a deployed UWASN and low-cost AUV (Fig. 1). In both cases, we outperform the types of hand-designed rule-based systems commonly deployed for AUV mission planning.

II. Related Work

Most existing AUV data harvesting methods plan shortest touring paths between sensor nodes, with no consideration of navigation, localisation, or node data contents uncertainty [4], [9]. To deploy these methods in the real world, high-quality localisation could be achieved using costly, bulky and power-hungry inertial navigation units and currents velocity sensors. Alternatively, one could receive position feedback by covering the operating area with acoustic localisation infrastructure. Again, this is expensive, often impractical, and wasteful since precise localisation is unnecessary in areas with no sensor nodes present. Our planning method is therefore designed to take advantage of sensor nodes that can themselves provide time-of-flight acoustic “ping” localisation to the AUV.

Similarly to our method, some existing works make use of kinematic navigation simulators [5]. We build probabilistic *policies*, rather than fixed plans, based on these uncertain models of the AUV and environment. By explicitly considering uncertainty in the outcomes of AUV navigation actions, our method allows the AUV to reason about localisation only as far as it needs to to carry out its data retrieval mission.

Furthermore, to the best of our knowledge no other methods are able to reason about collecting data from a *smart* sensor network with uncertain amounts of different values of data present. Even where heterogeneous data types are considered [4], [5], it is assumed that the data contents of sensor nodes is known before planning the mission. This is clearly not the case when sensor nodes have no persistent communication link to the outside world: the application for which AUV data retrieval is best suited.

When the AUV is underwater, its position is uncertain and partially observable via time-of-flight pings from sensor nodes. A partially observable MDP (POMDP) [10] approach to underwater navigation could be to define a grid of (lat, lon) positions underwater, and maintain a belief over the AUV’s grid position. The Adaptive Belief Tree (ABT) [11] algorithm uses this formulation as an example domain. For our task, this grid MDP contains many states with no utility as they would not be in contact with any sensor node. Our approach limits the state space size by only defining underwater states at *waypoints* where the vehicle can communicate with sets of beacons. One existing work [12] similarly defines underwater waypoints and carries out time-dependent planning between these. However, this planner explicitly considers only execution time uncertainty; it must replan online when actions do not result in the assumed outcome.

POMDP planning is highly intractable, and scalable POMDP methods such as ABT generally carry out online planning. As pre-computing a policy has large advantages in terms of the power and compute resources required on-board the AUV, we avoid online POMDP belief planning. Instead we take a novel approach and measure navigation success by whether the AUV is able to communicate with the target nodes it was trying to find. The AUV controller carries out line-of-sight navigation control alongside position estimation via an extended Kalman filter (EKF). The EKF is well suited to this type of trilateration-based position estimation problem [8].

III. Preliminaries

A. Continuous-Time Markov Decision Processes

We will use continuous-time Markov decision processes (CTMDPs) to model the transmission of data from sensor nodes to the AUV. A CTMDP is a tuple $\mathcal{Q} = \langle S, in, A, \Delta, \mathcal{R} \rangle$, where S is a finite set of states; $in \in Dist(S)$ is the initial state distribution; A is a finite set of actions; $\Delta : S \times A \times S \to \mathbb{R}_{\geq 0}$ is the rate transition function; and $\mathcal{R} : S \times A \to \mathbb{R}_{\geq 0}$ is the reward rate function. In a CTMDP, when action a is taken at state s, a *race condition* between n processes occur, one process for each state s' such that $\Delta(s, a, s') > 0$. The duration of the process that results in a transition to s' is modelled as an exponential distribution with rate $\Delta(s, a, s')$. Thus, defining $E(s, a) = \sum_{s' \in S} \Delta(s, a, s')$, the probability of the CTMDP evolving to s' given action a was taken in state s is given by $\Delta(s, a, s')/E(s, a)$ and the *sojourn time* in state s given that action a was taken is exponentially distributed with a rate $E(s, a)$. Reward is accumulated at the rate $\mathcal{R}(s, a)$ while action a is taken in state s.

B. (Discrete) Timed Markov Decision Processes

To model the navigation of the AUV, we will use a (discrete) timed MDP (TMDP) [13], which assumes a discrete distribution over the duration of actions. TMDPs provide a simplified model of action duration, which can easily be encoded into the state space, yielding a (discrete-time) MDP, for which standard techniques such as value iteration can be used. We will exploit this and define the global planning model as a TMDP too, by discretising the data transmission CTMDPs into MDPs, as described in Section V-C.

A TMDP is defined as a tuple $\mathcal{M} = \langle S, \overline{s}, A, \delta, T, \Theta, R\rangle$, where S is a finite set of discrete states; $\overline{s} \in S$ is the initial state; A is a finite set of actions; $\delta : S \times A \times S \to [0, 1]$ is a probabilistic transition function where $\delta(s, a, s')$ is the probability of moving to state s', given that action a was executed at state s; $T = \{t_1, \ldots, t_{|T|}\} \subset \mathbb{N}_{\geq 0}$ is a finite set of discrete action execution times, which we assume to be in increasing order; $\Theta = \{\theta_{s,a,s'} \mid \delta(s, a, s') > 0\}$, where each $\theta_{s,a,s'} : T \to [0, 1]$ is a probability distribution over integer durations, representing the time taken (duration) to execute action a from s and finish in s'; and $R : S \times A \to \mathbb{R}_{\geq 0}$ is a reward function. During a given mission the AUV selects actions using a *policy* π which maps state-action histories in the TMDP to the next action. We consider TMDP problems with a finite *time-bound* $\beta \in \mathbb{N}_{>0}$. After the time bound, executing actions and receiving additional reward is not possible.

For a time-bound β, a TMDP can be converted into an MDP $\mathcal{M}^T_\beta = \langle S^T_\beta, A, \delta^T_\beta, R^T_\beta\rangle$, where the state is augmented with the current timestep: $S^T = \{(s, t) \in S \times \mathbb{N} \mid t \leq \beta + t_{|T|}\}$; the transition function is augmented to consider the uncertainty over durations and the time-bound:

$$\delta^T_{\mathbf{b}}\left((s,t), a, (s', t+k)\right) = \begin{cases} \delta(s, a, s')\theta_{s,a,s'}(k) & \text{if } t \leq b \\ 0 & \text{otherwise;} \end{cases} \tag{1}$$

and $R^T_{\mathbf{b}}((s, t), a) = R(s, a)$. The planning problem is formulated as maximising the expected total accumulated reward in $\mathcal{M}^T_\beta$, which yields a policy $\pi : S^T \to A$.

IV. PROBLEM SETUP

In our setting the UWASN consists of "smart" sensor nodes which adaptively choose when to record measurement data based on the environment's behaviour [5], [14]. For example, a node might use a higher sampling rate when readings change rapidly. Alternatively, nodes may detect low-probability, high-value events such as passing marine life. These sensor nodes know the value, or *utility* of the data they contain.

The UWASN consists of a set $\Phi = \{\phi_1, \ldots, \phi_{|\Phi|}\}$ of underwater *sensor nodes*. For $\phi \in \Phi$, we denote the location of ϕ as $loc(\phi) = (lat, lon, depth) \in \mathbb{R}^3$. The location of all nodes is known. These nodes are able to communicate with the AUV via acoustic communication, and potentially with each other if they are within some maximum communications distance. Nodes can send 3 types of acoustic packet: a *data transfer* packet of size b^d, a *localisation* packet of size b^l which provides the AUV with a distance estimate to the node, and a *data statistics* packet of size b^s describing the data it contains. The probability of the AUV successfully receiving the packet is a function the distance from the node to the AUV, the acoustic packet size, and environmental conditions.

There is also a set $D = \{d_1, ..., d_{|D|}\}$ of *data types*. A function $U : D \to \mathbb{R}_{>0}$ maps each data type d to its information utility per byte $U(d)$. Finally, we define $L : \Phi \times D \to \mathbb{Z}_{\geq 0}$ such that $L(\phi, d)$ is the number of bytes of data type d in ϕ. The total utility of data stored on a node ϕ is therefore $\sum_{d \in D} U(d) \cdot L(\phi, d)$. The AUV starts with a prior over the value of L, and only knows its actual value with certainty when it receives a data statistics packet from the node. Given a bound β on total mission time, the AUV's goal is to collect as much information utility U from the sensor nodes as possible.

The problem above has several sources of uncertainty. Key sources are the a priori unknown sensor node content; the navigation and localisation uncertainty; and the varying communication rate between the AUV and the sensor nodes. This rate emerges from stochastic receipt probabilities of individual data transfer packets.

V. MODEL CONSTRUCTION

This section details the construction of our TMDP planning model $\mathcal{M}$. As illustrated in Figure 3, $\mathcal{M}$ is the product of a single *navigation TMDP* $\mathcal{M}^n$ (Section V-A) and one *data retrieval CTMDP* Q_ϕ (Section V-B) for each sensor node ϕ.

A. Navigation Model

We start by defining an *AUV topological map*. We exploit the communication between the AUV and the sensor nodes, and define topological waypoints based on which sensors the AUV is able to communicate with (i.e. reliably receive data packets from). The AUV topological map is defined as a tuple $\mathcal{T} = \langle V, E\rangle$, where $V = V^s \cup V^u$ is the set of waypoints. V^s is a set of *surface waypoints* and V^u is a set of *underwater waypoints*. When the AUV is at the surface, it receives a GPS fix and its coordinates are known with high confidence. Thus, each waypoint $v^s \in V^s$ is associated with a location $loc(v^s) = (lat, lon, 0)$. One can take several approaches to discretise the surface into a set of waypoints, e.g. building a grid map or utilising the underwater waypoint locations to predict the areas that the AUV will likely surface

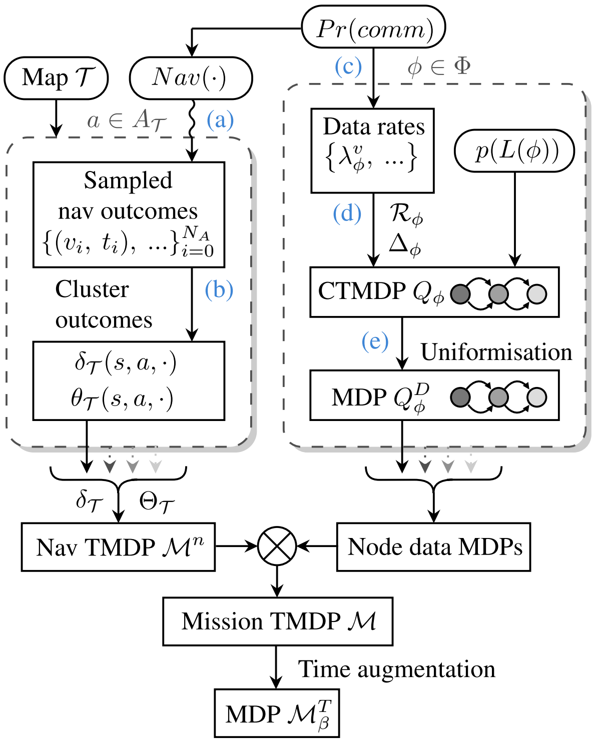

Fig. 3. High-level flow diagram depiction of the TMDP model construction and solution process. Blocks with dotted line edges and shadows represent multiple of the same type of component.

often. The set of underwater waypoints is defined based on which sensor nodes the AUV can communicate with. Thus, $V^u = \{\Phi' \subseteq \Phi \mid \forall_{\phi \in \Phi'}, \; \|loc(\phi), loc(\Phi')\|_2 < d^{comm}_{\max}\}$, where $loc(\Phi)$ is defined as the centroid of the locations of each sensor node $\phi' \in \Phi'$. Each underwater waypoint $v^u \subseteq \Phi$ is composed of a set of sensor nodes for which the AUV can be at a distance $d^{comm}_{\max}$ that ensures the AUV has a minimum probability of communicating with all $\phi \in v^u$. Edges in $\mathcal{T}$ connect waypoints that are within a predefined distance $d^{nav}_{\max}$ of each other. Furthermore, since we assume AUVs are not able to navigate reliably along the water surface due to the effects of waves and shipping collision hazards, edges between surface waypoints are not allowed. Formally, $E = \{(v, v') \in V \times V \mid \|loc(v), loc(v')\|_2 < d^{nav}_{\max}$ and $(v, v') \notin V^s \times V^s\}$.

There are two modes of *navigation between waypoints*. First, if the target waypoint is a surface waypoint, the AUV is instructed to navigate towards its location. When the AUV reaches the surface, it gets a GPS fix and we update its current waypoint to be surface waypoint closest to the AUV current location. Second, if the target waypoint is an underwater waypoint, the AUV is instructed to navigate towards the centroid of the sensor nodes that compose the target. Once the AUV's current localisation estimate is within a predefined distance to the centroid, the AUV starts pinging the sensor nodes. Its current waypoint is then updated to the waypoint that corresponds to the sensor nodes that pinged back, i.e. that the AUV managed to communicate with.

As the results of navigation and data collection are not readily known, we assume access to two black-box simulators. Firstly, the *comms model* $Pr(comm \mid dist, b)$ gives the probability of an acoustic packet of size b being successfully received at range $dist$. Secondly, the *navigation simulator* $Nav\,(v', t \mid v, e, Pr(comm \mid \cdot))$ is a kinematic simulator of the AUV and its localisation and guidance system. Given an initial location v, and target edge e, the navigation simulator simulates the vehicle attempting to navigate to the target waypoint specified by the edge. As Nav is a stochastic generative simulator, it returns samples $\tilde{v'}, \tilde{t} \sim Nav(\,\cdot \mid ...)$ of the outcome waypoint v' and time taken t rather than closed form probability distributions. $Nav()$ samples the probability of localisation ping success using the comms model. It also samples possible water currents dynamics and vehicle dynamics and control errors to encompass all navigation uncertainty and outcomes.

We can now define the *navigation TMDP*. Given an AUV topological map $\mathcal{T} = \langle V, E \rangle$, we define $\mathcal{M}_{\mathcal{T}} = \langle S_{\mathcal{T}}, \overline{s_{\mathcal{T}}}, A_{\mathcal{T}}, \delta_{\mathcal{T}}, T_{\mathcal{T}}, \Theta_{\mathcal{T}} \rangle$, where $S_{\mathcal{T}} = V$, i.e. the states correspond to topological map waypoints; $\overline{s_{\mathcal{T}}} \in V^s$, i.e. the AUV starts at a specified surface waypoint; and $A_{\mathcal{T}} = E$, i.e. actions correspond to attempting to navigate along topological map edges. The duration and state transition outcomes of navigation actions are stochastic. If the target waypoint v' is composed of more than one sensor node, it may not be possible to communicate with all sensor nodes in v' due to unsuccessful acoustic communication attempts. Alternatively, the vehicle's true path may diverge from its estimates due to lack of successful localisation pings. The AUV will either end up at another underwater waypoint, or will be completely lost and forced to surface. In this case it transitions to the nearest surface waypoint. The transition and duration distributions are estimated for each $e = (v, v')$ by sampling, from the navigation simulator, N_A attempts of navigating from v towards v'. This is shown as (a) in Figure 3, and yields dataset $x_{v,e} = \{(v_i, t_i)\}^{N_A}_{i=0}$. Then, for each $v'' \in S_{\mathcal{T}}$, the transition distribution $\delta_{\mathcal{T}}(v, e, v'')$ is obtained by calculating the frequency of data points (v_i, t_i) in $x_{v,e}$ such that $v_i = v''$. The duration distribution $\Theta_{\mathcal{T}}$ is similarly estimated by first clustering the times t_i in the dataset, yielding a cluster set $T^e = \{t^e_1, \dots, t^e_{|T^e|}\}$. This takes place at (b) in Figure 3. Then, for $t^e \in T^e$, $\theta_{v,e,v''}(t^e)$ is obtained by calculating the frequency of data points (v_i, t_i) in $x_{v,e}$ such that $v_i = v''$ and t^e is the element in T^e closest to t_i.

B. Data Retrieval CTMDPs

We use CTMDPs to represent data retrieval as a discretised stochastic counting process, in the form shown in Figure 4. The higher the number of states in the CTMDP, the closer the model matches the true underlying behaviour of individual bytes of data being probabilistically retrieved from the node, with probabilities given by the comms model.

Recall that sensor node ϕ has some a priori unknown data function $L : \Phi \times D \to \mathbb{Z}_{\geq 0}$ such that $L(\phi, d)$ is the number of bytes of data type d in ϕ. For each $\phi \in \Phi$, $d \in D$, we model the transfer of data of type d from ϕ to the AUV as a CTMDP $\mathcal{Q}_{\phi,d} = \langle S_{\phi,d}, in_{\phi,d}, A_{\phi,d}, \Delta_{\phi,d} \rangle$. $S_{\phi,d} = \{s^0_{\phi,d}, \dots, s^k_{\phi,d}\}$, where each state $s^i_{\phi,d},\ i > 0$ represents a "chunk" of size $b^i_{\phi,d}$ bytes of data type d stored on the node, and $s^0_{\phi,d}$ represents that node ϕ has no more data of type d; $in_{\phi,d} : S_{\phi,d} \to [0,1]$ is the initial state distribution, which we will define later; $A_{\phi,d} = \{a_v \mid v \in V^u \text{ and } \phi \in v^u\}$, i.e. actions correspond to retrieving data type d from sensor node ϕ at waypoint v^u; $\Delta_{\phi,d} : S_{\phi,d} \times A_{\phi,d} \times S_{\phi,d} \to \mathbb{R}_{\geq 0}$ is the transition rate function and $\mathcal{R}_{\phi,d} : S_{\phi,d} \times A_{\phi,d} \to \mathbb{R}_{\geq 0}$ is the reward rate.

Using the comms model, we can calculate the expected retrieval rate λ^v_ϕ, in bytes per second, from sensor node ϕ when the vehicle is at waypoint v. This is shown by arrow (c) in Figure 3. When collecting data of type d, the reward rate is straightforwardly the rate of utility value retrieval, i.e. $\mathcal{R}(s^i_{\phi,d}, a_v)_{\phi,d} = U(d) \cdot \lambda^v_\phi$. A transition occurring from $s^i_{\phi,d}$ represents the node being exhausted of $b^i_{\phi,d}$ bytes of that type of data, meaning that the next transition must be to $s^{i-1}_{\phi,d}$. When the AUV is retrieving data at waypoint v, the transition rate from $s^i_{\phi,d}$ to $s^{i-1}_{\phi,d}$ is therefore equal to $\lambda^v_\phi / b^i_{\phi,d}$. Formally, for all $i \in \{1, \ldots, k\}$ and $a_v \in A_{\phi,d}$, $\Delta(s^i_{\phi,d}, a_v, s^{i-1}_{\phi,d}) = \lambda^v_\phi / b^i_{\phi,d}$, and $\Delta(s, a_v, s') = 0$ for all other s, s' and a. Therefore, the transition and reward rates (Figure 3, (d)) are dependent on the expected retrieval rate.

A full joint model of the data state for all of a node's data types d would require the parallel composition between the CTMDPs for each $d \in D$. However, this would lead to an exponential blow-up of the number of states. Furthermore, the best harvesting policy for the AUV is to first retrieve the highest utility available data, then move on the next highest utility data, and so on. Thus, we model the data transmission of each node as a *chain* of the CTMDPs for each $d \in D$, starting with the highest utility data type and transitioning to the lower levels. In Figure 4, we present the data transmission CTMDP for sensor node ϕ assuming two data types d_1 and d_2 such that $U(d_1) > U(d_2)$ for simplicity. The generalisation for more data types is straightforward. Let $\mathcal{Q}_{\phi,d_1} = \langle S_{\phi,d_1}, in_{\phi,d_1}, A_{\phi,d_1}, \Delta_{\phi,d_1} \rangle$ and $\mathcal{Q}_{\phi,d_2} = \langle S_{\phi,d_2}, in_{\phi,d_2}, A_{\phi,d_2}, \Delta_{\phi,d_2} \rangle$ be the data transmission CTMDPs for sensor node ϕ and data type d_1 and d_2. The CTMDP for sensor node ϕ is defined as $\mathcal{Q}_\phi = \langle S_\phi, in_\phi, A_\phi, \Delta_\phi \rangle$, where: $S_\phi = (S_{\phi,d_1} \setminus \{s^0_{\phi,d_1}\}) \cup S_{\phi,d_2}$; the initial state distribution is defined as:

$$in_\phi(s) = \begin{cases} in_{\phi,d_1}(s) & \text{if } s \in S_{\phi,d_1} \\ in_{\phi,d_1}(s^0_{\phi,d_1}) in_{\phi,d_2}(s) & \text{if } s \in S_{\phi,d_2}; \end{cases} \quad (2)$$

$A_{\phi,d} = A_{\phi,d_1}$ (note that $A_{\phi,d_1} = A_{\phi,d_2}$); and the transition rates are defined as:

$$\Delta_\phi(s, s') = \begin{cases} \Delta_{\phi,d_1}(s) & \text{if } s, s' \in S_{\phi,d_1} \\ \Delta_{\phi,d_1}(s, s^0_{\phi,d_1}) in_{\phi,d_2}(s') & \text{if } s = s^1_{\phi,d_1} \text{ and} \\ & s' \in S_{\phi,d_2} \\ \Delta_{\phi,d_2}(s) & \text{if } s, s' \in S_{\phi,d_2} \\ 0 & \text{otherwise.} \end{cases} \quad (3)$$

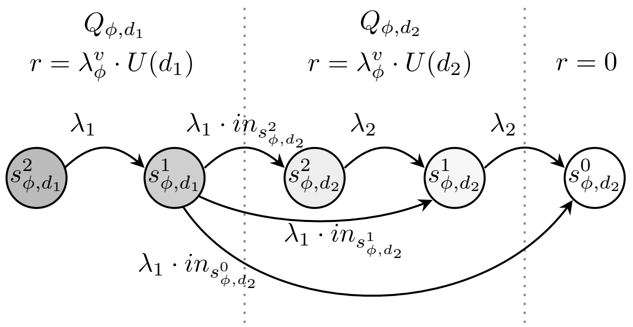
Q_{ϕ,d_1}
$r = \lambda^v_\phi \cdot U(d_1)$
Q_{ϕ,d_2}
$r = \lambda^v_\phi \cdot U(d_2)$
$r = 0$
λ_1
$\lambda_1 \cdot in_{s^2_{\phi,d_2}}$
λ_2
λ_2
s^2_{ϕ,d_1}
s^1_{ϕ,d_1}
s^2_{ϕ,d_2}
s^1_{ϕ,d_2}
s^0_{ϕ,d_2}
$\lambda_1 \cdot in_{s^1_{\phi,d_2}}$
$\lambda_1 \cdot in_{s^0_{\phi,d_2}}$

Fig. 4. A data retrieval CTMDP chain for two types of data, where ϕ is only contactable from one waypoint. The initial state distribution in_ϕ is defined across all states in the CTMDP chain but is omitted here for clarity.

This model assumes the AUV will start by retrieving the highest utility data in node ϕ. In the example in Figure 4, ϕ is only contactable from one waypoint so only a single data collection action is defined. The data contents for data types 1 and 2 are represented by two states each, each representing b_1 and b_2 bytes of data respectively. $\lambda_1 = \lambda^v_\phi / b_1$ and $\lambda_2 = \lambda^v_\phi / b_2$. Earlier in the data retrieval process, the reward rate is higher as $U(d_1) > U(d_2)$. When the node is empty (state s^0_{ϕ,d_2}), the reward rate is zero and no transitions are enabled.

The CTMDP chain simplification is a source of approximation: we cannot fully consider the information about the data contents in sensor node ϕ since we are only able to transition to an initial state according to in_ϕ. Broadly speaking, we can transition to the correct state representing the correct amount of the highest utility data in ϕ, but then the model can only transition to the states representing the next highest utility data types according to the prior over the data in ϕ. As shown later, this approximation still allows for efficient behaviour, since it still considers the information about the amount of data of the highest utility data type accurately.

C. Mission TMDP Construction

Let $\mathcal{M}_\mathcal{T}$ be a navigation TMDP and $\{\mathcal{Q}_{\phi,d}\}_{\phi \in \Phi}$ be the data retrieval CTMDPs. The AUV mission planning TMDP is a product composition of these models. We must first transform the data retrieval CTMDPs into a discrete MDP where all actions pass one navigation TMDP timestep.

We carry out a process similar to uniformisation [15] to transform each CTMDP Q_ϕ into an equivalent MDP Q^D_ϕ, given navigation TMDP timestep t. This occurs at arrow (e) in Figure 3. Uniformisation models a CTMDP with a discrete time MDP by adding self-loop action outcomes to states. As the timestep is given by the TMDP, rather than defined by the fastest rate of the CTMDP as is standard in uniformisation, we must carry out an adjustment of discrete time reward values to ensure they remain identical in expectation.

Let us define $P_\lambda(t \leq T)$ as the CDF of the first arrival time in a Poisson process, i.e. an exponential distribution. If, during one timestep of data retrieval from the node, the data for the current data state is not exhausted (i.e. a self-loop outcome in the discrete MDP), the data reward collected by the AUV in the timestep is simply $r_0 = \mathcal{V}_j \cdot \mu_i \cdot \tau$. If the data was exhausted during that timestep, the data value collected in the timestep is $r_{ex} = \mathcal{V}_j \cdot \mu_i \cdot \tau_{ex}$ where τ_{ex} is the expectation of when during the time period τ the data was exhausted. From Bayes' rule, we can define the probability that the first Poisson arrival took place within T_1 seconds given that it took place within $T_2 > T_1$ seconds:

$$P_\lambda(t \leq T_1 \mid t \leq T_2) = \frac{P_\lambda(t \leq T_2 \mid t \leq T_1) P_\lambda(t \leq T_1)}{P_\lambda(t \leq T_2)} \tag{4}$$

$$= \frac{P_\lambda(t \leq T_1)}{P_\lambda(t \leq T_2)} = \frac{1 - e^{-\lambda \cdot T_1}}{1 - e^{-\lambda \cdot T_2}}. \tag{5}$$

Differentiating with respect to T_1 to find the PDF,

$$p_\lambda(t = T_1 \mid t \leq T_2) = \frac{\lambda \cdot e^{-\lambda \cdot T_1}}{1 - e^{-\lambda \cdot T_2}}, \tag{6}$$

the expectation τ_{ex} can then be calculated:

$$\tau_{ex} = \int_{t'=0}^{\tau} t' \cdot p_{\lambda_{ij}}(t \leq t' \mid t \leq \tau)\; dt' \tag{7}$$

$$= \frac{1 - e^{-\lambda_{ij} \cdot \tau} \cdot (\lambda_{ij} \cdot \tau + 1)}{1 - e^{-\lambda_{ij} \cdot \tau} \cdot \lambda_{ij}}. \tag{8}$$

The mission TMDP $\mathcal{M}$ represents the dynamics of the entire system. Formally, $\mathcal{M} = \langle S, \overline{s}, A, \delta, T, \Theta, R \rangle$, where $S = V \times_{\phi \in \Phi} S^{+}_{\phi,d}$, where $S^{+}_{\phi,d} = S_{\phi,d} \cup \{\mathit{no\_comm}\}$ represents the amount of data in sensor node ϕ, plus a special initial state that represents that the AUV has yet to communicate with ϕ; $A = E \cup \{c_\phi\}_{\phi \in \Phi}$, where actions in $(v, v') \in E$ represent navigation actions and actions c_ϕ represent transfer of data from sensor node ϕ; for $s = (v, s_{\phi_1}, \ldots, s_{\phi_{|\Phi|}})$, $s' = (v', s'_{\phi_1}, \ldots, s'_{\phi_{|\Phi|}})$ and $a \in A$, the transition function is defined as:

$$\delta(s, a, s') = \begin{cases} \prod_{\phi \in \Phi^c_{v'}} p^{a,\phi}_{v \to v'} \cdot in_\phi(s'_\phi) & \text{if } a = (v, v'') \text{ and} \\ & \forall_{\phi \in \Phi \setminus \Phi^c_{v'}},\; s_\phi = s'_\phi \\ p_{s_{\phi_i} \to s'_{\phi_i}} & \text{if } a = c_{\phi_i} \text{ and} \\ & v = v' \text{ and} \\ & \forall_{\phi \in \Phi^{-i}},\; s_\phi = s'_\phi \\ 0 & \text{otherwise,} \end{cases} \tag{9}$$

where $\Phi^c_{v'} = \{\phi \in v' \mid s_{\phi_i} = \mathit{no\_comm}\}$ and $\Phi^{-i} = \Phi \setminus \{\phi_i\}$; and the reward $r_{\phi_i}(s_{\phi_i}, c_{\phi_i})$ is the uniformised discrete timestep reward from carrying out a data collection action from the relevant node.

For the first case in (9), navigation actions in the TMDP have some probability $p^{a,\phi}_{v \to v'}$ of receiving a data statistics message from node ϕ, either directly or via network state "gossiping". These probabilities for each action can be estimated using a more complex network communications model than we describe here, or given a constant rate per unit time when within localisation range of the node.

For the second case in (9), when a data collection action is selected in the TMDP, the transitions probabilities are defined by the transition probabilities in the relevant sensor node MDP. The waypoint and the state of all other node MDPs do not change.

VI. Experiments

In our experimental scenario, an UWASN with 5 nodes collects two types of data: *events* where $U(d_e) = 5$ and *measurements* where $U(d_m) = 1$. For ease of comparison, we use the same physical layout of sensor nodes and waypoints for simulated experiments as were used for the real field trial runs. This layout is as shown in Figure 2. For all results presented here, the mission length was 75 minutes. Each node was modelled with $|S_{\phi,d_e}| = 4$ and $|S_{\phi,d_m}| = 2$, where the expected number of bytes of measurements data is 400 and between 0 and 1000 bytes of events data, with higher probability assigned to lower numbers of events bytes.

The *baseline* comparison method is inspired by common industrial AUV rule-based mission specification. The baseline method travels between each single-node waypoint in a shortest touring path, having estimated travel times from distances. The baseline chooses departure times such that the expected time spent at each node is proportional to the expected total data utility at the node defined by the prior belief. For experiments in this paper, all nodes were assigned an identical data belief distribution, resulting in the baseline aiming to spend an equal amount of time at each node.

Finally, statistics messages are transmitted by the sensor nodes every 30 seconds. These have the same effective range and delivery probability as localisation pings from the node. For the comms model parameters used in these experiments, this was an effective maximum range of ∼400m. The comms model was provided by the developers of the Nanomodem acoustic modems [1] used in the field trial. The TMDP was solved using value iteration, with PRISM [16].

A. Real-World Robot Experiments

The AUV used for real-world experiments was an eco-SUB [17]. The policy was implemented as an SQLite database, allowing rapid action lookup in a ∼1GB policy with limited hardware. The localisation ping period (the time between attempts by the AUV to ping nodes to request a localisation ping) was 3 seconds. The primary source of uncertainty was therefore the unknown data distribution in the network.

Figure 5 shows one run each of the baseline (scoring data utility = 5256) and policy (scoring data utility = 8860) for the same data contents scenario. The plots show an example of intelligent adaptive behaviour from the policy. The policy-running AUV has decided that the additional event utility on the furthest away node (ID=4) is less valuable than remaining in the rest of the network closer to the goal location. It has therefore not visited that node during the mission. In a subsequent mission, the data value belief for this node would then be higher and the policy would more likely prioritise visiting it. For statistical comparison, running this experimental scenario in simulation (Section VI-B) with 20 repeats gives a policy mean score of 7939 (standard deviation (std) = 800) and baseline mean score of 5527 (std = 1454).

B. Experiments in Simulation

For statistical evaluation, simulations were run using the navigation and communication models. Both simulators'

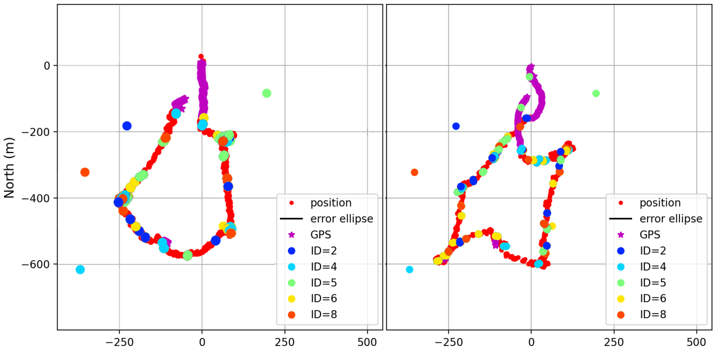

Fig. 5. AUV paths (estimated with acoustic localisation) in two real-world field trial missions. LHS: policy execution. RHS: baseline execution. Trajectory point colour corresponds to received acoustic localisation from a specific sensor node.

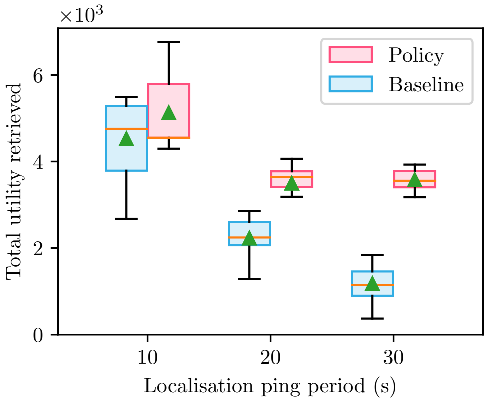

Fig. 6. Data collection performance vs localisation difficulty for policy and baseline on the same data collection scenario. 20 simulated runs per plotted box. Means shown as green triangles.

parameters were tuned to reasonably closely match the dynamics experienced during the real-world trials. Figure 6 illustrates the effects of decreasing localisation performance (and therefore increasing localisation and navigation uncertainty) on the performance of our method and the baseline. Each plotted box shows 20 simulated runs with the same fixed data distribution scenario. Despite a decreasing trend in retrieved utility value, the policy clearly outperforms the baseline by a greater margin as the uncertainty in the data collection mission grows.

VII. CONCLUSIONS

We have proposed a novel method for planning AUV data retrieval from sensor networks under uncertainty, and demonstrated its effectiveness in real and simulated experiments. To the best of our knowledge, ours is the only approach able to effectively plan in this problem setting.

The scalability of the approach described is limited primarily by the TMDP solution method, which produces a policy covering the entire state space. The state space is exponential in the number of nodes considered and the size of node CTMDPs. Analysing and improving scalability will therefore be addressed in future work. One avenue would be to design a hierarchical MDP model, where a node in our model would represent a cluster of physical sensor nodes which communicate between each other or which are all communicable from the same waypoint. A heuristic search solution method would also improve solution efficiency and produce more compact policies for loading onto the AUV.

ACKNOWLEDGMENTS

This work received EPSRC funding via the ORCA hub [EP/R026173/1] and the "From Sensing to Collaboration" programme grant [EP/V000748/1]. Matthew Budd is supported by an Amazon Web Services Lighthouse scholarship.

REFERENCES

[1] N. Morozs, P. D. Mitchell, Y. Zakharov, R. Mourya, Y. R. Petillot, T. Gibney, M. Dragone, B. Sherlock, J. A. Neasham, C. C. Tsimenidis, *et al.*, "Robust TDA-MAC for practical underwater sensor network deployment: Lessons from USMART sea trials," in *ACM International Conference on Underwater Networks & Systems*, 2018.

[2] J. Heidemann, M. Stojanovic, and M. Zorzi, "Underwater sensor networks: applications, advances and challenges," *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences*, vol. 370, 2012.

[3] W. E. Johns, L. Beal, M. Baringer, J. Molina, S. Cunningham, T. Kanzow, and D. Rayner, "Variability of shallow and deep western boundary currents off the Bahamas during 2004–05: Results from the 26 n RAPID–MOC array," *Journal of Physical Oceanography*, 2008.

[4] R. Duan, J. Du, C. Jiang, and Y. Ren, "Value-based hierarchical information collection for AUV-enabled internet of underwater things," *IEEE Internet of Things Journal*, vol. 7, 2020.

[5] J. Yan, X. Yang, X. Luo, and C. Chen, "Energy-efficient data collection over AUV-assisted underwater acoustic sensor network," *IEEE Systems Journal*, vol. 12, 2018.

[6] M. Dunbabin, P. Corke, I. Vasilescu, and D. Rus, "Data muling over underwater wireless sensor networks using an autonomous underwater vehicle," in *ICRA*, 2006.

[7] F. B. Teixeira, N. Moreira, R. Campos, and M. Ricardo, "Data muling approach for long-range broadband underwater communications," in *WiMob*, 2019.

[8] D. Fenucci, A. Munafo, A. B. Phillips, J. Neasham, N. Gold, J. Sitbon, I. Vincent, and T. Sloane, "Development of smart networks for navigation in dynamic underwater environments," in *IEEE/OES Autonomous Underwater Vehicle Workshop (AUV)*, 2018.

[9] G. A. Hollinger, S. Choudhary, P. Qarabaqi, C. Murphy, U. Mitra, G. S. Sukhatme, M. Stojanovic, H. Singh, and F. Hover, "Underwater data collection using robotic sensor networks," *IEEE Journal on Selected Areas in Communications*, vol. 30, 2012.

[10] L. P. Kaelbling, M. L. Littman, and A. R. Cassandra, "Planning and acting in partially observable stochastic domains," *AIJ*, vol. 101, 1998.

[11] H. Kurniawati and V. Yadav, "An online POMDP solver for uncertainty planning in dynamic environment," in *Robotics Research*, 2016.

[12] M. Cashmore, M. Fox, T. Larkworthy, D. Long, and D. Magazzeni, "Auv mission control via temporal planning," in *ICRA*, 2014.

[13] B. Lacerda, D. Parker, and N. Hawes, "Multi-objective policy generation for mobile robots under probabilistic time-bounded guarantees," in *ICAPS*, 2017.

[14] C. T. Chou, R. Rana, and W. Hu, "Energy efficient information collection in wireless sensor networks using adaptive compressive sensing," in *IEEE Conference on Local Computer Networks*, 2009.

[15] C. Baier, H. Hermanns, J.-P. Katoen, and B. R. Haverkort, "Efficient computation of time-bounded reachability probabilities in uniform continuous-time markov decision processes," *Theoretical Computer Science*, vol. 345, no. 1, pp. 2–26, 2005.

[16] M. Kwiatkowska, G. Norman, and D. Parker, "PRISM 4.0: Verification of probabilistic real-time systems," in *CAV*, 2011.

[17] A. B. Phillips, N. Gold, N. Linton, C. A. Harris, E. Richards, R. Templeton, S. Thuné, J. Sitbon, M. Muller, I. Vincent, *et al.*, "Agile design of low-cost autonomous underwater vehicles," in *OCEANS*, 2017.