Learning to Communicate Underwater

An Exploration of Limited Mobility Agents

Steven Porretta
Carleton University, School of Computer Science
Ottawa, Ontario

Michel Barbeau
Carleton University, School of Computer Science
Ottawa, Ontario

Joaquin Garcia-Alfaro
SAMOVAR, TELECOM SudParis
CNRS, Paris-Saclay University
91011 Evry, France

Evangelos Kranakis
Carleton University, School of Computer Science
Ottawa, Ontario

ABSTRACT

Underwater environmental parameters vary with time and can negatively impact the quality of communication. The adaptive control system suggested attempts to improve communication in underwater networks where environmental conditions are stochastic and time-variant. Adaptive depth control is explored in limited mobility underwater acoustic sensor networks. The adaptive control seeks to leverage the acoustic properties along the thermocline sensors anchored to the seabed in order to improve link stability in underwater networks. The sensor is allowed two directions of movement, either surface or dive, in order to avoid physical phenomena that cause faults. The algorithm presented is capable of adapting to the optimal performance depth in unimodal stochastic stationary and non-stationary environments.

ACM Reference Format:
Steven Porretta, Michel Barbeau, Joaquin Garcia-Alfaro, and Evangelos Kranakis. 2017. Learning to Communicate Underwater: An Exploration of Limited Mobility Agents. In *WUWNET'17: WUWNET'17: International Conference on Underwater Networks & Systems, November 6–8, 2017, Halifax, NS, Canada.* ACM, New York, NY, USA, 5 pages. https://doi.org/10.1145/3148675.3148709

1 INTRODUCTION

Mobile Ad-hoc Networks (MANETs) are a prominent topic of research in terrestrial networks. The applications possible for MANETs in underwater environments is vast and can range from naval security to seabed mining operations. Currently, underwater applications of MANETs are limited, in part due to the physical difficulties in establishing wireless communication networks underwater. The stochastic nature of underwater acoustic environments creates difficulties for communication [1]. Much of the existing work in the routing of underwater communications focuses on fixed rule algorithms for transmission that do not leverage the mobility of agents, or utilize the opportunity to change their rule structure with changes in the acoustic environment [9]. An existing problem in underwater communication is the fact that the acoustic qualities of oceanic media have a tendency to change both seasonally and with local weather phenomena [1, 11]. Although, these acoustic properties are unavoidable, it is the goal of this work to demonstrate that it may be possible for depth-varying anchored Limited Mobility Agents (LMAs) to utilize adaptive rule learning strategies to alter their depth in order to improve communication. Currently, many of the sensor networks which allow for variable depth of anchoring rely on human interaction to change node depths, a costly process which results in few movements of the sensor with respect to sensor operating lifespan. Replacing that human interaction with a motorized component would allow limited mobility to network elements which can be leveraged for sensing tasks and communications alike. Ultimately, the work herein seeks to provide a basis for leveraging acoustic sound speed changes along the thermocline of an underwater acoustic environment with the intent of improving link stability in an Underwater Acoustic Sensor Network (UASN).

A learning strategy that allows LMAs to adapt to the best depth of operation may not only be a tactic for fault avoidance in UASN, but could also be an acceptable strategy to avoid collisions in UASNs, since an adaptive strategy would not discriminate against the cause of faults in the network. A significant motivator of using learning strategies to avoid faults in UASN is the fact that the agents in UASNs are economically expensive. The cost of UASN agents inspire network sparsity, and it is this agent sparsity which motivates a scheme that can avoid faults, or link failures, in underwater communication.

Partan et al. [10] describes a series of physical limitations to underwater acoustic communication, among which are concerns including shadow zones, multipath interference, and near surface bubble cloud regions. These physical properties not only cause link failure in UASNs, but are demonstrably characteristic of a time-varying stochastic environment [2, 3]. Although the observation of stochastic features like bubble clouds, caustic regions, and shadow regions seems trivial, it is an important observation when considering the work of Narendra et al. into learning automata techniques to improve communication in terrestrial telephone networks [5, 6, 8]. Narendra et al. [5, 6, 8] use a Mean Action Learning Automaton (M-automaton) to create adaptive rule routing in a stochastic demand telephone network and demonstrate improvements in performance over traditional fixed rule routing. In a similar way, M-automaton can be leveraged in 3D underwater network topologies where agent

WUWNET'17, November 6–8, 2017, Halifax, NS, Canada
© 2017 Association for Computing Machinery.
ACM ISBN 978-1-4503-5561-2/17/11…$15.00
https://doi.org/10.1145/3148675.3148709

depth can be varied as described by Akyildiz et al. [1] in order to leverage the properties of acoustics along the thermocline of a body of water. At this time, the authors of this work are unaware of any publication which seeks to take advantage of such properties of underwater environments.

In the discussion of 3D underwater networks Akyildiz et al. [1] proposes to anchor the depth varying network agents on the seabed to improve surreptitious monitoring capabilities or to reduce the hazard of collision with passing vessels. A further alteration to this network architecture would be to allow the agents to autonomously alter their depth of operation while anchored to the seabed. In other words, allow the node itself to take action regarding the depth of operation. This way a network agent may avoid faults caused by physical phenomena, like caustics, air bubbles, and sound speed profile changes along the thermocline [4, 11]. This level of autonomous control is achieved by equipping each agent in the network with a M-automaton. This application of a M-automaton can perform up to three distinct tasks simultaneously. First, a M-automaton operates by determining mean actions, therefore it can maintain a probability vector of the approximate mean correlation between the depth of sensor operation and the probability of link existence. Secondly, the probability vector inherently contains information for the depth or depths corresponding to the highest probability of link existence. Finally, the M-automaton is able to avoid faults without ever requiring change to the routing protocol used by the network since it manipulates the link-state, thus reducing implementation cost. The Maximum Award Stationary Transmission (MaSt) algorithm, solves the best depth location, but does not maintain an accurate probability vector of the approximate mean correlation between depths and link existence, instead it maintains a vector of the confidence in the best depth with respect to time.

The theory of Learning Automata (LA) is discussed in Section 2. A network topology consisting of stationary network elements and LMAs is discussed in Section 3. In Section 4 the MaSt algorithm is evaluated against a non-stationary stochastic environment in a simulation. The simulation conducted consists of two stationary stochastic environments, one unimodally distributed and the other bimodally distributed applied to a Markovian Switching Process (MSE) to create a non-stationary stochastic environment. Finally, concluding remarks and open questions regarding the MaSt algorithm are discussed in Section 5.

2 BACKGROUND ON LEARNING AUTOMATA

A common model in reinforcement learning is one in which there is an environment, E, sometimes called the teacher, and a learner, or LA. The environment is responsible for rewarding or penalizing the LA. The purpose of the LA is to minimize the exposure to penalties by learning to select the action most likely to be rewarded. In the paradigm used in this work a M-automaton selects an action, $\alpha_r \in \{\alpha_0, \alpha_1, \alpha_2\}$, corresponding to movement control: dive, idle, and surface, respectively. The environment responds depending on the probability of a transmission timeout occurring along the unit interval $\beta(n) \in [0, 1]$, as seen in Figure 1 [6, 7]. This type of environment is called a S-model environment. S-model environments are desirable for adaptive control since the unit interval can correspond to a normalized continuous performance value index [7]. In order to control the depth of an underwater LMA a M-automaton would have possible action probabilities correspond to control states and the environmental response acts as a predictor of the next action. In a fixed-rule control system, such a LMA may have two states, dive and surface. However, the adaptive control counterpart would have three control states: dive, surface, and idle in order to allow for a surjective relationship between control states and LA actions, as discussed in subsequent sections.

The application of reinforcement learning in adaptive control of the operating depth of LMAs in an UASN requires that the LA determines rewards by the successful reception of a confirmation message, a type of message that must originate from a Fusion Centre (FC), and is propagated to all agents in the network. A penalty, then, is determined by a timeout failure. Timeout counters are only initiated after a data message is sent by an agent, in this way it does not matter which agent has originated the message. A further discussion of the algorithm which utilizes reinforcement learning in this manner is found in Section 3.

2.1 The Linear Reward-Penalty Scheme

There are many schemes for LA and the scheme being used in this work is a type of Variable-Structure Stochastic Automata (VSSA) known as a Linear Reward-Penalty ($L_{R,P}$) scheme. This scheme is chosen for several reasons, primarily because it is ergodic. It allows the collection of useful metrics that can implicitly map the environment as a probability vector; something that fixed-structure automata schemes could not provide. Since the $L_{R,P}$ scheme is ergodic it allows adaptation to changing environments including seasonal changes, or the natural drift of FCs and LMA caused by currents in the underwater environment [7, 11].

The $L_{R,P}$ scheme, and all VSSA automata, can be represented as the set of $\{\Phi, \alpha, \beta, A, G\}$, where Φ is the set of control states of the automaton, α is the set of actions, β is the set of environmental responses, A is the updating algorithm, and G is the output function such that $G : \Phi \to \alpha$. The function A, concerns how penalty probabilities ought to be updated. A general reinforcement scheme for a $L_{R,P}$ LA operating in a S-model environment such that $g_i(n)$ and $h_i(n)$ are continuous functions for all n, with n representative of iterable time. The function $g_i(n)$ can be viewed as the reward function, while $h_i(n)$ can be viewed as the penalty function. Essentially, if the environment rewards α_i by outputting $\beta(n) \leq 0.5$ the reward function increases the probability of selecting α_i in future action requests, and the function G iterates over remaining actions $\alpha_j \neq i$ reducing the probability of selecting those actions proportionally to the increase awarded to α_i. Generally, G is chosen such that the mapping from the set of control states Φ to the set of action probabilities α is surjective, allowing the functions $g_i(n)$ and $h_i(n)$ to be simplified as follows in Eq. (1) and Eq. (2). When examining the control states of the LMA in Section 3, this concept of surjective mapping will be exploited through the use of an idle control state. A state which could be viewed as redundant in fixed-rule control,

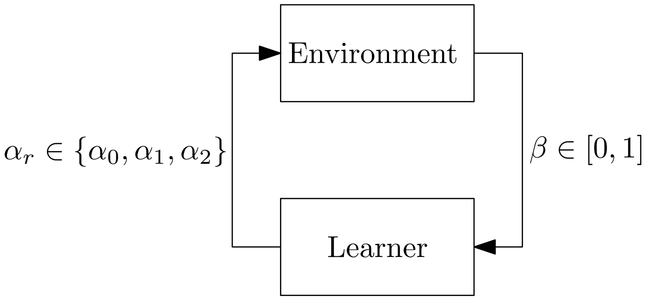


Figure 1: Reinforcement learning model.

but valuable in adaptive-rule automation.

$$p_i(n+1) = p_i(n) - (1 - \beta(n))g_i(p(n)) + \beta(n)h_i(p(n)) \quad \alpha(n) = \alpha_i \tag{1}$$

$$p_i(n+1) = p_i(n) + (1 - \beta(n)) \sum_{j \neq i} g_i(p(n) - \beta(n)) \qquad \alpha(n) \neq \alpha_i \tag{2}$$

The updating algorithm relies on one Initial Condition (IC), the initial action probability vector $p_i(n = 0)$, equivalently p_0, of the LA. In general LA theory, the value of p_0 can be any valid probability vector, however, in a practical application of LA this vector should be equal for all probabilities, since it is initially uncertain which action is the best action in adaptive rule control.

3 A REINFORCEMENT LEARNING STRATEGY FOR DEPTH CONTROL

We design a $L_{R,P}$ automaton with three actions $\alpha_i \in \{\alpha_1, \alpha_2, \alpha_3\}$ such that the automaton can respond to the environment by telling the LMA to choose one of three control states $\phi_i \in \{\phi_0, \phi_1, \phi_2\}$ corresponding to dive, idle, or surface, respectively such that control maps onto actions as: $\phi_0 \longrightarrow \alpha_0$ corresponding the dive command, $\phi_1 \longrightarrow \alpha_1$ corresponding to an idle command, and $\phi_2 \longrightarrow \alpha_2$ corresponding to a surface command. Since the automaton is ergodic, essentially all actions are transient, however, we design the automaton to inevitably tend toward α_2, idle, at the depth corresponding to the maximum reward *max* $(\beta(n))$. This allows the LMA to find the most stable link in a stochastic environment through adaptation and is referred to as the MaSt algorithm; inspired by the procedure of raising and lowering the masts of sailing vessels, seen in Algorithm 1. The MaSt algorithm is ergodic, and therefore there are no conditions for termination of this algorithm. The goal in the design of this algorithm is not to have the LMA absorb an action, but instead to continuously adapt as the operating environment evolves with time. As long as the automaton is sending data the MaSt algorithm is capable of learning, since an internal timeout is reached. The use of a timeout prevents inconsistent links from forcing the algorithm to fail to complete a state translation. Forcing the timeout, allows dependency to be shifted from the reception of a message onto the sending of a message, ensuring that a learning action occurs at all iterations of the algorithm.

Algorithm 1 Maximum Award Stationary Transmission (MaSt)

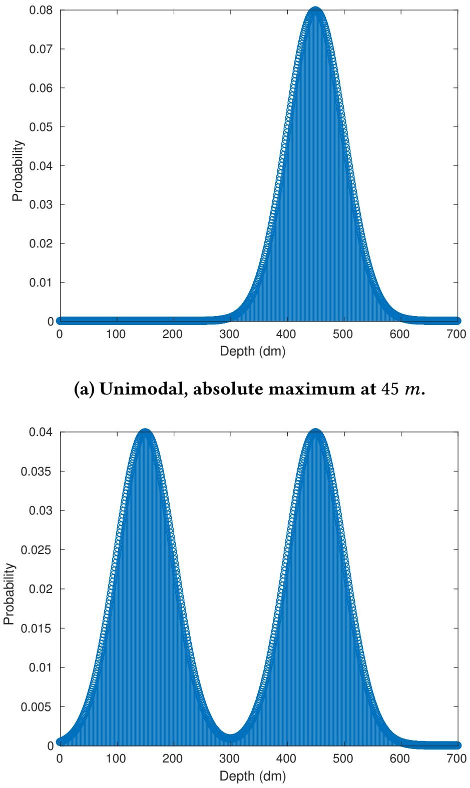
```
INITIALIZE
$lastWait \leftarrow timeoutMax$
SENDING
beginTimer()
if $currentWait = timeoutMax$ then
    $doPenalty(currentAction)$
end if
$lastAction \leftarrow currentAction$
$currentAction \leftarrow pickNextAction()$
move(currentAction)
$currentAction \leftarrow \alpha_2$
RECEIVING
if $currentWait \leq lastWait$ then
    $doReward(currentAction)$
else
    $doPenalty(currentAction)$
end if
$lastAction \leftarrow currentAction$
$currentAction \leftarrow pickNextAction()$
move(currentAction)
```

3.1 The automaton of the MaSt algorithm

The MaSt automaton maintains knowledge of the best depth so far and the last action chosen. Eventually, and with respect to the environmental reward $\beta(n) \in [0, 1]$ the action probability vector $p_i(n)$ will be biased to favour the action α_2 for the depth where $\beta(n) = \max(\beta)$. Considering our assumptions, it is not necessary that the timeout value we choose is actually output on the unit interval. What matters is that it is possible to normalize this output to fit the unit interval. This becomes clear when we consider that the S-model environment must have a valid Probability Density Function (PDF).

4 SIMULATION RESULTS

In the current version of the MaSt algorithmn the LMA moves a constant depth increment after selecting either ϕ_0 or ϕ_2, note that the automaton need not move for ϕ_1 since it is the idle state. Prior to discussing movement intervals it is prudent to note that it appears counter-intuitive to assign a control system state for an idle action; this state is chosen to allow a surjective mapping from control states on to LA actions, a necessary simplification. Constant movement increments provide an expedient solution to the initial problem of finding the depth of maximum performance in stochastic environments where the environment PDF is either an unimodal distribution, or a multimodal distribution as depicted in Figure 2. Constant movement increments also solve the problem of non-stationary stochastic environments. Ultimately, the practical goal is to have variable movement of the LMA to expedite the search process in a manner similar to that of telephone network learning systems [6]. In addition to expedience in solving the unique maximum or local maxima problem, the constant movement interval retains an intuitive error window. For a constant movement interval of k meters, the error window simply becomes ±k of the learned depth. The actual best depth of operation may lay along the interval



(a) Unimodal, absolute maximum at 45 m.

(b) Bimodal, absolute maxima at 15 m and 45 m.

Figure 2: Solved environment PDFs.

of one dive or one surfacing away from the learned depth. Any learning outside of this range would be a failure. The performance metric used for learning in this simulation is the maximum power. The maximum power may not be the most effective performance metric for all applications, and there may be situations where another metric is more desirable. The MaSt algorithm is designed to allow for customization in the selection of the learning metric. The question of which metrics are the most effective for the algorithm remains an open question.

4.1 Learning operating depth in a non-stationary environment

The MaSt algorithm is set against a non-stationary stochastic environment modelled as a Markov switching process with two stationary environments which switch at 10 0000 units of iterable time. In this context the non-stationary environment describes a learning environment, however it is also symbolic of an acoustic environment that experiences a change in acoustic properties after 10 000 time units. This significantly represents common changes in underwater environments and could represent a change in season, or variation in background noise from human interactions like shipping.

The LMA is placed at a random depth in the integer interval [0..70] representative of a range between the surface at 0 m and a seabed at 70 m. The movement function is set to a constant value of 1 m creating a suitably narrow error window of ±1 m from the learned depth. Initially, the automaton is exposed to the environment of Figure 2a where the best depth is situated at 45 m and the decay from the maximum is normally distributed. Figure 3 shows that the MaSt algorithm has learned the best depth to be 44 ± 1 m . Despite the fact that the true best depth exists at 45 m, this result is still correct within the error window, since the automaton is limited to movement intervals of 1 m any learned depth varying by 1 m from the true depth is a success. At n = 10 0000 time units the MSE switches the environment to have two equal maxima, one at 15 m and the other at 45 m, Figure 2b, such an environment could be used to represent a special condition of operation like shipping noise, or perhaps a variation in season.

Due to the expedient movement function, a multimodal distribution with multiple true maxima may not generate an equal confidence for all maxima in a non-infinite ensemble. This results since linear movement may create a latching effect where the automata latches to the local maxima it was initially nearest to. However, since the best depth distribution in this case is that of Figure 2b the automaton may latch to either 15 m or 45 m, which is correct because both maxima are absolute maxima. This latching is a result of the expedient movement function, which can be replaced without altering the decision process of the MaSt algorithm. Consequently, a larger ensemble would also correct this issue. From Figure 4 it is observed that the automata correctly chooses 15 ± 1 m as the optimum communication depth. It also stakes a claim to a relatively high confidence in 39 ± 1 m. This fits the expectation that the automata should produce higher confidence in the neighbourhood of local maxima. This confidence, statistically, is the proportion of ensemble experiments which have converged to that location such that a LMA converging to 44 ± 1 m with a confidence of approximately 0.81 indicates the proportion of ensemble experiments converging to 44 ± 1 m.

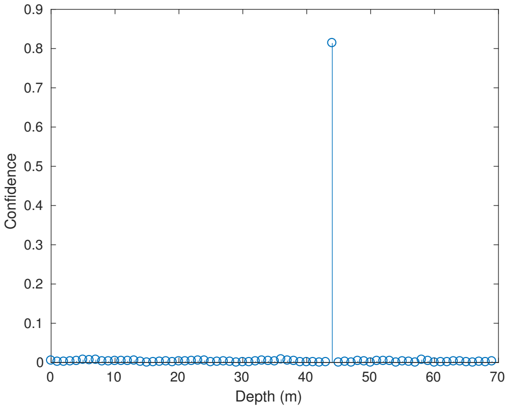


Figure 3: Learned best depth in the unimodal environment.

The limitation posed by constant movement is understood to be solvable through the use of a variable movement function. However, there is one additional question that the non-stationary environment implicitly poses. What happens when the environment

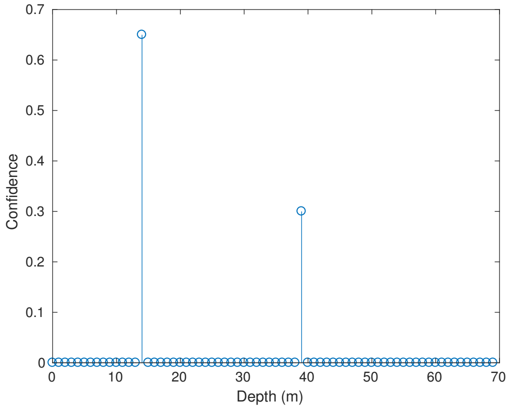


Figure 4: Learned best depth in the bimodal environment.

changes rapidly, how does rapid variation in environmental properties pose a limitation the ergodic reinforcement scheme used? An observation is that an environment which changes PDF in fewer discrete-time units than the expected value of the MaSt automaton, heuristically observed to be 13 discrete-time movements, would result in the LA learning the maximum of the joint PDF of the set learning environments. Further, it is observed that the automata begins to bias towards the idle action very early within the ensemble which is noted by the action probability evolution, Figure 5a. Observe the action evolution of Figure 5b, immediately after the MSE switches environments at $n = 10\,000$ the automaton experiences a transient event before rapidly converging to the new best depth.

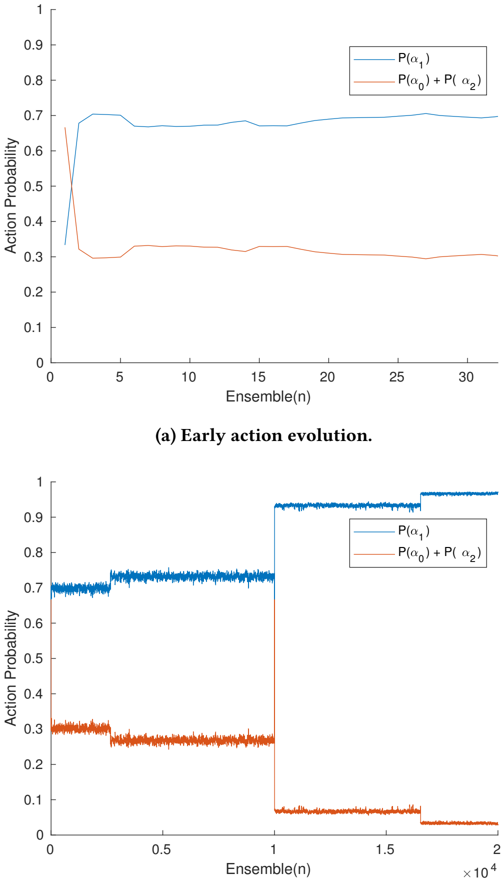
There is a practical benefit of having rapid increase in $P(\alpha_1)$, indicating that the automaton quickly biases towards a predisposition to remain idle, since movement requires power, it would be best to move as little as possible. Additionally, there is a statistical significance of noting that the sum of all action probabilities is one,$\sum_{\alpha_i} P(\alpha_i) = 1$. An indication that the plotted action probabilities of the simulation, Figure 5, are indeed probability vectors, validating that the reinforcement scheme is correct. Further, Figure 5 shows the idle action, α_1, probability asymptotically approaches 1, indicating both the ergodicity, non-recurrence, of the LA and the convergence to a learned position with high probability.

5 CONCLUSION

The MaSt algorithm is able to adapt to the optimal-performance depth of environments where the PDF is unimodal and environments with multimodal PDF. The algorithm utilizes an ergodic LA allowing it to adapt in non-stationary stochastic Markovian environments. The algorithm utilizes a simple constant movement function. Although, constant movement is expedient it is not necessarily ideal. However, this shortcoming does not effect the decision process. The MaSt algorithm is a hopeful step toward adaptive link-state control in UASNs. The source code of the simulation is avaiable at: **github.com/0xSteve/detection_learning**.

REFERENCES

[1] Ian F Akyildiz, Dario Pompili, and Tommaso Melodia. 2004. Challenges for efficient communication in underwater acoustic sensor networks. *ACM Sigbed Review* 1, 2 (2004), 3–8.
[2] Andrew H Bass and Christopher W Clark. 2003. The physical acoustics of underwater sound communication. In *Acoustic communication.* Springer, 15–64.
[3] Josko A Catipovic. 1990. Performance limitations in underwater acoustic telemetry. *IEEE Journal of Oceanic Engineering* 15, 3 (1990), 205–216.
[4] Paul C Etter. 2013. *Underwater acoustic modeling and simulation.* CRC Press.
[5] Kumpati S Narendra and Philip Mars. 1983. The use of learning algorithms in telephone traffic routing - A methodology. *Automatica* 19, 5 (1983), 495–502.
[6] Kumpati S Narendra and Mandayam AL Thathachar. 1980. On the behavior of a learning automaton in a changing environment with application to telephone traffic routing. *IEEE Transactions on Systems, Man, and Cybernetics* 10, 5 (1980), 262–269.
[7] Kumpati S Narendra and Mandayam AL Thathachar. 2012. *Learning automata: an introduction.* Courier Corporation.
[8] Kumpati S Narendra, E Allen Wright, and Lorne G Mason. 1977. Application of learning automata to telephone traffic routing and control. *IEEE Transactions on Systems, Man, and Cybernetics* 7, 11 (1977), 785–792.
[9] Roald Otnes and Svein Haavik. 2013. Duplicate reduction with adaptive backoff for a flooding-based underwater network protocol. In *OCEANS-Bergen, 2013 MTS/IEEE.* IEEE, 1–6.
[10] Jim Partan, Jim Kurose, and Brian Neil Levine. 2007. A survey of practical issues in underwater networks. *ACM SIGMOBILE Mobile Computing and Communications Review* 11, 4 (2007), 23–33.
[11] Robert J Urick. 1967. *Principles of underwater sound* (second ed.). McGraw-Hill.



(a) Early action evolution.

(b) Complete action evolution.

Figure 5: Action evolution in a non-stationary environment.