Universität Bremen

Learning and Exploration in Autonomous Agents

adaptive and active prospect

Bailu Si

December 2006

Learning and Exploration in Autonomous Agents

adaptive and active prospect

Vom Fachbereich für Physik und Elektrotechnik

der Universität Bremen

zur Erlangung des akademischen Grades

Doktor der Naturwissenschaften

genehmigte Dissertation von

Bailu Si

aus P. R. China

1. Gutachter: Prof. Dr. Klaus Pawelzik

2. Gutachter: Prof. Dr. Wolfram Burgard

Eingereicht am: 21.12.2006

Tag des Promotionskolloquiums: 22.03.2007

Abstract

Place learning and exploration in autonomous agents is important for understanding and building intelligent agents. Experimental and computational studies from the areas of psychology, computer science and neuroscience have achieved great successes. This thesis provides a theoretical investigation into three major problems in place learning and exploration, namely, localization, mapping, and action selection.

Two benchmark models are introduced to analyze the basic aspects of place learning and exploration. The checkerboard maze is a stochastic grid-type environment. Exploration performance of an agent is evaluated by the sensory prediction capability of the agent. The evaluation does not require the knowledge of the internal representation of the agent. Furthermore, the checkerboard maze is reduced to the classical multi-armed bandit model in order to analyze the action selection problem in detail. The exploration performance in the multi-armed bandit is quantified by the estimation error of the reward means.

Place learning and exploration is modelled as a Partially Observable Markov Decision Process (POMDP), and is implemented by a Bayesian network with internal dynamics. The map of the environment is represented by the observation probability of the POMDP, and is stored in the weights of a generative model. The belief state is tracked by Bayes filtering. The distribution of the sensory input is predicted by the generative model. Through the minimization of prediction errors by an on-line adaptive multiplicative gradient descent rule, the mapping between locations and sensory inputs is learned.

In the n-armed bandit, the optimal exploration policy in the sense of total mean squared error is proved to be gain-maximization exploration. The sample complexity of the proposed ideal gain-maximization exploration policy can be $O(n)$ as small as the counter-based and the error-based policies, both in the sense of total mean squared error and expected $0/1$ loss. For realistic situations where the reward variances are unknown, a realistic gain-maximization exploration policy is derived using upper confidence limits of the reward variances.

Gain-maximization is a general principle unifying a wide range of exploration strategies including counter-based and error-based policies. By generalizing the total mean squared error, the counter-based and error-based exploration policies are shown to result from the gain-maximization principle with respect to different variants of the general objective measure.

Formulating the exploration in reward maximization as the learning of the differences between the reward means, we derive gain-maximization selection policies both in the ideal case and for realistic situations. Through a simple linear trade-off, gain-based reward-maximization policies achieve smaller regret on fixed data sets, as compared to classical strategies like interval estimation methods, ϵ-greedy strategy, and upper confidence bound policy.

The action selection in the full place learning problem is implemented by a network maximizing the agent's intrinsic rewards. Action values are learned in a similar way as Q-learning. Based on the results of local gain adaptation and multi-armed bandit,two gain functions are defined as the agent's curiosity. Moreover, an estimation of a global exploration performance measure is found to model competence motivation. Active exploration directed by the proposed intrinsic reward functions not only outperforms random exploration, but also produces curiosity behavior which is observed in natural agents.

Acknowledgments

The results presented in this thesis were conducted during my work at the Institute for Theoretical Neurophysics in the University of Bremen. It is my honer and pleasure to work with my colleagues, who are so friendly and supportive to help me all the time.

First of all, I would thank Klaus Pawelzik to open the exciting research field for me. He is always thought provoking, insightful, and guiding me to explore scientific questions. His generous support, kind encouragement, and warm-hearted friendship through all these years make my pursuit of science possible to be realized. I am also grateful to Prof. Schwegler, Prof. Bornholdt and Prof. Jaeger for their insightful comments and discussions. I especially thank Prof. Burgard for reviewing my thesis.

I feel very thankful to Michael Herrmann for his constructive cooperation and discussions during these years. He is always ready to provide full strength help. I own many thanks to Agnes Janssen, who creates an amazingly friendly working atmosphere for us. I feel gratitude to Udo Ernst and David Rotermund, who maintain the computation facilities available anytime. I would thank my colleagues Matthias Bethge, Christian Eurich, Axel Etzold, Nadja Schinkel, Erich Schulzke, Onno Boehler, Juan Ochoa, Stefan Braunewell, Maria Davidich, Joerg Reichardt, Roland Rothenstein, Hedinn Steingrimsson, Andreas Thiel, Frank Emmert-Streib, Jens Otterpohl, Ronald Bormann, and Klaus Franke.

Last but not least, I want to thank my whole family for their constant love, support and encouragement. I will never thank them enough.

Contents

1 Introduction **1**
1.1 Mind and Intelligence 2
1.2 Curiosity, Learning and Exploration 2
1.3 Place Learning 3
1.4 Questions and Approaches 5
1.5 Contributions and Overview 10

2 Checkerboard Maze **14**
2.1 Definition of Environment and Agent 14
2.2 Measuring Prediction Capability 16

3 Local Gain Adaptation in Constrained Gradient Descent **19**
3.1 Additive and Multiplicative Gradient Descent 20
3.2 Learning Rate Adaptation 21
3.3 Adaptive Multiplicative Gradient Descent for Linear Networks 24
3.3.1 Adaptive Gradient Descent in Probability Space 25
3.3.2 Derivation of Local Gain Adaptation 26

4 Place Learning in Partially Observable Environments **30**
4.1 Overview of Probabilistic Place Learning Methods 31
4.2 Modelling Exploration by POMDP 32
4.3 Learning through Anticipation and Adaptation 34
4.3.1 Prediction 35
4.3.2 Learning Observation Probability 35
4.3.3 Belief Update 38
4.4 Simulations 38
4.4.1 Performance of Additive Gradient Descents 42
4.4.2 Performance of Multiplicative Gradient Descents 43

5 Exploration as Global Learning **48**
5.1 Exploration Strategies 49
5.2 Exploration in Multi-armed Bandits 50
5.3 Exploration Policies in Multi-armed Bandits 52

5.4 Optimal Exploration by Gain Maximization 53
5.4.1 Performance Measure and Gain Definition 54
5.4.2 Asymptotic Behavior of the Ideal Gain-maximization Exploration Policy 54
5.4.3 Optimality of the Gain-maximization Exploration 55
5.5 Active Exploration by Gain Maximization 58
5.5.1 Confidence Interval for the Variance 58
5.5.2 Realistic Gain-maximization Exploration Policy 59
5.6 Simulations 61
5.6.1 Power Law Distributed Parameters 62
5.6.2 Uniformly Distributed Parameters 65

6 A Unified View on Exploration 68
6.1 Generalized Performance Measure 68
6.2 Optimality of the Unified Exploration Policy 70
6.3 Probability Measure of Exploration Performance 71
6.3.1 Expected 0/1 Loss 71
6.3.2 Upper Bound on the Sample Complexity 72
6.3.3 Competitive Analysis on the Upper Bounds 75

7 Exploration in Reward Maximization 78
7.1 Exploration for Exploitation 78
7.2 Gain-based Exploration in Selection Task 80
7.2.1 Exploration Performance Measure 80
7.2.2 Optimal Policies 81
7.2.3 Performance Comparison 83
7.2.4 Realistic Gain-maximization Selection Policy 84
7.2.5 Numerical Simulations 85
7.3 Sample Complexity of Exploration in Selection Task 87
7.3.1 Upper Bound on the Sample Complexity 88
7.3.2 Competitive Analysis of the Upper Bounds 90
7.4 Exploration Exploitation Tradeoff 93
7.4.1 Gain-based Reward Maximization Policy 93
7.4.2 Empirical Evaluation of Reward Maximization Policies . 95

8 Active Exploration in Partially Observable Environments 111
8.1 Intrinsic Rewards from Self Evaluation 112
8.1.1 Local Error Reduction 113
8.1.2 Local Learning Rate 114
8.1.3 Global Performance Estimation 114
8.2 Action Selection by Maximizing Intrinsic Rewards 118
8.2.1 Action Selection 118
8.2.2 Value Update 119

8.3 Simulation Results 120
8.3.1 Active Exploration with Additive Gradient Descents . . 120
8.3.2 Active Exploration with Multiplicative Gradient Descents 123

9 Conclusions and Outlooks 136
9.1 Conclusions 136
9.2 Outlooks 138

A Calculation of Hessian-vector Product 140

B Mean Squared Error of Exploration Policies 143
B.1 Performance Ratio of Exploration Policies for Optimal Learning 143
B.1.1 Performance Ratio with Power Law Distributed Reward Standard Deviation 145
B.1.2 Performance Ratio with Uniformly Distributed Reward Standard Deviation 147

C Sample Complexity of Exploration Policies 148
C.1 Proof of Lemma 6.3.1 148
C.2 Union Bound 149
C.3 Chebyshev Inequality 149
C.4 Optimal Section of the Mean Distance for Optimal Learning . . 150
C.5 Proof of Theorem 6.3.4 151

D Sample Complexity of Selection Policies 153
D.1 Performance Ratio of Selection Polices 153
D.1.1 Performance Ratio with Power Law Distributed Reward Standard Deviation 154
D.1.2 Performance Ratio with Uniformly Distributed Reward Standard Deviation 155
D.2 Optimal Section of the Mean Distance for Exploitation 155
D.3 Proof of Theorem 7.3.3 156

Bibliography 159

List of Figures

1.1 A POMDP is represented by a Bayesian network 8

2.1 An agent with short-range sensors in a checkerboard maze environment 15

4.1 Examples of transition probability 34
4.2 Occupancy probabilities of the locations in two of the environments used in simulations 38
4.3 Average prediction error $E_{SQ}(t)$ of the additive rule and the counting rule under random exploration with different motion noise levels 39
4.4 Maps constructed for environment a at the end of random exploration using additive gradient descents 40
4.5 Maps learned for environment b at the end of random exploration using additive gradient descents 41
4.6 Prediction error $E_{KL}(t)$ of the multiplicative rule and the adaptive multiplicative rule under random exploration with different motion noise levels 44
4.7 Maps constructed for environment a at the end of random exploration using multiplicative gradient descents 45
4.8 Maps constructed for environment b at the end of random exploration using multiplicative gradient descents 46

5.1 Decay rates of the exploration for policies shown in Table 5.1 . . 60
5.2 Sensitivity of the realistic gain-maximization exploration policy and the t-interval exploration policy with respect to the parameter α when $\mu(a)$ and $\sigma(a)$ are power-law distributed 62
5.3 Comparison of exploration strategies with power-law distributed $\mu(a)$ and $\sigma(a)$ 63
5.4 The dependency of the realistic gain-maximization exploration policy and the t-interval exploration policy on the parameter α with uniformly distributed $\mu(a)$ and $\sigma(a)$ 65

5.5 Comparison of exploration strategies with uniformly distributed $\mu(a)$ and $\sigma(a)$ 66

7.1 Performance comparison of gain-based, error-based and counter-based selection policies 86
7.2 Reward maximization and exploration performance of the ideal gain-based reward-maximization policy with uniformly distributed reward standard deviations 97
7.3 Reward maximization and exploration performance of the realistic gain-based reward-maximization policy with uniformly distributed reward standard deviations 98
7.4 Reward maximization and exploration performance of the z-interval policy with uniformly distributed reward standard deviations 99
7.5 Reward maximization and exploration performance of the t-interval policy with uniformly distributed reward standard deviations 100
7.6 Reward maximization and exploration performance of the ϵ-greedy policy with uniformly distributed reward standard deviations 101
7.7 Performance comparison of reward-maximization policies with uniformly distributed reward standard deviations 102
7.8 Reward maximization and exploration performance of the ideal gain-based reward-maximization policy with power-law distributed reward standard deviations 104
7.9 Reward maximization and exploration performance of the realistic gain-based reward-maximization policy with power-law distributed reward standard deviations 105
7.10 Reward maximization and exploration performance of the z-interval policy with power-law distributed reward standard deviations 106
7.11 Reward maximization and exploration performance of the t-interval policy with power-law distributed reward standard deviations 107
7.12 Reward maximization and exploration performance of the ϵ-greedy policy with power-law distributed reward standard deviations 108
7.13 Performance comparison of reward-maximization policies with power-law distributed reward standard deviations 109

8.1 Active exploration using the counting rule compared with random exploration under different motion noise levels 121

8.2 Occupancy maps constructed from prediction records $q_t(s|z)$ for two environments under different exploration schemes using the counting rule 122
8.3 Exploration of environment a using the counting rule when there is no motion noise, $\sigma_M = 0$ 124
8.4 Exploration of environment a using the counting rule under large motion noises ($\sigma_M = 0.2$) 125
8.5 Exploration of environment b using the counting rule with accurate odometry ($\sigma_M = 0$) 126
8.6 Exploration of environment b using the counting rule with large motion noises ($\sigma_M = 0.2$) 127
8.7 Active explorations using the adaptive multiplicative rule compared with random exploration under different locomotion noise levels 129
8.8 Maps constructed from prediction records $q_t(s|z)$ for two environments under different exploration schemes using the adaptive multiplicative rule 130
8.9 Exploration of environment a using the adaptive multiplicative rule with accurate odometry ($\sigma_M = 0$) 131
8.10 Exploration of environment a using the adaptive multiplicative rule under high motion uncertainty ($\sigma_M = 0.2$) 132
8.11 Exploration of environment b using the adaptive multiplicative rule with accurate odometry ($\sigma_M = 0$) 133
8.12 Exploration of environment b using the adaptive multiplicative rule under high motion uncertainty ($\sigma_M = 0.2$) 134

List of Tables

3.1 Comparison of gain adaptation methods 24

5.1 Comparison of error-based exploration strategies and gain-maximization exploration strategies 60

6.1 The exponent β of the general performance measure 69

Chapter 1

Introduction

The question of what is intelligence has been intriguing and puzzling since ancient time. There are different interpretations and definitions for intelligence. In terms of behavior, intelligence is generally considered as "the ability to learn or understand or to deal with new or trying situations" (from Merriam-Webster dictionary).

In order to understand and create intelligence, researchers invented a new paradigm, called Artificial Intelligence (AI) in the 1950s. The aim of Artificial Intelligence is to create machines that are able to solve problems and achieve goals as well as humans. During the past fifty years, AI has achieved great success, in areas like pattern recognition, computer vision, and robotics.

However, Artificial Intelligence is far away from human intelligence and animal intelligence. Artificial Intelligence research was influenced by behaviorists for a long period in the 20th century. Behaviorists claim that cognition can be completely specified by high level functions, therefore can be realized by multiple implementations [Put75]. One famous proposition to distinguish intelligent systems is the Turing test [Tur88]. If the agent could successfully pretend to be human to a knowledgeable observer, then it is considered to be intelligent. Led by behaviorists, researchers are more concerned with whether an agent is able to perform some tasks, like face recognition or chess playing, instead of with the agent's internal structures and intrinsic mind. The behaviorism paradigm often results in passive machines without mind-related consciousness, like curiosity, motivation, and emotion.

1.1 Mind and Intelligence

The deficit of Artificial Intelligence research in modeling mind has been recognized. Marvin Minsky said that "the question is not whether intelligent machines can have any emotions, but whether machines can be intelligent without emotions" [Min86]. Results from neuroscience, psychology and physiology show that emotion, curiosity, motivation and other mind-related feelings play important roles in learning and decision making [Pan98]. Damasio showed that frontal-lobe disorders that impair the ability to feel yield impaired ability to make decisions [Dam94]. Voss found that curiosity is consistently linked to creativity [VK83]. These experimental results support the idea that mind is necessary for intelligence .

Motivated by these findings, researchers expand traditional Artificial Intelligence architecture to affective computing [Pic97]. At the first level, efforts have been put to create natural interfaces between human and agent. Being able to recognize, understand the affective states of human beings, and to synthesize its own affective expressions, the agent interacts with human beings in a human-like way [CZZZ05; Kot05; ONPS05]. Furthermore, the agent may have its own internal feelings, like emotion, curiosity and motivation. For example, in [Sch91; Her01], curiosity is modelled to guide the agent in data or action selection. Singh et al. combine intrinsic motivation with extrinsic rewards of the traditional reinforcement learning framework, and demonstrate that the agent improves the efficiency in developing hierarchical skills [SBC04]. Ahn et al. take a similar idea, assuming the existence of internal reward from cognition and emotion to include intrinsic motivation in learning and decision making [AP05].

The prominent feature of these systems is the agents' abilities to understand or generate human emotions, or even to produce their own emotions in learning and decision making. The agent is not a passive machine, but being autonomous. From this viewpoint, if it is exaggerating to say "intelligence comes to nothing when the emotions hold away" [Gol95], it is still valid to conclude that it is boring intelligence when emotions are held away.

1.2 Curiosity, Learning and Exploration

Among all the feelings in the mind, curiosity is especially important. Curiosity is an intrinsically motivated desire for specific information [Low94]. There are several possible accounts of the underlying cause of curiosity, such as the novelty of stimuli, boredom, competence motivation, and the need for sense

making [Low94]. Curiosity is considered as the driving force for exploration, and is widely observed to result in exploration [Fow65; Les69].

Exploration is the process of knowledge acquisition. Exploration is considered as a form of intelligent behavior. "Exploratory behavior is a major determinant for the development of intelligence" [VK83]. During exploration, information is gained and curiosity is satisfied. For example, in a maze learning task, rats show "perceptual curiosity" for novel or complex stimuli, and result in exploratory behaviors [BS69].

Between curiosity and exploration, what is happening? The agent learns during exploration. A place is novel if it does not have a representation in the brain [ON78]. The novelty reflects the incongruity between the agent's prediction and observation. The agent integrates new information into its model, and tries to reduce the violation. This learning process fulfills the function of exploration, resulting in a world model interpreting new information. Learning is a sense-making process that organizes the agent's internal representation, and improves its prediction for future events. This memory-prediction framework is the major mechanism for intelligence [HB04].

1.3 Place Learning

Among all types of exploratory behaviors, space exploration attracts much attention. The question of how place is learned has been studied by philosophers, psychologists, neuroscientists, and computer scientists [ON78]. In the past decades, place learning becomes an important interdisciplinary research area, where scientists with different backgrounds meet each other.

There are at least two motivations for studying place learning. First, the concept of space is primary both for natural and artificial agents, and plays a role in all behaviors of agents. Therefore, place learning is a prerequisite for agents to be autonomous. Second, understanding the place learning in natural agents provides insights on understanding the mechanism of learning and memory, and provides possible guides for building artificial agents. Spatial behaviors such like navigation and homing suggest the existence of a localization and mapping system in the brain. The findings of place cells, head direction cells, and grid cells of rodent suggest that Hippocampus and related cortex are the neural substrates for place learning and representation [OD71; TMR90; FMW+04; HFM+05]. Place cells are neurons in the hippocampus that fire strongly whenever an animal is in a specific location of its environment [OD71]. The activity of had direction cells is found to be tuned by the animal's heading direction [TMR90]. In recent years, neurons are found

in the entorhinal cortex that fire strongly when an animal is in specific locations in an environment. These neurons are called grid cells, because each grid cell has multiple firing fields, scattered in a highly regular hexagonal pattern [FMW+04; HFM+05]. The computational mechanisms of these experimental findings are remained to be revealed. However, they point out possible directions to develop place learning algorithms.

The three major issues in place learning are localization, mapping and action selection. Localization deals with the question of "where I am", i.e. to determine the current location of the agent according to its observations and actions in the past. Mapping is to build a model for future predictions, describing the dependency between the sensory information and the locations of the environment, so that the question of "where I have been" could be answered. Action selection is the problem of "where I should go next". Local action selections generate global exploration patterns.

Plenty of computational models for place learning have been proposed both in neuroscience and computer science [MBG+96; MBJ+06; Thr02; TBF05; DWB06a; DWB06b].

Some works give biologically plausible explanations for the coding in place learning [BA96; ZGMS98; OB05; GG07]. The location of the rat can be reconstructed from the activity of its place cells by probabilistic methods [ZGMS98]. Some models are motivated by physiology facts, and give hypothetical computation of place learning [BW94; WTR94; TS95; BO96; WLC+05]. Continuous attractor models are successful in explaining the localized activity of places cells and head direction cells [MBJ+06]. It is an extension of the discrete attractor network, like Hopfield network [Hop82], to maintain a stable activity over time, and to propagate by asymmetrical connections. A continuous attractor model has the ability to represent multiple maps in one network, which fits the observations from natural agents. However, attractor models require the knowledge of the true location, therefore they are restricted to explain path integration and mapping systems in the natural agents.

Another stream of research aims at practical solutions of mapping and localization systems for robots [TBF98; HPG99; DNDW+01; CN01]. As variants of Bayesian filters, these methods are implemented with different representations, including grid-based, features-based, and topological representations [LMS+03].

Grid-based methods represent the environment as a set of locally connected discrete cells [Mor88; BFHS96; FBT99a]. Each cell has a random variable indicating the probability of being occupied by an object. The location of the agent is tracked by a Bayesian filter. Since the full Bayesian filter is implemented, grid-based methods are able to represent general forms of probability

distributions. Grid-based methods are robust against noisy sensors and can deal with unstructured environments. Combined with the Rao-Blackwellized particle filter, grid-based methods can efficiently solve the localization and mapping problem [EP03; HFBT03; GSB07]. Grid map is a straightforward representation of the environment. However, it has disadvantages in sensory prediction and map learning. First, grid maps do not support the prediction of sensory observations directly. The observations of range sensors are normally predicted by ray-tracing operations. Second, sensory measurements are normally integrated into the map only for the most likely location. Neglecting too much uncertainty in the location distribution may produce inconsistent maps. This problem is most evident in loop closing, when the maximum likelihood location estimation may be a wrong prediction due to the accumulated odometry noises. This problem is alleviated in FastSLAM methods by maintaining multiple maps at the cost of additional computation burdens.

Feature-based methods represent the environment as a set of detected landmarks [SSC90; DNDW+01]. Assuming all probability distributions to be Gaussian, the Bayesian filter is approximated by an extended Kalman filter (EKF). The EKF gives accurate estimations of the location of the agent and the landmarks, and the convergence is guaranteed by estimation-theoretic analysis [DNDW+01]. There are three disadvantages of the feature-based methods. First, the matching of the sensory observations to the landmark estimations (called data association) is quite rigid, and this makes feature-based methods strongly dependent on object recognition. Second, the noise variances in sensor model and motion model need to be known in advance. Third, the computation complexity of the feature-based methods is quadratic to the number of landmarks.

Topological methods learn the structure of the environment, instead of a metric map [SK97; CN01]. The compact graph representation is suitable for high level planning, and need to be combined with metric model for low level predictions and controls.

1.4 Questions and Approaches

Current research in probabilistic place learning methods raises two questions for further improvements. First, online adaptation of the motion model and sensor model is necessary to accommodate different noise levels in different environments, instead of hand tuning of parameters. Second, data association needs to include the uncertainty of the location distribution. The data association, or more generally the object recognition and information fusion problem, is essentially probabilistic. It is quite often that multiple hypotheses cause the

same observation. Therefore, it is less robust to integrate observations only into the maximum likely hypothesis.

Taking into account of the advantages as well as disadvantages of previous methods, this thesis addresses localization, mapping and action selection problem under a coherent framework. Especially, the following questions are underscored.

- *What are the principles underlying place learning?*

  It is shown by Hebb that behavior can be accounted by synapse adaptation [Heb49]. Since then, revealing the principles underlying behaviors becomes a major scientific interest, echoing the metaphysics initiated by ancient philosophers.

  We focus on two related principles. First, to account for the agent's ability of learning and memory, the error minimization principle is applied to derive learning rules. Through on-line adaptation according to the resulting learning rules, the agent minimizes its sensory prediction error for a given location. Moreover the adaptation should require as less data as possible, and also need to obey possible constraints.

  Second, motivated by the fact that exploration is caused by curiosity, we model the curiosity of the agent by its gain in place learning. The gain maximization principle is used to organize the exploration behavior of the agent. Defined as the reduction of local prediction error, gain is the performance improvement. Reinforcing by the gain in place learning, the agent optimizes place learning in a global sense. Local error minimization is incompetent to produce exploration behavior, because reinforcing small errors results in sticking to the best explored location.

  The error minimization principle and the gain maximization principle guide the agent to achieve learning and exploration respectively. The agent thereby acquires two characteristics of intelligent behavior: adaptive and active.

  As to active exploration, some recent works successfully apply information gain-maximization principle to minimize the uncertainty in position estimation [BMW+03], uncertainty in map estimation [SB03; SGB05], or to maximize global exploration performance [SPH04]. The information gain-maximization is essentially the mutual information-maximization principle [Lin88; Lin92]. Mutual information is the reduction of uncertainty after a new piece of data is observed [CT91]. Therefore, mutual information-maximization is a gain-maximization principle in information-theoretic terms. Recent analysis shows that if the entropy being minimized is not equivalent to the objective function in learning,

the entropy measure might not be consistent in evaluating prediction performance [TK05].

- *What computational frameworks are suitable for place learning?*

  Neuroscientists accumulated many evidences and hypotheses about the mechanism of place learning [ON78; MBG+96; MBJ+06]. Computer scientists developed many practical solutions for real world applications [DWB06a; DWB06b]. It is worthwhile to reconsider the fundamental question of what computational frameworks are suitable for place learning.

  We address place learning and exploration in a memory-prediction framework. More specifically, exploration is modeled as a Partially Observable Markov Decision Process (POMDP). The POMDP is represented by a Bayesian network with internal dynamics [OS98; NJ02]. As depicted in Fig. 1.1, the network has a set of states, denoted by i. Analogous to the activity of the place cells, the activity of the states is the agent's belief of its location, called belief state. Between the states, there are several sets of lateral connections, which are the estimations of the true transition probabilities. These connections are sparse and localized to model the agent's motion. From the states, there are two sets of connections to the sensory nodes and action nodes respectively. The connections to the sensory nodes store an estimation of the observation probability of the POMDP. This set of connections is an implicit map of the environment, since the observation probability is the causal relationship between locations and sensory inputs. The connections from internal states to action nodes store an estimation of the action values of action-state pairs.

  Place learning and exploration are achieved in two iterative phases, namely prediction and adaptation. First, action values are predicted through the connections from the states to action nodes. The action with the maximal value is selected in order to maximize the discounted reward obtained after choosing this action. Accordingly, the belief state is propagated by the estimated transition probability. A prediction of the new sensory observations after the selected action is generated through the connections from the internal states to the sensory nodes. The prediction error is evaluated by comparing the sensory prediction with the true observation. Minimizing sensory prediction errors, the estimated observation probability and transition probabilities are adapted by gradient descent. The belief state is updated according to Bayes' rule to incorporate sensory information using the new estimation of the observation probability. The expected error reduction in learning observation probability is defined as an intrinsic reward signal to measure the curiosity of the agent. The action values are also updated in a gradient descent

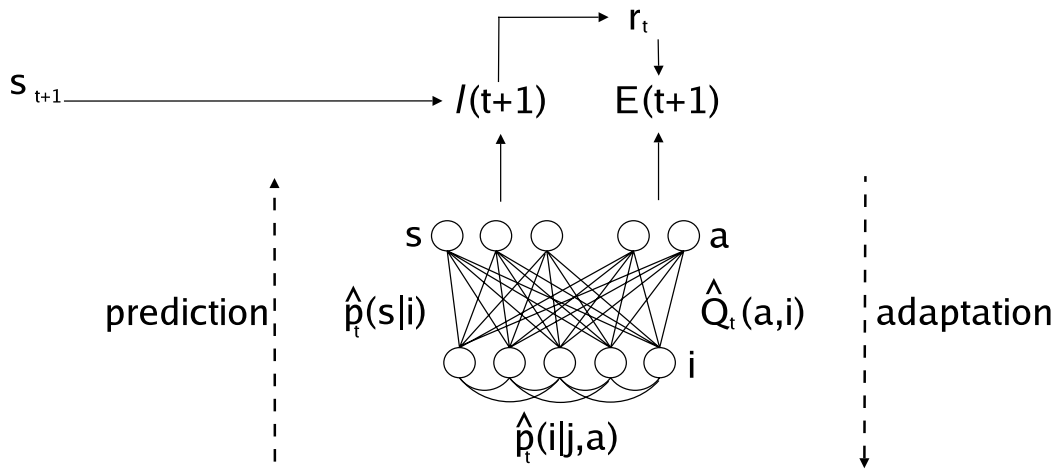


Figure 1.1: A POMDP is represented by a Bayesian network. The activity of the states i is the agent's belief of its location, called belief state. The belief state is propagated by the lateral connections $\hat{p}_t(i|j, a)$, which are the estimations of the transition probabilities in the POMDP. The estimated observation probability $\hat{p}_t(s|i)$ is the connections from state i to sensory nodes s, and is the agent's map representation. The estimated action values $\hat{Q}_t(a, i)$ are stored in the connections from i to action nodes a. The network generates sensory predictions and action value predictions. The observation probability, transition probabilities are learned by minimizing sensory prediction errors $\ell(t+1)$. Action values are learned by minimizing action value prediction errors $E(t+1)$. Action selection is directed by intrinsic reward signals r_t derived from sensory prediction error $\ell(t+1)$.

manner to optimize action value predictions.

One important feature of the proposed network representation is that all the models in the network are adaptable. The error measure for sensory prediction is the basic objective function of the network. The reward signal is derived from the sensory prediction errors, therefore the action selection model cooperates with the sensor prediction model to minimize the overall sensory prediction errors of the network.

The proposed framework does not maintain an explicit map, instead a generative model of the sensory distribution. The generative model represents the environment as a function, which is essentially the observation probability of the POMDP. This functional representation differs from grid-based and feature-based maps, which are treated as random variables. Therefore, in these methods, the task is to estimate the joint distribution over the states and the maps. In our method, the task is to estimate the distribution over the states only. But in order to do so, the

observation probability is learned incrementally.

The disadvantage of the proposed implicit functional map representation is that an explicit map needs to be generated additionally if exploitation of the map is of interest.

However, the advantages of the proposed framework are big enough to assure its value for practical applications. First, implicit multihypothesis data association is achieved by the modulation effect of the the belief state. The uncertainty in the belief state is introduced into the map estimation, and in turn increases the robustness in localization and the the consistency in map learning. The second advantage of the proposed framework is that it facilitates sensory predictions. This is a natural result of the generative model. Prediction is an important ability of learning systems [BSG03]. As long as the agent is able to make good sensory predictions for every location in the environment, the agent can be considered to have learned the environment.

- *What is the complexity of the computation in place learning?* Complexity quantifies the nature of computation. Computational complexity is considered more relevant for cognition than function is [Eli02], since different representations of the same behavior may require qualitatively different computation complexities. The learning and recognition abilities of natural agents probably result from small computation complexity and appropriate representations. For realistic applications, complexity is especially of practical importance. We aim at computation which requires linear complexity.

- *What benchmark models are helpful to evaluate the computational frameworks for place learning?*

  As a highly interdisciplinary research area, place learning is a meeting point of methods with different principles, frameworks and representations. It is beneficial to define benchmarks for evaluations. Simple models which contain the most essential elements in place learning are especially valuable to give clear comparisons between different methods. An ideal agent, who knows the true information of the benchmark models, need to be defined. Since the performance of the ideal agent provides a bound for all other agents who require learning.

  We propose two benchmarks. The first one is a checkerboard maze to evaluate the performance of place learning and exploration. The essential elements include partial observability, stochastic observations, uncertainty in motions, and action selections.

  The second benchmark model is the multi-armed bandit [Rob52]. The multi-armed bandit is a classical model to analyze exploration-

exploitation trade-off for reward maximization [BF89; Git89]. In this thesis, we formulate the multi-armed bandit as a benchmark for action selection strategies involved in exploration. The essential elements include uncertainty in observation and real-valued observations.

The multi-armed bandit is a simpler formulation of the checkerboard maze. It has one state with multiple actions. Therefore, the multi-armed bandit benchmark can be considered as one local model in the checkerboard maze.

1.5 Contributions and Overview

Here the main contributions of the thesis are summarized, and the organization of the thesis is outlined.

In chapter 2, a checkerboard maze is defined as a benchmark model for place learning and exploration. It keeps the most important elements of place learning and exploration, and provides objective measures to compare between different agents and with the ideal agent. The comparison is independent of the internal representations of the agents.

In chapter 3, an adaptive multiplicative gradient descent rule is derived for linear networks with probability constraints. It is an extension of the stochastic meta descent method [Sch99; VSS06], which is an additive gradient descent method with local learning adaptation. The proposed adaptive multiplicative rule has a wide application in learning probabilistic networks, like the generative model used here. First, the adaptive multiplicative rule locally adapts its learning rates, resulting fast convergence speed. It is a second order gradient descent method, with linear computation complexity $O(N)$, where N is the number of weights in the network. Second, it adapts probability distributions, obeying the non-negativity and normalization constraints automatically. Third, it is an on-line learning method, and is capable to deal with slowly changing environments.

In chapter 4, a Bayesian network with internal dynamics is proposed for place learning in partially observable environments. The learning of the observation probability is accomplished by the adaptive multiplicative rule derived in chapter 3, together with several other variants of gradient descent rules. The performance of these learning rules are compared under the random exploration strategy.

Chapter 5-7 study exploration strategies in the context of the multi-armed bandit, where the reward distributions of each action need to be learned. In chapter 5, exploration is formulated as optimal learning of reward distribu-

tions of the bandit. Under the ideal assumption that the reward variances are known, an optimal exploration strategy is derived from the gain maximization principle, minimizing the total mean squared error of the estimators. Analytical result shows that the proposed ideal gain maximization exploration policy achieves minimal total mean squared error, outperforming the classical counter based and error based exploration strategies. The realistic application of the gain maximization principle is approached by finding an upper confidence bound of the gain function. The resulting realistic gain maximization exploration policy achieves the ideal gain maximization exploration policy asymptotically, and is shown to outperform the counter based and error based policies with a fairly small number of data samples. The two gain maximization exploration policies proposed in this chapter are novel, and have a wide range of applications in active learning and reinforcement learning, where counter based exploration and error based exploration are widely used.

Chapter 6 shows that the gain maximization principle has more general forms than it appears at first sight. First, it is demonstrated that under a generalization of the total mean squared error introduced in chapter 5, a general class of gain maximization exploration policies is derived following the same manner as in chapter 5. The general class of policies naturally includes the ideal gain maximization exploration policy, the counter based exploration policy and the error based exploration policy compared in chapter 5. The general gain maximization exploration strategy is a gain maximization exploration strategy minimizing different variants of the general objective measure. Thereby, a unification of three types of exploration policies is achieved, implying that gain maximization results in different exploration behaviors in the aim of optimizing respective objective functions of interest. The unification is novel, and sheds light on further theoretical approaches to exploration. Second, an upper bound on the sample complexity of the general gain maximization exploration strategy is established. Previous analysis of the upper bound on the sample complexity is mainly restricted to distributions on a bounded range[EDMM02; ACBFS03; MT04; EDMM06]. Here, our analysis applies to unbounded distributions. The unbounded distribution assumption provides a broader range to model the gain as intrinsic reward signal of the agent. Third, the competitive analysis of the upper bounds proves that the ideal gain maximization exploration policy has the potential to achieve a sample complexity $O(n)$ times as small as those of the counter based and error based exploration policies, evaluated in terms of $0/1$ loss. This ratio is the same as the ratio obtained under the measure of the total mean squared error, implying that the presented sample complexity analysis in the sense of probability measure is consistent with the analysis in the sense of mean squared error. Therefore, gain maximization principle is consistently superior both under the measure of the total mean squared error and the measure of the expected $0/1$ loss.

In chapter 7, the classical objective of the multi-armed bandit problem, i.e. reward maximization, is addressed. On-line trade-off between exploration and exploitation is required in reward maximization. First, the pure exploration aspect is analyzed, which is equivalent to selecting the best arm. Here it is shown that exploration in the selection task is equivalent to learning the differences of the reward means involved in the pairwise comparisons. Thereby exploration in the selection task is formulated as a normal learning task. This formulation gives rise to a novel objective measure, minimizing which in a similar way as shown in chapter 5-6 leads to gain-based selection policies both under the ideal case and realistic cases. The proposed selection policies are useful for the problems of selecting or ranking the alternatives like in mixture of experts and discrete-event simulations [JJNH91; PKM96; And98; SJY03]. Second, the upper bound on the sample complexity of the selection policies arrives naturally through a similar analysis of the unified general selection strategy. Third, the proposed gain maximization selection policies help to achieve efficient exploration-exploitation trade-off. Numerical simulations on finite bandit data sets demonstrate that, through the classical linear weighting trade-off, the proposed gain-based reward maximization policies outperform the traditional reward maximization policies like confident-interval methods, ϵ-greedy policy, and Upper Confidence Bound policy (UCB) [ACBF02]. The UCB policy is proved to have logarithmic regret, which is the best achievable regret bound for a wide range of distributions [LR85]. The UCB policy does not have any free parameters, and is derived under worst cases of observation histories. This is the reason why the UCB policy appears worse in practical situations, where the data set is fixed, and where the policies are allowed to tune parameters. Never the less, the gain based reward maximization policies have a potential to achieve optimal regret bound if an adaptive trade-off factor is further developed.

In chapter 8, three intrinsic reward functions are defined to direct active exploration. A gradient descent method similar as Q-learning [WD92] is used to learn action values. First, two gain functions are defined for place learning based on the results from local gain adaptation and multi-armed bandit respectively. Both of the gain functions have theoretical bases as developed in chapter3 or in chapter 5. They are directly related to the loss function of the network, therefore are coherent with the error minimization objective of the network. Second, a global exploration performance is approximated and is intended to be minimized during exploration. Third, learning rules presented in chapter 4 combined with active exploration display performance improvement as compared with the results obtained under random exploration. More importantly, curiosity behavior is produced by the agent automatically. The agent incrementally expands its exploration frontier, and organizes its action selection systematically.

Finally, conclusions and future works are outlined in chapter 9. Both philosophical and technical observations are provided for a broad research in neuroscience, Artificial Intelligence, psychophysics, and behavioral research.

Chapter 2

Checkerboard Maze

In order to compare different place learning methods and exploration strategies on a common basis, it is necessary to define a benchmark problem containing the most essential elements. Previously [Mur99] introduces a simple benchmark problem for exploration and place learning. There also exists various real world data sets and implementations of place learning methods based on different representations (see [DWB06a] for a summary). Here, we define checkerboard maze, a generic benchmark problem for a quantitative comparison of different methods. It includes range sensors and additive motion noises to account for realistic sensors and motors. One important feature of the benchmark is that the evaluation is independent of the representations of the environment.

2.1 Definition of Environment and Agent

The agent explores in a 2-D stochastic environment with the ability to select its actions and observe sensory information through its sensors. The task of the agent is to explore actively, localize itself and learn a map of the environment as fast as possible. We assume that the agent is allowed to go to any cell freely, neglecting the problem of object avoidance.

The environment has a x-y coordinate system. Each location of the environment has a certain probability to be occupied by objects. This is because some objects in the environments are changing, e.g. chairs, door, and people walking around. For simplicity, the occupancy probability is defined by a discrete grid (i.e. a piecewise constant function), although continuous and smooth functions are also possible. The environment is divided into a grid with unitary squares as cells, indexed by z. Each cell of the grid has a number indicating its probability of being occupied by objects. Figure 2.1(a) shows one instance of

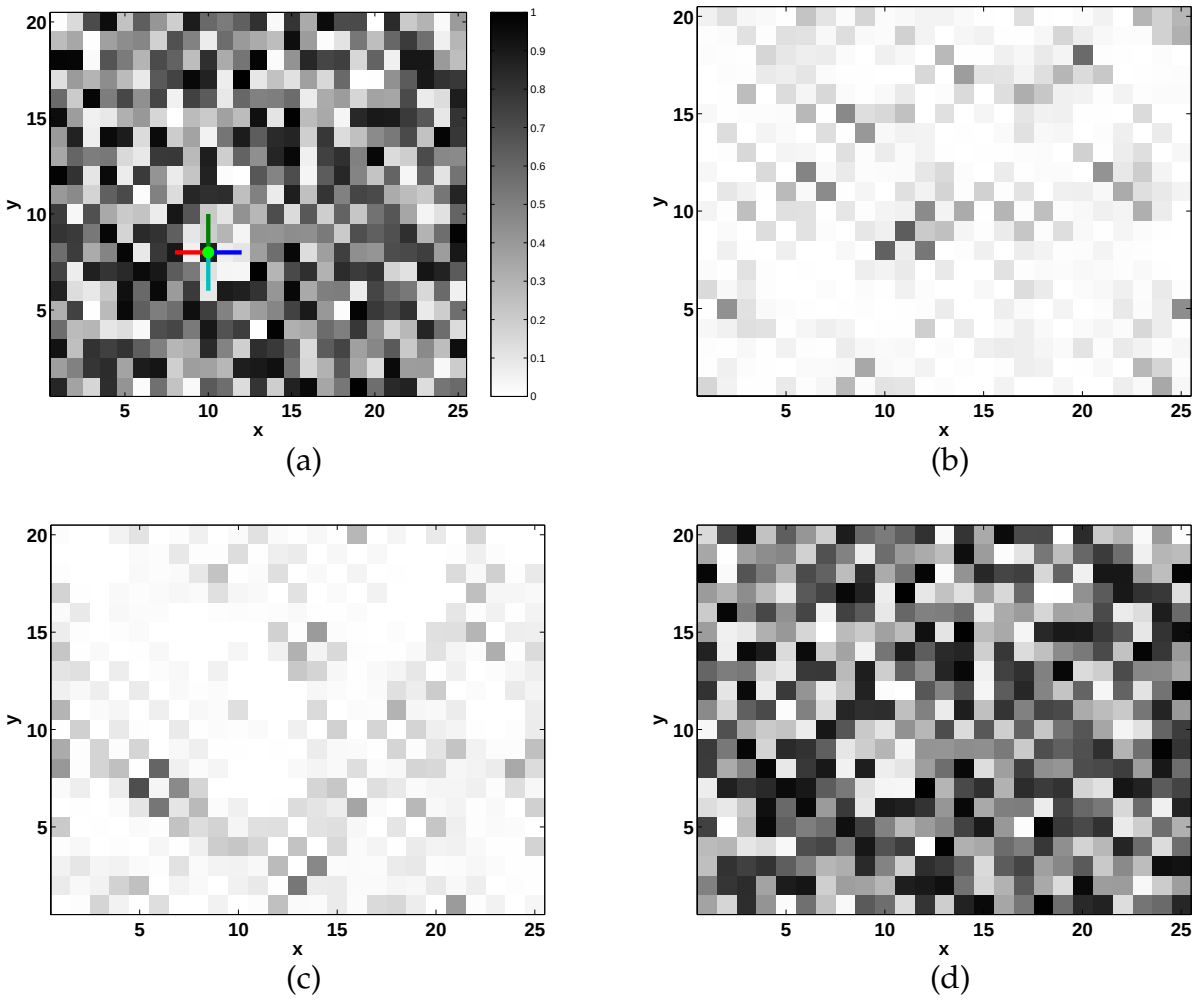


Figure 2.1: An agent with short-range sensors in a checkerboard maze environment. (a) One example of the checkerboard maze. The occupancy probabilities are randomly sampled from the uniform distribution on range $[0, 1]$. The main concern of the agent (depicted as a disk) is place learning and exploration instead of object avoidance, therefore the agent is allowed to move freely to any cell; (b) The probability of observing no objects. The agent's current coordinate is $(10, 8)$. All the neighboring cells have low probability of being occupied by objects. Therefore, the probability of not observing any objects in location $(10, 8)$ is high. The four sensors reach their maximal ranges; (c) The probability of observing four objects. In location $(5, 7)$, the probability of observing four objects is high, because its four neighboring cells are occupied by objects with high probabilities; (d) The marginal probability of observing objects by the right sensor. It is a left-shifted map of the environment. Aligning the four marginal probabilities reconstructs the maze.

the environments, with dark color being high probability of objects, and light color being low probability.

The agent has four short-range sensors, pointing to east, north, west, and south respectively. This is because in realistic robotic systems, range sensors like infrared sensor, sonar, laser range-finder are widely used. The maximal detection range of each sensor is 2. Therefore, rather than sensing the cell where the agent is currently in, the agent observes whether or not there are some objects in the four neighboring cells. Due to the stochastic nature of the environment, at each location z the agent observes a random vector containing four binary variables, resulting $S = 2^4 = 16$ different observations $s \in \{1, \cdots, S\}$. The observation s is the input from the environment to the agent, and is dependent on the sensors the agent uses.

The mapping from location to sensory observation is fully described by a probabilistic function $p(s|z)$. Figure 2.1(b)-(c) show two of the 16 observation probabilities $p(s|z)$. Since the 16 observation probabilities are the joint probability of the four sensory bits, by marginalization the probabilities for individual sensors to detect objects can be calculated. From the four marginal probability distributions, a map of the maze can be reconstructed (Figure 2.1(d)).

For sensors, the environment has periodic boundaries, so that the observations on the boundaries are still defined. However, for motions, the environment has reflexive (mirroring) boundaries. New positions beyond the boundaries will be mirrored back with respect to the boundaries.

The agent has five actions, namely left, right, forward, backward, and stay. Since the agent has omnidirectional sensors, the rotation action is not considered. The agent's true position at time t is denoted by its real-valued coordinates (x_t, y_t), which evolve according to the action it takes

$$\begin{aligned} x_{t+1} &= x_t + m_x(t) + \omega(t) \\ y_{t+1} &= y_t + m_y(t) + \psi(t) \end{aligned}, \tag{2.1}$$

where $m_x(t), m_y(t) \in \{-1, 0, 1\}$ are the offsets of action a_t. For example, $m_x(t) = 0, m_y(t) = 1$ when a_t is forward. $\omega(t)$ and $\psi(t)$ are additive motion noises to model the odometry errors encountered in realistic applications. $\omega(t)$ and $\psi(t)$ are independently sampled from a uniform distribution over the range$[-\sigma_M/2, \sigma_M/2]$.

2.2 Measuring Prediction Capability

How to evaluate the performance of exploration and place learning? From the viewpoint of an ideal observer, if an agent is able to predict its future sensory

inputs when navigating in its environment, then the agent is considered to know this environment.

Starting from this assumption, we quantify the exploration performance of an agent by its prediction capability. The prediction capability is the overall ability of the agent in localization and mapping.

First, the sensory predictions of the agent are recorded by an ideal observer into a *prediction record* defined by

$$q_t(s|z) = \begin{cases} \frac{1}{S}, & z \text{ has never been visited} \\ \hat{p}_{\tau(z)}(s), & z] \text{ was visited last time at } \tau(z) \end{cases}, \tag{2.2}$$

where z is the index of the cells in the maze, and $\hat{p}_{\tau(z)}(s)$ is the agent's prediction at time $\tau(z)$.

$q_t(s|z)$ is the up-to-date prediction recorded for the agent. If the agent has been to place z, its latest prediction at z is recorded into $q_t(s|z)$. For unvisited locations, the uniform prior $1/S$ is used. $q_t(s|z)$ is the sensory distribution constructed by the ideal observer, since the agent does not know its true location z.

There are two important features of the prediction record $q_t(s|z)$. First, it only depends on the agent's predictions, and not on the way the predictions were made. This gives the possibility to evaluate place learning methods implemented from different prospects. Second, $q_t(s|z)$ is defined for each location z of the environment. Therefore, it is the agent's knowledge about the whole environment. This gives the possibility to distinguish lazy agents, who are lack of exploration.

The uncertainty in $p(s|z)$ results only from uncertain observations. However, the uncertainty in the prediction record $q_t(s|z)$ come from both localization and observation. Due to inaccurate motions, the belief state of the agent has certain uncertainty to account for odometry errors. Other than uncertainty in localization, the uncertainty in stochastic observations is contained in the estimated observation probability $\hat{p}_t(s|i)$ (i is the internal representation of location z). Since the agent predicts the future sensory inputs based on its belief and the estimated observation probability, the uncertainty in prediction record $q_t(s|z)$ is larger than the uncertainty in $p(s|z)$.

Second, the agent's prediction capability is quantified by the distance from the prediction record $q_t(s|z)$ to the true observation probability. It is possible to compare the estimated observation probability $\hat{p}_t(s|i)$ with the true observation probability $p(s|z)$. However, this comparison requires the correspondence between the true location z and it internal representation i. This correspondence is normally dependent on how the agent represents the environment.

To be independent of internal representations, we define objective measures based on prediction record $q_t(s|z)$ to quantify the prediction capability of the agent.

When the agent uses a quadratic loss function (e.g. Eq.4.4), an objective measure is the square distance between the true observation probability $p(s|z)$ and the prediction record $q_t(s|z)$

$$E_{SQ}(t) = \sum_{s,z} \Big(p(s|z) - q_t(s|z) \Big)^2. \tag{2.3}$$

Another possible objective measure is the Kullback-Leibler divergence [CT91] between $p(s|z)$ and $q_t(s|z)$, suitable for place learning methods with Kullback-Leibler divergence as a loss function (e.g. Eq.4.8)

$$E_{KL}(t) = \sum_{s,z} p(s|z) \log \frac{p(s|z)}{q_t(s|z)}. \tag{2.4}$$

Chapter 3

Local Gain Adaptation in Constrained Gradient Descent

Gradient descent is a function optimization method which approaches a local minimum of the function by taking steps proportional to the negative gradient of the function at the current point in the domain [Bal95]. Gradient descent is the basis for many algorithms, such as Backpropagation [RHW86], Q-learning [WD92], least squares, and Kalman filter [Hay86]. In these algorithms, gradient descent is used to adapt the parameters of a model to best approximate the function defined by real data. Gradient descent can be applied both in batch learning and on-line learning.

In batch learning, the gradient is the sum of the gradients caused by each individual training sample. Therefore, batch gradient descent requires the storage and process of the whole data set before any parameters can be changed. In on-line learning, the training samples come one by one. The parameters of the model are updated by the gradient of the loss function caused by the current training sample. The gradient in on-line learning is a a stochastic approximation of the true gradient used in batch learning. However, on-line gradient descent is more suitable for stream-based information processing systems, like autonomous agents.

This chapter is organized as follows. First, we give an overview on main types of gradient decent methods and learning rate adaptation methods in section 3.1 and 3.2 respectively. Then in section 3.3 we derive an adaptive multiplicative gradient descent rule, which is a learning rate adaptation method for gradient descent subject to non-negativity and normalization constraint.

3.1 Additive and Multiplicative Gradient Descent

Consider a learning problem where a parameterized model $\hat{\mathbf{y}} = \mathbf{f}(\mathbf{x}; \boldsymbol{w})$ is trained to approximate the relationship between input-output data pairs $(\mathbf{x}_k, \mathbf{y}_k), k = 1, \cdots, T$. $\boldsymbol{w} = [w(1), w(2), \cdots, w(N)]$ is a N-dimensional weight vector. The learning system searches in the weight space in order to minimize a loss function $\ell(\boldsymbol{w})$. The loss function $\ell(\boldsymbol{w})$ is a differentiable distance measure between the predictions $\hat{\mathbf{y}}_k$ of the system and the target values $\mathbf{y}_k$. The local search is guided by the gradient of the loss function ℓ with respect to the current weight

$$\boldsymbol{w}_{t+1} = \boldsymbol{w}_t - \eta \frac{\partial \ell(\boldsymbol{w})}{\partial \boldsymbol{w}}\Big|_{\boldsymbol{w}=\boldsymbol{w}_t}, \tag{3.1}$$

where η is a small positive number, called learning rate. The gradient is denoted by

$$\mathbf{d}\boldsymbol{w}_t = \frac{\partial \ell(\boldsymbol{w})}{\partial \boldsymbol{w}}\Big|_{\boldsymbol{w}=\boldsymbol{w}_t}. \tag{3.2}$$

Adapted by the additive gradient descent (Eq.3.1), the weights $\boldsymbol{w}$ can be positive or negative. Often, the weights are constrained to be non-negative. Under these situations, the weights can be adapted by exponentiated gradient [KW97]

$$\boldsymbol{w}_{t+1} = \boldsymbol{w}_t \cdot \exp\left(-\eta \mathbf{d}\boldsymbol{w}_t\right), \tag{3.3}$$

where $\cdot$ denotes component-wise multiplication.

Another choice is to do gradient descent in the log domain of the weights

$$\log \boldsymbol{w}_{t+1} = \log \boldsymbol{w}_t - \eta \frac{\partial \ell(\boldsymbol{w})}{\partial \log \boldsymbol{w}}\Big|_{\boldsymbol{w}=\boldsymbol{w}_t}, \tag{3.4}$$

or equivalently

$$\boldsymbol{w}_{t+1} = \boldsymbol{w}_t \cdot \exp\left(-\eta \boldsymbol{w}_t \cdot \mathbf{d}\boldsymbol{w}_t\right). \tag{3.5}$$

For some constrained problems (e.g. linear model with non-negative input and Kullback-Leibler divergence as the loss function), the gradient $\mathbf{d}\boldsymbol{w}_t$ are always negative. Therefore, the additive gradient descent rule is still valid to use. A multiplicative gradient descent rule can be obtained by multiplying the gradient by the weight itself, converting additive steps into geometric steps [ERP07]

$$\begin{aligned} \boldsymbol{w}_{t+1} &= \boldsymbol{w}_t - \eta \boldsymbol{w}_t \cdot \mathbf{d}\boldsymbol{w}_t \\ &= \boldsymbol{w}_t \cdot \left(1 - \eta \mathbf{d}\boldsymbol{w}_t\right). \end{aligned} \tag{3.6}$$

In Eq.3.6, the gradient is premultiplied by a positive definite diagonal matrix formed by $\boldsymbol{w}_t$, therefore convergence is guaranteed. Eq.3.6 can be considered as a linearization of the exponentiated gradient (Eq.3.3) at point 0.

3.2 Learning Rate Adaptation

The gradient descent methods take advantage of the geometrical information of the loss function, therefore they are generally more efficient than stochastic search methods like genetic algorithm. However, if a fixed global learning rate η is used, it should be sufficiently small for gradient descent methods to converge. Moreover, the convergence can be slower if the weights display some geometrical structures like valleys or plateaus. Therefore, different weights should have adaptive and local learning rates instead of a fixed global one.

In the aim of fast convergence, methods are devised to adapt learning rates, or to choose better gradient descent directions. According to how they use second order gradient information, learning rate adaptation methods can be classified into two classes: model based or experience based methods. Learning rate adaptation is also called gain adaptation, since the learning rate is effectively related to the gain in loss reduction. The change in loss function is approximated quadratically by Taylor expansion

$$\begin{aligned} \Delta\ell(\boldsymbol{w}) &= \ell(\boldsymbol{w}+\Delta\boldsymbol{w}) - \ell(\boldsymbol{w}) \\ &\approx \left(\frac{\partial\ell(\boldsymbol{w})}{\partial\boldsymbol{w}}\right)^{\mathsf{T}} \Delta\boldsymbol{w} + \frac{1}{2}\Delta\boldsymbol{w}^{\mathsf{T}}\mathbf{H}\Delta\boldsymbol{w}, \end{aligned} \tag{3.7}$$

where T is the transpose operator, and $\mathbf{H}$ is the Hessian matrix defined by

$$\mathbf{H} = \frac{\partial^2\ell(\boldsymbol{w})}{\partial\boldsymbol{w}\boldsymbol{w}^{\mathsf{T}}}. \tag{3.8}$$

The maximal loss reduction is found by minimizing Eq.3.7 with respect to $\Delta\boldsymbol{w}$. The optimal change of weights in the sense of quadratic error is given by

$$\Delta\boldsymbol{w} = -\mathbf{H}^{-1}\frac{\partial\ell(\boldsymbol{w})}{\partial\boldsymbol{w}}. \tag{3.9}$$

This give the Newton's method [Bis95]

$$\boldsymbol{w}_{t+1} = \boldsymbol{w}_t - \mathbf{H}_t^{-1}\mathbf{d}\boldsymbol{w}_t. \tag{3.10}$$

As a second order geometric model, the Hessian matrix reveals the correlation between input dimensions. Therefore, Newton's method is a model-based gain adaptation method. Normalizing the gradient by the Hessian matrix, Newton's method transforms the approximated elliptical loss surface into a circular surface. It reaches the minimum within one step if the loss function is

quadratic. The computation complexity of Newton's method is $O(N^3)$, since it requires inverting the $N \times N$ Hessian matrix.

Avoiding matrix inversion, quasi-Newton methods such as the BFGS method approximate the inverse Hessian matrix, and reduce the complexity to $O(N^2)$ [Bis95]. Additionally, they use line search minimization along the approximated Newton step to avoid big jumps.

In the conjugate gradient descent method, the computation complexity is further reduced to $O(N)$ [Bis95]. The conjugate gradient descent uses second order information implicitly. It finds a set of conjugate search directions which are orthogonal under the transformation of the Hessian matrix. Along the conjugate directions, the gradients have no component along the previous search direction. Therefore, the new search direction will not oscillate in previous directions. The conjugate gradient method requires a set of line minimizations along the search directions. Both quasi-Newton methods and the conjugate gradient method are not suitable for on-line learning, since stochastic approximation is too noisy for line minimizations to be robust.

In Newton's method, the gradient is premultiplied by the inverse of a local curvature matrix under quadratic approximation. Better convergence results may be achieved by choosing better gradient directions. In natural gradient descent method, the weight space is viewed as a Riemannian space, and the conventional gradient is premultiplied by the inverse of the matrix tenser of the Riemannian space [Ama98]. If the loss function is defined as the negative log likelihood of the data, then the matrix tenser is the Fisher information matrix. The natural gradient includes curvature information of the underlying Riemannian matrix structure, and it is the deepest descent direction in the coordinate system defined by the weights $\boldsymbol{w}$. If the coordinate system is Euclidean and orthogonal, then the natural gradient reduces to the conventional gradient. The natural gradient can be applied both for bath learning and on-line learning problems. The natural gradient on-line learning is proved to be Fisher efficient [Ama98]. Combined with adaptive approximation techniques, the computation complexity of the natural gradient can be $O(N^2)$ [PAF00].

Computing the inverse curvature matrix is to build the second order geometric model for clever gradient descent. However, this not only requires at least $O(N^2)$ computation, but also may be strongly dependent on local quadratic approximation or the input data distribution.

Instead of maintaining an explicit model, experience-based gain adaptation methods directly accumulate and use the second order gradient information for each weight [Sut92b; Sut92a; Sch99; VSS06]. This leads to meta gradient descent methods, where learning rates are also adapted by gradient descent. Additional to a gradient descent in the base level, the meta level gradient de-

scent works in the logarithm domain to guarantee the non-negativity of the learning rates

$$\begin{aligned} \boldsymbol{w}_{t+1} &= \boldsymbol{w}_t - \boldsymbol{\eta}_t \cdot \mathbf{d}\boldsymbol{w}_t && (3.11) \\ \boldsymbol{\eta}_t &= \boldsymbol{\eta}_{t-1} \cdot \exp\left(-\mu \frac{\partial \ell(\boldsymbol{w})}{\partial \log \boldsymbol{\eta}}\Big|_{\boldsymbol{w}=\boldsymbol{w}_t}\right) \\ &= \boldsymbol{\eta}_{t-1} \cdot \exp\left(-\eta \mathbf{d}\boldsymbol{w}_t \cdot \boldsymbol{v}_t\right), && (3.12) \end{aligned}$$

where $\boldsymbol{\eta}_t = \left[\eta_t(1), \cdots, \eta_t(i), \cdots, \eta_t(N)\right]^T$ is an adaptive positive learning rate vector, with component $\eta_t(i)$ corresponding to the weight $w_t(i)$. μ is a positive meta learning rate. $\boldsymbol{v}_t$ is a memory of how each weight changes with respect to the change of its learning rate

$$\boldsymbol{v}_t \equiv \frac{\partial \boldsymbol{w}_t}{\partial \log \boldsymbol{\eta}} = \left[\frac{\partial w_t(1)}{\partial \log \eta(1)}, \cdots, \frac{\partial w_t(i)}{\partial \log \eta(i)}, \cdots, \frac{\partial w_t(N)}{\partial \log \eta(N)}\right]^T. \tag{3.13}$$

The partial derivative with respect to $\log \boldsymbol{\eta}$ without a time index represents the derivative with respect to an infinitesimal change in $\log \boldsymbol{\eta}$ at all time steps [Sut92b; Sch99]. The computation of $\partial \boldsymbol{w}_t / \partial \log \boldsymbol{\eta}$ depends not only on the base level learning rule, which governs how learning rates affect the weights, but also on how long the past history will be considered to evaluate the correlation between learning rates and weights. For example, $\boldsymbol{v}_t$ can be implemented as an exponential average of all past changes of $\eta_{t-j}(i)$ on weights $w_t(i)$,$j = 0, 1, 2, \cdots$. [Sch99; VSS06]

$$\boldsymbol{v}_{t+1} \equiv (1-\beta) \sum_{j=0}^{\infty} \beta^j \frac{\partial \boldsymbol{w}_{t+1}}{\partial \log \boldsymbol{\eta}_{t-j}},$$

where the discount parameter $\beta \in [0, 1)$ controls the time window of the history, and the base level gradient descent rule (Eq.3.11) determines the update rule for $\boldsymbol{v}_{t+1}$ via $\boldsymbol{w}_{t+1}$. Since the trace vector v_{t+1} can be computed within $O(N)$ computation, meta gradient descent methods have $O(N)$ computation complexity, and are suitable for on-line learning.

Previous meta gradient descent methods are developed for additive gradient (Eq.3.11)[Sut92b; Sut92a; Sch99; VSS06]. In the next section, we present a meta descent rule for multiplicative gradient (Eq.3.6) with additional non-negativity and normalization constraints. The proposed gradient descent method is called the adaptive multiplicative gradient descent rule, and is suitable for learning probability distributions, which subject to non-negativity and normalization constraints. Table 3.1 gives a comparison between gain adaptation methods.

Method name	Type	Computation	Application
Newton's method	model	$O(N^3)$	unconstrained, batch
Quasi-Newton methods	model	$O(N^2)$	unconstrained, batch
Conjugate gradient descent	model	$O(N)$	unconstrained, batch
Natural gradient descent	model	$O(N^2)$	unconstrained, batch, on-line
Meta gradient descent	experience	$O(N)$	unconstrained, on-line
Adaptive multiplicative gradient	experience	$O(N)$	probability constrained, on-line

Table 3.1: Comparison of gain adaptation methods.

3.3 Adaptive Multiplicative Gradient Descent for Linear Networks

How to adapt learning rates for gradient descent in constrained weight space is the problem we are going to solve here. We consider a linear network with weights defined in probability space (ref Fig.1.1)

$$\hat{p}_{t+1}(s) = \sum_s \hat{p}_t(s|i)\hat{p}_{t+1}(i), \quad 1 \le s \le S, 1 \le i \le N \tag{3.14}$$

or equivalently

$$\hat{\mathbf{y}}_{t+1} = \boldsymbol{W}_t \mathbf{x}_{t+1}, \tag{3.15}$$

where the input vector $\mathbf{x}_{t+1}$ and weight matrix $\boldsymbol{W}_t$ subject to probability constraints

$$x_{t+1}(i) = \hat{p}_{t+1}(i) \ge 0, \qquad \sum_i x_{t+1}(i) = \sum_i \hat{p}_{t+1}(i) = 1 \tag{3.16}$$

$$w_t(s,i) = \hat{p}_t(s|i) \ge 0, \qquad \sum_s w_t(s,i) = \sum_s \hat{p}_t(s|i) = 1. \tag{3.17}$$

The weight matrix of the linear network can be decomposed into S weight vectors indexed by $k = 1, 2, \cdots, S$

$$\boldsymbol{W}_t = \left[\boldsymbol{w}_t(1)^T, \cdots, \boldsymbol{w}_t(k)^T, \cdots, \boldsymbol{w}_t(S)^T\right]^T, \tag{3.18}$$

$$\boldsymbol{w}_t(k) = [w_t(k,1), \cdots, w_t(k,i), \cdots, w_t(k,N)]^T. \tag{3.19}$$

Therefore, the linear network is a set of parallel networks, which have a common input $\mathbf{x}_{t+1}$ but their own weight vectors $\boldsymbol{w}_t(k)$.

We take the loss function as the Kullback-Leibler divergence between the observation and prediction

$$\ell(\boldsymbol{W}_t) = D_{KL}(\boldsymbol{y}_{t+1} \| \hat{\boldsymbol{y}}_{t+1}) = \sum_k y_{t+1}(k) \log \frac{y_{t+1}(k)}{\hat{y}_{t+1}(k)}. \tag{3.20}$$

The gradient is denoted by

$$\begin{aligned} \mathbf{d}\boldsymbol{w}_t(k) &= \frac{\partial \ell(\boldsymbol{W}_t)}{\partial \boldsymbol{w}_t(k)} = \left[\frac{\partial \ell(\boldsymbol{W}_t)}{\partial w_t(k,1)}, \cdots, \frac{\partial \ell(\boldsymbol{W}_t)}{\partial w_t(k,i)}, \cdots, \frac{\partial \ell(\boldsymbol{W}_t)}{\partial w_t(k,N)} \right]^T \\ &= \left[-\frac{y_{t+1}(k)}{\hat{y}_{t+1}(k)} x_{t+1}(1), \cdots, -\frac{y_{t+1}(k)}{\hat{y}_{t+1}(k)} x_{t+1}(i), \cdots, -\frac{y_{t+1}(k)}{\hat{y}_{t+1}(k)} x_{t+1}(N) \right]^T \end{aligned} \tag{3.21}$$

3.3.1 Adaptive Gradient Descent in Probability Space

Since the gradient of the Kullback-Leibler divergence with respect to the weight vector is always positive (Eq.3.21), the multiplicative gradient descent (Eq.3.6) is used to adapt $\boldsymbol{w}_t(k)$

$$\boldsymbol{w}_{t+1}(k) = \mathbf{c}_t \cdot \boldsymbol{w}_t(k) \cdot \Big(1 - \boldsymbol{\eta}_t \cdot \mathbf{d}\boldsymbol{w}_t(k)\Big), \tag{3.22}$$

where $\boldsymbol{\eta}_t = \Big[\eta_t(1), \cdots, \eta_t(i), \cdots, \eta_t(N)\Big]^T$ is the adaptive learning rate vector for $\boldsymbol{w}_t(k)$'s, and $\mathbf{c}_t$ is the normalization vector, whose i-th component is

$$c_t(i) = \frac{1}{1 - \eta_t(i) \sum_k w_t(k,i) dw_t(k,i)}. \tag{3.23}$$

The common learning rate vector $\boldsymbol{\eta}_t$ is adapted in a similar way as Eq.3.12

$$\boldsymbol{\eta}_t = \boldsymbol{\eta}_{t-1} \cdot \exp\Big(-\mu\big(\sum_k \mathbf{d}\boldsymbol{w}_t(k) \cdot \boldsymbol{v}_t(k) + \epsilon \boldsymbol{\eta}_{t-1} \cdot \mathbf{x}_{t+1} \big) \Big) \tag{3.24}$$

Here μ and ϵ are two positive learning rates. Compared with the original meta gradient descent rule (Eq.3.12), there are two differences. First, the gradient of the loss function with respect to the logarithm of the learning rate is a sum of the contributions from all sub-networks. This results from the chain rule of the partial derivative. Second, we add a self-decaying term $\boldsymbol{\eta}_{t-1} \cdot \mathbf{x}_{t+1}$ weighted by the input vector $\mathbf{x}_{t+1}$. This helps the learning rate decrease when the average gradient vanishes (i.e. when the base gradient descent steps converge), and thus reduces the variance of the base gradient descent. This is especially helpful in our case, because the magnitude of the stochastic gradient $\mathbf{d}\boldsymbol{w}_t(k)$

is generally not zero even after converge (due to binary observation vectors). Smaller learning rates stabilize the base gradient descent steps.

As before, $\boldsymbol{v}_t(k)$ is a memory of how much the weights $\boldsymbol{w}_t(k)$ change with respect to the logarithm of the learning rates, and is defined as an exponential average of all past changes of $\eta_{t-j}(i)$ on weights $\boldsymbol{w}_t(k)$

$$\boldsymbol{v}_{t+1}(k) \equiv (1-\beta)\sum_{j=0}^{\infty}\beta^j\frac{\partial \boldsymbol{w}_{t+1}(k)}{\partial \log \boldsymbol{\eta}_{t-j}} \tag{3.25}$$

$$= \mathbf{c}_t\Big(\beta\boldsymbol{v}_t(k) - \mathbf{u}_{t+1}(k) + \boldsymbol{w}_{t+1}(k)\sum_{j=1}^{S}\mathbf{u}_{t+1}(j)\Big), \tag{3.26}$$

with $\mathbf{u}_{t+1}(k)$ defined by

$$\mathbf{u}_{t+1}(k) = \boldsymbol{\eta}_t \cdot \Big[\Big(\boldsymbol{w}_t(k) + \beta\boldsymbol{v}_t(k)\Big)\cdot \mathbf{d}\boldsymbol{w}_t(k) + \beta\boldsymbol{w}_t(k)\cdot \mathbf{H}_t(k)\boldsymbol{v}_t(k)\Big]. \tag{3.27}$$

Here $\mathbf{H}_t(k)$ is the $N \times N$ Hessian matrix given by

$$\mathbf{H}_t(k) \equiv \frac{\partial^2 \ell(\boldsymbol{W})}{\partial \boldsymbol{w}(k)\boldsymbol{w}(k)^T}\Big|_{\boldsymbol{W}=\boldsymbol{W}_t}.$$

The new trace $\boldsymbol{v}_{t+1}(k)$ is updated recursively. $\boldsymbol{v}_{t+1}(k)$ contains three parts. The first part is the old trace $\boldsymbol{v}_t(k)$ discounted by β. The second part $\mathbf{u}_{t+1}(k)$ is the change from unnormalized gradient step $\boldsymbol{\eta}_t \cdot \boldsymbol{w}_t(k) \cdot \mathbf{d}\boldsymbol{w}_t(k)$. The third part $\boldsymbol{w}_{t+1}(k)\sum_{j=1}^{S}\mathbf{u}_{t+1}(j)$ comes from the normalization factor $\mathbf{c}_t$. $\mathbf{u}_{t+1}(k)$ contains first and second-order gradient terms, rendering the adaptive multiplicative gradient descent as a second order gradient method.

In summary, Equation (3.22)-(3.27) give the adaptive multiplicative gradient descent for linear network defined in probability space. The Hessian-vector product $\mathbf{H}_t(k)\boldsymbol{v}_t(k)$ can be calculated in $O(N)$ computation [Bis95], and all other operations are component-wise, therefore the computation complexity of the adaptive multiplicative gradient descent is $O(N)$. We are going to give the scheme of the derivation in the next section.

3.3.2 Derivation of Local Gain Adaptation

To keep the non-negative property of the local learning rates $\eta(i)$, they are adapted by exponentiated gradient descent

$$\log \eta_t(i) = \log \eta_{t-1}(i) - \mu\frac{\partial \ell(\boldsymbol{W}_t)}{\partial \log \eta(i)}$$

$$= \log \eta_{t-1}(i) - \mu\sum_k \frac{\partial \ell(\boldsymbol{W}_t)}{\partial w_t(k,i)}\frac{\partial w_t(k,i)}{\partial \log \eta(i)},$$

which can be written in the form of component-wise vector multiplication

$$\boldsymbol{\eta}_t = \boldsymbol{\eta}_{t-1} \cdot \exp\Big(-\mu \sum_k \mathbf{d}\boldsymbol{w}_t(k) \cdot \boldsymbol{v}_t(k) \Big). \tag{3.28}$$

In Eq.3.28, when the averaged gradient $\sum_k \mathbf{d}\boldsymbol{w}_t(k) \cdot \boldsymbol{v}_t(k)$ is zero, the learning rate $\boldsymbol{\eta}_t$ is unchanged. Therefore, we add a decaying term to localize the search steps when approaching to convergence, resulting in Eq.3.24

$$\boldsymbol{\eta}_t = \boldsymbol{\eta}_{t-1} \cdot \exp\Big(-\mu\big(\sum_k \mathbf{d}\boldsymbol{w}_t(k) \cdot \boldsymbol{v}_t(k) + \epsilon \boldsymbol{\eta}_{t-1} \cdot \mathbf{x}_{t+1} \big) \Big).$$

We are going to see that with certain approximations (Eq.3.31), the form of learning rate adaptation (Eq.3.24) has no influence on how the gradient trace $\boldsymbol{v}_{t+1}(k)$ is calculated. Therefore, the self-decaying term is only for performance purpose and fits well with other rules.

$\boldsymbol{v}_{t+1}(k)$ is calculated part by part after putting the weight update rule (Eq.3.22) into into its definition Eq.3.25. The old weights $\boldsymbol{w}_t(k)$ give rise to the discounted old trace $\beta \boldsymbol{v}_t(k)$

$$\begin{aligned}
&(1-\beta) \sum_{j=0}^{\infty} \beta^j \frac{\partial \boldsymbol{w}_t(k)}{\partial \log \boldsymbol{\eta}_{t-j}} \\
&= (1-\beta) \sum_{j=1}^{\infty} \beta^j \frac{\partial \boldsymbol{w}_t(k)}{\partial \log \boldsymbol{\eta}_{t-j}} + (1-\beta) \frac{\partial \boldsymbol{w}_t(k)}{\partial \log \boldsymbol{\eta}_t} \\
&= \beta(1-\beta) \sum_{j=0}^{\infty} \beta^j \frac{\partial \boldsymbol{w}_t(k)}{\partial \log \boldsymbol{\eta}_{t-j-1}} \\
&= \beta \boldsymbol{v}_t(k).
\end{aligned} \tag{3.29}$$

By considering the second order gradient, the gradient of $\mathbf{d}\boldsymbol{w}_t(k)$ with respect to $\log \boldsymbol{\eta}$ is the product of the Hessian matrix and the discounted trace vector

$$\begin{aligned}
&(1-\beta) \sum_{j=0}^{\infty} \beta^j \frac{\partial \mathbf{d}\boldsymbol{w}_t(k)}{\partial \log \boldsymbol{\eta}_{t-j}} \\
&= (1-\beta) \sum_{j=0}^{\infty} \beta^j \sum_i \frac{\partial}{\partial w_t(k,i)} \left(\frac{\partial \ell(\boldsymbol{W}_t)}{\partial \boldsymbol{w}_t(k)} \right) \frac{\partial w_t(k,i)}{\partial \log \eta_{t-j}(i)} \\
&= \mathbf{H}_t(k)(1-\beta) \sum_{j=0}^{\infty} \beta^j \frac{\partial \boldsymbol{w}_t(k)}{\partial \log \boldsymbol{\eta}_{t-j}} \\
&= \beta \mathbf{H}_t(k) \boldsymbol{v}_t(k).
\end{aligned} \tag{3.30}$$

The gradient of the learning rate $\boldsymbol{\eta}_t$ with respect to $\log \boldsymbol{\eta}$ can be approximated by itself

$$
\begin{aligned}
& (1-\beta)\sum_{j=0}^{\infty}\beta^j \frac{\partial \boldsymbol{\eta}_t}{\partial \log \boldsymbol{\eta}_{t-j}} \\
= \; & (1-\beta)\sum_{j=0}^{\infty}\beta^j \frac{\partial e^{\log \boldsymbol{\eta}_t}}{\partial \log \boldsymbol{\eta}_t}\frac{\partial \log \boldsymbol{\eta}_t}{\partial \log \boldsymbol{\eta}_{t-j}} \\
\approx \; & (1-\beta)\sum_{j=0}^{\infty}\beta^j \boldsymbol{\eta}_t \qquad (3.31) \\
= \; & \boldsymbol{\eta}_t. \qquad (3.32)
\end{aligned}
$$

The approximation in Equation (3.31) assumes that $\partial \log \eta_t(i)/\partial \log \eta_{t-j}(i) = 1, \forall j \geq 0$. This means that the meta gradient descent (Eq.3.24) is small enough, therefore, the form of the meta descent rule is not important.

The gradient of the unnormalized gradient step $\boldsymbol{\eta}_t \cdot \boldsymbol{w}_t(k) \cdot \mathbf{d}\boldsymbol{w}_t(k)$ with respect to $\log \boldsymbol{\eta}$ is defined as

$$
\begin{aligned}
\mathbf{u}_{t+1}(k) &\equiv (1-\beta)\sum_{j=0}^{\infty}\beta^j \frac{\partial}{\partial \log \boldsymbol{\eta}_{t-j}}\Big(\boldsymbol{\eta}_t \cdot \boldsymbol{w}_t(k) \cdot \mathbf{d}\boldsymbol{w}_t(k)\Big) \qquad (3.33) \\
&= \boldsymbol{\eta}_t \cdot \boldsymbol{w}_t(k) \cdot \mathbf{d}\boldsymbol{w}_t(k) + \beta\boldsymbol{\eta}_t \cdot \boldsymbol{v}_t(k) \cdot \mathbf{d}\boldsymbol{w}_t(k) + \beta\boldsymbol{\eta}_t \cdot \mathbf{H}_t(k)\boldsymbol{v}_t(k) \cdot \boldsymbol{w}_t(k) \\
&= \boldsymbol{\eta}_t \cdot \Big[\big(\boldsymbol{w}_t(k) + \beta\boldsymbol{v}_t(k)\big) \cdot \mathbf{d}\boldsymbol{w}_t(k) + \beta\boldsymbol{w}_t(k) \cdot \mathbf{H}_t(k)\boldsymbol{v}_t(k)\Big].
\end{aligned}
$$

The gradient of the normalization vector $\mathbf{c}_t$ with respect to $\log \boldsymbol{\eta}$ turns out to be a weighted sum of $\mathbf{u}_{t+1}(k)$

$$
\begin{aligned}
& (1-\beta)\sum_{j=0}^{\infty}\beta^j \frac{\partial \mathbf{c}_t}{\partial \log \boldsymbol{\eta}_{t-j}} \\
&= (1-\beta)\sum_{j=0}^{\infty}\beta^j \Big[\mathbf{c}_t^2 \cdot \sum_k \frac{\partial}{\partial \log \boldsymbol{\eta}_{t-j}}\Big(\boldsymbol{\eta}_t \cdot \boldsymbol{w}_t(k) \cdot \mathbf{d}\boldsymbol{w}_t(k)\Big)\Big] \\
&= \mathbf{c}_t^2 \cdot \sum_k \mathbf{u}_{t+1}(k) \qquad (3.34)
\end{aligned}
$$

Combining Equation (3.22), (3.25), (3.29), (3.33) and (3.34) gives a recursive

update rule for the gradient trace $\boldsymbol{v}_{t+1}(k)$

$$\begin{aligned}
\boldsymbol{v}_{t+1}(k) &= (1-\beta)\sum_{j=0}^{\infty}\beta^{j}\frac{\partial}{\partial \log \boldsymbol{\eta}_{t-j}}\Big[\mathbf{c}_t\cdot\Big(\boldsymbol{w}_t(k)-\boldsymbol{\eta}_t\cdot\boldsymbol{w}_t(k)\cdot\mathbf{d}\boldsymbol{w}_t(k)\Big)\Big] \\
&= \mathbf{c}_t\cdot\Big(\beta\boldsymbol{v}_t(k)-\mathbf{u}_{t+1}(k)\Big)+\Big(\boldsymbol{w}_t(k)-\boldsymbol{\eta}_t\cdot\boldsymbol{w}_t(k)\cdot\mathbf{d}\boldsymbol{w}_t(k)\Big)\cdot\mathbf{c}_t^2\cdot\sum_{j=1}^{S}\mathbf{u}_{t+1}(j) \\
&= \mathbf{c}_t\Big(\beta\boldsymbol{v}_t(k)-\mathbf{u}_{t+1}(k)+\boldsymbol{w}_{t+1}(k)\sum_{j=1}^{S}\mathbf{u}_{t+1}(j)\Big). \qquad (3.35)
\end{aligned}$$

Following standard techniques, the Hessian-vector product $\mathbf{H}_t(k)\boldsymbol{v}_t(k)$ can be calculated in $O(N)$ steps. The calculation details is shown in Appendix A.

Chapter 4

Place Learning in Partially Observable Environments

Exploration is a distinguishing feature of an autonomous agent. There are two cognitive functions actively involved in exploration behaviors, namely, place learning and decision making. On the one hand, the agent gains understandings of the new place, and may recognize it in the future. On the other hand, the agent also decides where to explore or whether to stop exploration, being satisfied with the knowledge already learned. In place learning, the agent answers the questions "where I am" (localization) and "where I have been" (mapping). Based on partial result of place learning, the agent makes decisions about "where I should go next" (action selection).

In order to build artificial autonomous agents, robotics researchers have developed a lot of place learning methods [SC87; LM97; NPB95; BFHS96; YL96; HPG99]. The majority of early works focus on localization or mapping individually, while later works solve them simultaneously (called Simultaneous Localization and Mapping). In recent years, action selection is integrated into place learning to achieve robot systems capable of active exploration.

In this chapter, we address the place learning problem, i.e. localization and mapping. The action selection problem will be discussed in chapter 5-8. The remaining of this chapter is organized as follows. In section 4.1, probabilistic place learning methods are reviewed briefly. In section 4.2, exploration is modelled by a Partially Observable Markov Decision Process, and a generative model is introduced to implement a Bayesian network for place learning. In section 4.3, two gradient descent rules are proposed to learn the weights in the network. In section 4.4, the performance of the proposed learning rules is compared with standard gradient descent methods.

4.1 Overview of Probabilistic Place Learning Methods

Probabilistic place learning methods enjoy great success in robotics [SC87; SSC90; NPB95; BFHS96; TBF98; HPG99; DNDW+01; CN01]. According to the representation types of world models, probabilistic place learning methods roughly fall into three categories: grid-based, features-based, and topological methods [LMS+03]. Most of these methods use variants of Bayes filters to estimate the map of the environment and the agent's location in the map.

Grid-based methods represent the world as a discrete grid with cells indicating the occupancy probability [Mor88; BFHS96; FBT99b]. The agent's belief on location is a multimodal distribution, which makes it robust and capable for global localization [GBFK98; TFBD00]. Grid representation does not rely on predefined landmarks, but works directly with sensor data. Therefore, it is versatile for general types of environment. Due to the naive discretization, the resolution of the map is restricted and it is expensive both in memory and computation to maintain the state space. To improve computation efficiency, Monte Carlo sampling methods such as particle filters are developed to represent the belief as weighted samples [TFBD00; FTBD01].

The second category is feature-based methods, which represent the world as a set of detected landmarks. Based on the linear Gaussian assumptions, the full Bayesian filter reduces to an extended Kalman filter (EKF) [SSC90; DNDW+01]. The belief state is a multivariate Gaussian distribution over the locations of the agent and the landmarks. The mean of the Gaussian distribution is an estimate of the positions of the landmarks and the agent. The covariance matrix gives a measure of the uncertainty in estimation. The unimodal belief representation makes feature-based methods efficient for localization and mapping in environments with hundreds of landmarks. There are several disadvantages of feature-based methods. First, the linear Gaussian assumptions restrict the belief state to be unimodal, therefore make it hard to relocalize if the agent gets lost in tracking its position. Second, EKF assumes that the noises in observation and motion are Gaussian distributions, and needs the true noise variances to converge. Third, the convergence is crucially dependent on correct data association, i.e. on finding the correspondence between the observations and the internal landmark representations [SFL98; DNDW+01; NT01]. Incorrect data association can cause irrecoverable divergence. So the EKF-based methods are less robust than methods implementing the full Bays filter [GBFK98]. Forth, the computation complexity of EKF-based methods is quadratic in the number of landmarks. Therefore approximated filters [Pas03; TLK+04] or submap representations [LF99; GN02; ENT05] are designed to improve the scalability.

Instead of generating a metric map, topological methods learn the structure of an environment in a high-level way. The topology of the environment is explicitly modeled as Hidden-Markov Models or by graphs [NPB95; SK97; CN01]. Generally, topological representations are suitable for structured environments, where landmarks can be defined or learned [Thr98]. Topological representations are concise and can be combined easily with planning algorithms [SK95]. However, they normally lack metric information.

Some hybrid methods combine different representations to take the advantages from the individuals. The FastSLAM algorithm combines the Rao-Blackwellized particle filter and the extended Kalman filter to maintain sample based belief state and a feature-based map for each sample [MTKW02; MT03]. Therefore, the FastSLAM algorithm overcomes the limitation of unimodal location distribution and keeps the compactness of its map representation. The Rao-Blackwellized particle filter is also combined with grid maps to bypath the difficulty in landmark definition and recognition [EP03; HFBT03; GSB07].

There are two common problems in most of the current probabilistic place learning methods. First, the sensor model and motion model are normally fixed. For dynamic environments, an adaptive framework is important [HPG99; SPH04]. Second, data association methods neglect too much uncertainty in the location belief. Full probabilistic method is necessary to deal with multihypothesis data association.

To address these two problems, we propose a Bayesian network implementing a Partially Observable Markov Decision Process. The models in the framework are adaptive. Furthermore, the map is represented as a function, and is learned under the modulation of the uncertainty of the belief state. The proposed framework also has the ability to learn the optimal action selection strategy in exploration.

4.2 Modelling Exploration by POMDP

Partially Observable Markov Decision Process (POMDP) provides necessary elements to model the learning and decision making involved in exploration. The agent predicts its sensory observations and builds a sensor model by minimizing its sensory prediction errors. The sensor model is an estimation of the true observation probability, and serves as a map of the environment. The agent tracks its position according to the estimated motion model, and localizes itself with respect to the sensor model. Along with place learning, the agent actively selects its next action to maximize the improvement of its prediction performance in a global sense (see chapter 8).

We define a finite POMDP consisting of the following elements:

- A set of states $i \in \{1, 2, \cdots, N\}$, representing places in the environment.
- A set of actions $a \in \{1, 2, \cdots, A\}$.
- A set of sensory observations $s \in \{1, 2, \cdots, S\}$.
- Agent's belief of its location $p_t(i) := p(i|s_1, \cdots, s_t, a_1, \cdots, a_{t-1})$, which is a probability distribution over the states i. $p_t(i)$ is called belief state.
- Sensor model $\hat{p}_t(s|i)$. It is the internal estimation of the true observation probability $p(s|z)$ (see chapter 2), which is the conditional probability of observing s in cell z. The sensor model is initialized as a uniform distribution, and is learned by the agent during its exploration in the environment. $\hat{p}_t(s|i)$ is an internal representation of the causal relationship between location z and sensory input s. Therefore, it gives a functional representation of the map in the environment.
- Motion model $T^a(i, j) := \hat{p}(i|a, j)$ for each action a. It is the estimation of the true transition probability $p(i|a, j)$, which is the probability that the agent goes from state j to state i under action a. We assume that the agent has already learned the true transition probabilities according to a similar gradient descent method presented in this chapter. Figure 4.1 shows motion models for the right action with different motion noise levels (see Eq.2.1).
- Intrinsic reward signal $r_t \in R$ for the action a_t chosen by the agent at time t.
- Action value $\hat{Q}_t(a, i)$. It is the estimated utility of taking action a when the agent believes to be in state i. Based on $\hat{Q}_t(a, i)$, action is selected to greedily maximize the discounted rewards the agent receives

$$a_t = \arg\max_a \sum_i \hat{Q}_t(a, i) p_t(i). \tag{4.1}$$

  The definition of reward r_t and active exploration methods based on $\hat{Q}_t(a, i)$ will be covered in chapter 8. The simulations in this chapter are restricted to random action selection strategy.

As the estimation of the true observation probability $p(s|z)$, the sensor model $\hat{p}_t(s|i)$ contains the agent's memory on perceiving the environment. The sensor model realizes a cognitive map of the environment, and allows functions like sensory predictions, path planning and place recognitions. The value

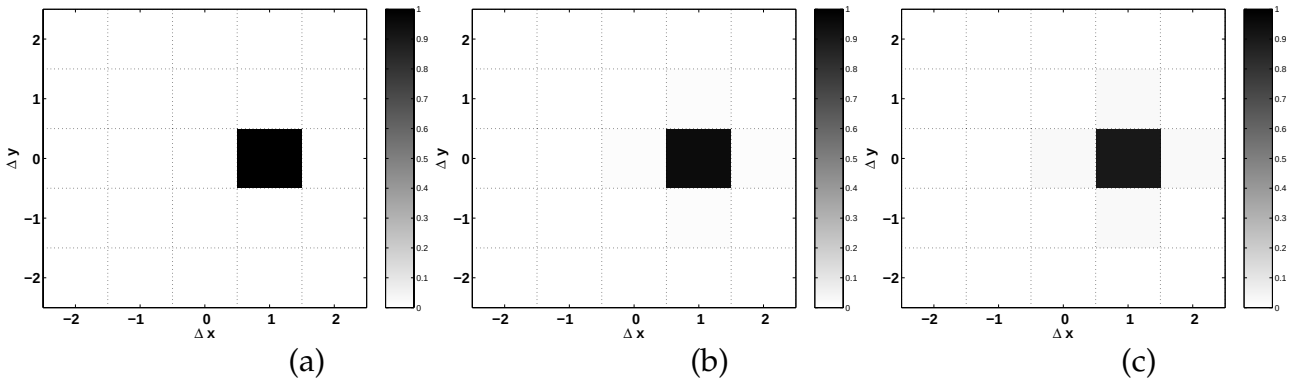


Figure 4.1: Examples of transition probability. Each panel shows the probability of arriving in neighboring cells (indexed by discrete offset Δx, Δy) when the agent takes the action "right" with unitary length from the current cell (0,0) under different noise levels. The transition probability matrices are discrete approximations of the state equation (2.1). (a) Accurate odometry $\sigma_M = 0$; (b) Motion noise spread $\sigma_M = 0.1$, or 10% noise; (c) Motion noise spread $\sigma_M = 0.2$, or 20% noise.

function $\hat{Q}_t(a, i)$ is analogous to the agent's internal feelings, like curiosity or motivation.

In summary, place learning involves the learning of the sensor model $\hat{p}_t(s|i)$, and the update of location belief $p_t(i)$. Action selection involves estimating the value function $\hat{Q}_t(a, i)$. In the next section, we will show how $\hat{p}_t(s|i)$ is learned online by gradient descent methods and how the belief state is updated by Beyes' rule. The definition of intrinsic rewards and the learning of the value function $\hat{Q}_t(a, i)$ will be discussed in chapter 8.

4.3 Learning through Anticipation and Adaptation

At time step t, the agent selects an action $a_t \in \{1, 2, \cdots, A\}$ according to some exploration strategy, and predicts its next observation based on past experiences. After executing action a_t, the agent gets new sensory information $s_{t+1} \in \{1, 2, \cdots, S\}$ from the environment, and evaluates its prediction performance with a loss function. Minimizing the prediction loss through gradient decent in the sensor model $\hat{p}_t(s|i)$, the agent integrates new sensory information into its internal representations, thereby improves future prediction performance. The agent's belief of its location is updated according to Bayes' rule incorporating the new sensory and motion information. This learning cycle is explained in the following sections.

4.3.1 Prediction

After choosing action a_t, the agent propagates the internal belief of its location by the motion model

$$\begin{aligned} \hat{p}_{t+1}(i) &:= p(i|s_1, \cdots, s_t, a_1, \cdots, a_t) \\ &= \sum_j T^{a_t}(i,j) p_t(j). \end{aligned} \tag{4.2}$$

The sensory input at the new position is predicted

$$\hat{p}_{t+1}(s) = \sum_i \hat{p}_t(s|i) \hat{p}_{t+1}(i). \tag{4.3}$$

4.3.2 Learning Observation Probability

As a result of action a_t, the agent reaches the next position, and perceives new sensory information s_{t+1}. The prediction loss is evaluated according to some measure between the real sensory observation $p_{t+1}(s)$ and the predicted one $\hat{p}_{t+1}(s)$. We consider square distance and Kullback-Leibler divergence as two possible loss functions, and derive gradient descent rules for these two loss functions respectively.

Quadratic Loss and Additive Gradient Descents

The quadratic loss is a commonly used loss function

$$\ell_{SQ}(t+1) = \frac{1}{2} \sum_s \Big(\hat{p}_{t+1}(s) - p_{t+1}(s) \Big)^2. \tag{4.4}$$

By combining Eq.4.3 with Eq.4.4, the gradient of the square loss with respect to $\hat{p}_t(s|i)$ is given by

$$\frac{\partial \ell_{SQ}(t+1)}{\partial \hat{p}_t(s|i)} = \Big(\hat{p}_{t+1}(s) - p_{t+1}(s) \Big) \hat{p}_{t+1}(i). \tag{4.5}$$

A straightforward gradient descent rule is the standard *additive rule*

$$\hat{p}_{t+1}(s|i) = \hat{p}_t(s|i) - \eta \Big(\hat{p}_{t+1}(s) - p_{t+1}(s) \Big) \hat{p}_{t+1}(i), \tag{4.6}$$

where η is a fixed positive learning rate.

In order to achieve better asymptotic result, decreasing learning rates can be realized by summing the believes, giving rise to the *counting rule*

$$\begin{aligned} \hat{p}_{t+1}(s|i) &= \hat{p}_t(s|i) - \eta_t(i)\Big(\hat{p}_{t+1}(s) - p_{t+1}(s)\Big)\hat{p}_{t+1}(i) \\ \eta_t(i) &= \frac{\mu}{1 + \sum_{\tau=1}^{t+1} \hat{p}_\tau(i)} \end{aligned}, \tag{4.7}$$

where μ is a positive learning rate.

Both the additive rule (Eq.4.6) and the counting rule (Eq.4.7) are self-normalizing. However, $\hat{p}_{t+1}(s|i)$ can become negative or larger than one if the learning rate is too large. When this happens, $\hat{p}_{t+1}(s|i)$ are projected to the boundary values 0 or 1 respectively and are renormalized in order to conserve the probability.

Interestingly, the belief $\hat{p}_{t+1}(i)$ appears as a multiplicative factor in the gradient term (Eq. 4.5). The belief state acts as a self adapting learning rate, since the discrepancy $\Big(\hat{p}_{t+1}(s) - p_{t+1}(s)\Big)$ takes effect under the modulation of the belief state. The effect that sensory observations are integrated into internal estimations proportional to the location likelihood is called belief modulation. Therefore multihypothesis data association is realized by keeping the uncertainty of location in the map estimation.

Relative Entropy and Multiplicative Gradient Descents

Another interesting loss function is the Kullback-Leibler divergence or relative entropy between the true sensory observation $p_{t+1}(s)$ and the predicted sensory distribution $\hat{p}_{t+1}(s)$

$$\ell_{KL}(t+1) = \sum_s p_{t+1}(s) \log \frac{p_{t+1}(s)}{\hat{p}_{t+1}(s)}. \tag{4.8}$$

When the target coding scheme is used, where $p_{t+1}(s) = \delta_{s,s_{t+1}}$, the loss function $\ell_{KL}(t+1)$ is simplified to

$$\ell_{KL}(t+1) = -\log \hat{p}_{t+1}(s_{t+1}), \tag{4.9}$$

which leads to the gradient of $\ell_{KL}(t+1)$ with respect to $\hat{p}_t(s|i)$

$$\frac{\partial \ell_{KL}(t+1)}{\partial \hat{p}_t(s|i)} = -\frac{\delta_{s,s_{t+1}} \hat{p}_{t+1}(i)}{\hat{p}_{t+1}(s_{t+1})}. \tag{4.10}$$

Since the negative gradient is always non-negative, the multiplicative gradient decent is applicable, leading to the *multiplicative rule*

$$\begin{aligned}\hat{p}_{t+1}(s|i) &= c_t(i)\hat{p}_t(s|i)\Big(1+\eta\frac{\delta_{s,s_{t+1}}\hat{p}_{t+1}(i)}{\hat{p}_{t+1}(s_{t+1})}\Big)\\ c_t(i) &= \frac{1}{1+\eta\frac{\hat{p}_t(s_{t+1}|i)\hat{p}_{t+1}(i)}{\hat{p}_{t+1}(s_{t+1})}}\end{aligned}, \tag{4.11}$$

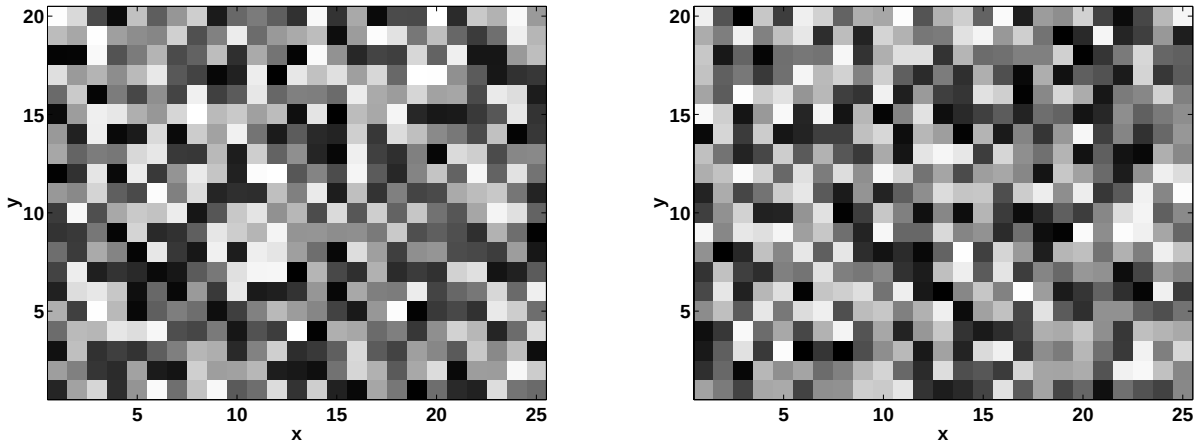
where the learning rate η is a positive constant, and $c_t(i)$ is a normalization factor.

For better convergence, the learning rate of the multiplicative rule could be adapted by gradient descent in a meta level, resulting in the *adaptive multiplicative rule* (see chapter 3)

$$\begin{aligned}\hat{p}_{t+1}(s|i) &= c_t(i)\hat{p}_t(s|i)\Big(1+\eta_t(i)\frac{\delta_{s,s_{t+1}}\hat{p}_{t+1}(i)}{\hat{p}_{t+1}(s_{t+1})}\Big)\\ c_t(i) &= \frac{1}{1+\eta_t(i)\frac{\hat{p}_t(s_{t+1}|i)\hat{p}_{t+1}(i)}{\hat{p}_{t+1}(s_{t+1})}}\\ \eta_t(i) &= \eta_{t-1}(i)\exp\Big(-\mu\Big(-\frac{\hat{p}_{t+1}(i)v_t(s_{t+1},i)}{\hat{p}_{t+1}(s_{t+1})}+\epsilon\eta_{t-1}(i)\hat{p}_{t+1}(i)\Big)\Big)\\ v_{t+1}(s,i) &= c_t(i)\Big(\beta v_t(s,i)-u_{t+1}(s,i)+\hat{p}_{t+1}(s|i)u_{t+1}(s_{t+1},i)\Big)\\ u_{t+1}(s,i) &= \frac{\eta_t(i)\hat{p}_{t+1}(i)}{\hat{p}_{t+1}(s_{t+1})}\Big[-\Big(\hat{p}_t(s|i)+\beta v_t(s,i)\Big)\\ &\quad+\beta\frac{\hat{p}_t(s|i)}{\hat{p}_{t+1}(s_{t+1})}\sum_j v_t(s,j)\hat{p}_{t+1}(j)\Big]\delta_{s,s_{t+1}}\end{aligned}, \tag{4.12}$$

where μ and ϵ are positive meta learning rates. $\beta \in [0,1]$ is a discounting factor. The trace $v_t(s,i)$ is initialized as 0.

Since the loss function is the negative log likelihood of the observed data, the resulting multiplicative rule and the adaptive multiplicative rule are maximum likelihood estimation methods. The belief $\hat{p}_{t+1}(i)$ also appears as a multiplicative factor in the gradient term (Eq.4.10). Therefore, these two rules are also capable of multihypothesis data associations due to belief modulation.



(a) Environment a (b) Environment b

Figure 4.2: Occupancy probabilities of the locations in two of the environments used in simulations. Black encodes probability 1, and white represents probability 0.

4.3.3 Belief Update

After adapting the sensor model, the agent updates its belief into a posterior distribution through Bayes' rule to include the newest information

$$\begin{aligned} p_{t+1}(i) &:= p(i|s_1, \cdots, s_t, s_{t+1}, a_1, \cdots, a_t) \\ &= \frac{\hat{p}_{t+1}(s_{t+1}|i)\hat{p}_{t+1}(i)}{\sum_j \hat{p}_{t+1}(s_{t+1}|j)\hat{p}_{t+1}(j)}. \end{aligned} \tag{4.13}$$

Through anticipation and adaptation, the agent incrementally localizes itself by maintaining its belief up-to-date, and maps its observations into the sensor mode.

4.4 Simulations

Additive gradient descents (Eq.4.6 and Eq.4.7) and multiplicative gradient descents (Eq.4.11 and Eq.4.12) are compared under random exploration. Five environments are randomly generated by choosing the occupancy probabilities from the uniform distribution in the range $[0, 1]$. Fig.4.2 depicts two of them. Each learning rule is tested in the five environments, with two different starting positions. Average performance of the ten runs is compared.

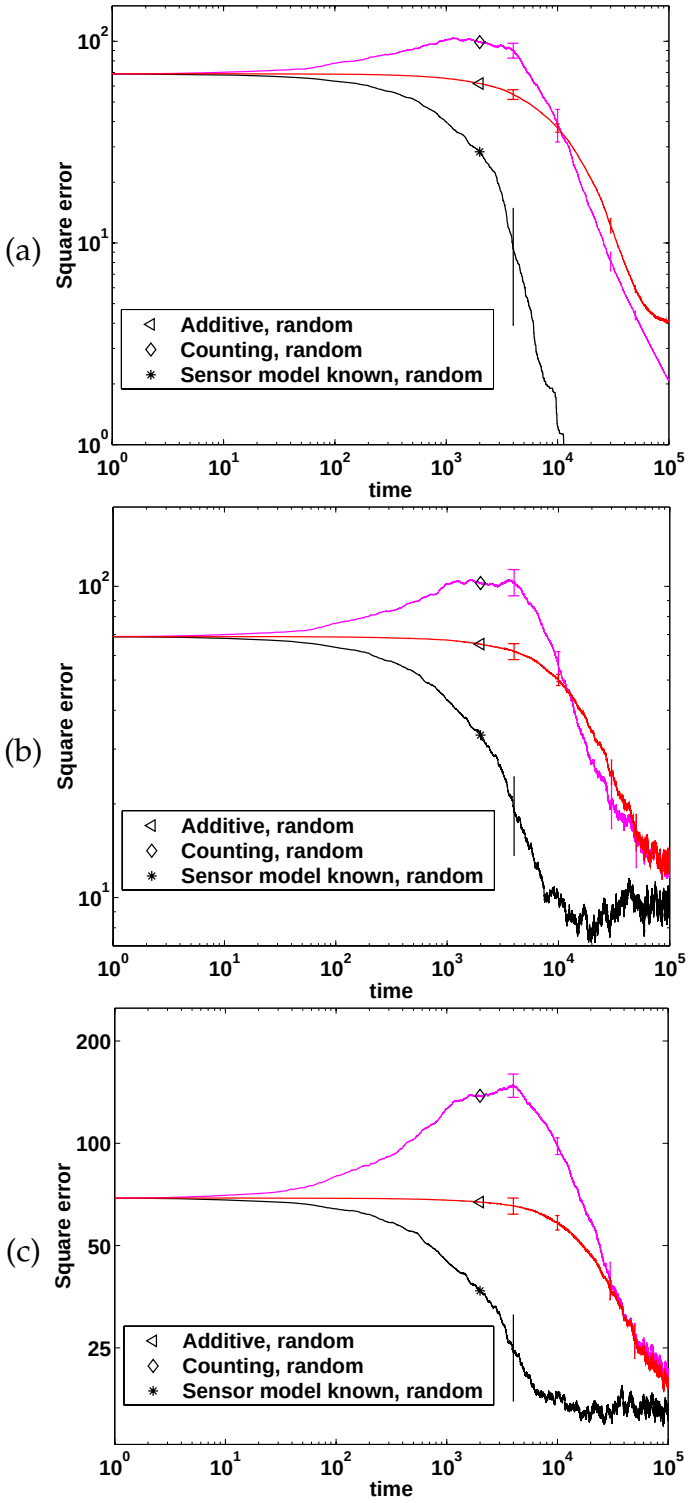


Figure 4.3: Average prediction error $E_{SQ}(t)$ of the additive rule and the counting rule under random exploration with different motion noise levels. The error bars shows $\pm$ standard deviations. (a) No motion noise ($\sigma_M = 0$). The learning rates are $\eta = 0.02, \mu = 1$; (b) Medium motion noises ($\sigma_M = 0.1$). The learning rates are $\eta = 0.02, \mu = 1$; (c) Large motion noises ($\sigma_M = 0.2$). The learning rates are $\eta = 0.02, \mu = 3$.

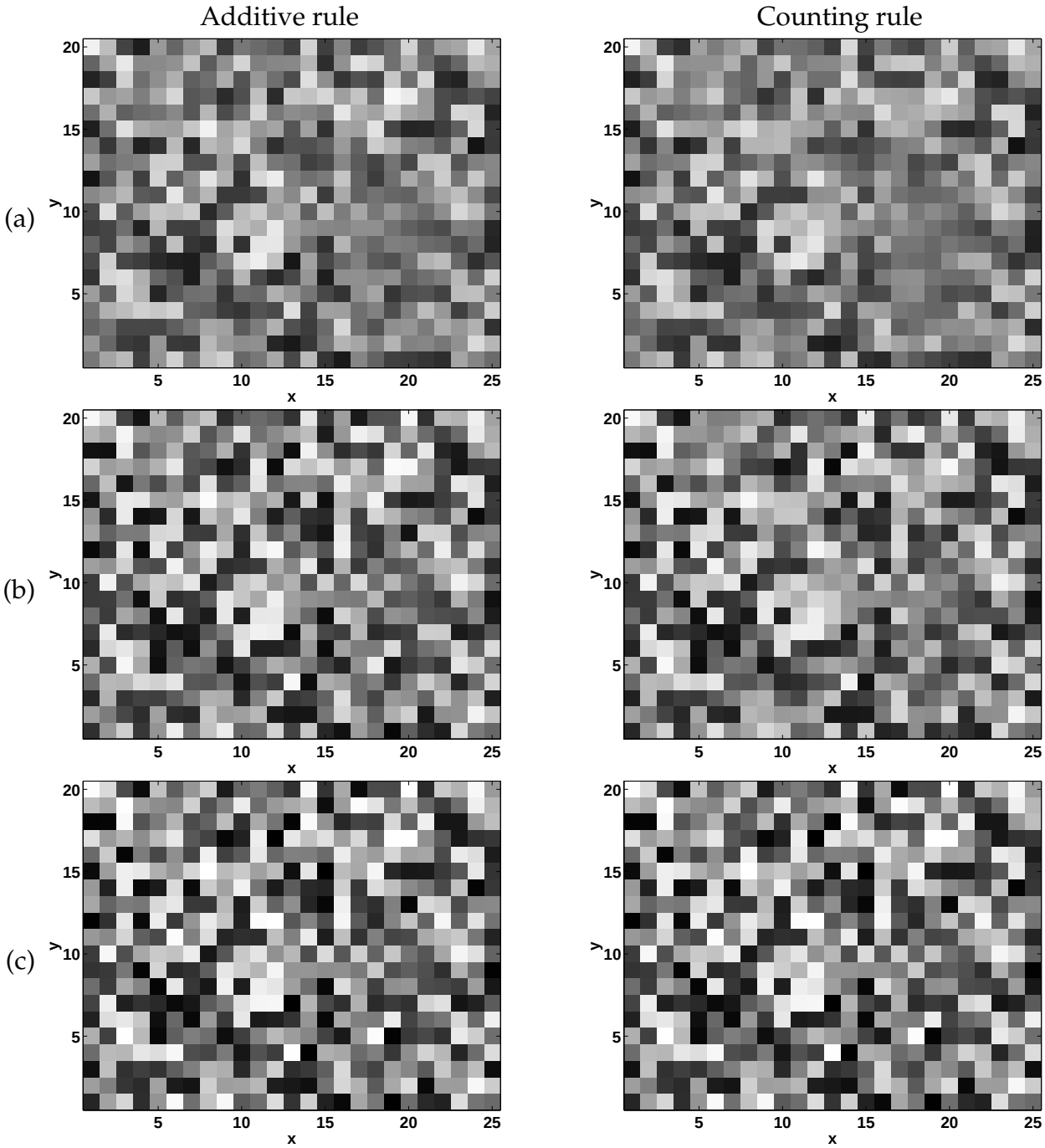


Figure 4.4: Maps constructed for environment a at the end of random exploration using additive gradient descents ($\sigma_M = 0.2$). The total number of steps is $t = 10^5$. (a) Maps constructed from the prediction records $q_t(s|z)$; (b) Maps constructed from the sensor models $\hat{p}_t(s|i)$; (c) True map of environment a. The maps constructed from the prediction records are more blurred than those from the sensor models.

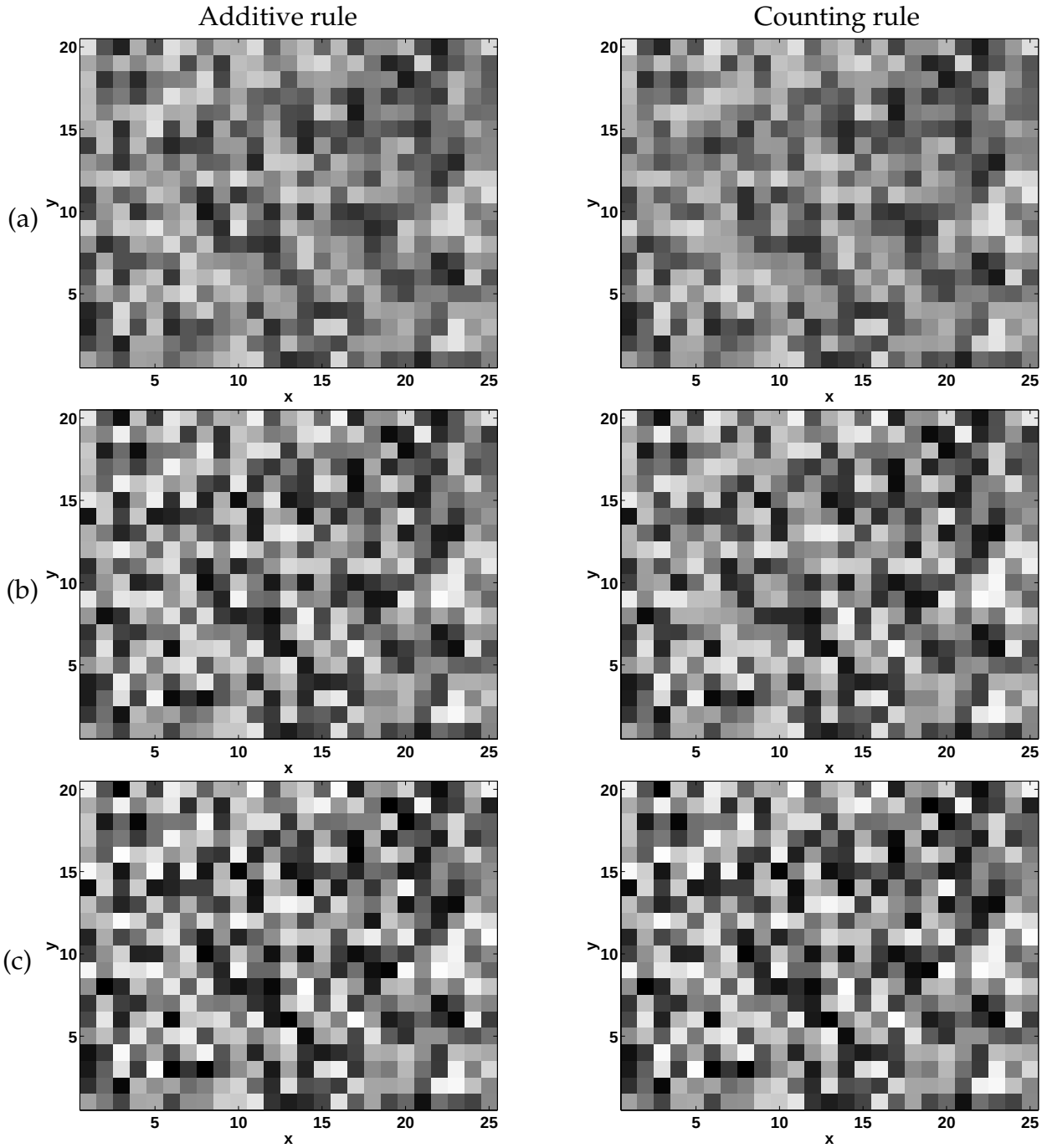


Figure 4.5: Maps learned for environment b at the end of random exploration using additive gradient descents ($\sigma_M = 0.2$). The total number of steps is $t = 10^5$. (a) Maps constructed from the prediction records $q_t(s|z)$; (b) Maps constructed from the sensor models $\hat{p}_t(s|i)$; (c) True map of environment b. The maps constructed from the prediction records are more blurred than those from the sensor models.

4.4.1 Performance of Additive Gradient Descents

First, we compare the prediction errors of the additive rule (Eq.4.6) and the counting rule (Eq.4.7) with respect to the average square error $E_{SQ}(t)$ (Eq.2.3), under different levels of motion noises σ_M. To show how large the prediction error results from noisy motions, we also test the performance of an ideal agent, who already knows perfect knowledge, i.e. the true sensor model, and who navigates randomly in the environment without any learning.

Fig.4.3(a) plots the averaged square error $E_{SQ}(t)$ when the motion is accurate. The curve marked by star is for the ideal agent. Since the ideal agent has already known the correct observation probability, after one traverse of the environment (about 10^4 steps), the agent makes perfect predictions for all locations. Therefore, the prediction record $q_t(s|z)$ recorded by the ideal observer perfectly matches the true observation probability $p(s|z)$, and the error measure $E_{SQ}(t)$ becomes zero. The performance differences between the ideal agent and the learning agents are caused by the lack of knowledge about $p(s|z)$.

The curve marked by triangle corresponds to the performance of the learning agent with the additive rule ($\eta = 0.02$). The asymptotic error is about 4. Larger learning rates lead to larger asymptotic errors (not shown), because the gradient decent steps oscillate around the optimal point.

As a contrast, the learning agent with the counting rule (diamond mark) achieves a smaller error of 2 at the end of 10^5 steps. Its error can be further reduced, because its decaying learning rate scheme leads the gradient descent steps to convergence to the optimal point. On average, the counting rule outperforms the additive rule by 27%.

Under inaccurate motions, the agents achieve worse learning performance. As shown in Fig.4.3(b), when the motion noises is 10% the ideal agent settles down in a higher asymptotic error, which is about 10. When the motion noise is 20% the asymptotic prediction error of the ideal agent is about 15 (Fig.4.3(c)). This asymptotic error can not be reduced further, because it results from the motion noises. With perfect observation probability, the ideal agent has no problem in localizing itself. The belief state contains necessary uncertainty to account for the odometry noises. As a result, the sensory prediction of the agent is an integration of the sensory information in the environment. Therefore, the prediction record is a blurred version of the true observation probability.

Similarly, the learning agents also achieves higher asymptotic prediction errors under inaccurate odometry. Both the additive rule and the counting rule are quite robust against motion noises. Under large motion noises even up to 20% motion noise ($\sigma_M = 0.2$), the counting rule is still as good as the additive rule.

It is interesting to look into the occupancy maps constructed from the prediction records and from the observation probabilities learned by the agents. Fig.4.4(a)-(b) are the maps constructed for environment a. Fig.4.5(a)-(b) are the maps constructed for environment b. In both cases, the prediction records and the estimated observation probabilities produce occupancy maps consistent with the environments (shown in Fig.4.4(c) and 4.5(c) respectively). This demonstrates that the agents build correct internal models of the environments during exploration.

The occupancy maps constructed from the estimated observation probabilities (Fig.4.4(b) and 4.5(b)) are blurred version of the true occupancy maps (Fig.4.4(c) and 4.5(c)). This is due to the belief modulation effect. The sensory observations are integrated into locations proportional to the location likelyhood. The occupancy maps constructed from the prediction records (Fig.4.4(a) and 4.5(a)) are more blurred than those constructed from the estimated observation probabilities. This is because the sensory prediction is an integration of the sensor model by the location belief. The uncertainty of position estimation is included in the prediction record. As a result, the prediction record $q_t(s|z)$ is dependent on the joint performance of localization and mapping, thereby is a good quantity to evaluate place learning.

4.4.2 Performance of Multiplicative Gradient Descents

In the second simulation, the prediction errors of the multiplicative rule (Eq.4.11) and the adaptive multiplicative rule (Eq.4.12) are compared with respect to the averaged Kullback-Leibler divergence $E_{KL}(t)$ (Eq.2.4).

Fig.4.6 depicts the averaged prediction error $E_{KL}(t)$ under different motion noise conditions. The advantage of the adaptive multiplicative rule is most evident when the odometry noise is small. With accurate motion, the adaptive multiplicative rule outperforms the multiplicative rule more than 50%. It achieves perfect convergence due to adaptive learning rates. On the contrary, the asymptotic performance of the multiplicative rule saturates on a fixed level.

The advantage of the adaptive multiplicative rule is maintained with motion noise up to 20% (Fig.4.6c). Under large motion noises, the asymptotic performance of the adaptive multiplicative rule is little worse than the multiplicative rule.

The simulations demonstrate that learning rate adaptation is important for fast learning and asymptotic convergence. The proposed adaptive multiplicative rule adopts an adaptive gain adaptation method, therefore, it can better deal

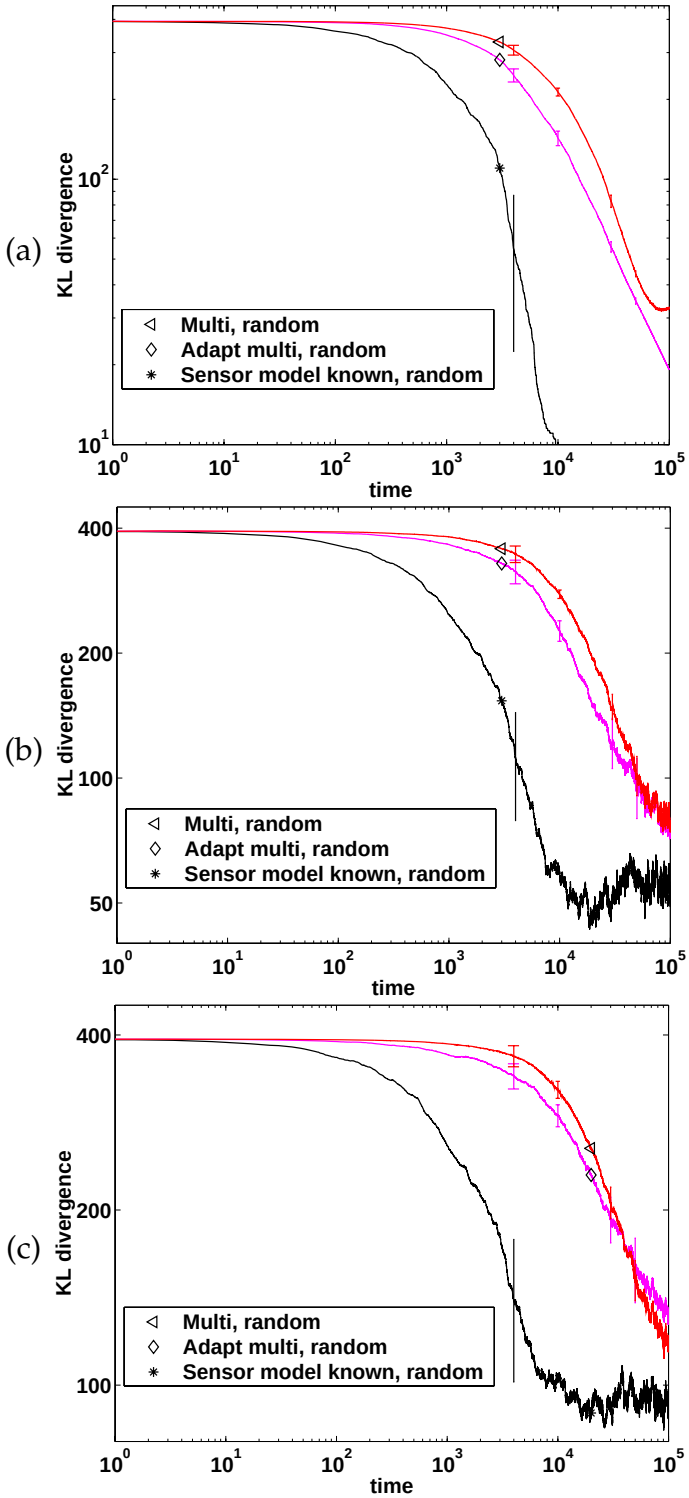


Figure 4.6: Prediction error $E_{KL}(t)$ of the multiplicative rule ($\eta = 0.02$) and the adaptive multiplicative rule ($\eta_0 = 0.1, \mu = 2, \beta = 0.1, \epsilon = 0.3$) under random exploration with different motion noise levels. The error bars show $\pm$ standard deviations. (a) No motion noise ($\sigma_M = 0$); (b) Medium motion noises ($\sigma_M = 0.1$); (c) Large motion noises ($\sigma_M = 0.2$).

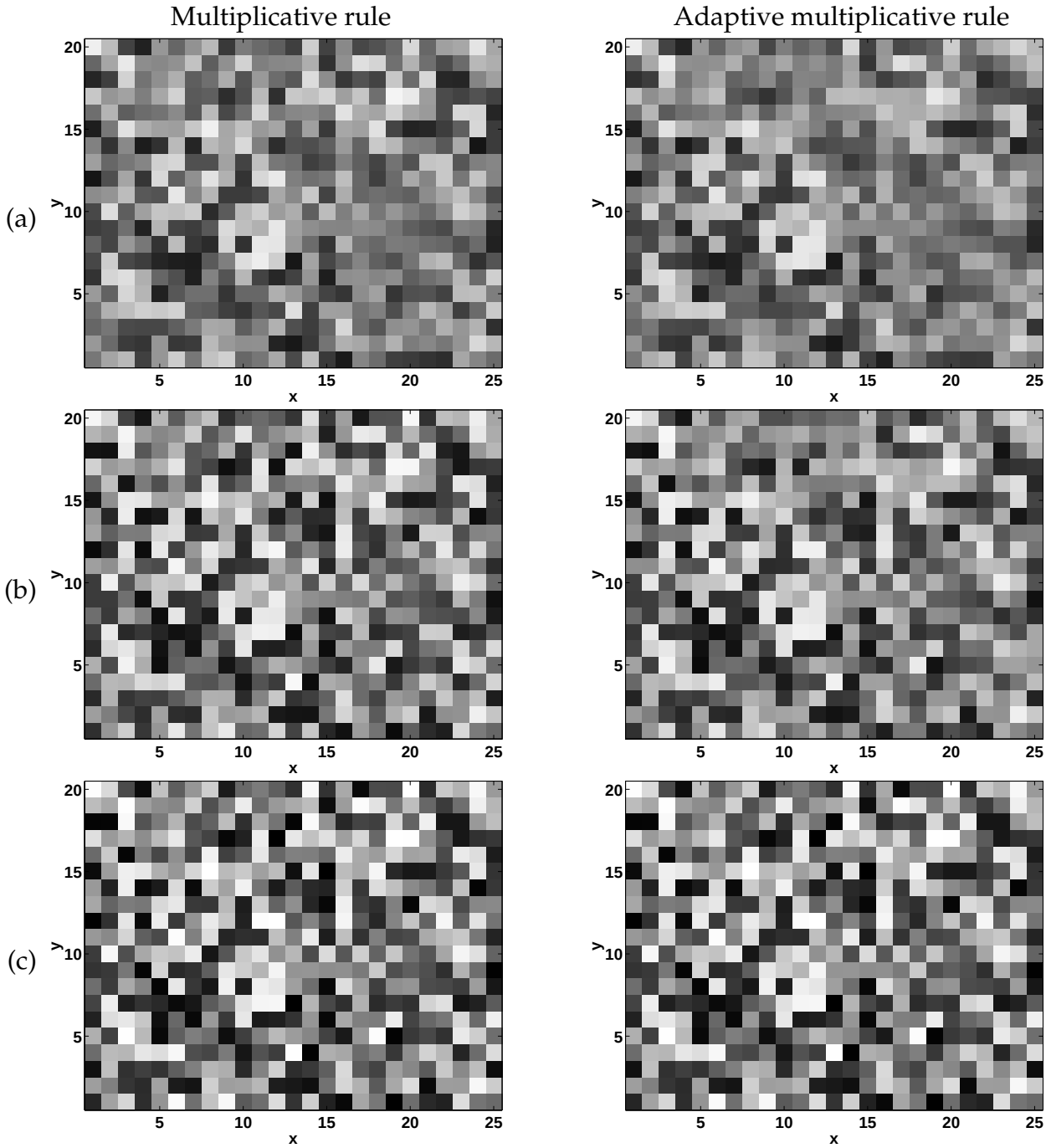


Figure 4.7: Maps constructed for environment a at the end of random exploration using multiplicative gradient descents ($\sigma_M = 0.2, t = 10^5$). (a) Maps constructed from the prediction records $q_t(s|z)$; (b) Maps constructed from the sensor models $\hat{p}_t(s|i)$. (c) True map of environment a. The maps constructed from the prediction records are more blurred than those from the sensor models.

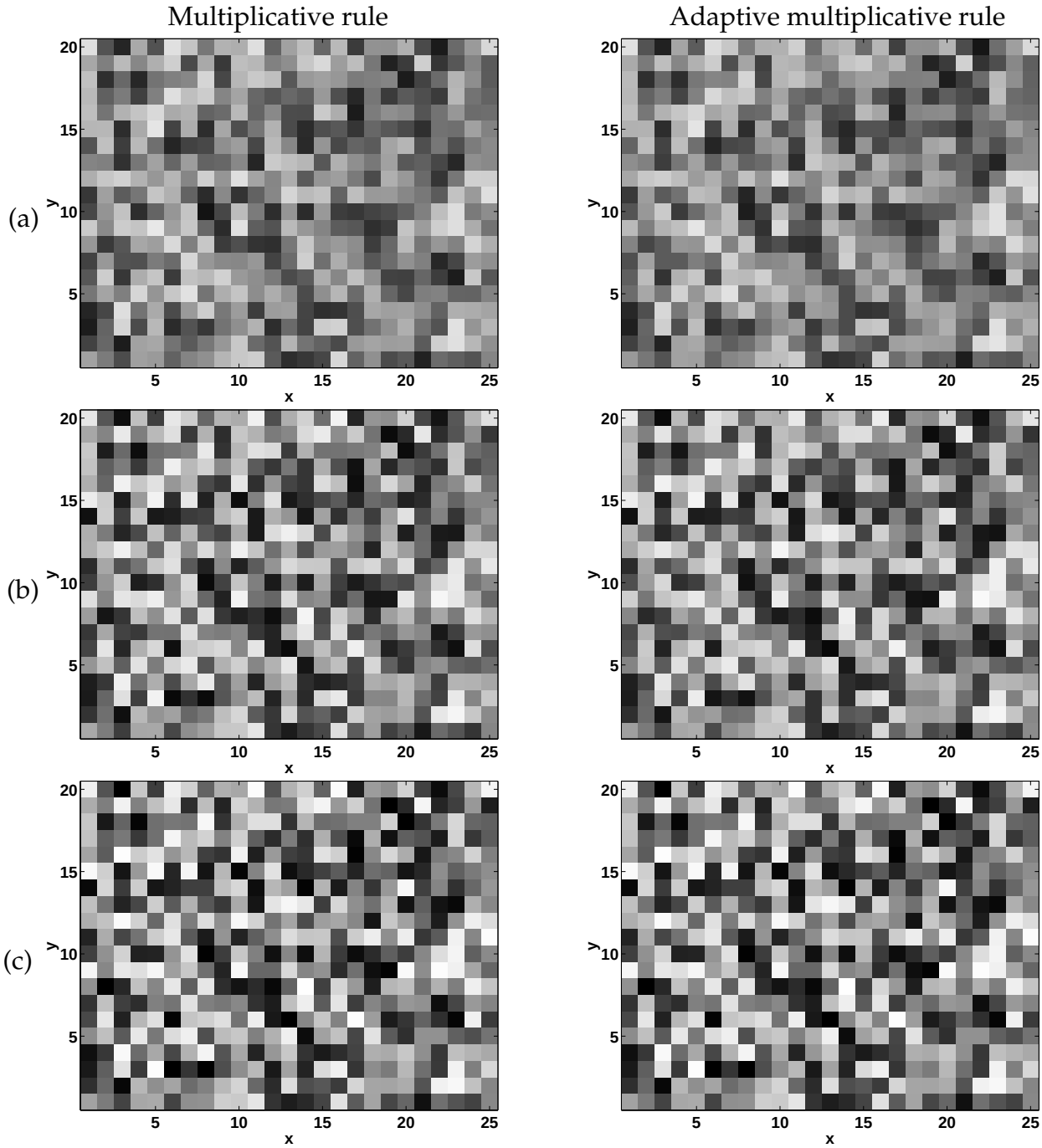


Figure 4.8: Maps constructed for environment b at the end of random exploration using multiplicative gradient descents ($\sigma_M = 0.2, t = 10^5$). (a) Maps constructed from the prediction records $q_t(s|z)$; (b) Maps constructed from the sensor models $\hat{p}_t(s|i)$. (c) True map of environment b. The maps constructed from the prediction records are more blurred than those from the sensor models.

with slowly changing environment than the counting rule, which uses a fixed scheme for learning rate decaying.

At the end of exploration, both adaptive multiplicative rule and the multiplicative rule successfully learn good maps of the environments (Fig.4.2), even with large odometry errors (Fig.4.7 and 4.8). The maps constructed from the sensor models are blurred compared with the true occupancy maps. The uncertainty in senor model is introduced by the belief modulation of the learning rules. It is an internal representation containing uncertainties both in observations and motions. The maps constructed from the prediction records are external representations of the true occupancy maps. The uncertainty in prediction record comes from the sensor model (mapping) and the belief state (localization).

Chapter 5

Exploration as Global Learning

Exploration is to sample data in the input space, so that the learning agent performs well over the whole range of the input space. Exploration directs the agent to the subspace where local learning is most necessary. As a consequence, the agent achieves global learning efficiently. Active exploration strategies play a critical role for learning systems to better solve problems like classification, regression, clustering, robot navigation and etc [SOS92; Coh96; TM92; Sch91; Her01]. Active exploration gives rise to two distinctive learning paradigms: reinforcement learning and active learning.

In reinforcement learning, the agent explores the action-state space in order to estimate a value function or policy, so that the currently obtained knowledge improves the efficiency of future exploitation [SB98b]. Meanwhile, the agent need to dynamically tradeoff between discovering putatively better actions (exploration) and immediate reward maximization (exploitation).

Active data selection in active learning systems is another type of exploration. In the traditional passive learning paradigm, training data is randomly sampled according to a given probability distribution on the input data space. On the contrary, an active learning agent actively selects training samples according to strategies that are developed for specific problems, such as classification, regression, robot navigation and etc [SOS92; Coh96; TM92; Sch91; Her01]. By exploring the space of input data, the agent may not only achieves better asymptotic results, but also may increase efficiency on a finite data set.

In section 5.1, exploration strategies are categorized into three broad classes. To compare different exploration strategies, in section 5.2 we introduce the multi-armed bandit as a benchmark, and formulate exploration as the problem of global learning. After reviewing the exploration policies in multi-armed bandits in section 5.3, we derive an optimal exploration strategy in the senses of mean squared error under the known reward variances assumption (section

5.4). In section 5.5, the unknown reward variances situation is discussed, and an active exploration policy is proposed, which achieves the optimal exploration strategy asymptotically. In section 5.6, the performance of exploration policies is compared numerically.

5.1 Exploration Strategies

According to the underlying principles, exploration strategies can be categorized into three classes: counter-based, error-based and gain-based strategies.

In counter-based exploration, a register for each sample is kept updated and the least selected samples are chosen first. Counter-based exploration is more efficient than random exploration in terms of costs and time, because less-informative and redundant samples are avoided [Thr92]. However, counter-based exploration does not distinguish the estimation performance in action/data selection.

Error-based exploration prefers states where the estimation error is large [MB99; TM92]. This heuristic assumes that the larger the estimation error is, the more one needs to know about it. Error-based exploration strategies allocate the amount of samples directly according to the estimation error. However, the optimal amount of exploration is not necessarily proportional to the estimation error. It is both theoretically interesting and practically worthwhile to find the optimal amount of exploration, especially when the estimation error is large.

The third type of exploration is gain-based exploration. Gain-based exploration strategies choose the sample which causes maximal performance improvement. In these policies, an optimality measure is defined to be minimized [Mac92], and samples are selected to maximize the gain of data or action selection, defined by the reduction of the measure. For example, the entropy of the posterior probability distribution on the weight space was considered as a measure of uncertainty for classification problems [SOS92]. The entropy reduction after querying one example is defined as the information gain of the query. Training examples are queried to maximize the entropy reduction. For regression problems considered in [Coh96; CGJ96], the mean squared error is a measure of the prediction performance of the network. For an unbiased learner, minimizing the mean squared error is equivalent to minimizing the variance, so the variance reduction of the network is appropriate to guide data selection. By organizing the exploration behavior according to the gain maximization principle, the learning system achieves some sort of autonomy. In this aspect, gain is the curiosity or attention of the learning system. For exam-

ple, in [Sch91] the change of prediction reliability is defined as the curiosity goal of the control system, and actions are chosen to maximize a combination of rewards and curiosity. In robot exploration tasks, the reduction of the averaged prediction error works as the curiosity of the autonomous agents [Her01]. Curiosity drives the agent to the place where it learns most, and to avoid the place where it is too easy or too difficult to learn.

5.2 Exploration in Multi-armed Bandits

Due to model-specific assumptions and approximations, the action/data selection rules used in learning systems are rather different. A standard model and a common measure are necessary for a manifesting comparison between different exploration strategies. To this end, we consider the multi-armed bandit [Rob52] as a prototypical benchmark for such an evaluation. The multi-armed bandit is an analogy for a gambler, who is trying to maximize his cumulative reward in a casino. It is a general model of decision making, shared in fields like machine learning, game theory, and clinical experiment design. The multi-armed bandit is suitable for fair comparison of exploration strategies because it captures the essential aspects of reinforcement learning and active learning, for example action/data selection under uncertainty, and exploration-exploitation trade-off [KLM96; Coh96]. Furthermore, the simplicity of the multi-armed bandit makes analytical evaluation possible, and thus provides theoretical foundations for different exploration strategies.

Before we go into the details of exploration strategies, we introduce the multi-armed bandit and define the problem of exploration.

Formally, an n-armed bandit is a one-state Markov Decision Process with the following elements.

- A set of actions (arms) $\mathcal{A} = \{1, 2, \cdots, n\}$.
- An action selection policy $\pi_{t+1}(a)$ specifying an action a_{t+1} to be played at the next time step.
- Distributions of rewards $p(r|a)$. At time t a reward r_t is drawn from $p(r|a_t)$ corresponding to the action at time t.

The agent is faced with a learning problem: to learn the n reward distributions $p(r|a), a \in \mathcal{A}$. Assuming $p(r|a)$ to be normally distributed with mean $\mu(a)$ and variance $\sigma^2(a)$, the agent estimates the parameters of $p(r|a)$, i.e. of the mean $\mu(a)$ and the variance $\sigma^2(a)$, by

$$\hat{\mu}_t(a) = \frac{1}{\tau(a)} \sum_{k=1}^{t} r_k \delta_{a,a_k}, \qquad \tau(a) \geq 1, \tag{5.1}$$

$$\hat{\sigma}_t^2(a) = \frac{1}{\tau(a)-1} \sum_{k=1}^{t} [r_k - \hat{\mu}_t(a)]^2 \delta_{a,a_k}, \qquad \tau(a) \geq 2. \tag{5.2}$$

Here δ_{a,a_k} is the Kronecker delta function. And the sampling number $\tau(a)$ is the number of times action a has been selected. For the sake of simplicity of notation, we drop the dependency of $\tau(a)$ on t.

$$\tau(a) = \sum_{k=1}^{t} \delta_{a,a_k}, \tag{5.3}$$

$$\sum_{a=1}^{n} \tau(a) = t. \tag{5.4}$$

Given this learning problem, the agent has the freedom to choose which action to sample next. This gives the agent the possibility to achieve optimal learning in a global sense. We define exploration as the agent's adaptive sampling behavior to achieve the minimal overall estimation error for the given learning task (Eq. 5.1).

More specifically, the agent has to find a policy $\pi_t(a)$ that minimizes the expected total mean squared error

$$E_L(t) = \sum_{a=1}^{n} \left\langle \left(\hat{\mu}_t(a) - \mu(a) \right)^2 \right\rangle, \tag{5.5}$$

where the expectation $\langle \cdot \rangle$ is with respect to the probability distribution of the reward history $H = \{r_1, r_2, \cdots, r_t\}$.

$$\langle x \rangle = \int p(r_1, r_2, \cdots, r_t) x \, dr_1 dr_2 \cdots dr_t \tag{5.6}$$

Since $\mu(a)$ are the main quantities of interest in learning, the total estimation error of the corresponding estimators $\hat{\mu}_t(a)$ tells how much the agent knows about the truth. Therefore, the total mean squared error (Eq. 5.5) is a measure of learning performance from the global viewpoint.

5.3 Exploration Policies in Multi-armed Bandits

The traditional data selection rule used in passive learning is random exploration. As the simplest undirected exploration strategy [Thr92], *random exploration* selects the next sample according to the uniform distribution over input space.

$$a_{t+1} \sim U(\mathcal{A}) \tag{5.7}$$

In reinforcement learning, random action selection is widely used as the exploration part of a policy. For example, ϵ-greedy policy switches from greedy action selection to random action selection with specified probability [Wat89; ACBF02].

In multi-armed bandits, counter-based exploration is *the least-taken policy*, which chooses the action that is visited least (ties are broken randomly)

$$a_{t+1} = \arg\min_a \tau(a) = \arg\max_a \frac{1}{\tau(a)}. \tag{5.8}$$

Here $\tau(a)$ is the number of times action a has been selected (Eq. 5.3). In this chapter, we use the convention that $1/0 = +\infty$. The least-taken policy adopts an even sample allocation scheme. It is the asymptotic policy of the random exploration when the fluctuation in random sampling averages out. In the multi-armed bandit, there is no neighborhood structure, therefore the least-taken policy is the same as the recency-based exploration policy [Sut90], where uncertainty is assumed to increase with respect to the time elapsed since the last trial of the action in a certain state.

Error-based exploration is used in the interval estimation method [Kae93; Fon95; MB99]. *The z-interval policy*, chooses an action according to the upper limit of the confidence interval for the reward mean,

$$a_{t+1} = \arg\max_a \left[\hat{\mu}_t(a) + z_\alpha \frac{\sigma(a)}{\sqrt{\tau(a)}} \right]. \tag{5.9}$$

Here z_α is a quantile of the standard normal distribution $\mathcal{N}(0,1)$. α is the tail probability of a standard normally distributed random variable x, so that z_α upper bounds x with probability $(1-\alpha)$ [FW86].

$$P(x \leq z_\alpha) = 1 - \alpha \tag{5.10}$$

According to Eq. 5.9, one action is selected either the reward estimation $\hat{\mu}_t(a)$ is large (exploitation), or the estimation error $\sigma(a)/\sqrt{\tau(a)}$ is large (exploration, see Eq. 5.15). z_α is a weight to combine exploitation and exploration. The

second term $\sigma(a)/\sqrt{\tau(a)}$ implements an error-based exploration policy, which is called *the z-interval exploration policy* hereafter.

$$a_{t+1} = \arg\max_a \frac{\sigma(a)}{\sqrt{\tau(a)}} = \arg\max_a \frac{\sigma^2(a)}{\tau(a)} \tag{5.11}$$

When the reward standard deviations $\sigma(a)$ are unknown, the confidence interval for reward mean is derived from the student distribution, resulting in *the* t-*interval policy* [MB99]

$$a_{t+1} = \arg\max_a \left[\hat{\mu}_t(a) + t_{\alpha,\tau(a)-1} \frac{\hat{\sigma}_t(a)}{\sqrt{\tau(a)}} \right]. \tag{5.12}$$

Here $t_{\alpha,\tau(a)-1}$ is a quantile of the student distribution $\mathcal{T}\big(x|\tau(a)-1\big)$ with $(\tau(a)-1)$ degrees of freedom. $t_{\alpha,\tau(a)-1}$ is an upper bound for the student distributed random variable x with probability $(1-\alpha)$ [FW86].

$$P\left(x \leq t_{\alpha,\tau(a)-1}\right) = 1 - \alpha \tag{5.13}$$

Using the same exploration-exploitation tradeoff scheme as in the z-interval policy, the t-interval policy uses the following exploration strategy (called *the* t-*interval exploration policy* hereafter)

$$a_{t+1} = \arg\max_a \frac{t_{\alpha,\tau(a)-1}}{z_\alpha} \frac{\hat{\sigma}_t(a)}{\sqrt{\tau(a)}}. \tag{5.14}$$

Here the term $t_{\alpha,\tau(a)-1}/z_\alpha$ approaches 1 from above as $\tau(a)$ tends to ∞.

Both the z-interval exploration policy (Eq. 5.11) and the t-interval exploration policy (Eq. 5.14) have similar forms. The least-taken policy (Eq. 5.8) can also be interpreted as an error-based policy with a common variance for all actions.

Gain-based exploration has been widely used in actively learning. However, in reinforcement learning, gain-based exploration strategies attract less attention. In the next section, we are going to derive a gain-based exploration policy for multi-armed bandits.

5.4 Optimal Exploration by Gain Maximization

In this section, we temporally assume that the reward variance $\sigma^2(a)$ is known. Under this ideal condition, an optimal exploration policy is derived from the gain-maximization principle. The realistic situation of a priori unknown variance is dealt with in section 5.5.

5.4.1 Performance Measure and Gain Definition

As stated in section 5.2, exploration is an active sampling process to minimize the total expected estimation error. Since the mean squared error of the estimator $\hat{\mu}_t(a)$ is

$$E(a,t) = \left\langle \left[\hat{\mu}_t(a) - \mu(a)\right]^2 \right\rangle = \frac{\sigma^2(a)}{\tau(a)}, \tag{5.15}$$

the exploration performance measure becomes

$$E_L(t) = \sum_{a=1}^{n} \frac{\sigma^2(a)}{\tau(a)}. \tag{5.16}$$

For each action selection, the reduction of mean squared error $\Delta E(a)$ is its contribution in reducing the total error.

$$\Delta E(a,t) = E(a,t) - E(a,t+1) = \frac{\sigma^2(a)}{\tau(a)[\tau(a)+1]} \tag{5.17}$$

We refer to the reduction in mean squared error $g_t(a) = \Delta E(a,t)$ as the gain of an action selection step. Obviously, $g_t(a)$ is a decreasing function with respect to $\tau(a)$.

5.4.2 Asymptotic Behavior of the Ideal Gain-maximization Exploration Policy

Since each action is independent, the only factor that influences the total error is the gain of the actions. The *ideal gain-maximization exploration policy* directly relies on the gain function $g_t(a)$ for action selection.

$$a_{t+1} = \arg\max_a \frac{\sigma^2(a)}{\tau(a)\,[\tau(a)+1]} \tag{5.18}$$

In order to understand the behavior of the ideal gain-maximization exploration, we analyze the sampling frequency of each action and the total mean squared error of the estimators.

It is convenient to employ a continuous approximation of $\tau(a)$ which allows us to express the gain function by the negative gradient of $E_L(t)$

$$g_t(a) = -\frac{\partial E_L(t)}{\partial \tau(a)} = \frac{\sigma^2(a)}{\tau^2(a)}. \tag{5.19}$$

The ideal gain-maximization exploration policy becomes

$$a_{t+1} = \arg\max_a \frac{\sigma^2(a)}{\tau^2(a)} = \arg\max_a \frac{\sigma(a)}{\tau(a)}. \tag{5.20}$$

Under the equilibrium condition of the ideal gain-maximization exploration policy (5.20), the gain of all actions is maintained approximately at the same level. Otherwise the action with the dominating gain will be selected and its gain will be reduced. The equilibrium condition leads to the ratio of the sampling numbers:

$$\frac{\tau(b)}{\tau(a)} = \frac{\sigma(b)}{\sigma(a)} \qquad \forall a, b \in \mathcal{A}. \tag{5.21}$$

The sampling number of an action is thus a fraction of the total samples proportional to the reward standard deviation (cf. Eq. 5.4)

$$\tau(a) = t\frac{\sigma(a)}{\sum_{b=1}^{n} \sigma(b)}, \tag{5.22}$$

and the visiting frequency of action a is

$$p^*(a) = \frac{\tau(a)}{t} = \frac{\sigma(a)}{\sum_{b=1}^{n} \sigma(b)}. \tag{5.23}$$

Inserting Eq. 5.22 into 5.16 we obtain the total mean squared error achieved by the ideal gain-maximization exploration policy

$$E_L^*(t) = \sum_{a=1}^{n} \frac{\sigma^2(a)}{\tau(a)} = \sum_{a=1}^{n} \frac{\sigma^2(a)}{t\frac{\sigma(a)}{\sum_{b=1}^{n} \sigma(b)}} = \frac{1}{t}\left(\sum_{a=1}^{n} \sigma(a)\right)^2. \tag{5.24}$$

5.4.3 Optimality of the Gain-maximization Exploration

The optimality of the ideal gain-maximization exploration policy can be proved by formulating exploration as a constrained minimization problem with respect to the total mean squared error defined earlier.

$$\text{minimize} \quad E_L(t) = \sum_{a=1}^{n} \frac{\sigma^2(a)}{\tau(a)} \quad \text{subject to} \quad \sum_{a=1}^{n} \tau(a) = t \tag{5.25}$$

Lagrange multipliers allow us to obey the constraint by defining

$$L = \sum_{a=1}^{n} \frac{\sigma^2(a)}{\tau(a)} + \lambda[\sum_{a=1}^{n} \tau(a) - t]. \tag{5.26}$$

The condition for optimal sampling number $\tau(a)$ is given by

$$\begin{aligned} \frac{\partial L}{\partial \tau(a)} &= -\frac{\sigma^2(a)}{\tau^2(a)} + \lambda = 0 \qquad \forall a \in \mathcal{A} \\ \frac{\partial L}{\partial \lambda} &= \sum_{a=1}^{n} \tau(a) - t = 0. \end{aligned} \tag{5.27}$$

This yields

$$\frac{\sigma^2(a)}{\tau^2(a)} = \frac{\sigma^2(b)}{\tau^2(b)} = \lambda \qquad \forall a, b \in \mathcal{A}. \tag{5.28}$$

Rearranging the terms, and summing with respect to b lead to the optimal sampling number $\tau^*(a)$

$$\tau^*(a) = t \frac{\sigma(a)}{\sum_{b=1}^{n} \sigma(b)}. \tag{5.29}$$

Since the second derivatives of the loss function L are non-negative for all $a \in \mathcal{A}$

$$\frac{\partial^2 L}{\partial \tau(a)^2} = \frac{2\sigma^2(a)}{\tau^3(a)} \geq 0, \qquad \frac{\partial^2 L}{\partial \lambda^2} = 0, \tag{5.30}$$

L is a convex function. Therefore, the loss function L achieves its global minimum at the critical point $\tau^*(a)$. Previous analysis in section 5.4.2 shows that the ideal gain-maximization exploration policy allocates samples to each action exactly the same as the optimal sampling number (5.29), therefore it is the optimal policy achieving the minimal total mean squared error (5.24) among all policies.

A policy can be derived from the ratio of the optimal sampling number $\tau^*(a)$ and the current sampling number $\tau(a)$

$$a_{t+1} = \arg\max_a \frac{t}{\tau(a)} \frac{\sigma(a)}{\sum_{b=1}^{n} \sigma(b)}, \tag{5.31}$$

which expresses the necessity of sampling one action due to the relative lack of samples in comparison to the optimal sampling number. Removing the common terms, it results in the same local exploration policy as the ideal gain-maximization exploration policy (Eq. 5.20).

The advantage of the ideal gain-maximization exploration policy can be shown by comparing the total mean squared error $E_L(t)$ with the least-taken policy and the z-interval exploration policy.

The equilibrium condition of the least-taken exploration policy is $\tau(a) = \tau(b)$, $\forall a, b \in \mathcal{A}$, and due to the constraint (5.4) the sampling number is given by

$$\tau(a) = \frac{t}{n}. \tag{5.32}$$

Putting the sampling number $\tau(a)$ into Eq. 5.16, we get the total mean squared error for the least-taken policy

$$E_L^e(t) = \sum_{a=1}^{n} \frac{\sigma^2(a)}{\frac{t}{n}} = \frac{n}{t} \sum_{a=1}^{n} \sigma^2(a). \tag{5.33}$$

For the z-interval exploration policy, its equilibrium condition is $\sigma(a)/\sqrt{\tau(a)} = \sigma(b)/\sqrt{\tau(b)}, \forall a, b \in \mathcal{A}$. This leads to different sampling numbers

$$\tau(a) = t \frac{\sigma^2(a)}{\sum_{b=1}^{n} \sigma^2(b)}, \tag{5.34}$$

but the same total mean squared error as that of the least taken policy

$$E_L^e(t) = \sum_{a=1}^{n} \frac{\sigma^2(a)}{t \frac{\sigma^2(a)}{\sum_{b=1}^{n} \sigma^2(b)}} = \frac{n}{t} \sum_{a=1}^{n} \sigma^2(a). \tag{5.35}$$

According to Jensen's inequality and the convexity of the quadratic function x^2, $E_L^e(t)$ is always larger or equal to $E_L^*(t)$, the total mean square error of the ideal gain-maximization exploration policy.

$$\begin{aligned} E_L^e(t) &= \frac{n}{t} \sum_{a=1}^{n} \sigma^2(a) = \frac{n^2}{t} \left(\frac{1}{n} \sum_{a=1}^{n} \sigma^2(a) \right) \\ &\geq \frac{n^2}{t} \left(\frac{1}{n} \sum_{a=1}^{n} \sigma(a) \right)^2 = \frac{1}{t} \left(\sum_{a=1}^{n} \sigma(a) \right)^2 = E_L^*(t). \end{aligned} \tag{5.36}$$

The equality in Eq. 5.36 holds only when the reward standard deviations $\sigma(a)$ are equal for all a. In this case, the three policies compared become the same.

Eq. 5.36 shows that for any particular choice of $\sigma(a)$, $E_L^*(t)$ is smaller than or equal to $E_L^e(t)$. In appendix B.1, $\sigma(a)$ is considered as a random variable, and the ratio of the expected total mean squared errors is shown to be bounded for independently and identically distributed $\sigma(a)$. The larger the variance of the standard deviation $\mathrm{Var}[\sigma(a)]$ is, the more superior is the ideal gain-maximization exploration policy to the other two policies. The largest performance ratio is equal to the number of actions n, when $\sigma(a)$ follows a distribution with infinite variance. In this case, estimators differ dramatically with respect to the estimation error, and provide a big potential for exploration strategies to reduce the total estimation error fast (see Fig. 5.3a).

5.5 Active Exploration by Gain Maximization

The ideal gain-maximization exploration policy (Eq. 5.20) assumes that the reward variance $\sigma^2(a)$ is available. However, for practical situations, the variance is unknown. A realistic policy should be derived using the online variance estimator $\hat{\sigma}_t^2(a)$.

When the sample size is small, the estimation $\hat{\sigma}_t^2(a)$ is not accurate enough to direct exploration optimally. Underestimation of the variance may occur and will result in insufficient exploration, thus hindering the increase of sample size for further improvement in estimation. In order to explore each action sufficiently, we search for an upper bound $u(a, \alpha, \tau(a))$, which bounds the true variance $\sigma^2(a)$ from above with desired probability $(1 - \alpha)$. Simultaneously, the upper bound approaches the true variance when more and more samples being available.

$$\Pr\left[\sigma^2(a) \leq u(a, \alpha, \tau(a))\right] = 1 - \alpha \tag{5.37}$$

$$\lim_{\tau(a)\to\infty} u\left(a, \alpha, \tau(a)\right) = \sigma^2(a) \tag{5.38}$$

Replacing the variance by this upper bound, an realistic gain-maximization exploration policy is

$$a_{t+1} = \arg\max_a \frac{u(a, \alpha, \tau(a))}{\tau^2(a)}. \tag{5.39}$$

The upper bound is an optimistic estimation of the variance. Compared to the ideal gain-maximization exploration policy, the realistic gain-maximization exploration policy pays more exploration in order to get the accurate variance information necessary for optimal exploration. This additional exploration aims at acquiring knowledge about how to explore, and can be called meta-exploration.

5.5.1 Confidence Interval for the Variance

According to the confidence interval method in statistics [FW86], the sum of the squares of independent standard normal variables has the Chi-square distribution. The probability density function of the Chi-square distribution with d degree of freedom is

$$f(x|d) = \frac{1}{\Gamma(\frac{d}{2})2^{d/2}} x^{\frac{d}{2}-1} e^{-\frac{x}{2}}, \qquad x \geq 0, \tag{5.40}$$

where $\Gamma(k), k > 0$ is the standard Gamma function.

Eq. 5.2 implies that

$$\frac{[\tau(a)-1]\,\hat{\sigma}_t^2(a)}{\sigma^2(a)} = \sum_{k=1}^{t}\left[\frac{r_k-\hat{\mu}_t(a)}{\sigma(a)}\right]^2 \delta_{a,a_k} \tag{5.41}$$

is a sum of squares of $\tau(a)$ standardized normal variables, therefore it is distributed according to a Chi-square distribution with $\tau(a)-1$ degrees of freedom.

We now select a quantile $\chi^2_{\alpha,\tau(a)-1}$ of the Chi-square distribution with $(\tau(a)-1)$ degrees of freedom and confidence level $(1-\alpha)$, so that $\chi^2_{\alpha,\tau(a)-1}$ acts as a lower bound for the Chi-square distributed random variable x with probability $(1-\alpha)$.

$$P\left(x \geq \chi^2_{\alpha,\tau(a)-1}\right) = 1-\alpha \tag{5.42}$$

Then, since $\frac{[\tau(a)-1]\hat{\sigma}_t^2(a)}{\sigma^2(a)}$ is Chi-square distributed with $(\tau(a)-1)$ degrees of freedom, it satisfies Eq. 5.42.

$$P\left(\frac{[\tau(a)-1]\hat{\sigma}_t^2(a)}{\sigma^2(a)} \geq \chi^2_{\alpha,\tau(a)-1}\right) = P\left(\sigma^2(a) \leq \frac{\tau(a)-1}{\chi^2_{\alpha,\tau(a)-1}}\hat{\sigma}_t^2(a)\right) = 1-\alpha. \tag{5.43}$$

Defining

$$m_{\alpha,\tau(a)-1} = \sqrt{\frac{\tau(a)-1}{\chi^2_{\alpha,\tau(a)-1}}}, \tag{5.44}$$

we obtain a confidence interval for $\sigma^2(a)$ with confidence level $(1-\alpha)$, i.e. $m^2_{\alpha,\tau(a)-1}\hat{\sigma}_t^2(a)$ is an upper bound of $\sigma^2(a)$ with probability $(1-\alpha)$.

5.5.2 Realistic Gain-maximization Exploration Policy

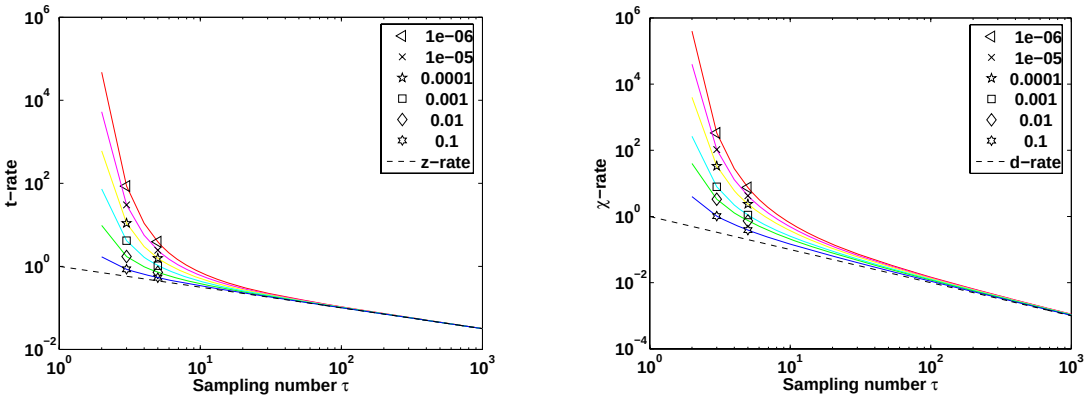
Using the confidence limit $m^2_{\alpha,\tau(a)-1}\hat{\sigma}_t^2(a)$ as an upper bound of the variance and taking the root, we get a *realistic gain-maximization exploration policy*

$$a_{t+1} = \arg\max_a \frac{m_{\alpha,\tau(a)-1}}{\tau(a)}\hat{\sigma}_t(a). \tag{5.45}$$

The realistic gain-maximization exploration policy (5.45) has a similar form as the z-interval exploration policy (Eq. 5.11), the t-interval exploration policy (5.14) and the ideal gain-maximization exploration policy (5.20). The essential difference among these policies is the amount of exploration indicted by the decay rate, cf. Table 5.1.

Policy name	$\pi_{t+1}(a)$	Decay rate
z-interval exploration policy	$a_{t+1} = \arg\max_a \frac{1}{\sqrt{\tau(a)}}\sigma(a)$	z-rate: $\frac{1}{\sqrt{\tau}}$
t-interval exploration policy	$a_{t+1} = \arg\max_a \frac{t_{\alpha,\tau(a)-1}/z_\alpha}{\sqrt{\tau(a)}}\hat{\sigma}_t(a)$	t-rate: $\frac{t_{\alpha,\tau-1}/z_\alpha}{\sqrt{\tau}}$
Ideal gain-maximization	$a_{t+1} = \arg\max_a \frac{1}{\tau(a)}\sigma(a)$	d-rate: $\frac{1}{\tau}$
Realistic gain-maximization	$a_{t+1} = \arg\max_a \frac{m_{\alpha,\tau(a)-1}}{\tau(a)}\hat{\sigma}_t(a)$	χ-rate: $\frac{m_{\alpha,\tau-1}}{\tau}$

Table 5.1: Comparison of error-based exploration strategies and gain-maximization exploration strategies. These strategies are a product of the reward standard deviation and a decay rate (shown in the third column). The gain-maximization strategies decay as $\frac{1}{\tau}$, faster than the decay rate of error-based strategies.



(a) (b)

Figure 5.1: Decay rates of the exploration for policies shown in Table 5.1. (a) The solid lines with markers are the t-rates for different parameter α. The dashed line is the z-rate, which is the limit of the t-rates. (b) The solid lines with markers are the χ-rates for different α. All χ-rates approach the d-rate, which is the dashed line. The amount of meta-exploration required for the t-interval exploration policy and the realistic gain-maximization exploration policy is controlled by the parameter α.

Fig. 5.1 plots the decay rates as functions of sampling number τ, with different confidence parameter α. The decay rate of the ideal gain-maximization exploration policy is $1/\tau$ (dashed line in Fig. 5.1b), and is faster than the decay rate $1/\sqrt{\tau}$ of the z-interval exploration policy (dashed lines in Fig. 5.1a).

As shown in Fig. 5.1, t-rates and χ-rates are larger than z-rate and d-rate respectively, since their numerators $t_{\alpha,\tau-1}/z_\alpha$ and $m_{\alpha,\tau-1}$ approach 1 from above. This additional exploration is the meta-exploration needed to get accurate variance estimations required in the exploration policies. As indicated by the curves of the t-rates and the χ-rates, the parameter α controls how fast the t-interval exploration policy (Eq. 5.14) and the realistic gain-maximization exploration policy (Eq. 5.45) approach their asymptotic policies respectively. The more amount of meta-exploration is required, the smaller the parameter α should be. However, α should always be smaller than 0.5. Otherwise z_α and $t_{\alpha,\tau-1}$ will become non-positive, and $m_{\alpha,\tau-1}$ will be smaller than 1, indicating insufficient exploration.

It should be noted that the amount of exploration of the optimal Bayesian policy decays as $\frac{1}{\tau}$ [Git89; MB99]. The gain-maximization policies derived here in a non-Bayesian setting have the same decaying rate of exploration as in a Bayesian setting.

5.6 Simulations

Exploration policies are compared on randomly generated 100-armed bandits. The reward distribution of each arm is a normal distribution $p(r|a) \sim N\left(\mu(a), \sigma^2(a)\right)$. In order to compare the average performance, the reward mean $\mu(a)$ and standard deviation $\sigma(a)$ are independently generated from two kinds of probability distributions, namely from the power law distribution and the uniform distribution. Each policy is tested on 100 random instances of 100-armed bandit. To average out the fluctuation, each policy is tested for 100 runs on the same bandit. In total, each policy is simulated for 10^4 runs.

Simulation is done as follows. For the initial 100 steps of each run, each arm of the bandit is sampled once in a random order, so that an estimation $\hat{\mu}_t(a)$ of the mean is available to calculate the average estimation error. Additionally, for the realistic gain-maximization exploration policy and the t-interval exploration policy, each arm is sampled once again for a second round, in order to get the variance estimation $\hat{\sigma}_t^2(a)$ required in these policies. After the initial stage, all simulations follow the policies as normal.

The learning performances of six exploration policies are compared in terms

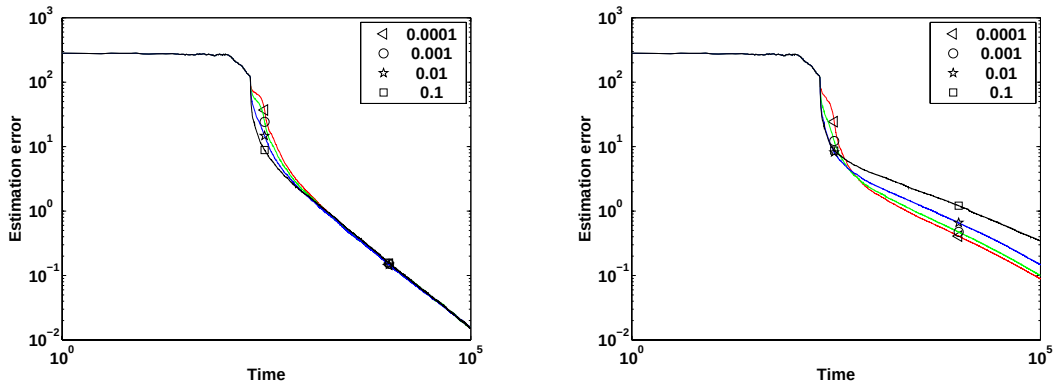


Figure 5.2: Sensitivity of the realistic gain-maximization exploration policy and the t-interval exploration policy with respect to the parameter α when $\mu(a)$ and $\sigma(a)$ are power-law distributed. (a) The performance of the realistic gain-maximization exploration policy is slightly dependent on the parameter α. For large α, the asymptotic performance is worse due to under-estimation of variances caused by insufficient meta-exploration in the beginning. (b) For the t-interval exploration policy, the asymptotic performance is influenced more strongly by α ($0 < \alpha < 0.5$).

of the averaged estimation error

$$\bar{e}_t = \left\langle \frac{\sum_{a=1}^{n} \left(\hat{\mu}_t(a) - \mu(a)\right)^2}{2} \right\rangle.$$

Here $\langle \cdot \rangle$ represents the average over 10^4 runs, and $\hat{\mu}_t(a)$ are initialized as 0.

The six policies compared are: the random exploration policy (5.7), the least-taken policy (5.8), the z-interval exploration policy (5.11), the t-interval exploration policy (5.14), the ideal gain-maximization exploration policy (5.20), and the realistic gain-maximization exploration policy (5.45).

5.6.1 Power Law Distributed Parameters

In the first simulation, the reward means $\mu(a)$ and standard deviations $\sigma(a)$ are both generated independently from a power law distribution given by

$$p(x) = Cx^{-2.5}, \; x \geq \frac{1}{6}, \tag{5.46}$$

with the normalization constant $C = 1.5 \times 6^{-1.5}$. Following this power law distribution, the minimal value of $\mu(a)$ and $\sigma(a)$ is $1/6$. The expectation values $\langle \mu(a) \rangle$ and $\langle \sigma(a) \rangle$ are both $1/2$.

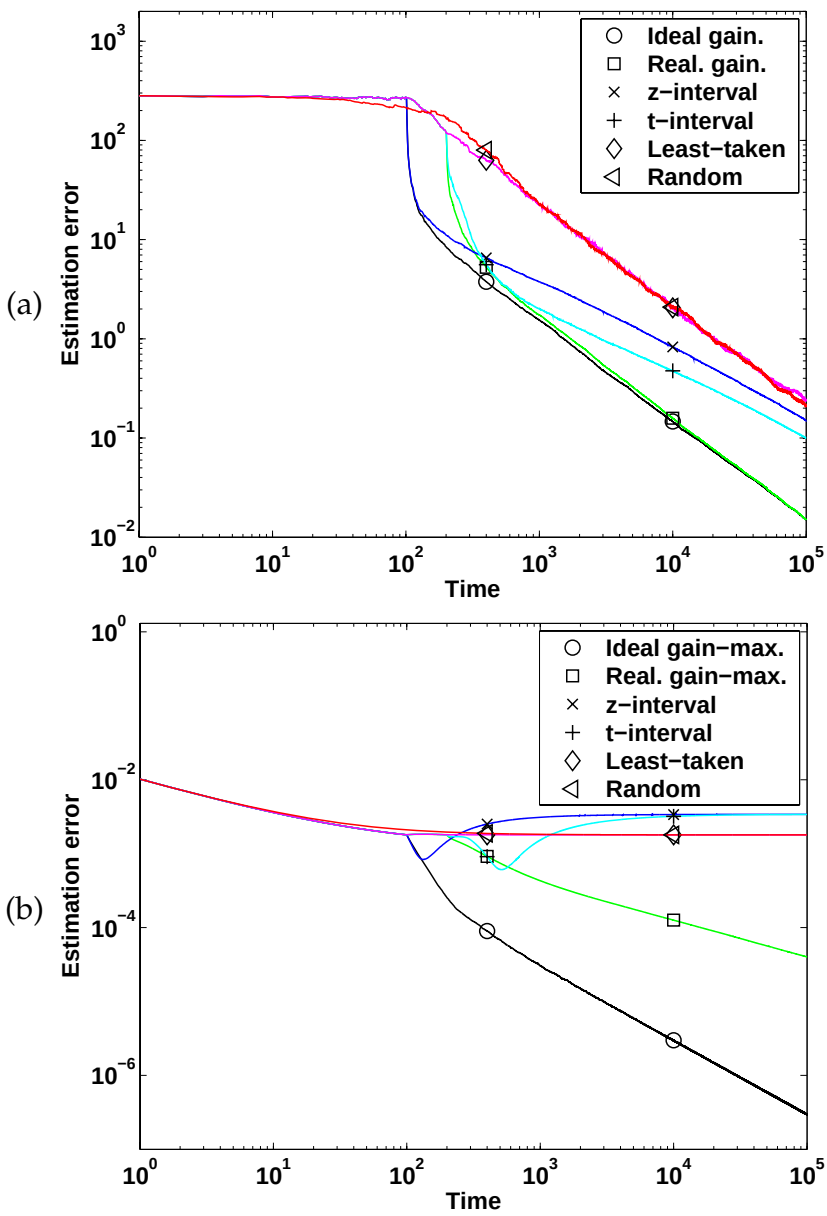


Figure 5.3: Comparison of exploration strategies with power-law distributed $\mu(a)$ and $\sigma(a)$. (a) Mean estimation error $\bar{e}_t$. The ideal gain-maximization exploration policy (circle marker) is about 10 times better than the least-taken policy (diamond marker). The realistic gain-maximization exploration policy ($\alpha = 0.1$, square marker) is always better than the t-interval exploration policy ($\alpha = 0.001$, $+$ marker), and outperforms the z-interval exploration policy ($\times$ marker) after 300 steps. (b) Difference $\bar{d}_t$ between the empirical visiting frequency and the optimal visiting frequency. $\alpha = 0.1$ for the realistic gain-maximization exploration policy and $\alpha = 0.001$ for the t-interval exploration policy. Gain-based explorations (circle and square markers) approach the optimal exploration behavior. The random exploration (triangle), the least-taken exploration (diamond) and error-based explorations ($\times$ and $+$ markers) stay different from the optimal exploration behavior.

Figure (5.2) shows the curves of the averaged estimation error $\bar{e}_t$ of the realistic gain-maximization exploration policy and the t-interval exploration policy with different error parameter α. In the beginning, the performances are better for larger α. However the asymptotic performances are worse for large α. This is because the meta-exploration is small for large α. Insufficient meta-exploration may result in under-estimation of the variances, so that these arms may not receive enough samples to reduce their estimation errors. The t-interval exploration policy is more vulnerable to the change of α.

Fig. 5.3a gives a performance comparison of the six strategies. The ideal gain-maximization exploration policy (circle marker) is the best, and the random exploration (triangle marker) is the worst. The performance of the least-taken policy (diamond marker) is similar as that of random exploration. Both the z-interval exploration policy ($\times$ marker) and the t-interval exploration policy ($+$ marker) are better than the least-taken policy in the beginning, but they approach the least-taken policy in the end. The realistic gain-maximization exploration policy ($\alpha = 0.1$, square marker) is better than the t-interval exploration policy ($\alpha = 0.001$). It is also better than curves of the t-interval exploration policy with other tested parameters (Fig.5.2b). Already after an average of about 3 samples per action, the realistic gain-maximization exploration policy outperforms the z-interval exploration policy, which requires known reward variances.

The performance of the ideal gain-maximization exploration policy is about 10 times better than the performance of the least-taken exploration. As analyzed in Appendix B.1.1, when the reward standard deviation follows a power law distribution with the exponent in the range $(2, 3]$, the reward standard deviation has a large variance. In this case, the mean estimation errors of the estimators vary a lot, therefore it is very beneficial to sample each action carefully in order to get the maximal gain out of each sample. Sampling strategies that are different from the ideal gain-maximization exploration will be much worse.

Fig. 5.3b shows how different the empirical visiting frequencies of each policies are as compared with the optimal visiting frequency $p^*(a)$ (Eq. 5.23). The averaged squared distance between the empirical and the optimal visiting frequency is denoted by $\bar{d}_t$

$$\bar{d}_t = \left\langle \frac{\sqrt{\sum_{a=1}^{n} \left(p^*(a) - p_t(a)\right)^2}}{n} \right\rangle. \tag{5.47}$$

Here $\langle\cdot\rangle$ represents the average over 10^4 runs, and $p_t(a)$ is the empirical visiting frequency given by $p_t(a) = \frac{1}{t}\sum_{k=1}^{t} \delta_{a,a_k}$.

After the initial phase, the distance between the optimal visiting frequency and the empirical visiting frequency of the ideal gain-maximization explo-

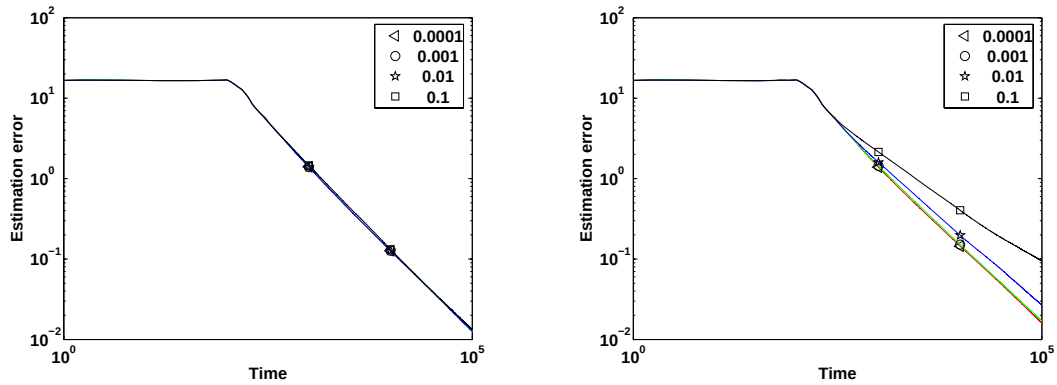


Figure 5.4: The dependency of the realistic gain-maximization exploration policy and the t-interval exploration policy on the parameter α with uniformly distributed $\mu(a)$ and $\sigma(a)$. (a) The realistic gain-maximization exploration policy is robust against the change of α. (b) For the t-interval exploration policy, the smaller the α ($0 < \alpha < 0.5$) is, the better the asymptotic performance is.

ration decreases toward zero (circle marker), showing that the ideal gain-maximization exploration approaches the optimal visiting behavior. The realistic gain-maximization exploration (square marker) takes longer time to approach the optimal action selection behavior. For other exploration strategies, their visiting frequencies stay different from the optimal visiting frequency.

5.6.2 Uniformly Distributed Parameters

We also test the six strategies with reward means $\mu(a)$ and standard deviations $\sigma(a)$ distributed uniformly in the range $[0, 1]$. The expectation values $\langle\mu(a)\rangle$ and $\langle\sigma(a)\rangle$ are both $1/2$, the same as in the power law distribution used in the first simulation.

Figure (5.4) depicts the curves of the averaged estimation error $\bar{e}_t$ of the realistic gain-maximization exploration policy and the t-interval exploration policy for different α. The realistic gain-maximization exploration is robust against the change of α in a large range. The performance of the t-interval exploration policy deteriorates quickly with increasing α.

Figure (5.5) compares the learning performance of different policies. The ideal gain-maximization exploration (circle marker) is the best among the six strategies. In the beginning, the realistic gain-maximization exploration (square marker, $\alpha = 0.01$) is almost as good as the t-interval exploration policy (

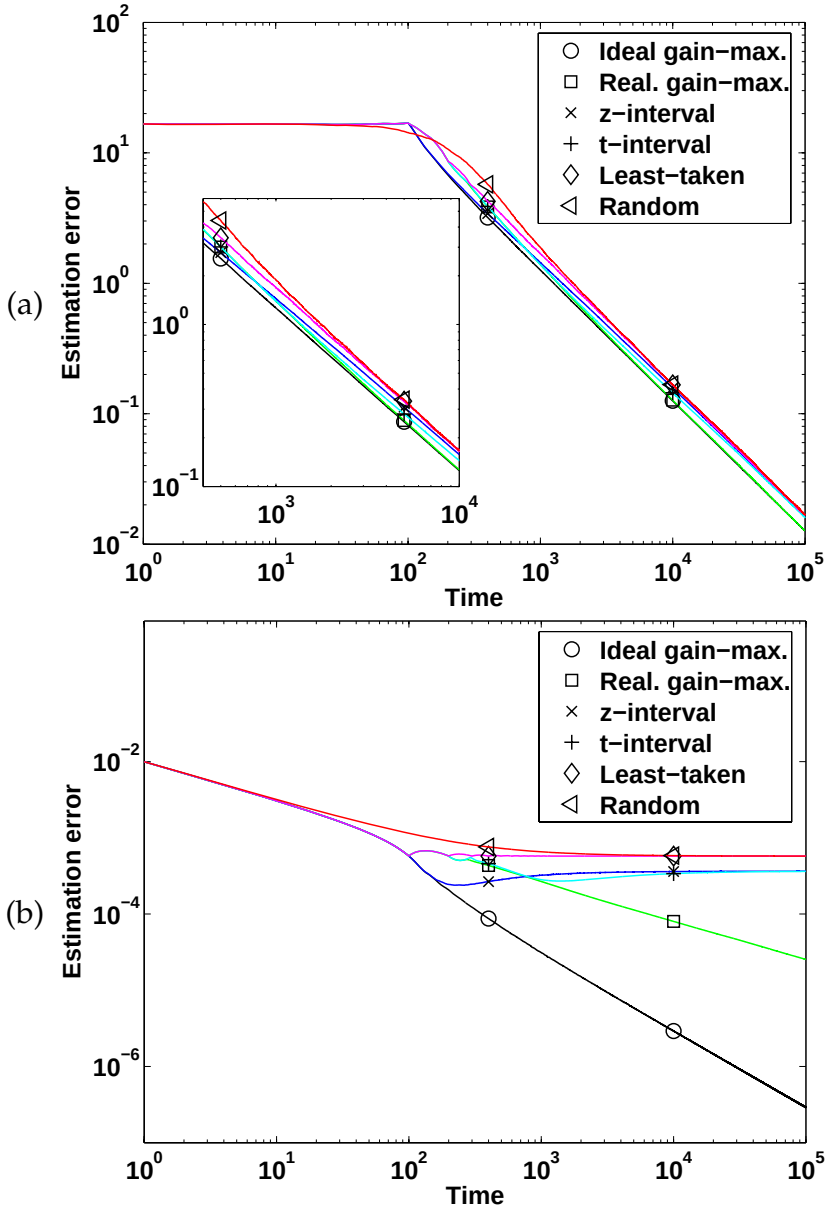


Figure 5.5: Comparison of exploration strategies with uniformly distributed $\mu(a)$ and $\sigma(a)$. (a) Mean estimation error $\bar{e}_t$. The parameter α is 0.01 for the realistic gain-maximization exploration policy, and $\alpha = 0.0001$ for the t-interval exploration policy. The performance ratio between the least-taken exploration (diamond marker) and the ideal gain-maximization exploration (circle marker) is about 1.3. (b) Difference $\bar{d}_t$ between the empirical visiting frequency and the optimal visiting frequency. For the realistic gain-maximization exploration policy, $\alpha = 0.01$. For the t-interval exploration policy, $\alpha = 0.0001$. The differences $\bar{d}_t$ for the random exploration, the least-taken exploration, and error-based exploration are smaller than those in Fig. 5.3b.

$\alpha = 0.0001$, + marker). But after 1000 steps, it outperforms both the t-interval exploration policy and the z-interval exploration policy, and approaches the ideal gain-maximization exploration policy. The performances of the t-interval exploration policy and the z-interval policy approach those of the least-taken exploration and the random exploration, which yields the worst performance among the six strategies.

In this simulation, the performance of the ideal gain-maximization exploration is about 1.3 times as good as the performance of the least-taken exploration. This simulation result is consistent with the performance ratio calculated in Appendix B.1.2. With uniformly distributed reward standard deviations, the advantage of the gain-maximization exploration is not significant, because the variance of the reward standard deviations $\sigma(a)$ is small. The uncertainty of each action does not differ much, and the mean error reduction of the estimator does not provide enough variation for the gain-maximization exploration to exploit.

Chapter 6

A Unified View on Exploration

Motivated by different principles, exploration strategies appear rather different. However, they are all error minimization processes. From this viewpoint, we show that the counter-based, error-based and gain-based exploration policies are all gain-based strategies, minimizing respective cases of a generalized error function.

6.1 Generalized Performance Measure

The objective function $E_L(t)$ (Eq.5.16) is the 1-norm of the mean estimation error vector $\Big[E(1,t), \cdots, E(a,t), \cdots, E(n,t)\Big]$. We consider its β-norm as a generalized error function

$$G_L(t) = \Big(\sum_{a=1}^{n} \frac{\sigma^{2\beta}(a)}{\tau^{\beta}(a)}\Big)^{\frac{1}{\beta}}, \quad \beta \in [0, \infty]. \tag{6.1}$$

When performing action a, the error reduction or the negative gradient $-\frac{\partial G(t)}{\partial \tau(a)}$ is the gain of one step action selection

$$g_t(a) = -\frac{\partial G_L(t)}{\partial \tau(a)} = \Big(\sum_{b=1}^{n} \frac{\sigma^{2\beta}(b)}{\tau^{\beta}(b)}\Big)^{\frac{1}{\beta}-1} \frac{\sigma^{2\beta}(a)}{\tau^{\beta+1}(a)}. \tag{6.2}$$

Note that Eq. 6.2 is based on the assumption that actions are independent and sampling numbers $\tau(a)$ are continuous, which will be made throughout this work.

Directly maximizing the gain $g_t(a)$ and neglecting the common term for all a,

Principle	β	$\tau(a)$	Policy $\pi_{t+1}(a)$	Total MSE $E_L(t)$
Even resource allocation	0	t/n	$a_{t+1} = \arg\max_a \frac{1}{\tau(a)}$	$\frac{n}{t}\sum_{a=1}^{n}\sigma^2(a)$
Total error minimization	1	$t\frac{\sigma(a)}{\sum_{b=1}^{n}\sigma(b)}$	$a_{t+1} = \arg\max_a \frac{\sigma(a)}{\tau(a)}$	$\frac{1}{t}\left(\sum_{a=1}^{n}\sigma(a)\right)^2$
Local error minimization	∞	$t\frac{\sigma^2(a)}{\sum_{b=1}^{n}\sigma^2(b)}$	$a_{t+1} = \arg\max_a \frac{\sigma^2(a)}{\tau(a)}$	$\frac{n}{t}\sum_{a=1}^{n}\sigma^2(a)$

Table 6.1: The exponent β of the general performance measure. Different principles have different values of β. The minimal total mean squared error is achieved when $\beta = 1$, which corresponds to the total error minimization principle.

results in a *general gain-maximization exploration strategy*

$$a_{t+1} = \arg\max_a \frac{\sigma^{2\beta}(a)}{\tau^{\beta+1}(a)} = \arg\max_a \frac{\sigma^{\frac{2\beta}{\beta+1}}(a)}{\tau(a)}. \tag{6.3}$$

In the equilibrium condition, the action selection indexes $\frac{\sigma^{\frac{2\beta}{\beta+1}}(a)}{\tau(a)}$ are kept at the same level, i.e. $\frac{\sigma^{\frac{2\beta}{\beta+1}}(a)}{\tau(a)} = \frac{\sigma^{\frac{2\beta}{\beta+1}}(b)}{\tau(b)}$, $\forall a, b \in \mathcal{A}$, which gives the solution of the sampling number

$$\tau^*(a) = t\frac{\sigma^{\frac{2\beta}{\beta+1}}(a)}{\sum_{b=1}^{n}\sigma^{\frac{2\beta}{\beta+1}}(b)}, \tag{6.4}$$

where t denotes the total number of samples up to now.

Table 6.1 gives a summary of the general gain-maximization exploration strategy (Eq.6.3) with $\beta = 0, 1$, and ∞. When the value of β is set to 0, the generalized performance measure (Eq.6.1) becomes the geometric mean of the estimation errors $\left(\prod_{a=1}^{n}\frac{\sigma^2(a)}{\tau(a)}\right)^{\frac{1}{n}}$. Since the variance product $\prod_{a=1}^{n}\sigma^2(a)$ is fixed for a specific bandit, the gain function is proportional to $1/\tau(a)$, resulting in the least-taken policy (Eq.5.8) and even sampling of every action.

When β increases then large estimation errors are weighted more than small ones due to the non-linearity of the power function. At $\beta \to \infty$ the general measure becomes the infinity norm $\max\left(\frac{\sigma^2(1)}{\tau(1)}, \frac{\sigma^2(2)}{\tau(2)}, \cdots, \frac{\sigma^2(n)}{\tau(n)}\right)$, which is a winner-take-all selection rule of the local estimation errors. Under this situation, the gain function vanishes for all actions except the one with the maximal mean squared error. The resulting policy focuses on local error minimization.

This error-based policy allocates samples proportional to the reward variances, as does the z-interval exploration policy (Eq.5.11).

When β is equal to 1, the general measure becomes the objective function $E_L(t)$ (Eq.5.16). The gain function becomes the negative gradient of mean squared error with respect to $\tau(a)$. The resulting ideal gain-maximization exploration policy allocates samples proportional to the reward standard deviations, and is optimal in minimizing the total mean squared error.

The fifth column of Table 6.1 gives a comparison of the total mean squared error for the three cases of the general gain-maximization strategy. It is derived by putting Eq. 6.4 into objective function $E_L(t)$

$$\begin{aligned} E_L^G(t) &= \sum_{a=1}^{n} \frac{\sigma^2(a)}{t\frac{\sigma^{\frac{2\beta}{\beta+1}}(a)}{\sum_{b=1}^{n}\sigma^{\frac{2\beta}{\beta+1}}(b)}} \\ &= \frac{1}{t}\sum_{b=1}^{n}\sigma^{\frac{2\beta}{\beta+1}}(b)\sum_{a=1}^{n}\sigma^{\frac{2}{\beta+1}}(a). \end{aligned} \tag{6.5}$$

6.2 Optimality of the Unified Exploration Policy

The general gain-maximization exploration strategy is optimal in minimizing the generalized performance measure (Eq.6.1), since its sampling number is the optimal solution to to following constrained minimization problem.

$$\text{minimize} \quad G_L(t) = \Big(\sum_{a=1}^{n}\frac{\sigma^{2\beta}(a)}{\tau^{\beta}(a)}\Big)^{\frac{1}{\beta}}, \ \text{subject to} \ \sum_{a=1}^{n}\tau(a) = t \tag{6.6}$$

According to the method of Lagrange multipliers, the Lagrangian function is

$$L = \Big(\sum_{a=1}^{n}\frac{\sigma^{2\beta}(a)}{\tau^{\beta}(a)}\Big)^{\frac{1}{\beta}} + \lambda\Big[\sum_{a=1}^{n}\tau(a) - t\Big]. \tag{6.7}$$

The condition for the optimal sampling number is when the gradients of the general performance measure are equal for all actions

$$\begin{aligned} \frac{\partial L}{\partial \tau(a)} &= -\Big(\sum_{b=1}^{n}\frac{\sigma^{2\beta}(b)}{\tau^{\beta}(b)}\Big)^{\frac{1}{\beta}-1}\frac{\sigma^{2\beta}(a)}{\tau^{\beta+1}(a)} + \lambda = 0 \qquad \forall a \in \mathcal{A} \\ \frac{\partial L}{\partial \lambda} &= \sum_{a}\tau(a) - t = 0. \end{aligned} \tag{6.8}$$

This results in the sampling number $\tau^*(a)$ in Eq.6.4. Since L is a convex function, the general gain-maximization exploration strategy derived from $\tau^*(a)$ is the optimal policy minimizing respective norms of the mean squared errors.

6.3 Probability Measure of Exploration Performance

Our previous analysis is focused on the total mean squared error $E_L(t)$, which is the expected quadratic loss of the estimators. In this section, we will examine the exploration policies with respect to a probability measure, namely, the total probability of wrong pairwise ordering.

Probability is the likelihood for a event to occur. Since the pairwise orderings are decisions inferred from learning, the probability of erring measures the performance of these decisions in a natural way. The sample complexity derived under the probability measure specifies the number of samples necessary before the algorithm stops and returns a correct answer with expected probability guarantee [Val84]. For example in [EDMM02; MT04], the sample complexities of exploration policies in multi-armed bandits were established with the probability of wrong selection as performance measure. In their cases, the purpose of exploration is reward maximization, therefore the probability of wrong selection is an appropriate measure. In this section, the purpose of exploration is optimal learning. Therefore, the appropriate probability measure is the total probability of wrong pairwise ordering, which is the expected $0/1$ loss. Based on this measure, we give an upper bound on the sample complexity of the general gain-maximization exploration strategy. In the next chapter, the exploration for the purpose of reward maximization is analyzed, and the sample complexity of gain-based exploration is established.

6.3.1 Expected 0/1 Loss

Without loss of generality, we assume that the true reward means are ordered as $\mu(1) \leq \mu(2) \leq \cdots \leq \mu(n)$. The differences between reward means are denoted as $\epsilon_{ik} = \mu(k) - \mu(i), 1 \leq i < k \leq n$. Since all pairs are of interest for learning, an appropriate probability measure is the total probability of wrong pairwise ordering, which is shown to be the total expected $0/1$ loss of $\hat{\epsilon}_{ik} = \hat{\mu}_t(k) - \hat{\mu}_t(i)$.

$$M_L(t) = \sum_{1 \le i < k \le n} P\Big(\hat{\mu}_t(k) - \hat{\mu}_t(i) < 0\Big) \tag{6.9}$$

$$= \sum_{1 \le i < k \le n} \Big\langle \Theta\Big(\hat{\mu}_t(i) - \hat{\mu}_t(k)\Big) \Big\rangle, \tag{6.10}$$

where $\Theta(\cdot)$ is the step or Heaviside function

$$\Theta(x) = \begin{cases} 1, & \text{if } x > 0 \\ 0, & \text{if } x \le 0 \end{cases} . \tag{6.11}$$

The 0/1-loss $\Theta\Big(\hat{\mu}_t(i) - \hat{\mu}_t(k)\Big)$ indicates a wrong pairwise ordering $\hat{\mu}_t(i) > \hat{\mu}_t(k)$ of the reward means (we assume that $\mu(i) \le \mu(k), 1 \le i < k \le n$). While $M_L(t)$ is the total expected 0/1 loss of the difference estimators $\hat{e}_{ik}$, our previous measure $E_L(t)$ is equivalent to the total expected quadratic loss of $\hat{e}_{ik}$.

$$\begin{aligned}
&\sum_{1 \le i < k \le n} \Big\langle \Big[\Big(\hat{\mu}_t(k) - \hat{\mu}_t(i)\Big) - \Big(\mu(k) - \mu(i)\Big)\Big]^2 \Big\rangle \\
&= \sum_{1 \le i < k \le n} \Big[\Big\langle \Big(\hat{\mu}_t(k) - \mu(k)\Big)^2 \Big\rangle + \Big\langle \Big(\hat{\mu}_t(i) - \mu(i)\Big)^2 \Big\rangle \Big] \\
&= (n-1) \sum_{a=1}^{n} \Big\langle \Big(\hat{\mu}_t(a) - \mu(a)\Big)^2 \Big\rangle \\
&= (n-1) E_l(t).
\end{aligned} \tag{6.12}$$

Eq.6.12 shows the connection between $E_L(t)$ and $M_L(t)$. Both are total expected errors of the reward difference estimators $(\hat{\mu}_t(k) - \hat{\mu}_t(i))$. $E_L(t)$ is based on quadratic loss, and is a smooth function. $M_L(t)$ is based on 0/1 loss, which is not smooth.

6.3.2 Upper Bound on the Sample Complexity

Based on the total probability of wrong pairwise ordering $M_L(t)$, we analyze the upper bound on the sample complexity of the unified strategy(Eq.6.3). Then we compare the upper bounds of three special cases, i.e. the ideal gain-maximization exploration policy, the least-taken policy, and the z-interval exploration policy.

We follow the PAC (Probably Approximately Correct) type of analysis [Val84; MT04; EDMM06]. The basic idea of PAC algorithms is to trade the performance guarantee (learning error) for learning efficiency (sample size). A

wrong pairwise ordering occurs when $\hat{\mu}_t(k) < \hat{\mu}_t(i)$, $1 \leq i < k \leq n$. However, if action i is not too worse than action k, then it is satisfactory to consider them approximately the same. Relaxing the accuracy of learning will require less samples.

More specifically, the actions whose reward means μ fall in the indifference zone of action k, i.e. $\mu > \mu(k) - \epsilon$, are defined to be indifferent from action k (ϵ is a positive number) [GOS79]. For any positive number $\epsilon > 0$, suppose action i is out of the indifference zone of action k, then

$$\epsilon_{ik} = \mu(k) - \mu(i) \geq \epsilon. \tag{6.13}$$

Then the problem is to determine a stopping time for an exploration strategy so that the total probability of wrong pairwise ordering is bounded when all the actions i fall out of the indifference zone of action k ($1 \leq i < k \leq n$).

The analysis in this and the next subsection does not assume normal distribution except the symmetry of the probability density function around the mean. Therefore, the result applies for a wide range of distributions. Similar analysis can also be applied to more special distributions, e.g. normal distribution or bernoulli distribution, to get better bounds.

First of all, we introduce a preliminary result about the upper bound on the probability of one wrong pairwise ordering.

Lemma 6.3.1. *Let $k > i$, and $\mu(k) \geq \mu(i)$ be the means of two distributions. Their distance is denoted as $\epsilon_{ik} = \mu(k) - \mu(i)$. The estimations of $\mu(k)$ and $\mu(i)$ are $\hat{\mu}_t(k)$ and $\hat{\mu}_t(i)$ respectively. Then for any free variable $\zeta_{ik} \in [0, \epsilon_{ik}]$,*

$$P\Big(\hat{\mu}_t(k) < \hat{\mu}_t(i)\Big) \leq P\Big(\mu(i) + \zeta_{ik} \leq \hat{\mu}_t(i)\Big) + P\Big(\hat{\mu}_t(k) \leq \mu(k) - (\epsilon_{ik} - \zeta_{ik})\Big). \tag{6.14}$$

The proof of Lemma 6.3.1 is standard, and is listed in Appendix C.1 for completeness. Lemma 6.3.1 separates the probability of one wrong pairwise ordering into the probabilities of estimation errors. It says that the probability of a wrong pairwise ordering is bounded by the total probability of overestimating of $\mu(i)$ by ζ_{ik} and underestimating of $\mu(k)$ by $(\epsilon_{ik} - \zeta_{ik})$. The distance $\mu(k) - \mu(i)$ is the total error the two estimators can make without causing wrong ordering decisions. Combined with appropriate probability inequalities, the right side of Eq.6.14 can be bounded tightest by choosing appropriate value for ζ_{ik}.

Previous results are mainly based on Hoeffding inequality by the merit of Bernoulli type reward assumptions[Fon95; EDMM02; MT04]. Since we don't have this assumption, we derive an upper bound of the sample complexity according to Chebyshev Inequality. The upper bound $T_L(\beta)$ on the sample complexity of the unified policy is given by the following Theorem.

Theorem 6.3.2. *Let $\epsilon > 0$ and $\delta \in (0,1)$ be two small positive numbers. $\hat{\mu}_t(i)$ are the empirical mean (ref. Eq.5.1) of any distribution, whose probability density function is symmetric around the true mean. Then it takes at least $T_L(\beta) = \frac{\sum_{j=1}^{n} \sigma^{\frac{2\beta}{\beta+1}}(j)}{2\delta\epsilon^2} \sum_{1\le i<k\le n} \left(\sigma^{\frac{2}{3(\beta+1)}}(k) + \sigma^{\frac{2}{3(\beta+1)}}(i)\right)^3$ samples for the sampling scheme Eq.6.4 to have the total probability of wrong pairwise ordering $M_L(t)$ smaller than δ, whenever the distance between any two means is significantly large, i.e. $\mu(k)-\mu(i) \ge \epsilon$, for $1 \le i < k \le n$.*

Proof. Since $\mu(k) - \mu(i) \ge \epsilon, \forall 1 \le i < k \le n$, we have

$$\epsilon \le \min_{1\le i<k\le n} \epsilon_{ik} = \min_{1\le i<k\le n} \big(\mu(k) - \mu(i)\big). \tag{6.15}$$

The total probability of wrong pairwise ordering is bounded by

$$\begin{aligned}
M_L(t) &= \sum_{1\le i<k\le n} P\Big(\hat{\mu}_t(k) < \hat{\mu}_t(i)\Big) \\
&\le \sum_{1\le i<k\le n} \Big[P\Big(\hat{\mu}_t(i) \ge \mu(i) + \zeta_{ik}\Big) + P\Big(\hat{\mu}_t(k) \le \mu(k) - (\epsilon_{ik} - \zeta_{ik})\Big)\Big] && (6.16)\\
&\le \frac{1}{2} \sum_{1\le i<k\le n} \Big[\frac{\sigma^2(k)}{(\epsilon_{ik} - \zeta_{ik})^2 \tau(k)} + \frac{\sigma^2(i)}{\zeta_{ik}^2 \tau(i)}\Big] && (6.17)\\
&= \frac{\sum_{j=1}^{n} \sigma^{\frac{2\beta}{\beta+1}}(j)}{2t} \sum_{1\le i<k\le n} \Big[\frac{\sigma^{\frac{2}{\beta+1}}(k)}{(\epsilon_{ik} - \zeta_{ik})^2} + \frac{\sigma^{\frac{2}{\beta+1}}(i)}{\zeta_{ik}^2}\Big] && (6.18)\\
&= \frac{\sum_{j=1}^{n} \sigma^{\frac{2\beta}{\beta+1}}(j)}{2t} \sum_{1\le i<k\le n} \frac{1}{\epsilon_{ik}^2} \Big(\sigma^{\frac{2}{3(\beta+1)}}(k) + \sigma^{\frac{2}{3(\beta+1)}}(i)\Big)^3 \\
&\le \frac{\sum_{j=1}^{n} \sigma^{\frac{2\beta}{\beta+1}}(j)}{2t\epsilon^2} \sum_{1\le i<k\le n} \Big(\sigma^{\frac{2}{3(\beta+1)}}(k) + \sigma^{\frac{2}{3(\beta+1)}}(i)\Big)^3 \\
&\le \delta && (6.19)
\end{aligned}$$

The inequality 6.16 is due to Lemma 6.3.1. Step 6.17 is obtained by applying Chebyshev Inequality (Eq.C.7). In Eq.6.18, ζ_{ik} is selected as

$$\zeta_{ik}^* = \epsilon_{ik} \frac{\sigma^{\frac{2}{3(\beta+1)}}(i)}{\sigma^{\frac{2}{3(\beta+1)}}(i) + \sigma^{\frac{2}{3(\beta+1)}}(k)}, \tag{6.20}$$

so that the quantity $\frac{\sigma^{\frac{2}{\beta+1}}(k)}{(\epsilon_{ik}-\zeta_{ik})^2} + \frac{\sigma^{\frac{2}{\beta+1}}(i)}{\zeta_{ik}^2}$ is minimized (shown in Appendix C.4), resulting in the tightest upper bound. ζ_{ik}^* sections the distance $\mu(k) - \mu(i)$ according to reward variances.

Then from Eq.6.19, the total probability of wrong pairwise ordering is at most δ, when the total number of samples t is at least

$$t \geq \frac{\sum_{j=1}^{n} \sigma^{\frac{2\beta}{\beta+1}}(j)}{2\delta\epsilon^2} \sum_{1\leq i<k\leq n} \left(\sigma^{\frac{2}{3(\beta+1)}}(k) + \sigma^{\frac{2}{3(\beta+1)}}(i)\right)^3.$$

□

It is possible to directly minimize an upper bound of the total probability of wrong pairwise ordering (e.g. Eq.6.17) to get an optimal sampling policy in the sense of probability. However, this depends strongly on the forms of the probability density functions of the reward distributions. Even for general bounds like Eq.6.17, there are still many correlated nonlinear terms, such as $(\epsilon_{ik}-\zeta_{ik})^{-2}$ and ζ_{ik}^{-2}, which make the minimization difficult. If these nonlinear terms are neglected, Eq.6.17 results in the simple objective function (Eq.5.16) based on the mean squared error measure. Therefore, the general exploration strategy derived from mean squared error is an approximate solutions for probability measure.

6.3.3 Competitive Analysis on the Upper Bounds

In the sense of mean squared error, the ideal gain-maximization exploration policy is better than the least-taken policy and the z-interval exploration policy. Is it a better policy also under the probability measure? The following theorem shows that the ideal gain-maximization exploration policy requires less samples to achieve the same probability guarantee.

Theorem 6.3.3. *For any reward standard deviations $\sigma(i) \geq 0$, $1 \leq i \leq n$, the ideal gain-maximization exploration policy has a smaller upper bound on sample complexity than the lest-taken policy and the z-interval exploration policy, i.e. $T_L(\infty) \geq T_L(0) \geq T_L(1)$ with the same choice of $\epsilon > 0$ and $\delta \in (0, 1)$.*

Proof. For the simplicity of notation, we denote the numerator of $T_L(\beta)$ as $C_L(\beta)$

$$C_L(\beta) = \sum_{j=1}^{n} \sigma^{\frac{2\beta}{\beta+1}}(j) \sum_{1\leq i<k\leq n} \left[\sigma^{\frac{2}{3(\beta+1)}}(k) + \sigma^{\frac{2}{3(\beta+1)}}(i)\right]^3. \tag{6.21}$$

$C_L(\beta)$ can be further simplified

$$C_L(\beta) = \sum_{j=1}^{n} \sigma^{\frac{2\beta}{\beta+1}}(j) \left[(n-4)\sum_{i=1}^{n} \sigma^{\frac{2}{\beta+1}}(i) + 3\sum_{k=1}^{n} \sigma^{\frac{4}{3(\beta+1)}}(k) \sum_{i=1}^{n} \sigma^{\frac{2}{3(\beta+1)}}(i)\right], \tag{6.22}$$

which exploits the fact that in a symmetrical matrix the sum of the off-diagonal elements is the half of the difference between the sum of all the elements and the sum of the diagonal elements.

Then it is sufficient to show that $C_L(\infty) \geq C_L(0) \geq C_L(1)$. We get this by showing that the difference $C_L(\infty) - C_L(0)$ and $C_L(0) - C_L(1)$ can be factorized into a sum of non-negative terms.

$$
\begin{aligned}
&C_L(\infty) - C_L(0) \\
&= 4n(n-1)\sum_{j=1}^{n}\sigma^2(j) - n\left[(n-4)\sum_{i=1}^{n}\sigma^2(i) + 3\sum_{k=1}^{n}\sigma^{\frac{4}{3}}(k)\sum_{i=1}^{n}\sigma^{\frac{2}{3}}(i)\right] \\
&= \frac{3}{2}n\sum_{i=1}^{n}\sum_{j=1}^{n}\left[\sigma^{\frac{2}{3}}(i) - \sigma^{\frac{2}{3}}(j)\right]\left[\sigma^{\frac{4}{3}}(i) - \sigma^{\frac{4}{3}}(j)\right] \\
&\geq 0
\end{aligned}
$$

$$
\begin{aligned}
&C_L(0) - C_L(1) \\
&= n\left[(n-4)\sum_{i=1}^{n}\sigma^2(i) + 3\sum_{k=1}^{n}\sigma^{\frac{4}{3}}(k)\sum_{i=1}^{n}\sigma^{\frac{2}{3}}(i)\right] \\
&\quad - \sum_{i=1}^{n}\sigma(i)\left[(n-4)\sum_{i=1}^{n}\sigma(i) + 3\sum_{k=1}^{n}\sigma^{\frac{2}{3}}(k)\sum_{i=1}^{n}\sigma^{\frac{1}{3}}(i)\right] \\
&= \frac{n-4}{2}\sum_{i=1}^{n}\sum_{j=1}^{n}\Big(\sigma(i) - \sigma(j)\Big)^2 \\
&\quad + \frac{3}{2}\sum_{i=1}^{n}\sigma^{\frac{2}{3}}(i)\sum_{i=1}^{n}\sum_{j=1}^{n}\Big(\sigma^{\frac{1}{3}}(i) - \sigma^{\frac{1}{3}}(j)\Big)\Big(\sigma(i) - \sigma(j)\Big) \qquad (6.23)
\end{aligned}
$$

Clearly, Eq.6.23 is non-negative for $n \geq 4$. A more detailed calculation shows that Eq.6.23 is non-negative also for $n = 2, 3$. □

From Theorem 6.3.3, it follows that for any reward standard deviations $\sigma(i) > 0$, $1 \leq i \leq n$, $\frac{T_L(\infty)}{T_L(1)} \geq 1$, and $\frac{T_L(0)}{T_L(1)} \geq 1$. The equalities hold when $\sigma(i)$ are the same for all $i \in \mathcal{A}$. The ratios $\frac{T_L(\infty)}{T_L(1)}$ and $\frac{T_L(0)}{T_L(1)}$ depend both on the number of actions n and the choices of reward standard deviations $\sigma(i)$. To simplify the analysis, we consider an ensemble of bandits, with reward standard deviations $\sigma(i)$ independently identically distributed. We can prove the following

theorem, showing that the advantage of the ideal gain-maximization exploration policy can be in the order of n, when the variation in $\sigma(i)$ is large (see Appendix C.5 for a detailed proof).

Theorem 6.3.4. *Let $\sigma(i) \geq 0$, $1 \leq i \leq n$, be independently identically distributed random variables, whose probability density function is denoted by $p(\sigma)$. The moments of $\sigma(i)$ are defined by $\langle \sigma^b \rangle = \int \sigma^b p(\sigma) d\sigma$, $b > 0$.*

Then if the ratios of the lower moments to the second moment vanish, i.e. $\frac{\langle \sigma^b \rangle}{\langle \sigma^2 \rangle} = 0$, $b \in (0, \frac{5}{3}]$, the ratios of the expected complexities are in the order of $O(n)$,

$$\frac{\langle T_L(0) \rangle}{\langle T_L(1) \rangle} = n, \qquad \frac{\langle T_L(\infty) \rangle}{\langle T_L(1) \rangle} = 4n.$$

If all the moments $\langle \sigma^b \rangle$,$b \in (0, 2]$, are finite, the asymptotic complexity ratios are finite,

$$\lim_{n \to \infty} \frac{\langle T_L(0) \rangle}{\langle T_L(1) \rangle} = \frac{\langle \sigma^2 \rangle + 3\langle \sigma^{\frac{4}{3}} \rangle \langle \sigma^{\frac{2}{3}} \rangle}{\langle \sigma \rangle^2 + 3\langle \sigma \rangle \langle \sigma^{\frac{2}{3}} \rangle \langle \sigma^{\frac{1}{3}} \rangle}, \qquad \lim_{n \to \infty} \frac{\langle T_L(\infty) \rangle}{\langle T_L(1) \rangle} = \frac{4\langle \sigma^2 \rangle}{\langle \sigma \rangle^2 + 3\langle \sigma \rangle \langle \sigma^{\frac{2}{3}} \rangle \langle \sigma^{\frac{1}{3}} \rangle}.$$

Theorem 6.3.4 analyzes the asymptotic ratios of the expected sample complexities along the two dimensions of the sample complexities, i.e. the number of actions n, and the variation of reward standard deviations. It shows that, depending on the variability of the reward standard deviations, the ideal gain-maximization exploration policy outperforms error-based and counter-based exploration strategy by n times. More specifically, the complexity ratio y falls in the region defined by the lines $y = 1$ and $y = n$ in the n-y coordinate system. The ideal gain-maximization exploration policy has large advantage when the variation of $\sigma(i)$ is large enough (i.e. the second moment $\langle \sigma^2 \rangle$ of $\sigma(i)$ approaches to infinity, but lower moments $\langle \sigma^b \rangle$, $b \in (0, \frac{5}{3}]$, are still comparatively small).

Chapter 7

Exploration in Reward Maximization

In previous chapters, we are mainly concerned with exploration for optimal learning in a global sense. As a process of knowledge acquisition, exploration is important not only for learning itself, but also for reward maximization. In this chapter, we analyze exploration in reward maximization, where exploration shall serve the the purpose of exploitation. We call it exploration for exploitation, because reward maximization requires to identify and exploit the best actions.

7.1 Exploration for Exploitation

In reward maximization, the reward function on the one hand is a quantity being learned, and on the other hand is a quantity being maximized. Therefore, an agent needs to tradeoff exploration and exploitation in his action selection behavior in order to obtain maximal cumulative rewards under uncertainty. For example, in reinforcement learning, the agent needs to associate value with each state-action pair, so that an action selection policy is derived to maximize the long term rewards [SB98a]. In clinical experiment design, the best treatment of a certain disease need to be selected among several alternatives with the fewest number of trials [Fed72]. Directly focusing on the reward-maximization objective, the most common performance measure of the agent's action selection is the cumulative reward the agent receives. Maximizing the cumulative reward is equivalent to minimizing the regret, which is the expected loss in reward caused by selecting sub-optimal actions [ACBF02; ACBFS03].

The performance of reward maximization is contingent on sufficient exploration. Therefore, an indirect solution to the reward-maximization problem is to find good exploration strategies, so that exploitation can be efficient enough at the cost of a given amount of exploration [MT04; EDMM06]. In this setting, instead of trading exploitation for exploration, exploitation is undertaken only after sufficient exploration is executed. The full action selection strategy includes an exploration strategy specifying a criterion of exploratory action selection, and a stopping condition indicating the switching time from exploration to exploitation. In this first-explore-then-exploit scheme, the pure exploration phase is the focus of analysis, and the performance of exploration is of main concern. Exploration performance is quantified by its sample complexity, i.e. the number of samples before probably correct decisions can be made.

Exploration for exploitation is widely observed in selecting or ranking a set of alternatives as in mixture of experts and discrete event simulations [JJNH91; PKM96; And98; SJY03]. Compared to analyzing reward maximization directly, devising pure exploration policies has two advantages. First, in reward maximization, the exploration is essentially an active learning problem for the purpose of exploitation. Therefore, analyzing exploration for exploitation is also helpful to understand active learning. Second, finding a good exploration policy may improve the reward-maximization performance of existing exploration-exploitation tradeoff schemes.

Like in previous chapters, our analysis is focused on multi-armed bandits. Two main contributions are presented in this chapter, focusing on gain-based exploration in multi-armed bandits. First, a general mean square error measure is introduced by showing that exploration is equivalent to learning reward differences. From this measure gain-based selection policies in multi-armed bandits are derived from a unified framework which also includes error-based and counter-based exploration. The proposed ideal gain-maximization selection policy is shown to outperform error-based and counter-based policies, both in the sense of mean squared error and probability measure. Related work in [EDMM06] mainly focused on eliminating suboptimal arms to reduce the action set. On each stage of the elimination process, each actions were evenly sampled. We keep the action set fixed and find an optimal exploration strategy in the sense of mean squared error by taking advantage of reward variances. Second, gain-based reward-maximization policies are proposed by linearly trading exploitation with gain-based exploration. Unlike previous methods which use adaptive tradeoff weights to avoid convergence to suboptimal actions [ACBF02], the proposed gain-based reward-maximization policies switch to exploitation under fixed tradeoff weights with a fast rate, achieving performance improvement on fixed sets of bandits.

In section 7.2 we first introduce a mean squared error measure to evaluate selection policies in multi-armed bandits. A generalization of this measure results in a unification of the proposed gain-based selection policy and error-based and counter-based selection policy. The realistic situation, where reward variances and the best action are unknown, is also addressed. A realistic gain-maximization selection policy is proposed, which achieves the optimal exploration strategy asymptotically. In section 7.3, an upper bound on the sample complexity of the general gain-maximization selection strategy is given in terms of probability measure. In section 7.4, the proposed gain-based exploration is linearly traded with exploitation, resulting in gain-based reward-maximization policies.

7.2 Gain-based Exploration in Selection Task

The objective of exploration in reward maximization is to correctly select the best action according to the exploitation rule $\arg\max_a \hat{\mu}_t(a)$ when exploration is finished. This is called the selection task. Without loss of generality, we assume that the true reward means are ordered as $\mu(1) \leq \mu(2) \leq \cdots \leq \mu(n)$. A wrong selection may only happen in pairwise comparisons between $\hat{\mu}_t(n)$ and $\hat{\mu}_t(i)$, $1 \leq i \leq n-1$. The pairwise comparisons are given by the reward mean difference estimators $\hat{\epsilon}_{in} = \hat{\mu}_t(n) - \hat{\mu}_t(i)$, therefore the accuracy of $\hat{\epsilon}_{in}$ determines the selection performance. Exploration is considered as an adaptive sampling behavior in reward estimation, so that the mean difference estimators $\hat{\epsilon}_{in}$ are accurate enough to select the best action.

7.2.1 Exploration Performance Measure

We introduce the total mean squared error of the distance estimators $\hat{\epsilon}_{in}$ to measure the performance of exploration in selection task.

$$\begin{aligned} E_E(t) &= \left\langle \sum_{i=1}^{n-1} \left[\left(\hat{\mu}_t(n) - \hat{\mu}_t(i) \right) - \left(\mu(n) - \mu(i) \right) \right]^2 \right\rangle \\ &= \sum_{i=1}^{n-1} \left[\left\langle \left(\hat{\mu}_t(n) - \mu(n) \right)^2 \right\rangle + \left\langle \left(\hat{\mu}_t(i) - \mu(i) \right)^2 \right\rangle \right] \\ &= \sum_{a=1}^{n-1} \frac{\sigma^2(a)}{\tau(a)} + (n-1)\frac{\sigma^2(n)}{\tau(n)}. \end{aligned} \tag{7.1}$$

In the measure $E_E(t)$, the mean squared error $\sigma^2(n)/\tau(n)$ of the best arm has a larger weight than the other arms. However, in the measure $E_L(t)$ (ref. Eq.5.16), the mean squared errors have the same weight. This is because learning the distances involves the best arm $(n-1)$ times more than the other arms.

$E_E(t)$ requires the knowledge of the index of the best action. It is important to note that this prior information here is employed only to show how learning should be organized optimally. Exploration performance is analyzed based on the result of learning. A realistic exploration policy will be proposed in section 7.2.4 without a priori information of reward variances or the index of the best action.

We also evaluate exploration performance under probability measure. This is because probability naturally measures the performance of decisions. The probability of wrong selection was considered in previous works [MT04; EDMM06]. In section 7.3, we will derive an upper bound on sample complexity and a stopping condition of the proposed exploration policies based on the probability of wrong selection.

In summary, the objective of the agent is to find a policy $\pi_{t+1}(a)$ that minimizes the expected total mean squared error $E_E(t)$ with the learning rules (5.1) and (5.2).

7.2.2 Optimal Policies

First we temporally assume that the reward variances $\sigma^2(a)$ and the index of the best action are known. The realistic case without assuming this information is dealt with in section 7.2.4.

Generalizing $E_E(t)$, we define a β-norm as a general performance measure $G_E(t)$ to be minimized

$$G_E(t) = \Big(\sum_{a=1}^{n-1} \frac{\sigma^{2\beta}(a)}{\tau^{\beta}(a)} + (n-1)\frac{\sigma^{2\beta}(n)}{\tau^{\beta}(n)} \Big)^{\frac{1}{\beta}}, \tag{7.2}$$

where $\beta \in [0, \infty]$ is an exponent parameter. $G_E(t)$ is the β-norm of the $2(n-1)$-dimensional vector $\left[\frac{\sigma^2(1)}{\tau(1)}, \cdots, \frac{\sigma^2(n-1)}{\tau(n-1)}, \frac{\sigma^2(n)}{\tau(n)}, \cdots, \frac{\sigma^2(n)}{\tau(n)}\right]$, with $\frac{\sigma^2(n)}{\tau(n)}$ repeating for $n-1$ times.

The gain of action selection is the negative gradient of $G_E(t)$ with respect to

$\tau(a)$

$$
\begin{aligned}
g_t(a) &= -\frac{\partial G_E(t)}{\partial \tau(a)} \\
&= \begin{cases} \Big(\sum_{b=1}^{n-1} \frac{\sigma^{2\beta}(b)}{\tau^{\beta}(b)} + (n-1)\frac{\sigma^{2\beta}(n)}{\tau^{\beta}(n)}\Big)^{\frac{1}{\beta}-1} \frac{\sigma^{2\beta}(a)}{\tau^{\beta+1}(a)} & \text{if } a \neq n \\ (n-1)\Big(\sum_{b=1}^{n-1} \frac{\sigma^{2\beta}(b)}{\tau^{\beta}(b)} + (n-1)\frac{\sigma^{2\beta}(n)}{\tau^{\beta}(n)}\Big)^{\frac{1}{\beta}-1} \frac{\sigma^{2\beta}(n)}{\tau^{\beta+1}(n)} & \text{if } a = n \end{cases}
\end{aligned} \tag{7.3}
$$

This gives rise to a general gain-maximization selection strategy by neglecting the common term

$$
\begin{aligned}
a_{t+1} &= \arg\max_a h_t(a), \\
h_t(a) &= \begin{cases} \frac{\sigma^{\frac{2\beta}{\beta+1}}(a)}{\tau(a)}, & \text{if } a \neq n \\ \frac{(n-1)^{\frac{1}{\beta+1}}\sigma^{\frac{2\beta}{\beta+1}}(n)}{\tau(n)}, & \text{if } a = n \end{cases}.
\end{aligned} \tag{7.4}
$$

Under the equilibrium condition, the gain of each arm is the same, and the sampling number of each arm has a fixed proportion in the total number of samples

$$
\tau^*(a) = \begin{cases} t\frac{\sigma^{\frac{2\beta}{\beta+1}}(a)}{\sum_{b=1}^{n-1}\sigma^{\frac{2\beta}{\beta+1}}(b) + (n-1)^{\frac{1}{\beta+1}}\sigma^{\frac{2\beta}{\beta+1}}(n)}, & \text{if } a \neq n \\ t\frac{(n-1)^{\frac{1}{\beta+1}}\sigma^{\frac{2\beta}{\beta+1}}(n)}{\sum_{b=1}^{n-1}\sigma^{\frac{2\beta}{\beta+1}}(b) + (n-1)^{\frac{1}{\beta+1}}\sigma^{\frac{2\beta}{\beta+1}}(n)}, & \text{if } a = n \end{cases}. \tag{7.5}
$$

According to the method of Lagrange multipliers, the general policy (Eq.7.4) is optimal in minimizing $G_E(t)$. $\tau^*(a)$ is the optimal allocation scheme of the samples.

The general gain-maximization selection strategy unifies three classes of policies. When $\beta \to \infty$, the resulting error-based exploration policy in selection task is the *z*-interval exploration policy (Eq.5.11).

$$
a_{t+1} = \arg\max_a \frac{\sigma^2(a)}{\tau(a)} = \arg\max_a \frac{\sigma(a)}{\sqrt{\tau(a)}}
$$

If β takes the special value of 0, the general policy (Eq.7.4) is a counter-based policy (called *the ideal counter-based selection policy* hereafter),

$$
\begin{aligned}
a_{t+1} &= \arg\max_a h_t(a), \\
h_t(a) &= \begin{cases} \dfrac{1}{\tau(a)}, & \text{if } a \neq n \\ \dfrac{n-1}{\tau(n)}, & \text{if } a = n \end{cases}.
\end{aligned} \tag{7.6}
$$

Letting β equal to 1, yields a gain-based policy (called *the ideal gain-maximization selection policy* hereafter), which is the optimal policy minimizing the total mean squared error (Eq.7.1) of the distance estimators

$$
\begin{aligned}
a_{t+1} &= \arg\max_a h_t(a), \\
h_t(a) &= \begin{cases} \dfrac{\sigma(a)}{\tau(a)}, & \text{if } a \neq n \\ (n-1)^{\frac{1}{2}} \dfrac{\sigma(n)}{\tau(n)}, & \text{if } a = n \end{cases}.
\end{aligned} \tag{7.7}
$$

The ideal counter-based selection policy and the ideal gain-maximization selection policy have different forms for the best arm n. Therefore, besides the reward standard variances, the knowledge of the best mean is also required.

7.2.3 Performance Comparison

Putting the sampling number (Eq.7.5) into Eq.7.1, we get the total mean squared error of the general gain-maximization selection strategy (Eq. 7.4)

$$
\begin{aligned}
E_E(t) &= \sum_{a=1}^{n-1} \frac{\sigma^2(a)}{\tau(a)} + (n-1)\frac{\sigma^2(n)}{\tau(n)} \\
&= \sum_{a=1}^{n-1} \frac{\sigma^2(a)}{t\dfrac{\sigma^{\frac{2\beta}{\beta+1}}(a)}{\sum_{b=1}^{n-1}\sigma^{\frac{2\beta}{\beta+1}}(b)+(n-1)^{\frac{1}{\beta+1}}\sigma^{\frac{2\beta}{\beta+1}}(n)}} + (n-1)\frac{\sigma^2(n)}{t\dfrac{(n-1)^{\frac{1}{\beta+1}}\sigma^{\frac{2\beta}{\beta+1}}(n)}{\sum_{b=1}^{n-1}\sigma^{\frac{2\beta}{\beta+1}}(b)+(n-1)^{\frac{1}{\beta+1}}\sigma^{\frac{2\beta}{\beta+1}}(n)}} \\
&= \frac{1}{t}\Big[\sum_{b=1}^{n-1}\sigma^{\frac{2\beta}{\beta+1}}(b) + (n-1)^{\frac{1}{\beta+1}}\sigma^{\frac{2\beta}{\beta+1}}(n)\Big]\Big[\sum_{a=1}^{n-1}\sigma^{\frac{2}{\beta+1}}(b) + (n-1)^{\frac{\beta}{\beta+1}}\sigma^{\frac{2}{\beta+1}}(n)\Big].
\end{aligned} \tag{7.8}
$$

When $\beta = 1$, Eq.7.8 gives the minimal total mean squared error $E_E^*(t)$, achieved by the ideal gain-maximization selection policy.

$$
E_E^*(t) = \frac{1}{t}\Big[\sum_{a=1}^{n-1}\sigma(a) + (n-1)^{\frac{1}{2}}\sigma(n)\Big]^2. \tag{7.9}
$$

As two extremes of the general gain-maximization selection strategy, the ideal counter-based selection policy and the z-interval exploration policy have the same total mean squared error, denoted by

$$E_E^e(t) = \frac{2(n-1)}{t} \sum_{a=1}^{n} \sigma^2(a). \tag{7.10}$$

For any choices of $\sigma(a)$, $E_E^e(t)$ is always larger than or equal to $E_E^*(t)$.

$$\begin{aligned}
E_E^e(t) &= \frac{2(n-1)}{t} \sum_{a=1}^{n} \sigma^2(a) = \frac{2}{t}\Big[(n-1)\sum_{a=1}^{n-1} \sigma^2(a) + (n-1)\sigma^2(n)\Big] \\
&\geq \frac{2}{t}\Big[\left(\sum_{a=1}^{n-1} \sigma(a)\right)^2 + (n-1)\sigma^2(n)\Big] \qquad (7.11) \\
&= \frac{1}{t}\Big[\left(\sum_{a=1}^{n-1} \sigma(a)\right)^2 + (n-1)\sigma^2(n) + \left(\sum_{a=1}^{n-1} \sigma(a)\right)^2 + (n-1)\sigma^2(n)\Big] \\
&\geq \frac{1}{t}\Big[2(n-1)^{\frac{1}{2}}\sigma(n)\left(\sum_{a=1}^{n-1} \sigma(a)\right) + \left(\sum_{a=1}^{n-1} \sigma(a)\right)^2 + (n-1)\sigma^2(n)\Big] \\
&= \frac{1}{t}\Big[\sum_{a=1}^{n-1} \sigma(a) + (n-1)^{\frac{1}{2}}\sigma(n)\Big]^2 \\
&= E_E^*(t)
\end{aligned}$$

Eq.7.11 is due to Jensen's inequality. $E_E^*(t)$ is equal to $E_E^e(t)$ when $\sigma(n) = (n-1)^{\frac{1}{2}}\sigma(a), 1 \leq a \leq n-1$.

In appendix D.1, the reward standard deviations $\sigma(a), (1 \leq a \leq n)$, are considered as independently and identically distributed random variables, and the maximal performance ratio of the ideal gain-maximization selection policy to the other two is in the order of $O(n)$, when the variance $Var[\sigma(a)]$ is large enough.

7.2.4 Realistic Gain-maximization Selection Policy

The ideal gain-maximization selection policy (Eq.7.7) requires known reward standard deviations $\sigma(a)$ and the index of the the best arm. For practical application of the gain-based policy, a posterior estimates take the place to form a realistic gain-maximization selection policy.

$$
\begin{aligned}
a_{t+1} &= \arg\max_a h_t(a) \\
h_t(a) &= \begin{cases} m_{\alpha,\tau(a)-1}\frac{\hat{\sigma}_t(a)}{\tau(a)}, & \text{if } a \neq \hat{a}_t^* \\ (n-1)^{\frac{1}{2}} m_{\alpha,\tau(\hat{a}_t^*)-1}\frac{\hat{\sigma}_t(\hat{a}_t^*)}{\tau(\hat{a}_t^*)}, & \text{if } a = \hat{a}_t^* \end{cases} \\
\hat{a}_t^* &= \arg\max_a \hat{\mu}_t(a).
\end{aligned}
\tag{7.12}
$$

Here, $m_{\alpha,\tau}$ is defined in Eq.5.44.

The reward mean estimations $\hat{\mu}_t(a)$ are required for the policy to determine the best reward mean. The change in the estimation of $\hat{a}_t^*$ will result in the change of the policy function for two arms, therefore the realistic gain-maximization selection policy is not local.

7.2.5 Numerical Simulations

Five exploration policies are compared in selection task. They are: the ideal counter-based selection policy (Eq.7.6), the z-interval exploration policy (Eq.5.11), the t-interval exploration policy (Eq.5.14), the ideal gain-maximization selection policy (Eq.7.7), and the realistic gain-maximization selection policy (Eq.7.12).

Performance is compared with respect to the averaged estimation error

$$
\bar{e}_t = \left\langle \frac{\sum_{a=1}^{n-1}\left(\hat{\mu}_t(a)-\mu(a)\right)^2 + (n-1)\left(\hat{\mu}_t(n)-\mu(n)\right)^2}{2} \right\rangle. \tag{7.13}
$$

Here $\langle\cdot\rangle$ represents the average over 10^4 runs, and $\hat{\mu}_t(a)$ are initialized as 0.

Power Law Distributed Parameters

In the first simulation, the reward means $\mu(a)$ and standard deviations $\sigma(a)$ are both generated independently from a power law distribution given by Eq.5.46.

Fig.7.1(a) plots the averaged estimation error $\bar{e}_t$ of the five policies. The ideal gain-maximization selection policy (circle marker) is the best among the five strategies. The performance of the realistic gain-maximization selection policy (square marker, $\alpha = 0.1$) is almost as good as that of the ideal gain-maximization exploration policy after 10^3 steps. The performances of the z-interval exploration policy ($\times$ marker) and the t-interval exploration policy (+

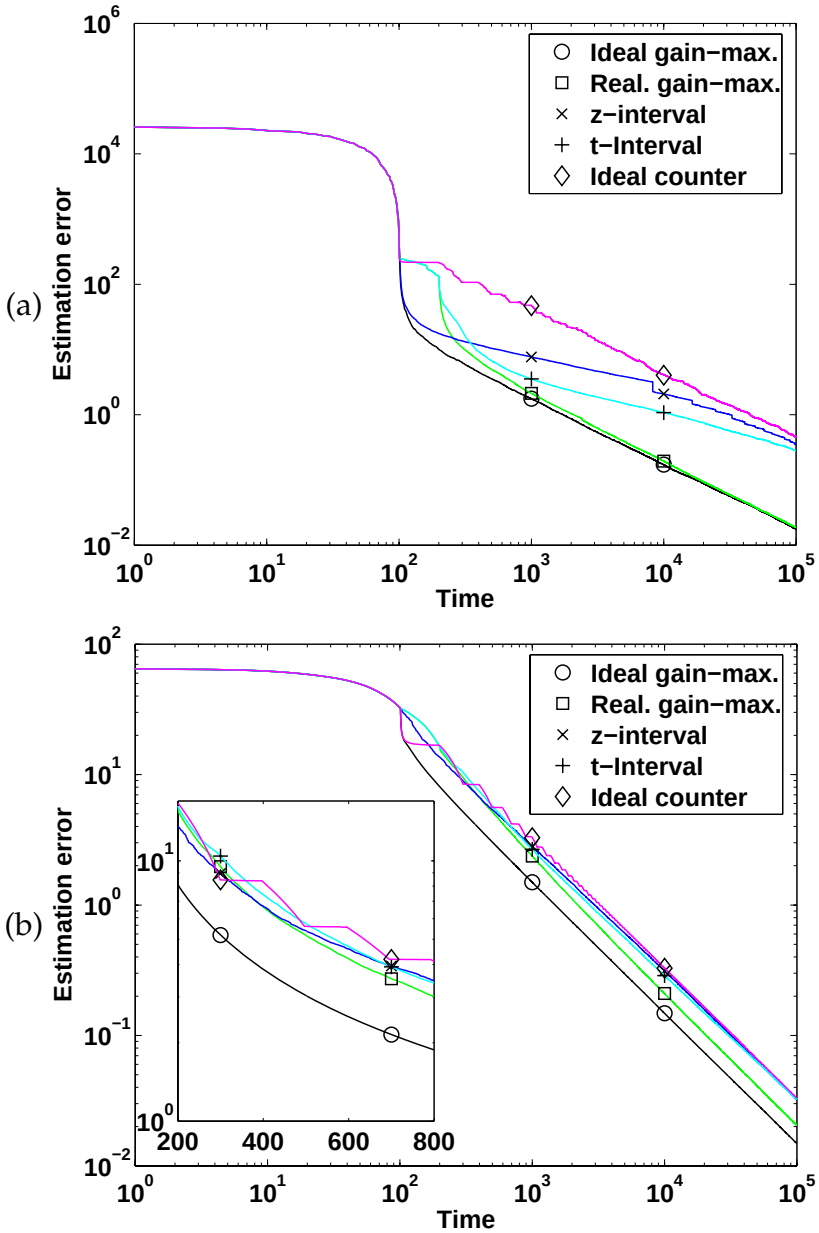


Figure 7.1: Performance comparison of gain-based, error-based and counter-based selection policies. (a) Mean estimation error $\bar{e}_t$ in the selection task with power-law distributed $\mu(a)$ and $\sigma(a)$. The performance of the realistic gain-maximization selection policy ($\alpha = 0.1$) quickly approaches that of the ideal gain-maximization selection policy. The asymptotic performance of the z-interval exploration policy, t-interval exploration policy ($\alpha = 0.001$), and the ideal counter-based selection policy are worse than the gain-based exploration. (b) $\mu(a)$ and $\sigma(a)$ are uniformly distributed. The confidence parameter α is 0.01 for the realistic gain-maximization selection policy, and $\alpha = 0.0001$ for the t-interval exploration policy.

marker, $\alpha = 0.001$) approach to the performance of the ideal counter-based selection policy (diamond marker). For the z-interval exploration policy, the averaged estimation error occurs large decreases after long simulation time. This comes from the estimation error of the best arm with small reward variances. Because in this case, according to the error-based exploration, the best arm can only receive very few samples after a long time, and the factor $(n-1)$ in the performance measure magnifies its error and causes big changes in the curve. On average, the performance of the ideal gain-based exploration is about 12 times as good as that of the counter-based or error-based exploration. This simulation result is in the same order as the performance ratio calculated in Appendix D.1.1.

Uniformly Distributed Parameters

The gain-based exploration also outperforms the counter-based and the error-based exploration, when the reward means $\mu(a)$ and standard deviations $\sigma(a)$ follow a uniform distribution in the range $[0, 1]$.

As shown in Fig.7.1(b), the averaged estimation error $\bar{e}_t$ of the ideal gain-maximization exploration policy (circle marker) is the minimum among the five strategies. The asymptotic performances of the z-interval exploration policy ($\times$ marker) and the t-interval exploration policy ($+$ marker, $\alpha = 0.0001$) are similar as that of the ideal counter-based selection policy (diamond marker). The performance of the ideal gain-maximization policy is about 2 times better than that of the error-based or counter-based exploration. This performance ratio is verified by the analytical calculation in Appendix D.1.2. The realistic gain-maximization exploration does not converge to the ideal gain-maximization exploration as close as achieved in Fig.7.1a. This is because the uniformly distributed reward means only spread out in the small range $[0, 1]$, therefore the best mean is not distinctive enough and selection errors happens quite easily, which makes it difficult to settle in the correct selection policy.

7.3 Sample Complexity of Exploration in Selection Task

For reward maximization, the probability of correct selection is of primary interest. Here, we show that the ideal gain-maximization selection policy has a higher probability to detect the best action than the z-interval exploration policy and the ideal counter-based selection policy. Therefore, from the probabilistic viewpoint, the ideal gain-maximization selection policy also achieves

better performance in exploration.

7.3.1 Upper Bound on the Sample Complexity

A wrong selection happens when there is some wrong pairwise orderings $\hat{\mu}_t(n) \leq \hat{\mu}_t(i)$, $1 \leq i \leq n-1$. However, if a sub-optimal action is not too worse than the best action, then selecting this sub-optimal action is also satisfactory. This is the basic idea underlying Probably Approximately Correct (PAC) algorithms [Val84; MT04; EDMM06]. More specifically, the actions whose reward means μ fall in the indifference zone $\mu > \mu(n) - \epsilon$ are defined to be ϵ-optimal actions (ϵ is a positive number) [GOS79; EDMM06]. Then the problem is to determine a stopping time for an exploration strategy so that the probability of returning a sufficiently worse action is small, or equivalently, the probability of returning an ϵ-optimal action is large.

For any positive number $\epsilon > 0$, suppose action i is not an an ϵ-optimal action (i.e. action i is out of the indifference zone of action n), then

$$\epsilon_{in} = \mu(n) - \mu(i) \geq \epsilon. \tag{7.14}$$

We need to bound the probability for the general gain-maximization selection strategy (Eq.7.4) to return such significantly worse actions by the amount $\delta \in (0, 1)$ from above. This is given in the following theorem.

Theorem 7.3.1. *Let $\epsilon > 0$ and $\delta \in (0, 1)$ are two small positive numbers. $\hat{\mu}_t(i)$ are the empirical mean (ref. Eq.5.1) of any distribution, whose probability density function is symmetric around the true mean. Then it takes at least $T_E(\beta) = \frac{\sum_{j=1}^{n-1} \sigma^{\frac{2\beta}{\beta+1}}(j) + (n-1)^{\frac{1}{\beta+1}} \sigma^{\frac{2\beta}{\beta+1}}(n)}{2\delta\epsilon^2} \sum_{i=1}^{n-1} \left[(n-1)^{-\frac{1}{3(\beta+1)}} \sigma^{\frac{2}{3(\beta+1)}}(n) + \sigma^{\frac{2}{3(\beta+1)}}(i)\right]^3$ samples for the sampling strategy (Eq.7.5) to output an ϵ-optimal action with probability at least $(1-\delta)$.*

Proof. Firstly, we bound the probability of wrong selections. A wrong selection happens when there is some wrong pairwise ordering $\hat{\mu}_t(n) < \hat{\mu}_t(i)$, $1 \leq i \leq n-1$.

For any significantly worse action i, we have $\mu(n) - \mu(i) \geq \epsilon, \forall 1 \leq i <\leq n-1$. Therefore,

$$\epsilon \leq \min_{1 \leq i \leq n-1} \epsilon_{in} = \min_{1 \leq i \leq n-1} \big(\mu(n) - \mu(i)\big) \tag{7.15}$$

The probability of wrong selection is bounded by

$$
\begin{aligned}
&P\Big(\text{Wrong selection}\Big) \\
&= P\Big(\exists i \neq n, \hat{\mu}_t(n) < \hat{\mu}_t(i)\Big) = P\Big(\cup_{i=1}^{n-1} \{\hat{\mu}_t(n) < \hat{\mu}_t(i)\}\Big) \\
&\leq \sum_{i=1}^{n-1} P\Big(\hat{\mu}_t(n) < \hat{\mu}_t(i)\Big) && (7.16) \\
&\leq \sum_{i=1}^{n-1} \Big[P\Big(\hat{\mu}_t(i) \geq \mu(i) + \zeta_{in}\Big) + P\Big(\hat{\mu}_t(n) \leq \mu(n) - (\epsilon_{in} - \zeta_{in})\Big)\Big] && (7.17) \\
&\leq \frac{1}{2}\sum_{i=1}^{n-1} \Big[\frac{\sigma^2(n)}{(\epsilon_{in} - \zeta_{in})^2 \tau(n)} + \frac{\sigma^2(i)}{\zeta_{in}^2 \tau(i)}\Big] && (7.18) \\
&= \frac{\sum_{j=1}^{n-1} \sigma^{\frac{2\beta}{\beta+1}}(j) + (n-1)^{\frac{1}{\beta+1}} \sigma^{\frac{2\beta}{\beta+1}}(n)}{2t} \sum_{i=1}^{n-1} \Big[\frac{(n-1)^{-\frac{1}{\beta+1}} \sigma^{\frac{2}{\beta+1}}(n)}{(\epsilon_{in} - \zeta_{in})^2} + \frac{\sigma^{\frac{2}{\beta+1}}(i)}{\zeta_{in}^2}\Big] && (7.19) \\
&= \frac{\sum_{j=1}^{n-1} \sigma^{\frac{2\beta}{\beta+1}}(j) + (n-1)^{\frac{1}{\beta+1}} \sigma^{\frac{2\beta}{\beta+1}}(n)}{2t} \sum_{i=1}^{n-1} \frac{1}{\epsilon_{in}^2} \Big[(n-1)^{-\frac{1}{3(\beta+1)}} \sigma^{\frac{2}{3(\beta+1)}}(n) + \sigma^{\frac{2}{3(\beta+1)}}(i)\Big]^3 \\
&\leq \frac{\sum_{j=1}^{n-1} \sigma^{\frac{2\beta}{\beta+1}}(j) + (n-1)^{\frac{1}{\beta+1}} \sigma^{\frac{2\beta}{\beta+1}}(n)}{2t\epsilon^2} \sum_{i=1}^{n-1} \Big[(n-1)^{-\frac{1}{3(\beta+1)}} \sigma^{\frac{2}{3(\beta+1)}}(n) + \sigma^{\frac{2}{3(\beta+1)}}(i)\Big]^3 && (7.20) \\
&\leq \delta && (7.21)
\end{aligned}
$$

Eq.7.16 is due to union bound (Eq.C.3). In Eq.7.17 Lemma 6.3.1 is applied. Eq.(7.18) is obtained by applying Chebyshev Inequality (Eq.C.7). Eq.7.19 employs the sample number (Eq.7.5). Then the quantity $\frac{(n-1)^{-\frac{1}{\beta+1}} \sigma^{\frac{2}{\beta+1}}(n)}{(\epsilon_{in} - \zeta_{in})^2} + \frac{\sigma^{\frac{2}{\beta+1}}(i)}{\zeta_{in}^2}$ is minimized by selecting ζ_{in} as (ref. Appendix D.2)

$$
\zeta_{in}^* = \epsilon_{in} \frac{\sigma^{\frac{2}{3(\beta+1)}}(i)}{(n-1)^{-\frac{1}{3(\beta+1)}} \sigma^{\frac{2}{3(\beta+1)}}(n) + \sigma^{\frac{2}{3(\beta+1)}}(i)}. \tag{7.22}
$$

Eq.7.20 is the result of applying Eq.7.15.

Then from Eq.7.21, it is clear that the probability of select significantly worse actions is at most $\delta \in (0, 1)$, or equivalently, the probability of selecting an ϵ-optimal action is at least $(1 - \delta)$, when the total number of samples t is at least

$$t \geq \frac{\sum_{j=1}^{n-1} \sigma^{\frac{2\beta}{\beta+1}}(j) + (n-1)^{\frac{1}{\beta+1}} \sigma^{\frac{2\beta}{\beta+1}}(n)}{2\delta\epsilon^2} \cdot \sum_{i=1}^{n-1} \left[(n-1)^{-\frac{1}{3(\beta+1)}} \sigma^{\frac{2}{3(\beta+1)}}(n) + \sigma^{\frac{2}{3(\beta+1)}}(i) \right]^3. \tag{7.23}$$

□

7.3.2 Competitive Analysis of the Upper Bounds

It is clear that the sample complexity of the general gain-maximization selection strategy is polynomial in $1/\delta$ and $1/\epsilon^2$. Therefore is is a PAC algorithm. The following theorem compares the difference of the upper bounds on sample complexity of the ideal gain-maximization selection policy, the ideal counter-based selection policy and the z-interval exploration policy.

Theorem 7.3.2. *For any reward standard deviations $\sigma(i) \geq 0$, $1 \leq i \leq n$, the ideal gain-maximization selection policy has a smaller upper bound on sample complexity than the ideal counter-based selection policy and the z-interval exploration policy, i.e. $T_E(\infty) \geq T_E(0) \geq T_E(1)$ with the same choice of ϵ and δ.*

Proof. For the simplicity of notation, we denote the numerator of $T_E(\beta)$ as $C_E(\beta)$

$$C_E(\beta) = \left[\sum_{j=1}^{n-1} \sigma^{\frac{2\beta}{\beta+1}}(j) + (n-1)^{\frac{1}{\beta+1}} \sigma^{\frac{2\beta}{\beta+1}}(n) \right] \cdot \sum_{i=1}^{n-1} \left[(n-1)^{-\frac{1}{3(\beta+1)}} \sigma^{\frac{2}{3(\beta+1)}}(n) + \sigma^{\frac{2}{3(\beta+1)}}(i) \right]^3. \tag{7.24}$$

Then it is sufficient to show that $C_E(\infty) \geq C_E(0) \geq C_E(1)$.

We first apply the following transformation to $\sigma(i)$

$$\sigma(i) = \begin{cases} \gamma(i), & \text{if } i \neq n \\ (n-1)^{\frac{1}{2}}\gamma(n), & \text{if } i = n \end{cases}. \tag{7.25}$$

Then $C_E(\beta)$ becomes

$$\begin{aligned}
C_E(\beta) &= \Big[\sum_{j=1}^{n-1}\gamma^{\frac{2\beta}{\beta+1}}(j)+(n-1)\gamma^{\frac{2\beta}{\beta+1}}(n)\Big]\sum_{i=1}^{n-1}\Big(\gamma^{\frac{2}{3(\beta+1)}}(n)+\gamma^{\frac{2}{3(\beta+1)}}(i)\Big)^3\\
&= \Big[\sum_{j=1}^{n-1}\gamma^{\frac{2\beta}{\beta+1}}(j)+(n-1)\gamma^{\frac{2\beta}{\beta+1}}(n)\Big]\Big[(n-1)\gamma^{\frac{2}{\beta+1}}(n)+\sum_{i=1}^{n-1}\gamma^{\frac{2}{\beta+1}}(i)\\
&\qquad +3\gamma^{\frac{2}{3(\beta+1)}}(n)\sum_{i=1}^{n-1}\gamma^{\frac{4}{3(\beta+1)}}(i)+3\gamma^{\frac{4}{3(\beta+1)}}(n)\sum_{i=1}^{n-1}\gamma^{\frac{2}{3(\beta+1)}}(i)\Big].
\end{aligned}$$

Finally, we show that the difference $C_E(\infty)-C_E(0)$ and $C_E(0)-C_E(1)$ can be factorized into a sum of non-negative terms.

$C_E(\infty)-C_E(0)$ is given by

$$\begin{aligned}
&C_E(\infty)-C_E(0)\\
&=8(n-1)\Big[\sum_{j=1}^{n-1}\gamma^2(j)+(n-1)\gamma^2(n)\Big]\\
&\quad -2(n-1)\Big[(n-1)\gamma^2(n)+\sum_{i=1}^{n-1}\gamma^2(i)+3\gamma^{\frac{2}{3}}(n)\sum_{i=1}^{n-1}\gamma^{\frac{4}{3}}(i)+3\gamma^{\frac{4}{3}}(n)\sum_{i=1}^{n-1}\gamma^{\frac{2}{3}}(i)\Big]\\
&=6(n-1)\sum_{i=1}^{n-1}\Big(\gamma^{\frac{4}{3}}(i)-\gamma^{\frac{4}{3}}(n)\Big)\Big(\gamma^{\frac{2}{3}}(i)-\gamma^{\frac{2}{3}}(n)\Big)\\
&\geq 0.
\end{aligned}$$

$C_E(0)-C_E(1)$ is equal to

$$\begin{aligned}
&C_E(0)-C_E(1)\\
&=2(n-1)\Big[(n-1)\gamma^2(n)+\sum_{i=1}^{n-1}\gamma^2(i)+3\gamma^{\frac{2}{3}}(n)\sum_{i=1}^{n-1}\gamma^{\frac{4}{3}}(i)+3\gamma^{\frac{4}{3}}(n)\sum_{i=1}^{n-1}\gamma^{\frac{2}{3}}(i)\Big]\\
&\quad -\Big[(n-1)\gamma(n)+\sum_{i=1}^{n-1}\gamma(i)+3\gamma^{\frac{1}{3}}(n)\sum_{i=1}^{n-1}\gamma^{\frac{2}{3}}(i)+3\gamma^{\frac{2}{3}}(n)\sum_{i=1}^{n-1}\gamma^{\frac{1}{3}}(i)\Big]\\
&\quad \cdot\Big[\sum_{j=1}^{n-1}\gamma(j)+(n-1)\gamma(n)\Big]
\end{aligned}$$

$$
\begin{aligned}
&= (n-1)\sum_{i=1}^{n-1}\Big(\gamma(n)-\gamma(i)\Big)\Big(\gamma^{\frac{1}{3}}(n)-\gamma^{\frac{1}{3}}(i)\Big)^3 + \frac{1}{2}\sum_{i=1}^{n-1}\sum_{j=1}^{n-1}\Big(\gamma(i)-\gamma(j)\Big)^2 \\
&\quad + \frac{3}{2}\gamma^{\frac{2}{3}}(n)\sum_{i=1}^{n-1}\sum_{j=1}^{n-1}\Big(\gamma^{\frac{1}{3}}(i)-\gamma^{\frac{1}{3}}(j)\Big)\Big(\gamma(i)-\gamma(j)\Big) \\
&\quad + \frac{3}{2}\gamma^{\frac{1}{3}}(n)\sum_{i=1}^{n-1}\sum_{j=1}^{n-1}\Big(\gamma^{\frac{2}{3}}(i)-\gamma^{\frac{2}{3}}(j)\Big)\Big(\gamma(i)-\gamma(j)\Big) \\
&\geq 0.
\end{aligned}
$$

□

From Theorem 7.3.2, it follows that $\frac{T_E(\infty)}{T_E(1)} \geq 1$, and $\frac{T_E(0)}{T_E(1)} \geq 1$ for any reward standard deviations $\sigma(i) > 0$, $1 \leq i \leq n$, . The equalities hold when $\sigma(n) = (n-1)^{\frac{1}{2}}\sigma(i)$, $1 \leq i \leq n-1$.

The ratios $\frac{T_E(\infty)}{T_E(1)}$ and $\frac{T_E(0)}{T_E(1)}$ depend both on the number of arms n and the choices of reward standard deviations $\sigma(i)$. To simplify the analysis, we consider an ensemble of bandits, with reward standard deviations $\sigma(i)$ independently identically distributed. The following theorem shows that the ratios $\frac{\langle T_E(\infty)\rangle}{\langle T_E(1)\rangle}$ and $\frac{\langle T_E(0)\rangle}{\langle T_E(1)\rangle}$ between the upper bounds are in the order of $O(n)$, when the variation of reward standard deviation is sufficiently large. The proof of Theorem 7.3.3 is shown in Appendix D.3.

Theorem 7.3.3. *Let $\sigma(i) \geq 0$, $1 \leq i \leq n$, are independently identically distributed random variables, whose probability density function is denoted by $p(\sigma)$. The moments of $\sigma(i)$ are defined by*

$$\langle \sigma^b \rangle = \int \sigma^b p(\sigma) d\sigma, b > 0. \tag{7.26}$$

Then if the ratios of the lower moments to the second moment vanish, i.e. $\frac{\langle \sigma^b \rangle}{\langle \sigma^2 \rangle} = 0$, $b \in (0, \frac{5}{3}]$, the ratios of the expected complexities are in the order of $O(n)$

$$\frac{\langle T_E(0)\rangle}{\langle T_E(1)\rangle} = n, \quad \frac{\langle T_E(\infty)\rangle}{\langle T_E(1)\rangle} = 4n.$$

If all the moments $\langle \sigma^b \rangle$, $b \in (0, 2]$, are finite, then the asymptotic complexity ratios are finite.

$$\lim_{n\to\infty} \frac{\langle T_E(0)\rangle}{\langle T_E(1)\rangle} = \frac{2\langle \sigma^2 \rangle}{\langle \sigma \rangle^2}, \quad \lim_{n\to\infty} \frac{\langle T_E(\infty)\rangle}{\langle T_E(1)\rangle} = \frac{8\langle \sigma^2 \rangle}{\langle \sigma \rangle^2}$$

7.4 Exploration Exploitation Tradeoff

There are several well-known heuristics to combine exploration and exploitation. In soft-max policy, exploration is controlled by the temperature parameter of the Boltzmann distribution [SB98b].

The ϵ-*greedy policy* mixes exploitation with random exploration according to a specified probability [Wat89; ACBF02].

$$\begin{aligned} a_{t+1} &= \begin{cases} \arg\max_a \hat{\mu}_t(a), & \text{with probability } 1-\epsilon_t \\ a \sim U(\mathcal{A}), & \text{with probability } \epsilon_t \end{cases} \\ \epsilon_t &= \min\left\{1, \frac{cn}{d^2 t}\right\} \end{aligned} \tag{7.27}$$

where $d = \mu(n) - \mu(n-1)$ is the difference of reward means between the best action and the second best action. $c > 0$ is a free parameter, which controls the amount of exploration.

In the z-interval policy (Eq.5.9) and the t-interval policy (Eq.5.12), exploration and exploitation is lineally combined [Fon95]

$$a_{t+1} = \arg\max_a \left[\hat{\mu}_t(a) + \kappa g_t(a)\right], \tag{7.28}$$

where $g_t(a)$ is a gain function of exploration, and κ is a positive weight, indicating one unit of gain in exploration is traded for κ unit of reward.

Because of the fixed tradeoff weight κ, action selection may converge to suboptimal actions if the rewards observed from the best action happen to be very bad. The best tradeoff weight is generally different for different bandit [Fon95]. To overcome this disadvantage, Auer et al. proposed *the Upper Confidence Bound (UCB) policy* [ACBF02], which trades exploitation with error-based exploration according to an adaptive weight

$$a_{t+1} = \begin{cases} a, & \text{if } \exists a, \tau(a) \leq 8\log t \\ \arg\max_a \left[\hat{\mu}_t(a) + 4\sqrt{\log(t-1)}\dfrac{\hat{\sigma}_t(a)}{\sqrt{\tau(a)}}\right], & \text{else} \end{cases} \tag{7.29}$$

The UCB policy and the ϵ-greedy policy are shown to have logarithmic regret bounds [ACBF02], which are the best possible bound achievable by a reward-maximization policy for a large family of reward distributions [LR85].

7.4.1 Gain-based Reward Maximization Policy

Motivated by the z-interval and the t-interval policy, we propose two gain-based reward maximization policies in the same fashion of linear tradeoff

(Eq.7.28).

Requiring the knowledge that arm n is the best arm, *the ideal gain-based reward-maximization policy* is

$$\begin{aligned} a_{t+1} &= \arg\max_a h_t(a) \\ h_t(a) &= \begin{cases} \hat{\mu}_t(a) + \kappa \dfrac{\sigma(a)}{\tau(a)}, & \text{if } a \neq n \\ \hat{\mu}_t(n) + \kappa (n-1)^{\frac{1}{2}} \dfrac{\sigma(n)}{\tau(n)}, & \text{if } a = n \end{cases} . \end{aligned} \tag{7.30}$$

Using a posterior estimator, *the realistic gain-based reward-maximization policy* is

$$\begin{aligned} a_t &= \arg\max_a h_t(a) \\ h_t &= \begin{cases} \hat{\mu}_t(a) + \kappa m_{\alpha,\tau(a)-1} \dfrac{\hat{\sigma}_t(a)}{\tau(a)}, & \text{if } a \neq \hat{a}_t^* \\ \hat{\mu}_t(\hat{a}_t^*) + \kappa (n-1)^{\frac{1}{2}} m_{\alpha,\tau(\hat{a}_t^*)-1} \dfrac{\hat{\sigma}_t(\hat{a}_t^*)}{\tau(\hat{a}_t^*)}, & \text{if } a = \hat{a}_t^* \end{cases} \\ \hat{a}_t^* &= \arg\max_a \hat{\mu}_t(a), \end{aligned} \tag{7.31}$$

where $\alpha \in [0, 1]$ is an error parameter. $m_{\alpha,\tau(a)-1}$ is defined in Eq.5.44.

The problem of the linear tradeoff scheme is how to choose an appropriate value for the free parameter κ. For small κ, exploration is weighted less. The policy may converge to an suboptimal action when the best action happens to have bad initial rewards due to its reward variation. On the one hand, in order to avoid the convergence to suboptimal actions, κ should be sufficiently large, and apparently bad actions are still explored to make sure that the inferiority in reward is not a result of fluctuation. On the other hand, however, if the exploration term decays too slowly, then it takes long time for the policy to converge to the optimal action selection, resulting in large regrets.

The solution to this problem given by the gain-based reward-maximization policies is that κ is chosen large enough, and that the exploration term decays fast enough with respect to $\tau(a)$. In this way, sufficient initial data are collected, and at the same time the switch is made fast enough to exploit the learned knowledge earlier. Therefore, the proposed gain-based reward-maximization policies demonstrate performance improvement in reward maximization as compared to error-based reward-maximization policies like the z-interval policy and the t-interval policy (ref. Fig. 7.7(a) and 7.13(a)). This exploration-exploitation tradeoff behavior is different from that of the UCB policy (7.29), where the exploration term is fixed as the error-based exploration, and where the tradeoff parameter increases with respect to the total

number of samples t. This solution can be considered as perpendicular to the one presented here.

In summary, the proposed gain-based reward-maximization policies are soft realizations of the first-explore-then-exploit scheme, with the parameter κ controlling the length of exploration phase and with a fast rate of switching to exploitation.

7.4.2 Empirical Evaluation of Reward Maximization Policies

Since the ϵ-greedy policy (Eq.7.27) and the UCB policy (Eq.7.29) are shown to achieve the optimal logarithmic regret bound, we compare the proposed ideal gain-based reward-maximization policy (Eq.7.30) and realistic gain-based reward maximization policy (Eq.7.31) with them, together with the similar z-interval policy (Eq.5.9) and the t-interval policy (Eq.5.12).

The policies are tested on 10 randomly generate 10-armed bandits, 100 trials for each bandit. Each bandit has the same worst case configuration of reward means $\mu(i)$, but differs in reward standard deviations $\sigma(i)$. More specifically, in each bandit the reward means of suboptimal arms are set to be equal $\mu(i) = 0, 1 \leq i \leq 9$, and the reward mean of the best action is set to $\mu(10) = 0.1$. This is the worst setting for the selection task in a bandit whose minimal reward mean difference is 0.1. The reward standard deviations $\sigma(i), 1 \leq i \leq 10$, are independently sampled from a power law distribution or a uniform distribution, in order to see how reward standard deviations influence the performance on average. The density function of the power law distribution is given by $p(x) = Cx^{-2.5},\ x \geq \frac{1}{6}, C = 1.5 \times 6^{-1.5}$. Its exponent is chosen as 2.5 because most of the power law distributions observed in the real life has an exponent in the range $(2, 3]$ [New05]. The uniform distribution used is distributed over the range $[0, 1]$. Both distributions have mean value $1/2$. Policies are compared on ensembles of such randomly generated bandits. In the beginning of every simulation run, each action is selected twice in order to get initial estimations of reward means and variances. After this initial phase, action selection is done according to the respective policies.

Performance of reward maximization is evaluated with respect to regret $R(t)$ and the percentage of optimal action selection $F(t)$, averaged on the 10^3 runs .

$$R(t) = t\mu(n) - \left\langle \sum_{\tau=1}^{t} r_\tau \right\rangle \tag{7.32}$$

$$F(t) = \left\langle \frac{1}{t} \sum_{\tau=1}^{t} \delta_{a_\tau, n} \right\rangle \tag{7.33}$$

In order to show the relationship between exploration and exploitation, the exploration performance is also compared in terms of estimation error $\bar{e}_t$ (Eq.7.13) and the probability of wrong selection $P_{WS}(t)$

$$P_{WS}(t) = P\Big(\exists i \neq n, \hat{\mu}_t(n) < \hat{\mu}_t(i)\Big). \tag{7.34}$$

Uniformly Distributed Parameters

In the first simulation, the reward standard deviations are uniformly distributed over the range $[0, 1]$.

Fig. 7.2 depicts the performance of the ideal gain based reward maximization policy. For small tradeoff parameters κ, the policy get stuck in an suboptimal arm, resulting in linear increase of regret (Fig.7.2(a)). This is because for small κ, the exploration term is weighted too less, and exploitation is favored based on inaccurate reward estimations (Fig.7.2(c)). Therefore, the probability of erring in detecting the optimal arm is quite high (Fig.7.2(d)). With increasing κ, the regret saturates asymptotically. This is because the probability of wrong selection decreases to zero. Note that once the probability of wrong selection is sufficiently small, it is better to switch to exploitation. Too much exploration will result in higher asymptotic regret, although the estimation error can be reduced better. Therefore, it is important to find an appropriate tradeoff scheme. The proposed method relies on tuning of the tradeoff parameter, which is generally different for different sets of bandits.

Fig.7.3-7.6 display similar situations for the realistic gain based reward maximization policy, the z-interval policy the t-interval policy, and the ϵ-greedy policy. Too greedy in exploration or too greedy in exploitation are both not optimal in minimizing the regret.

Unlike these policies, the UCB policy does not have free parameters. Fig.7.7(a) compares the performance of the UCB policy with these policies tuned to their best parameters. The regret of the ideal gain-based reward-maximization policy settles down on a low level very quickly. The regret of the realistic gain-based reward-maximization policy saturates on a higher level. The regrets of the other four policies are least 2 times as large.

The action selection behaviors of the six policies are different (Fig.7.7(b)). For the ideal gain-based reward-maximization policy, it contains prior information, therefore, it requires less exploration, resulting large percentage of selections of the best action (circle marker). Without any prior information, the realistic gain-based reward-maximization policy (square marker) pays more for exploration, showing a lower percentage of the best action in its action

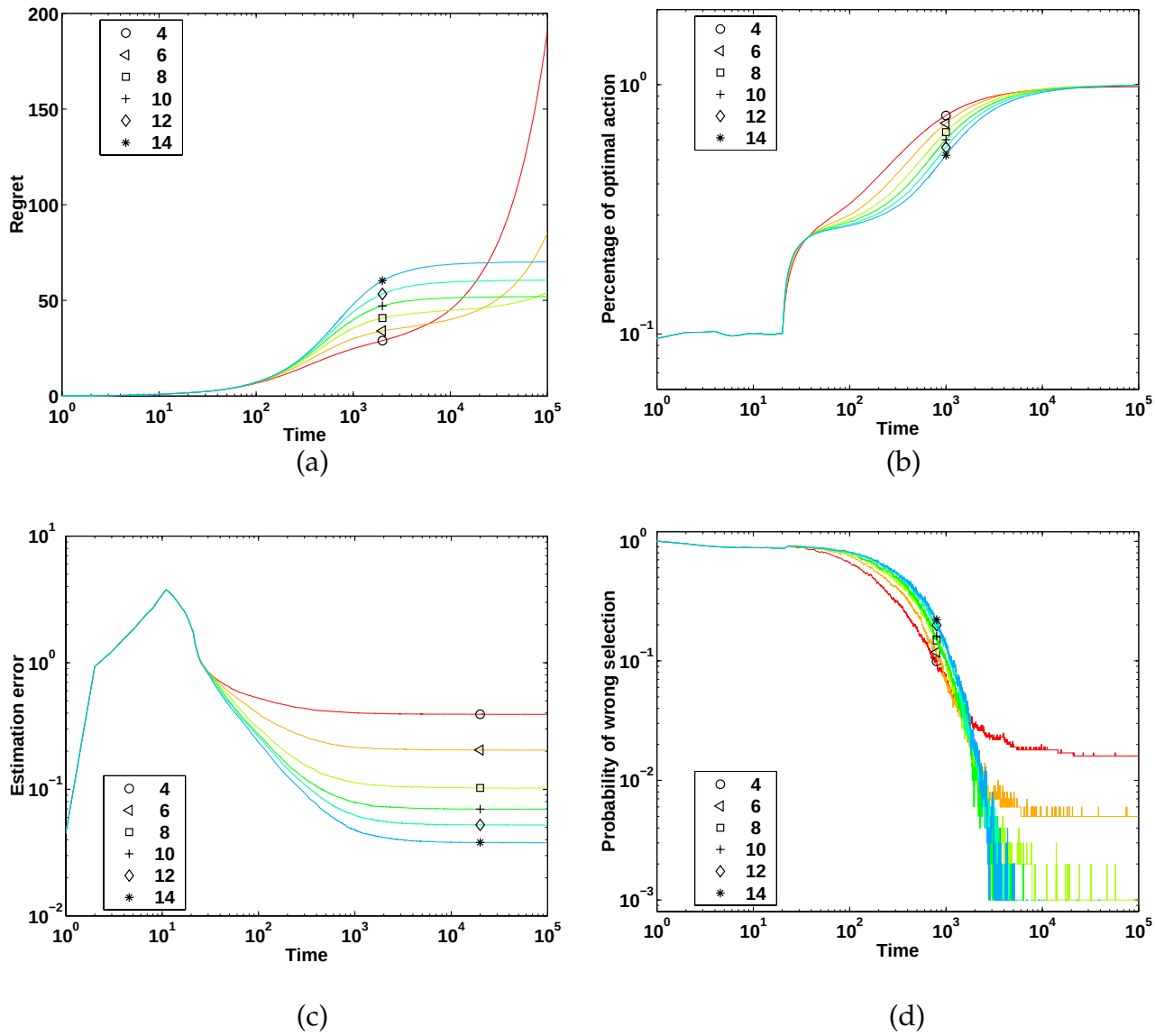


Figure 7.2: Reward maximization and exploration performance of the ideal gain-based reward maximization policy with uniformly distributed reward standard deviations. $\kappa \in \{4, 6, 8, 10, 12, 14\}$. The best value of κ is about 8. (a) Average regret $R(t)$; (b) Average percentage of the optimal action $F(t)$; (c) Average estimation error $\bar{e}_t$; (d) Probability of wrong selection $P_{WS}(t)$.

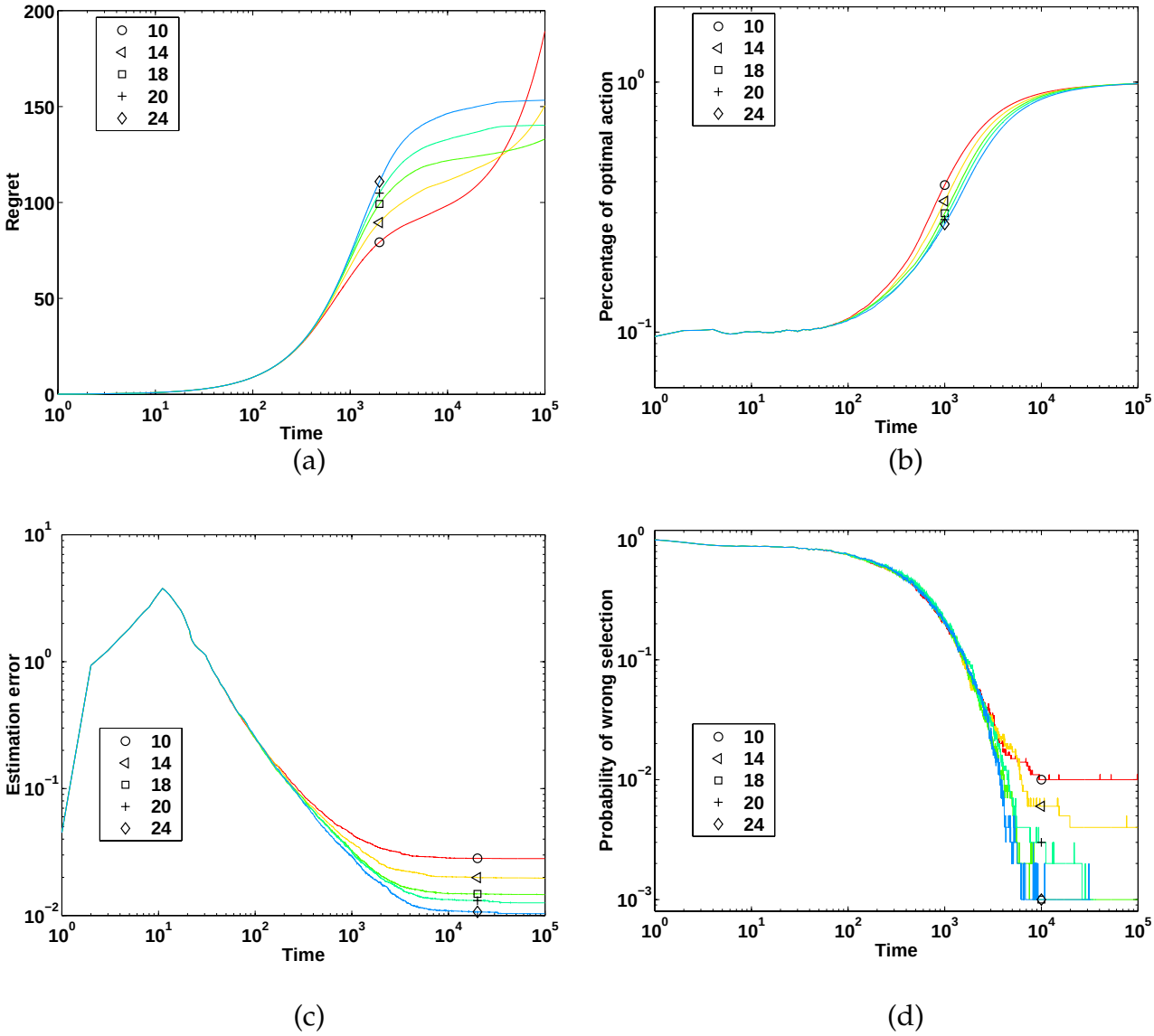


Figure 7.3: Reward maximization and exploration performance of the realistic gain-based reward-maximization policy with uniformly distributed reward standard deviations. $\kappa \in \{10, 14, 18, 20, 24\}$, $\alpha = 1e-4$. The best value of κ is between 18 and 20. (a) Average regret $R(t)$; (b) Average percentage of the optimal action $F(t)$; (c) Average estimation error $\bar{e}_t$; (d) Probability of wrong selection $P_{WS}(t)$.

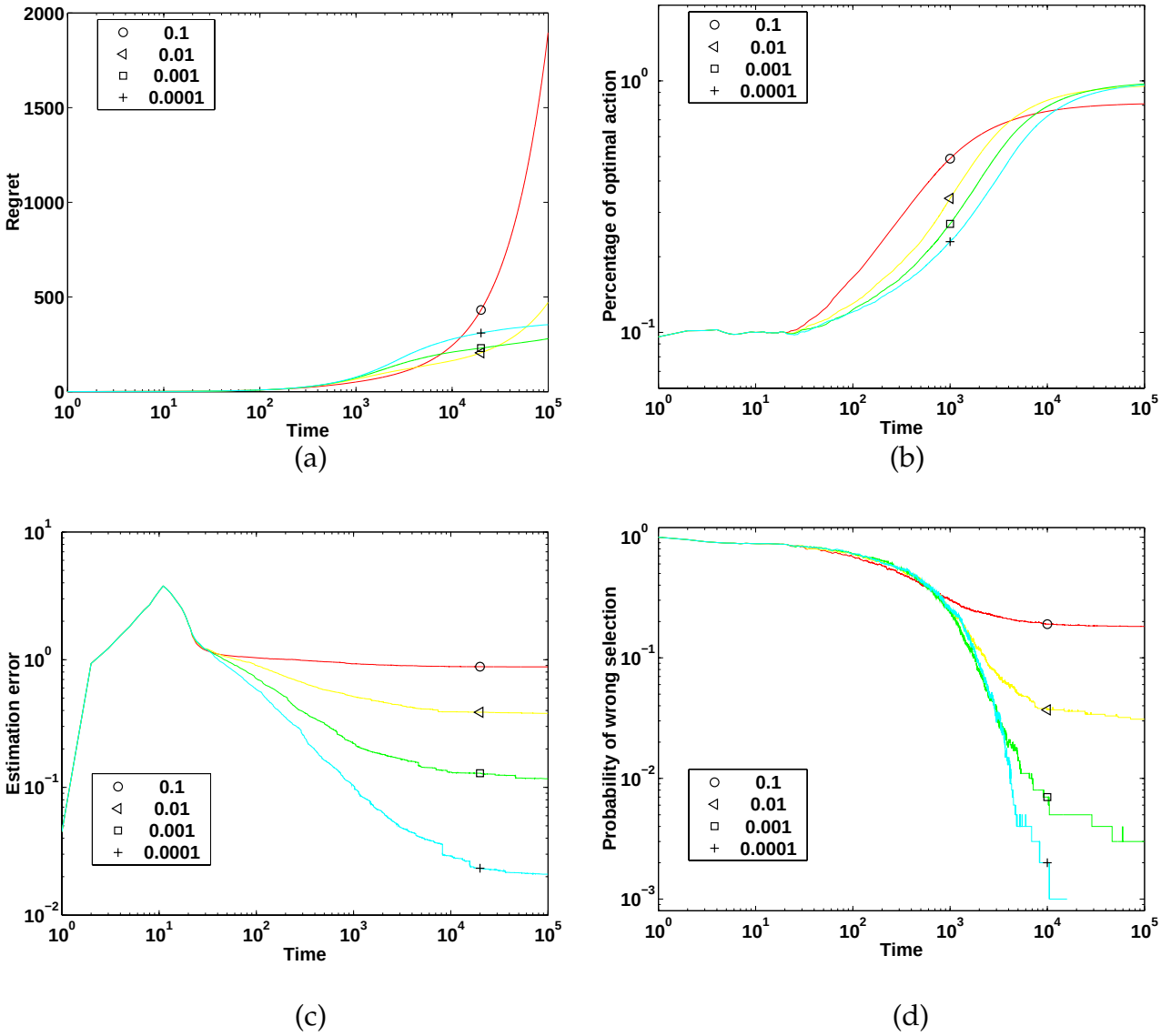


Figure 7.4: Reward maximization and exploration performance of the z-interval policy with uniformly distributed reward standard deviations. $\alpha \in \{0.1, 0.01, 1e-3, 1e-4\}$. The best value of α is 0.001. (a) Average regret $R(t)$; (b) Average percentage of the optimal action $F(t)$; (c) Average estimation error $\bar{e}_t$; (d) Probability of wrong selection $P_{WS}(t)$.

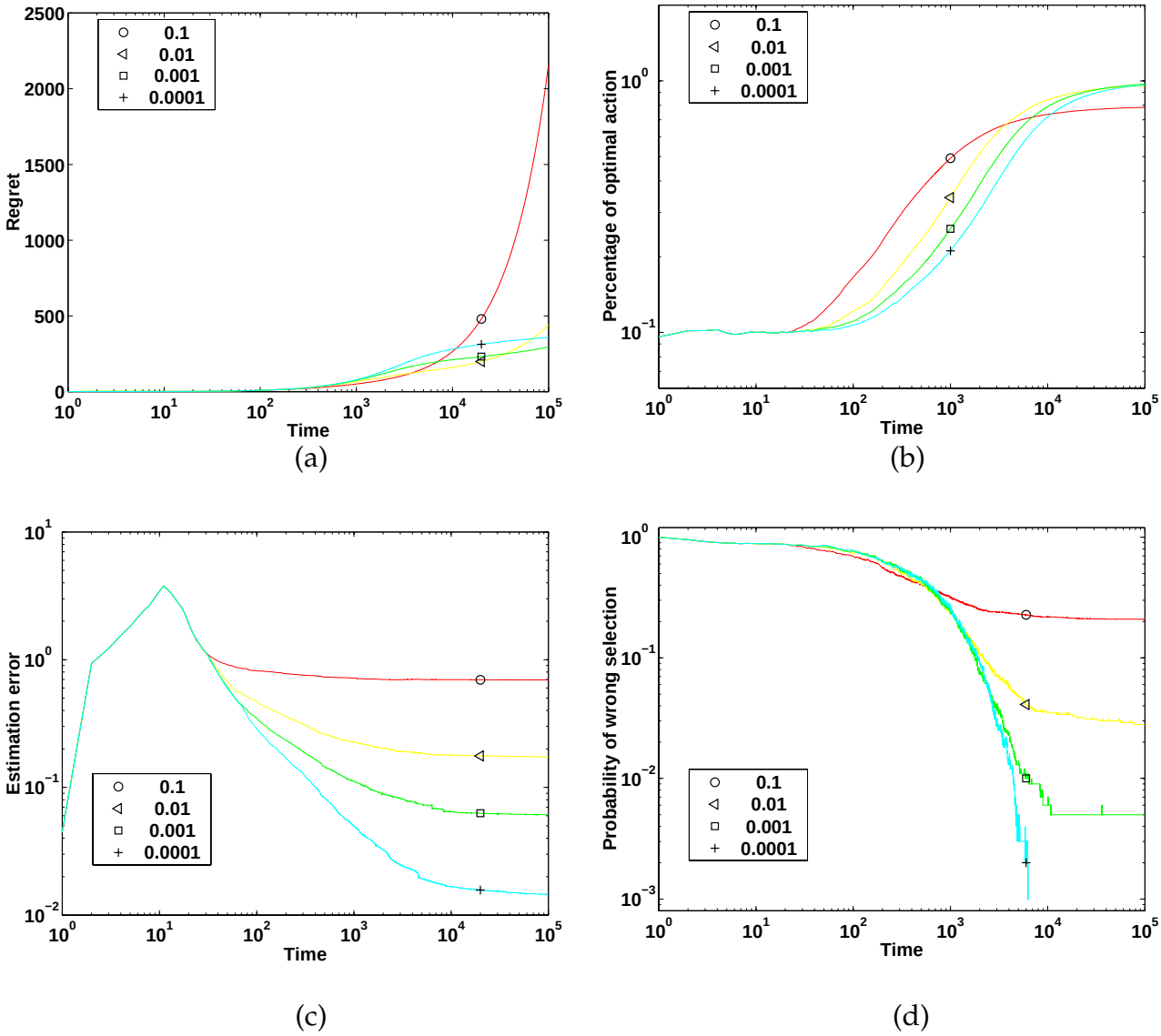


Figure 7.5: Reward maximization and exploration performance of the t-interval policy with uniformly distributed reward standard deviations. $\alpha \in \{0.1, 0.01, 1e-3, 1e-4\}$. The best value of α is 0.001. (a) Average regret $R(t)$; (b) Average percentage of the optimal action $F(t)$; (c) Average estimation error $\bar{e}_t$; (d) Probability of wrong selection $P_{WS}(t)$.

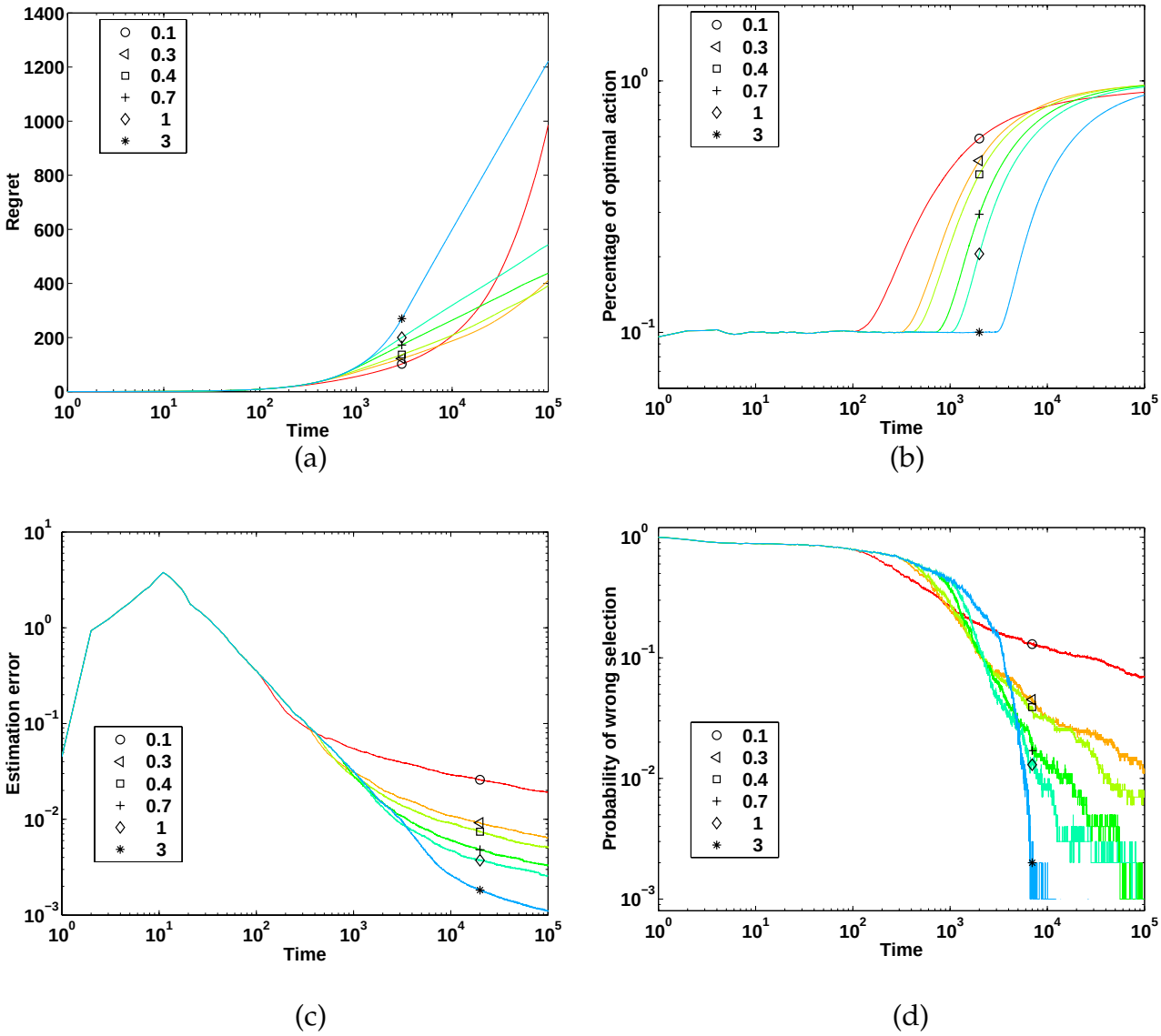


Figure 7.6: Reward maximization and exploration performance of the ϵ-greedy policy with uniformly distributed reward standard deviations. $c \in \{0.1, 0.3, 0.4, 0.7, 1, 3\}$. The best value of c is 0.4. (a) Average regret $R(t)$; (b) Average percentage of the optimal action $F(t)$; (c) Average estimation error $\bar{e}_t$; (d) Probability of wrong selection $P_{WS}(t)$.

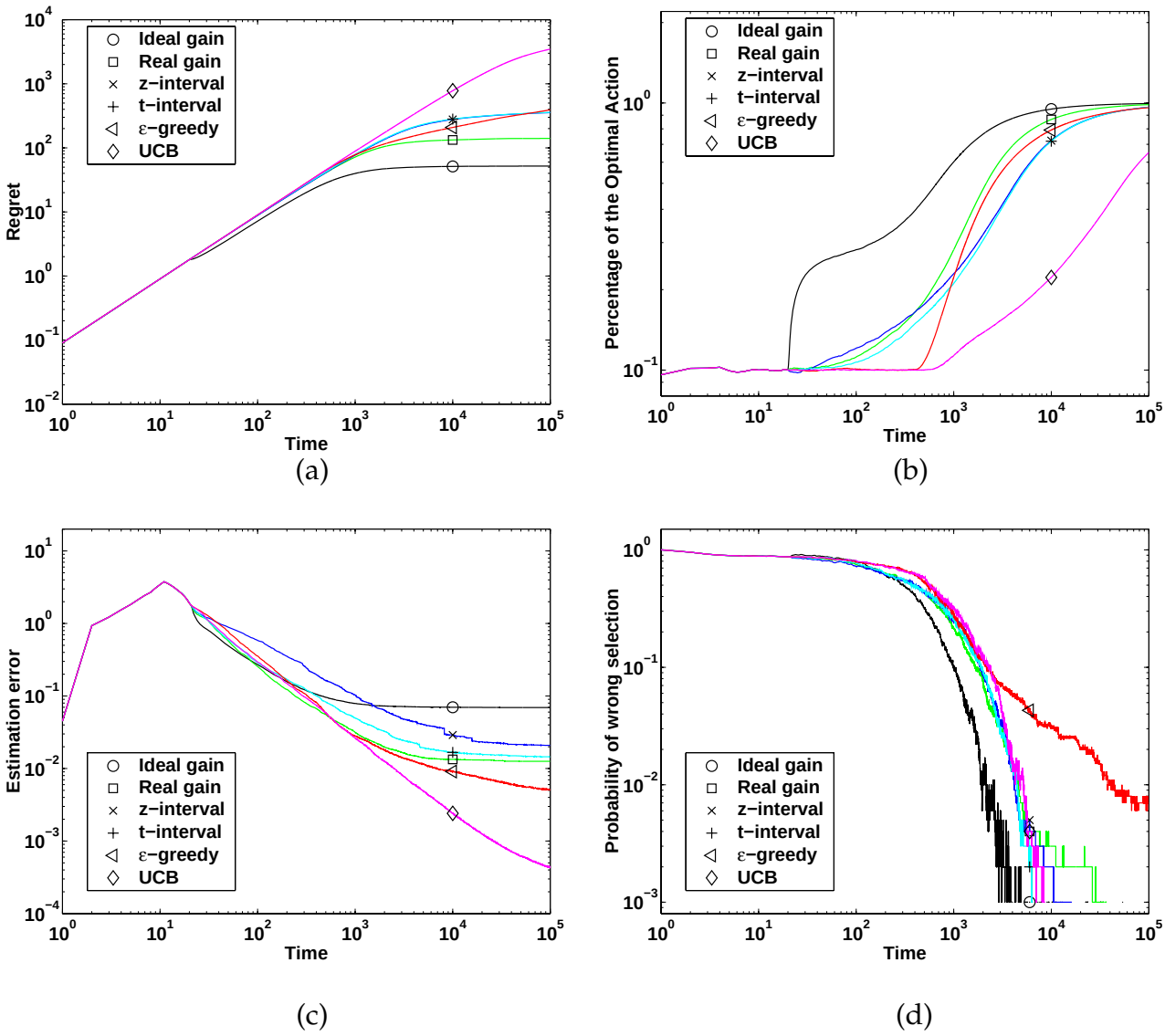


Figure 7.7: Performance comparison of reward-maximization policies with uniformly distributed reward standard deviations. The parameters are set to their best values. (a). Average regret $R(t)$; (b) Average percentage of the optimal action $F(t)$; (c) Average estimation error $\bar{e}_t$; (d) Probability of wrong selection $P_{WS}(t)$.

history, as compared to the ideal gain-based reward-maximization policy. A similar relationship is also evident between the z-interval policy ($\times$ marker) and its realistic counterpart the t-interval policy ($+$ marker). For the z-interval and t-interval policy, it takes 10^4 steps to settle down in the optimal action selection. The curves of the gain-based policies (circle and square markers) have larger slopes after about 300 time steps as compared with the z-interval policy and the t-interval policy. This results from the fact that the gain-based reward-maximization policies have a faster switching scheme. The ϵ-greedy policy (triangle marker) also has a fast rate of switching to exploitation. However, since random exploration of the ϵ-greedy policy is not efficient in detecting the optimal action, the ϵ-greedy policy has larger regret than the gain-based reward-maximization policies. For the UCB policy (diamond marker), it spends too much exploration, resulting in the most accurate reward estimation but the largest regret .

Power Law Distributed Parameters

In a second simulation, the reward standard deviations are sample from a power law distribution (the same as Eq.5.46), introducing more large reward uncertainties than in the first simulation. Therefore, more exploration is required within the time horizon considered. All policies appear logarithmic increases in regret (Fig.7.8-7.12). This seems to be characteristic for the early phase of reward maximization, when exploration is more than exploitation. Similar logarithmic increases of regret in the early phase can also be observed in the first simulation (Fig.7.7).

Compared with the first simulation, the ideal gain-based reward-maximization policy, the realistic gain-based reward-maximization policy, the z-interval policy, the t-interval policy, and the ϵ-greedy policy achieve their best performance at different parameters. Their best performance is compared with that of the UCB policy in Fig.7.13. The gain based reward maximization policies still outperform the other policies about 3 times.

The proposed gain-based reward-maximization policies outperform the other four compared policies empirically. In the linear tradeoff scheme, if the constant parameter κ is too small, the policy easily gets stuck in suboptimal actions, resulting linear regret. If κ is too large, surplus exploration induces large asymptotic regret. The z-interval policy and t-interval policy decrease slower in exploration than the gain-based reward-maximization policies. Therefore κ has longer transient effect and has a smaller range for the z-interval policy and t-interval policy to give satisfactory performance. The gain-based reward-maximization policies have a faster transient of exploration, therefore the parameter tuning is not so critical, as long as κ is larger than a certain threshold.

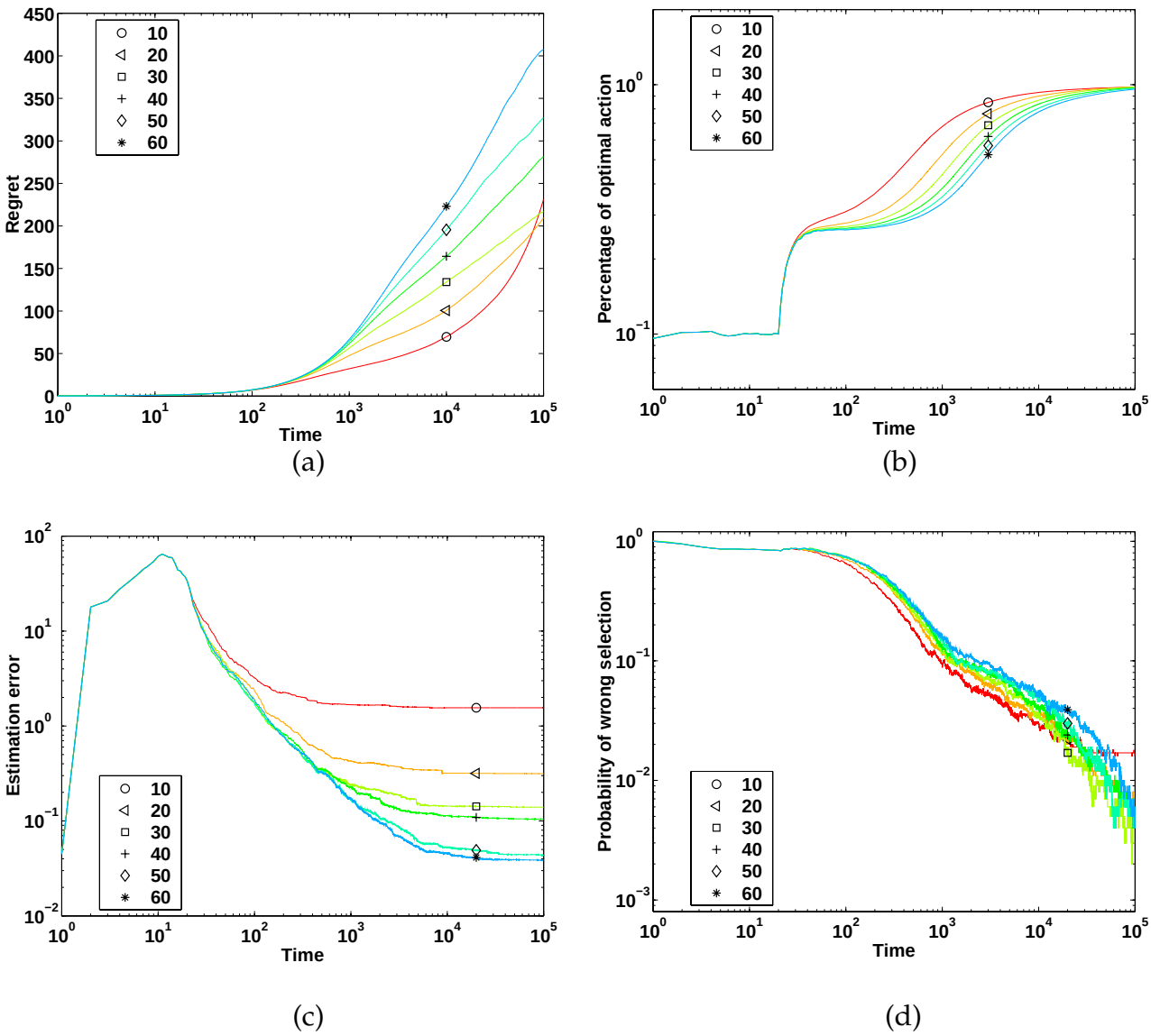


Figure 7.8: Reward maximization and exploration performance of the ideal gain-based reward-maximization policy with power-law distributed reward standard deviations. $\kappa \in \{10, 20, 30, 40, 50, 60\}$. The best value of κ is 20. (a) Average regret $R(t)$; (b) Average percentage of the optimal action $F(t)$; (c) Average estimation error $\bar{e}_t$; (d) Probability of wrong selection $P_{WS}(t)$.

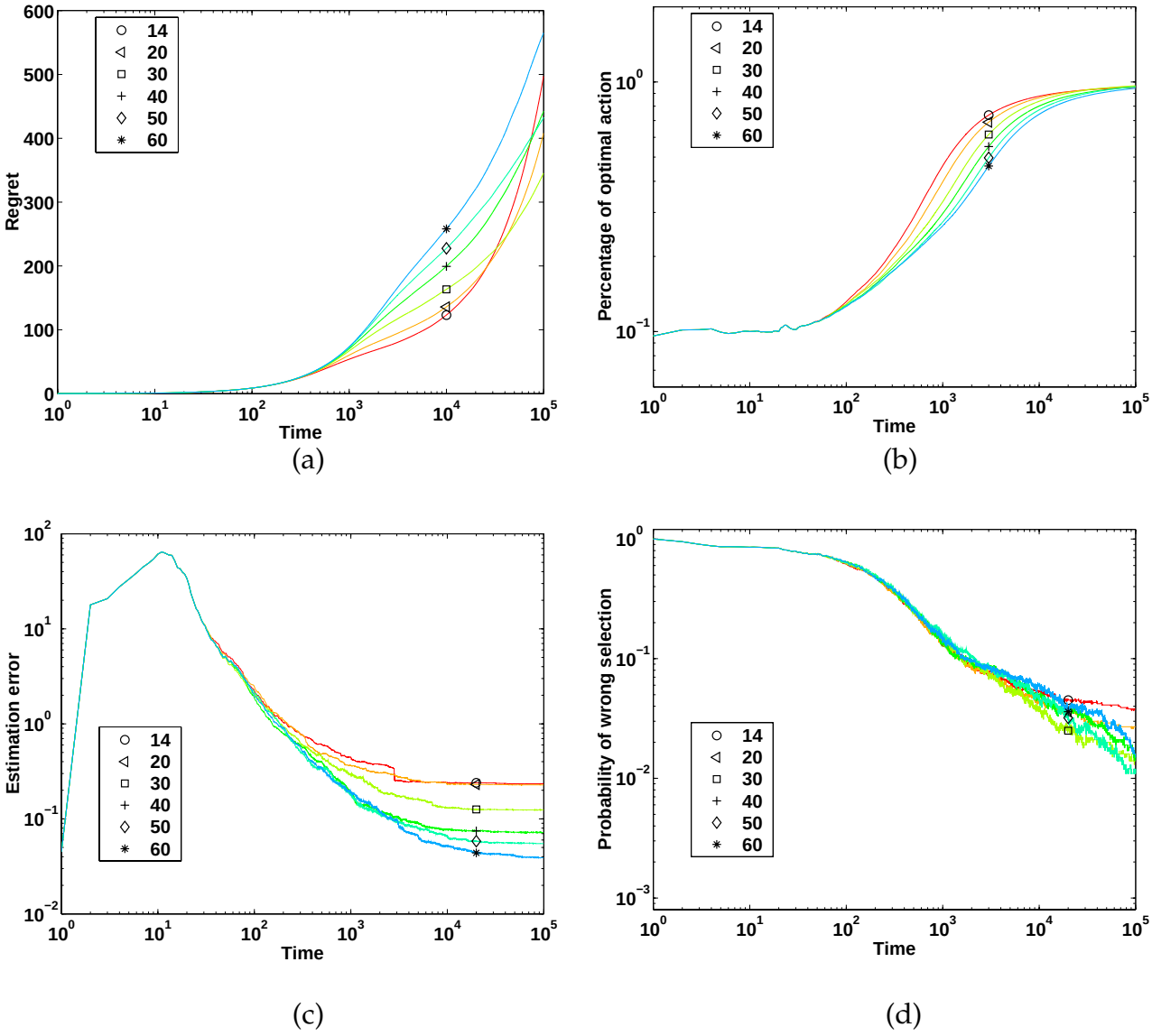


Figure 7.9: Reward maximization and exploration performance of the realistic gain-based reward-maximization policy with power-law distributed reward standard deviations. $\kappa \in \{14, 20, 30, 40, 50, 60\}, \alpha = 1e-3$. The best value of κ is 30. (a) Average regret $R(t)$; (b) Average percentage of the optimal action $F(t)$; (c) Average estimation error $\bar{e}_t$; (d) Probability of wrong selection $P_{WS}(t)$.

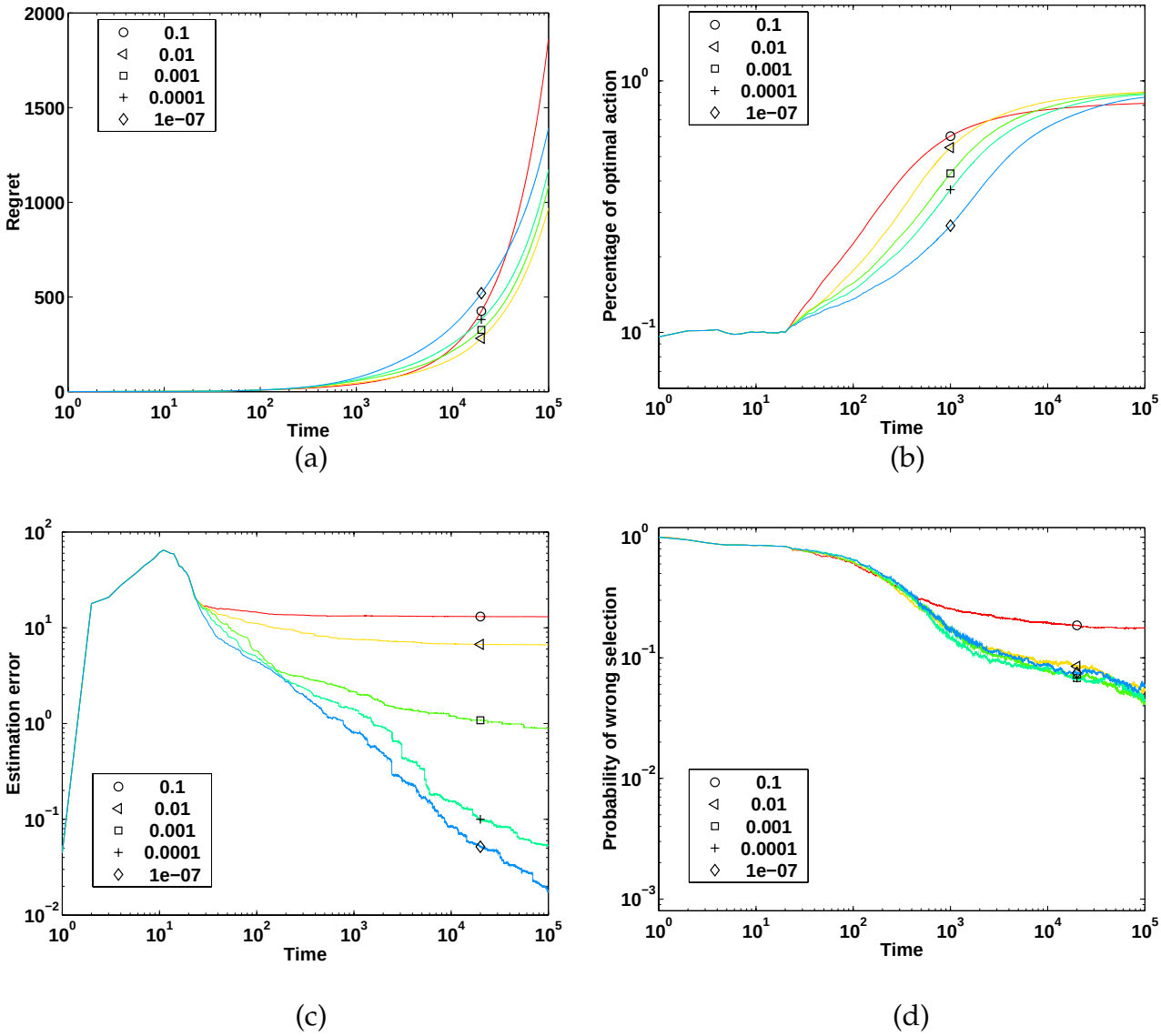


Figure 7.10: Reward maximization and exploration performance of the z-interval policy with power-law distributed reward standard deviations. $\alpha \in \{0.1, 0.01, 1e-3, 1e-4, 1e-7\}$. The best value of α is 0.01. (a) Average regret $R(t)$; (b) Average percentage of the optimal action $F(t)$; (c) Average estimation error $\bar{e}_t$; (d) Probability of wrong selection $P_{WS}(t)$.

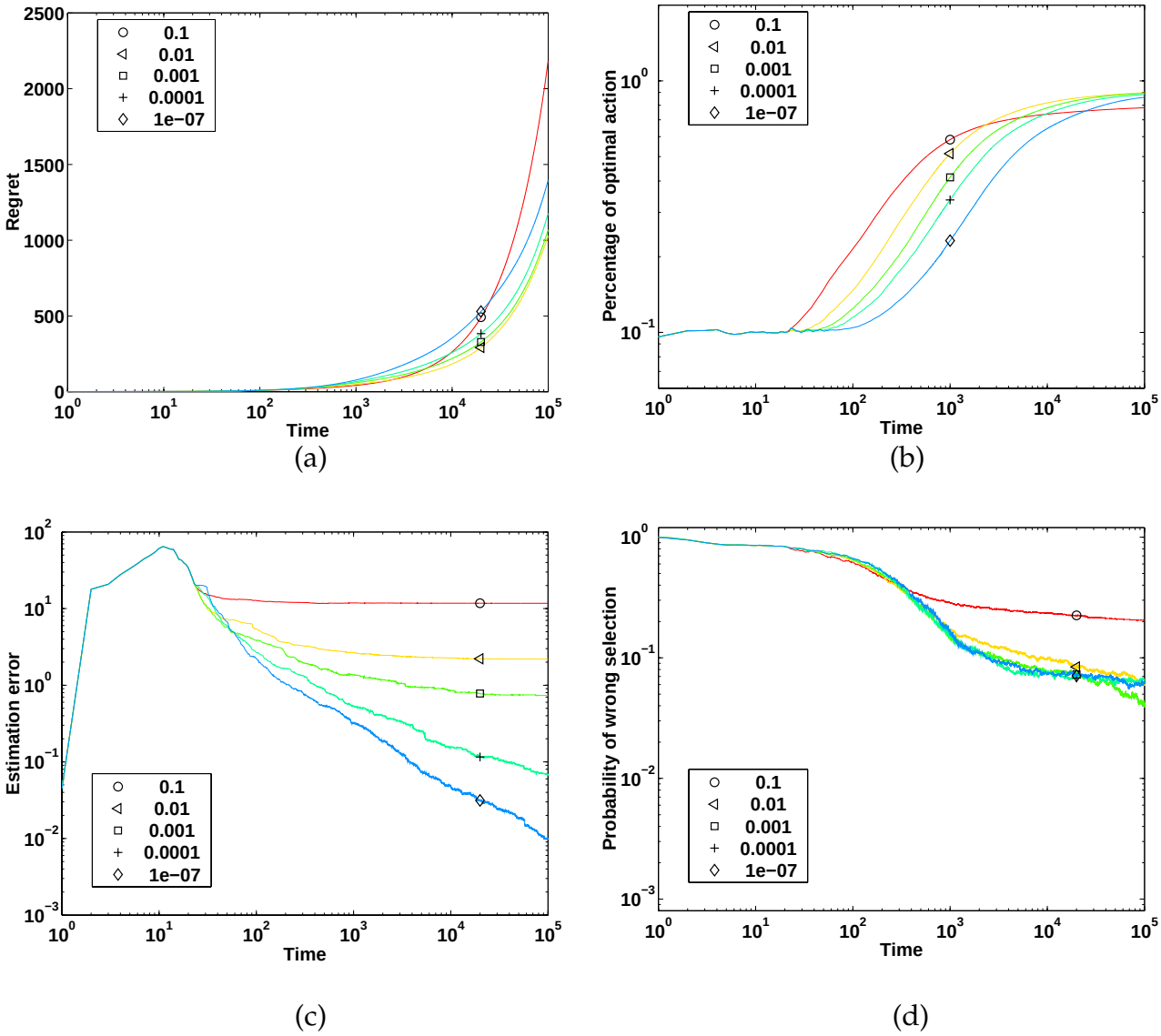


Figure 7.11: Reward maximization and exploration performance of the t-interval policy with power-law distributed reward standard deviations. $\alpha \in \{0.1, 0.01, 1e-3, 1e-4, 1e-7\}$. The best value of α is 0.01. (a) Average regret $R(t)$; (b) Average percentage of the optimal action $F(t)$; (c) Average estimation error $\bar{e}_t$; (d) Probability of wrong selection $P_{WS}(t)$.

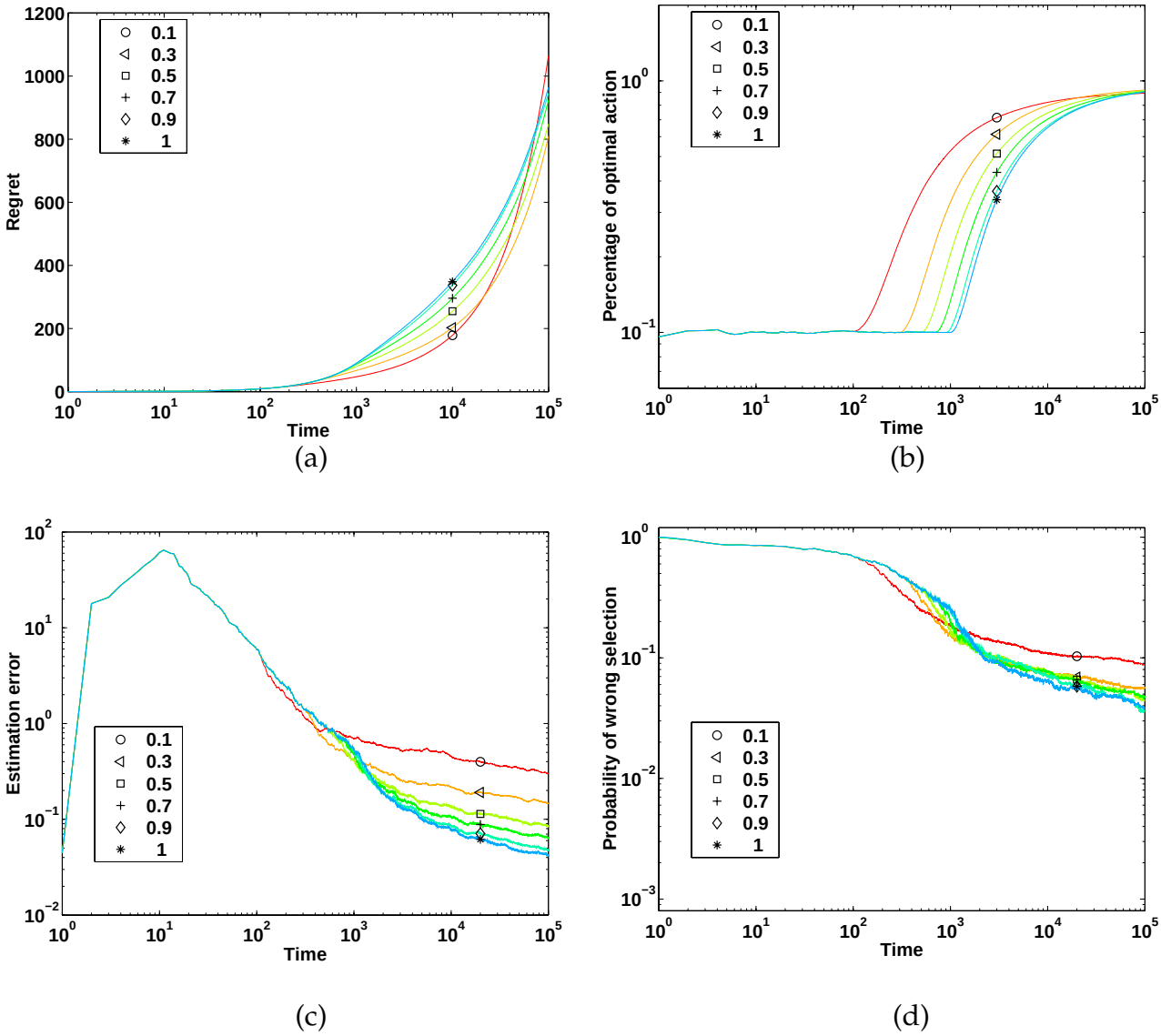


Figure 7.12: Reward maximization and exploration performance of the ϵ-greedy policy with power-law distributed reward standard deviations. $c \in \{0.1, 0.3, 0.5, 0.7, 0.9, 1\}$. The best value of c is 0.3. (a) Average regret $R(t)$; (b) Average percentage of the optimal action $F(t)$; (c) Average estimation error $\bar{e}_t$; (d) Probability of wrong selection $P_{WS}(t)$.

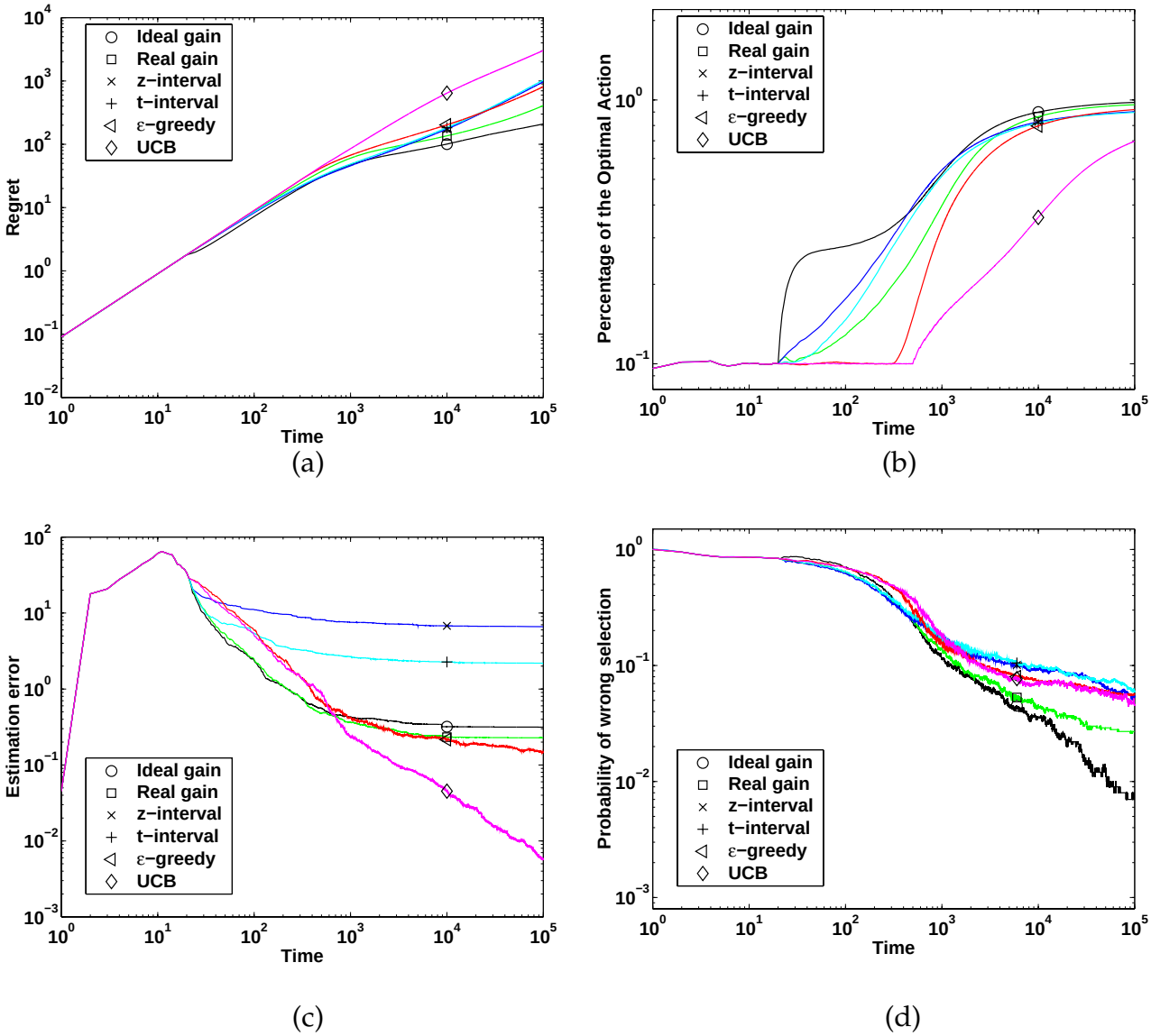


Figure 7.13: Performance comparison of reward-maximization policies with power-law distributed reward standard deviations. The parameters are set to their best values. (a). Average regret $R(t)$; (b) Average percentage of the optimal action $F(t)$; (c) Average estimation error $\bar{e}_t$; (d) Probability of wrong selection $P_{WS}(t)$.

The UCB policy provides an adaptive tradeoff weight, which has a theoretical performance guarantee for any data set. The gain-based reward-maximization policies presented here are attractive if the data set is not overwhelmingly large and parameter tuning is allowed. Their regret bounds need to be investigated in the future.

The proposed gain-based exploration and reward-maximization policies have possible applications in human decision behaviors [Art91; DOD+06], online convex optimization [FKM05], and related problems in economics [BV06], such as project management and portfolio selection [LK02]. In the next chapter, we extend the gain-based exploration to autonomous robot exploration tasks [MB99; SPH04; SHP07].

Chapter 8

Active Exploration in Partially Observable Environments

Curiosity is an intrinsic motivation to explore an environment in order to gain information from novel stimuli. Curiosity can be aroused by novel stimuli, boredom, competence motivation, and the need for sense making [Whi59; Low94]. Curiosity can be considered as intrinsic reward, since curiosity is observed to be reinforcing.

Curiosity distinguishes fundamentally from external rewards, such as food sources. External rewards are normally static. Curiosity, on the other hand, has two characteristics. First, curiosity is transient. Curiosity declines with time, because the agent will learn from the new stimuli, and be bored by a completely understood place. Second, curiosity is a reference-point phenomenon, meaning that the value of an alternative is defined relative to a reference point, which serves as the zero point of the value scale in decision making [KT79]. The same information may or may not evoke curiosity depending on one's reference point [Low94].

Since curiosity is the drive for exploration, it is important to measure curiosity when modelling exploration behavior of autonomous agents.

We measure curiosity in two ways. First, curiosity is measured by the gain in learning. As the improvement in exploration performance, the gain is a rational measure of curiosity, because curiosity is the cause of exploration behavior. Gain is essentially the derivative of the performance measure with respect to samples. Maximizing the gain, the agent maximizes the speed of learning, and therefore achieves efficient exploration. The gain function matches the characteristics of curiosity; the gain function is defined internally by the agent, asymptotically decreases with respect to the number of samples, and can be considered as self referencing.

Instead of local gain, curiosity can also be measured by a global exploration performance. Since the global exploration performance tells the agent's competence in exploration, it measures the competence motivation of the agent. The global exploration performance measure is consistent with the local gain measure, since gain maximization is equivalent to global error minimization.

The question follows: how to direct exploration by intrinsic rewards based on curiosity? We apply reinforcement learning to maximize intrinsic rewards. Reinforcement learning is learning good action selection strategies from experiences in order to maximize cumulative rewards [SB98b]. It has origins in animal behavior (e.g. conditioning), and found to have neuronal basis [SDM97]. Reinforcement learning models the interaction between a learning system and its environment by actions and rewards. Through its interaction experiences, the agent learns a value function, quantifying the utility for states or state-action pairs. From the value function, the agent infers a policy, indicating which actions are good to follow in each state. The agent can also directly searches a policy in a parameterized space by maximizing the rewards it experiences [SMSM00; Wil92]. The autonomy to learn good action selection strategies makes reinforcement learning a prominent method for many control and online optimization problems.

The remaining of this chapter is organized as follows. In section 8.1, three intrinsic reward functions are defined. Two of them are extensions of the gain functions in multi-armed bandits and local-gain adaptation. The third is an approximated global performance measure. In section 8.2, exploration is directed by a value function learned from the intrinsic rewards the agent experienced. In section 8.3, we compare the proposed active exploration with random exploration based on the rules presented in chapter4.

8.1 Intrinsic Rewards from Self Evaluation

Intrinsic motivations are normally based on self evaluations. In this section, three different measures for curiosity are derived from appropriate error measures. In section 8.1.1, an upper confidence bound on the mean squared error reduction is derived, based on the assumption that the POMDP is approximated by a set of local independent bandits. In section 8.1.2, the average local learning rate is defined as a reward function. This is because local learning rates indicate the necessity of further learning. In section 8.1.3, a global exploration performance measure is approximated internally. It gives a global performance estimation measuring the agent's competence in exploration.

8.1.1 Local Error Reduction

In chapter 5 and 6, we have studied the gain functions in the multi-armed bandit, which has only one state. In place learning, the agent has multiple states, therefore, the agent need to associate the rewards of different actions with different states.

One straightforward approximation is to model each state of the underlying POMDP as a multi-armed bandit [MB99]. After each transition between states, the agent switches to a new bandit. The gain in the new state is the reward the agent receives.

According to the result from multi-armed bandit, the gain function is the ratio of the variance over the squared sampling number (Eq.5.19). In place learning, the distributions being learned are Bernoulli distributions. Their variances and corresponding upper bounds need to be found.

For a Bernoulli distribution with p as the probability of success, its variance is $p(1-p)$. The estimator $\hat{p}_t$ of p is defined as the average of t trials. As a one-parameter distribution, the variance of the Bernoulli distribution is dependent on p. An upper bound of the variance can be formed from a confidence interval of p. A confidence interval $[L(\hat{p}_t, t), U(\hat{p}_t, t)]$ bounds the actual probability p with certain confidence. According to the Wilson interval, the confidence limits U and L are approximated by [FW86]

$$U(\hat{p}_t, t) = \frac{\hat{p}_t + \frac{z^2_{\alpha/2}}{2t} + z_{\alpha/2}\sqrt{\frac{\hat{p}_t(1-\hat{p}_t)}{t} + \frac{z^2_{\alpha/2}}{4t^2}}}{1 + \frac{z^2_{\alpha/2}}{t}}, \tag{8.1}$$

$$L(\hat{p}_t, t) = \frac{\hat{p}_t + \frac{z^2_{\alpha/2}}{2t} - z_{\alpha/2}\sqrt{\frac{\hat{p}_t(1-\hat{p}_t)}{t} + \frac{z^2_{\alpha/2}}{4t^2}}}{1 + \frac{z^2_{\alpha/2}}{t}}. \tag{8.2}$$

Here t is the number of sample used to estimate $\hat{p}_t$. $z_{\alpha/2}$ denotes the variate value from the standard normal distribution such that the area to the right of the value is $\alpha/2$.

Replacing the true probability p by its upper and lower confidence limits, gives rise to an upper bound of the variance

$$p(1-p) < U(\hat{p}_t, t)\big(1 - L(\hat{p}_t, t)\big). \tag{8.3}$$

Therefore, we get a gain function as intrinsic reward signal r_t for the action a_t

$$r_t := \hat{g}_{t+1} = \frac{\sum_s U\big(\hat{p}_{t+1}(s), \hat{\tau}_{t+1}\big)\Big(1 - L\big(\hat{p}_{t+1}(s), \hat{\tau}_{t+1}\big)\Big)}{\hat{\tau}^2_{t+1}}, \tag{8.4}$$

where $\hat{\tau}_{t+1}$ is the average sampling number

$$\hat{\tau}_{t+1} = \sum_i \tau_{t+1}(i)\hat{p}_{t+1}(i). \tag{8.5}$$

$\tau_{t+1}(i)$ is the counter of the beliefs, and is initialized as 1

$$\tau_{t+1}(i) = \tau_t(i) + \hat{p}_{t+1}(i). \tag{8.6}$$

Since the sensor model $\hat{p}_0(s|i)$ is initialized as $1/S$, the initial value of r_t is its maximal value, denoted by $R = SU(1/S, 1)\big(1 - L(1/S, 1)\big)$. r_t has a trend to decrease towards zero.

8.1.2 Local Learning Rate

In chapter 3 and 4, we show how the local learning rates are adapted by the gradient of the loss function with respect to the logarithm of the learning rates. The local learning rate is called gain because it controls how large the gradient steps take effect. A big local learning rate means that the convergence is still far away, and that more gradient descent steps are necessary. A small local learning rate indicates that the gradient steps are in the final stages. Therefore, the local learning rate is a natural gain signal in place learning.

The average local learning rate at the new location is defined as an intrinsic reward signal for action a_t

$$r_t := \sum_i \eta_t(i)\hat{p}_{t+1}(i), \tag{8.7}$$

where $\eta_t(i)$ is the local learning rate (Eq.4.12).

The initial value of $\eta_t(i)$ is denoted by $R = \eta_0$.

8.1.3 Global Performance Estimation

The previous two gain functions are defined as local gain in reducing prediction error. As shown in chapter 5, gain maximization is equivalent to global error minimization. Therefore, in this section, an approximation of the global error is directly minimized in action selection.

The Kullback-Leibler distance between the objective sensory-position distribution $p(s, z)$ and the recorded distribution $q_t(s, z)$ measures how well the agent

explores the environment and estimates the sensory input in information theoretic terms

$$E_{obj}(t) = \sum_{s,z} p(s,z) \log \frac{p(s,z)}{q_t(s,z)}, \tag{8.8}$$

where z is the location index in the environment, and s is the sensory input.

According to the chain rule. the objective distribution $p(s,z)$ is factorized into

$$p(s,z) = p(s|z)p(z), \tag{8.9}$$

where $p(s|z)$ is the true observation probability. $p(z)$ is the desired visiting frequency over location z. For exploration, the agent should cover all places, so $p(z)$ is a uniform distribution over all locations

$$p(z) = \frac{1}{N}, \tag{8.10}$$

with N as the number of locations/states. We assume that the number of states is the same as the number of locations.

By similar factorization, the empirical sensory-position distribution $q_t(s,z)$ is defined as

$$q_t(s,z) = q_t(s|z)q_t(z), \tag{8.11}$$

where $q_t(s|z)$ and $q_t(z)$ are the sensory prediction and visiting frequency recorded by an ideal observer

$$q_t(s|z) = \begin{cases} \frac{1}{S}, & \text{location } z \text{ has never been visited} \\ \hat{p}_{\tau(z)}(s), & \tau(z) \in [1,t] \text{ is the latest time location } z \text{ is visited} \end{cases},$$

$$q_t(z) = \frac{1}{t} \sum_{\tau=1}^{t} \delta_{z,z_\tau}. \tag{8.12}$$

Here z_τ is the agent's true location at time τ. Since z_τ is not available to the agent, $q_t(z)$ is not known for the agent. However, it can be approximated by a trace $f_t(i)$ of the location belief

$$f_t(i) = \frac{1}{t} \sum_{\tau=1}^{t} \hat{p}_\tau(i) \tag{8.13}$$

$$\approx (1-\xi) f_{t-1}(i) + \xi \hat{p}_\tau(i), \tag{8.14}$$

where $\xi \in (0,1)$ is a fixed averaging factor. $f_t(i)$ is initialized as $1/N$.

These definitions lead Eq.8.8 to be

$$\begin{aligned} E_{obj}(t) &= \sum_{s,z} p(s,z)\log p(s,z) - \frac{1}{N}\sum_{s,z} p(s|z)\log q_t(s|z) - \frac{1}{N}\sum_{s,z} p(s|z)\log q_t(z) \\ &= Cons. - \frac{1}{N}\sum_{s,z} p(s|z)\log q_t(s|z) - \frac{1}{N}\sum_{z}\log q_t(z). \end{aligned} \tag{8.15}$$

$E_{obj}(t)$ is the relative entropy of two distributions: the ideal joint distribution of homogeneous position distribution and the true observation probability, and the empirical visiting frequency $q_t(z)$ and the sensory prediction $q_t(s|z)$. It measures the distance of the agent's predictive behavior to the objective behavior. The objective of the agent is to minimize this measure.

Since the true position z and the observation probability $p(s|z)$ are only known by ideal observers, $E_{obj}(t)$ is not measurable by the agent. Therefore $E_{obj}(t)$ is called objective objective function. It needs to be approximated by the agent's internal statistics before the agent can use it as an exploration criterion.

The first term in Eq.8.15, the negative of the entropy of the objective sensory-position distribution, is a constant, therefore it is not interesting for optimization.

The last term is approximated by its corresponding internal representation

$$-\frac{1}{N}\sum_{z}\log q_t(z) \approx -\frac{1}{N}\sum_{i}\log f_t(i). \tag{8.16}$$

The second term of Eq.8.15 is approximated by e_t, replacing the sum over true location z by the sum over path history

$$\begin{aligned} e_t &= -\frac{1}{N}\sum_{s,z} p(s|z)\log q_t(s|z) \\ &= -\frac{1}{N}\sum_{s,z}\frac{q_t(z)}{q_t(z)}p(s|z)\log q_t(s|z) \\ &= = -\frac{1}{Nt}\sum_{s,z}\sum_{\tau=1}^{t}\frac{\delta_{z,z_\tau}}{q_t(z)}p(s|z)\log q_t(s|z) \\ &= -\frac{1}{Nt}\sum_{\tau=1}^{t}\sum_{s}\frac{1}{q_t(z_\tau)}p(s|z_\tau)\log q_t(s|z_\tau) \end{aligned}$$

$$\approx -\frac{1}{Nt}\sum_{\tau=1}^{t}\sum_{s}\frac{1}{q_t(z_\tau)}\delta_{s,s_\tau}\log\hat{p}_\tau(s) \tag{8.17}$$

$$= -\frac{1}{Nt}\sum_{\tau=1}^{t}\frac{1}{q_t(z_\tau)}\log\hat{p}_\tau(s_\tau)$$

$$\approx \frac{1}{Nt}\sum_{\tau=1}^{t}\frac{\ell_{KL}(\tau)}{\hat{\rho}_\tau} \tag{8.18}$$

$$\approx (1-\xi)e_{t-1} + \xi\frac{1}{N}\frac{\ell_{KL}(t)}{\hat{\rho}_t}. \tag{8.19}$$

Eq.8.17 is obtained by replacing the sensory distribution $p(s|z_\tau)$ by the observation δ_{s,s_τ}. This is true on average. Eq.8.18 is due to approximating the true visiting frequency by the internal belief trace

$$\hat{\rho}_t = \sum_i f_t(i)\hat{p}_t(i). \tag{8.20}$$

Eq.8.19 uses a fixed averaging rate ξ to forget the old prediction errors. The factor $\frac{\frac{1}{N}}{\hat{\rho}_\tau}$ is the ratio between the objective visiting frequency and the agent's local estimation of its visiting frequency, and is used to rectify the bias caused by local exploration. e_t is initialized as $\log S$, where S is the number of possible observations s.

Combining Eq.8.16 and 8.19, leads to an internal measure $E_{sub}(t)$

$$E_{sub}(t) = e_t - \frac{1}{N}\sum_i \log f_t(i). \tag{8.21}$$

Thus, the objective measure $E_{obj}(t)$ is internalized into a subjective objective function $E_{sub}(t)$. $E_{sub}(t)$ is large if the average prediction error e_t is large or if the belief trace $f_t(i)$ is very localized. Therefore, $E_{sub}(t)$ measures how well the agent predicts the sensory information and how broadly the agent explores the environment. Directly minimizing $E_{sub}(t)$, the agent achieves a balance between assuring good prediction and exploring the unknown parts of the environment.

An intrinsic reward signal to be maximized then becomes

$$r_t := -E_{sub}(t+1). \tag{8.22}$$

r_t is always negative, therefore the maximal reward R is defined as $R = 0$.

8.2 Action Selection by Maximizing Intrinsic Rewards

The intrinsic reward r_t defined in the previous section is the immediate reward for action a_t. It is possible that larger rewards appear in some other places. The best choice is just go to the place and select the action which has the maximal immediate reward. However, the agent need to know the trajectory leading to this place. After taking a action, the agent normally results in a different place.

Therefore, instead of maximizing the immediate rewards, the agent maximizes the immediate rewards and delayed rewards along its trajectory. This is done by two mechanisms. First, the agent discounts its future rewards by a factor $\gamma \in [0, 1)$, and evaluates the utility of an action by the combination of its immediate reward and discounted delayed rewards. Second, the agent maintains a trace of its trajectory, and propagates the current immediate reward backwards along the trajectory. In this way, when the agent visits the states in the trajectory next time, it knows the future rewards along this path.

8.2.1 Action Selection

The classical methods for POMDPs assume that the reward function is stationary [Cas98; Mur05]. Since they aim at exact solutions, the computation complexity is rather high. In place learning, the gain function is diminishing, therefore, classical methods are not suitable.

Here the action selection mechanism of the agent is implemented by a linear network, which follows the non-stationary reward signals. The weights of the network encode the estimated action values. The value of taking action a in place z is defined as the expected discounted reward the agent receives starting from position z following its policy π_t

$$Q_t(a, z) = E_{\pi_t}\left[r_t + \sum_{k=1}^{\infty} \gamma^k r_{t+k}\right], \tag{8.23}$$

where $\gamma \in [0, 1)$ is a discount factor indicating how important the future rewards are for the current action. The reward signal r_t is transient, therefore the true action value $Q_t(a, z)$ is also transient.

Since the agent does not know its true location, it maintains an internal estimation of action value, denoted by $\hat{Q}_t(a, i)$. At each time step, the agent predicts the action values according to its current belief $\hat{p}_t(i)$, and selects a greedy ac-

tion according to its current prediction

$$a_t = \arg\max_a \hat{Q}_t(a) \tag{8.24}$$

$$\hat{Q}_t(a) = \sum_i \hat{Q}_t(a,i)\hat{p}_t(i), \tag{8.25}$$

where $\hat{Q}_t(a,i)$ is the estimated action values. $\hat{Q}_t(a)$ is the agent's prediction of action values for current location, and $\hat{p}_t(i)$ is the belief state. $\hat{Q}_t(a,i)$ is initialized as the normalized value $R/(1-\gamma)$, since the sum of the weights in Eq.8.23 is $\sum_{k=0}^{\infty}\gamma^k = 1/(1-\gamma)$.

8.2.2 Value Update

After executing action a_t, the agent gets the reward r_t, and then the action value for a_t is updated similarly like Q-learning

$$\hat{Q}_{t+1}(a,i) = \hat{Q}_t(a,i) + \theta\Big(r_t + \gamma \max_a \hat{Q}_{t+1}(a) - \hat{Q}_t(a_t)\Big)e_t(a,i), \tag{8.26}$$

where $\theta \in [0,1]$ is a positive learning rate. $\hat{Q}_{t+1}(a)$ is the predicted action value for the new belief $\hat{p}_{t+1}(i)$ according to Eq.8.25. $e_t(i,a)$ is the eligibility trace, memorizing how recently the states are visited

$$e_t(a,i) = \gamma\lambda e_{t-1}(a,i) + \hat{p}_t(i)\delta_{a,a_t}. \tag{8.27}$$

The value update rule (Eq.8.26) is a gradient descent rule derived by minimizing the value prediction error

$$E_Q(t) = \frac{1}{2}\Big(r_t + \gamma \max_a \hat{Q}_{t+1}(a) - \hat{Q}_t(a_t)\Big)^2, \tag{8.28}$$

where $\hat{Q}_t(a_t)$ is the agent's previous prediction on the value of action a_t. $\big(r_t + \gamma \max_a \hat{Q}_{t+1}(a)\big)$ is a bootstrap approximation of the true value of a_t, since the action value is the sum of the current reward and the discounted future rewards (Eq.8.23).

The gradient of the prediction error $E_Q(t)$ with respect to $\hat{Q}_t(a,i)$ is

$$\begin{aligned}\frac{\partial E_Q(t)}{\partial \hat{Q}_t(a,i)} &= \frac{\partial E_Q(t)}{\partial \hat{Q}_t(a_t)}\frac{\partial \hat{Q}_t(a_t)}{\partial \hat{Q}_t(a,i)} \\ &= -\Big(r_t + \gamma \max_a \hat{Q}_{t+1}(a) - \hat{Q}_t(a_t)\Big)\hat{p}_t(i)\delta_{a,a_t}. \end{aligned} \tag{8.29}$$

Here $\hat{p}_t(i)\delta_{a,a_t}$ controls which state and action is responsible for the value prediction error. Since the current reward is a result of all previous actions and

states, the current time difference error $\left(r_t + \gamma \max_a \hat{Q}_{t+1}(a) - \hat{Q}_t(a_t)\right)$ can be credited to all these states and actions. This is done through the eligibility trace $e_t(a, i)$, resulting in the gradient descent rule of Eq.8.26.

The time difference error $\left(r_t + \gamma \max_a \hat{Q}_{t+1}(a) - \hat{Q}_t(a_t)\right)$ plays an important role. If it is positive, meaning that the new value $\left(r_t + \gamma \max_a \hat{Q}_{t+1}(a)\right)$ is larger than the old value $\hat{Q}_t(a_t)$, then the values of the actions in the eligibility trace will be increased. These actions have higher tendency to be selected in the future. Since the rewards r_t are transient, after repeated selections, the time difference error will become negative. According to the greedy action selection rule (Eq.8.24), the agent will switch to other less explored actions.

8.3 Simulation Results

The simulation condition is the same as in chapter 4. Active exploration is applied for the counting rule and the adaptive multiplicative rule. When the counting rule is used, the reward is the local error reduction (Eq.8.4). When the adaptive multiplicative rule is used, the reward is the averaged local learning rate (Eq.8.7) or the global performance estimation (Eq.8.22). The resulting active explorations are compared with the random explorations presented in chapter 4.

8.3.1 Active Exploration with Additive Gradient Descents

In the first experiment, the counting rule is used with the same learning rates as in chapter 4. The local error reduction (Eq.8.4) is maximized in the active exploration. As before, exploration performance is compared with respect to the averaged square error $E_{SQ}(t)$ of the ten simulated runs.

Fig. 8.1 shows that the active exploration (square marker) outperforms the same counting rule under random exploration (marked by diamond) by 25%. This advantage is still pronounced even when the motion noise is as large as 20% ($\sigma_M = 0.2$).

In the beginning, the active exploration with the counting rule is worse than random exploration with the additive rule (triangle marker). But after 10^4 steps, the active exploration is consistently better than the latter. This is because the initial learning rates used by the counting rule are larger than that of the additive rule, and pushes the estimation near the boundary values.

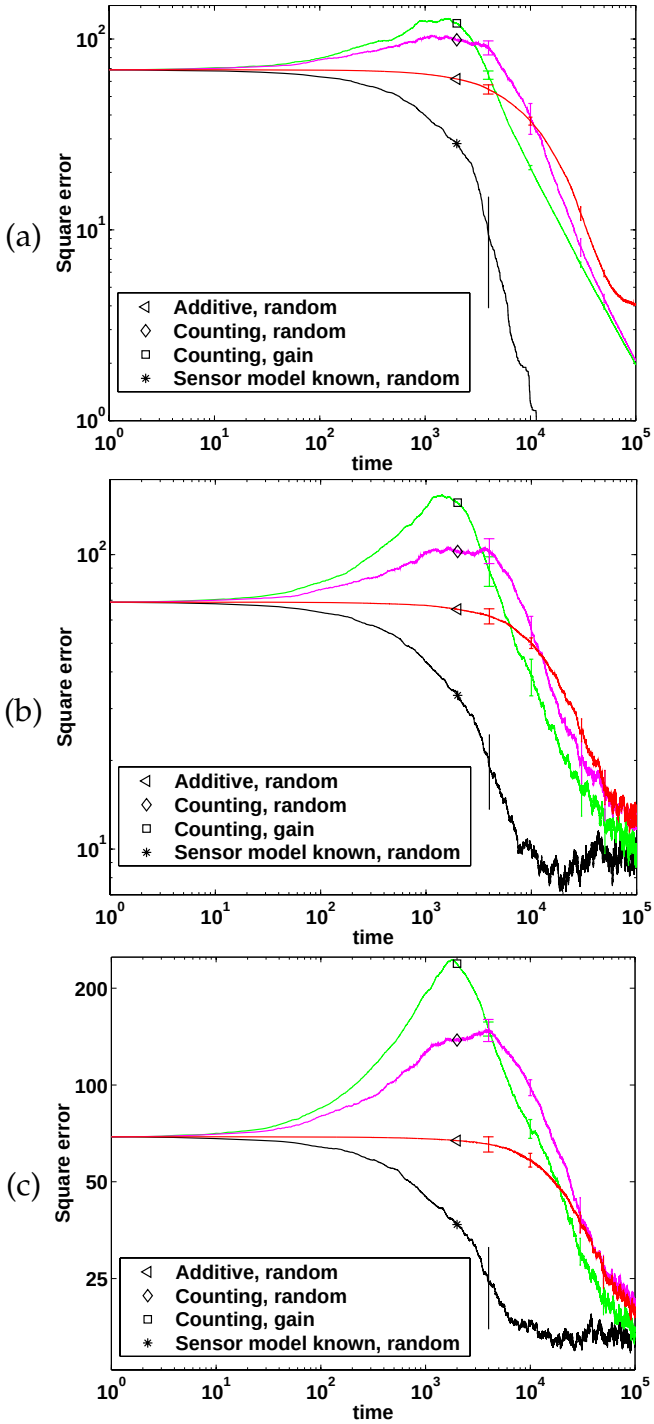


Figure 8.1: Active exploration using the counting rule compared with random exploration, under different motion noise levels. The error bars depict $\pm$ standard deviation. The learning rates η and μ used for sensor model are the same as in chapter 4. (a) No motion noise ($\sigma_M = 0$). Learning rates are $\theta = 0.1, \gamma = 0.95, \lambda = 0.1$; (b) Median motion noises ($\sigma_M = 0.1$). Learning rates are $\theta = 0.1, \gamma = 0.1, \lambda = 0.5$; (c) Large motion noises ($\sigma_M = 0.2$). Learning rates are $\theta = 0.1, \gamma = 0.9, \lambda = 0.1$.

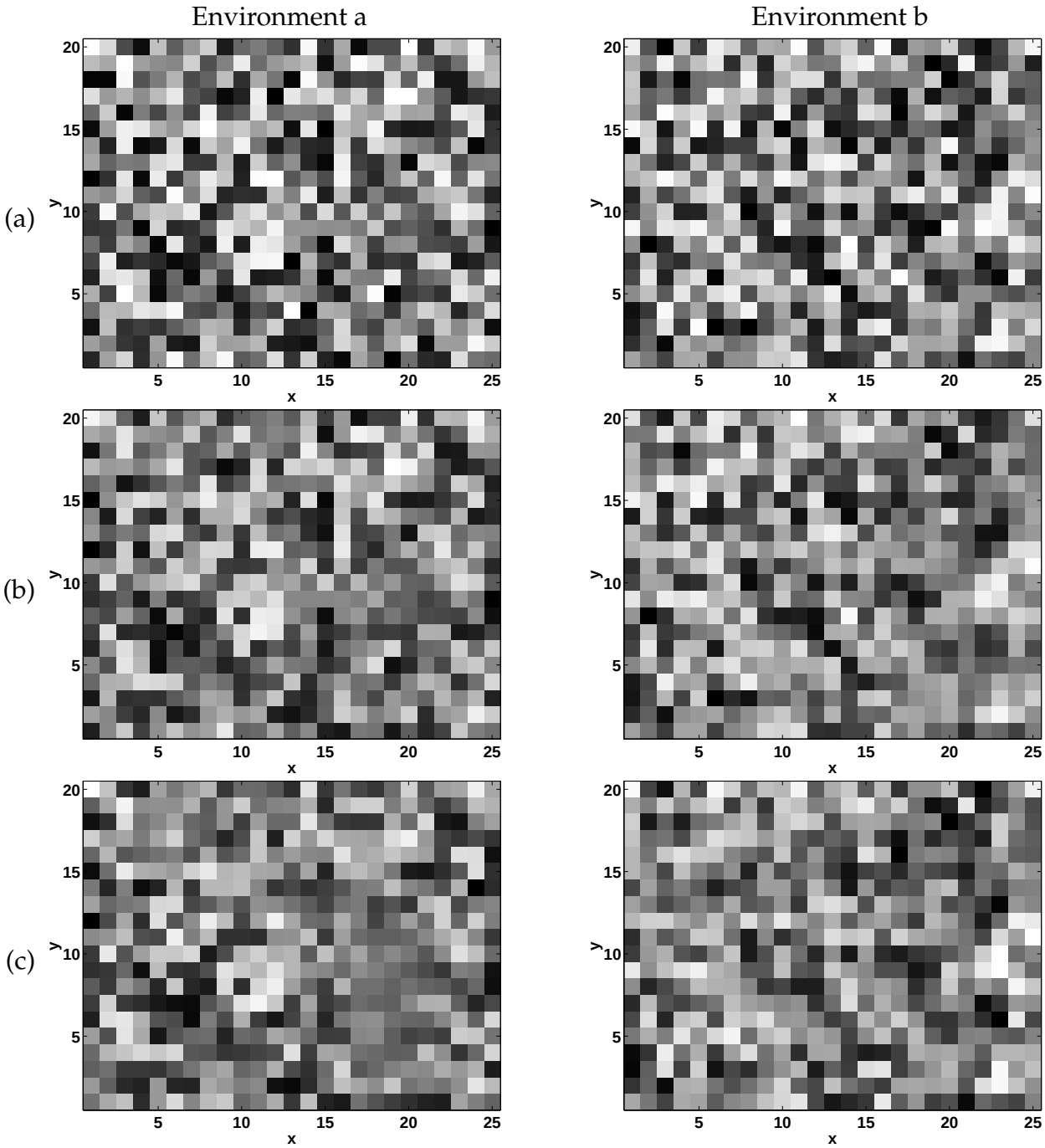


Figure 8.2: Occupancy maps constructed from prediction records $q_t(s|z)$ for two environments under different exploration schemes using the counting rule ($\sigma_M = 0.2, t = 10^5$). (a) True maps of the environments; (b) Active exploration directed by the local error reduction; (c) Random exploration. The maps learned by active exploration have less and uniform uncertainty. The uncertainty in maps learned by random exploration is not uniformly reduced.

In Fig. 8.2, each column lists the occupancy maps of the same environment constructed at the end of exploration (10^5 time steps). The maps learned under active exploration are sharper than those obtained by random exploration. The reason is that for random exploration some places do not receive sufficient samples, resulting in large uncertainty in some areas. However, the active exploration allocates samples in an efficient way. Therefore the uncertainty in the map is evenly reduced.

Interestingly, under the active exploration, the agent produces an exploration behavior similar to the frontier-based exploration [Yam97]. For example, Fig. 8.3 shows some snapshots of the exploration process in environment a under accurate odometry. With active exploration, the agent gradually expands its territory, forming a compact frontier. However, in random exploration, the exploration is not systematic, and the resulting exploration pattern is loosely connected and scattered in a random manner. Fig. 8.4 demonstrates a similar difference between exploration patterns when the motion noise is large ($\sigma_M = 0.2$)[1].

Fig. 8.5 and 8.6 compare the difference between exploration behaviors in environment b under accurate and highly noisy locomotion respectively. In this case, random exploration also forms a fairly compact exploration frontier. However, it is slower to traverse the whole environment as compared with the active exploration.

8.3.2 Active Exploration with Multiplicative Gradient Descents

In the second simulation, the adaptive multiplicative rule is used in learning, with the same learning rates as used in chapter 4. Two active explorations are evaluated, maximizing the local learning rate (Eq.8.7) or minimizing the global performance estimation (Eq.8.22) respectively. Exploration performance is compared with respect to the averaged prediction error $E_{KL}(t)$.

The two resulting active explorations have similar performance, with the active exploration directed by the local learning rate (square marker) slightly better than the active exploration directed by the global performance estimation (circle marker).

Compared with random exploration (diamond marker) using the same adaptive multiplicative rule, active explorations are consistently better than random exploration under different levels of motion noise. The biggest advantage

[1]videos are available from `http://www.neuro.uni-bremen.de/~bailu/video/index.html`

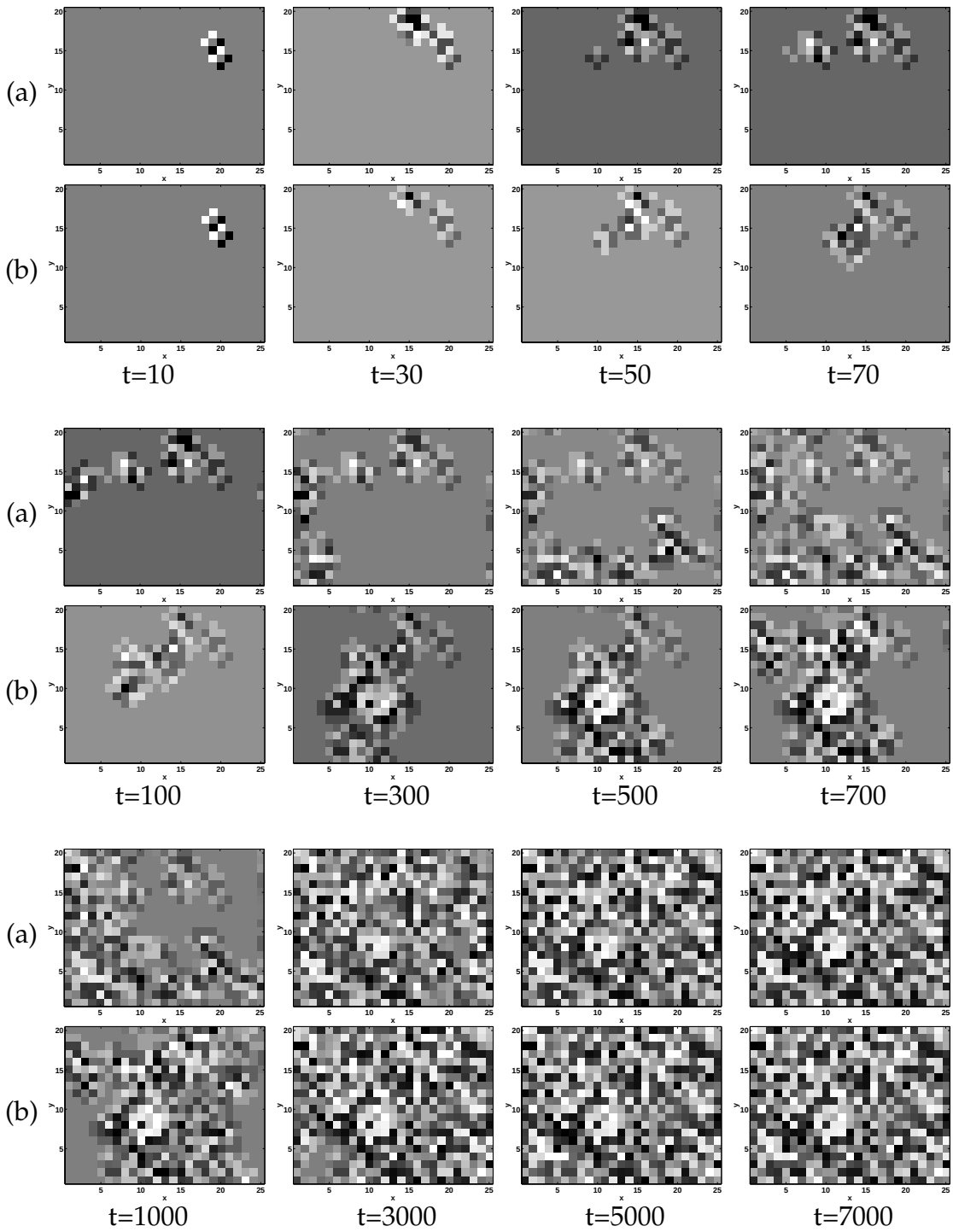


Figure 8.3: Exploration of environment a using the counting rule when there is no motion noise ($\sigma_M = 0$). The evolution of the constructed occupancy maps is shown in rows with respect to time. The maps are scaled to use the full color codes (not shown). (a) Random exploration; (b) Active exploration directed by local error reduction.

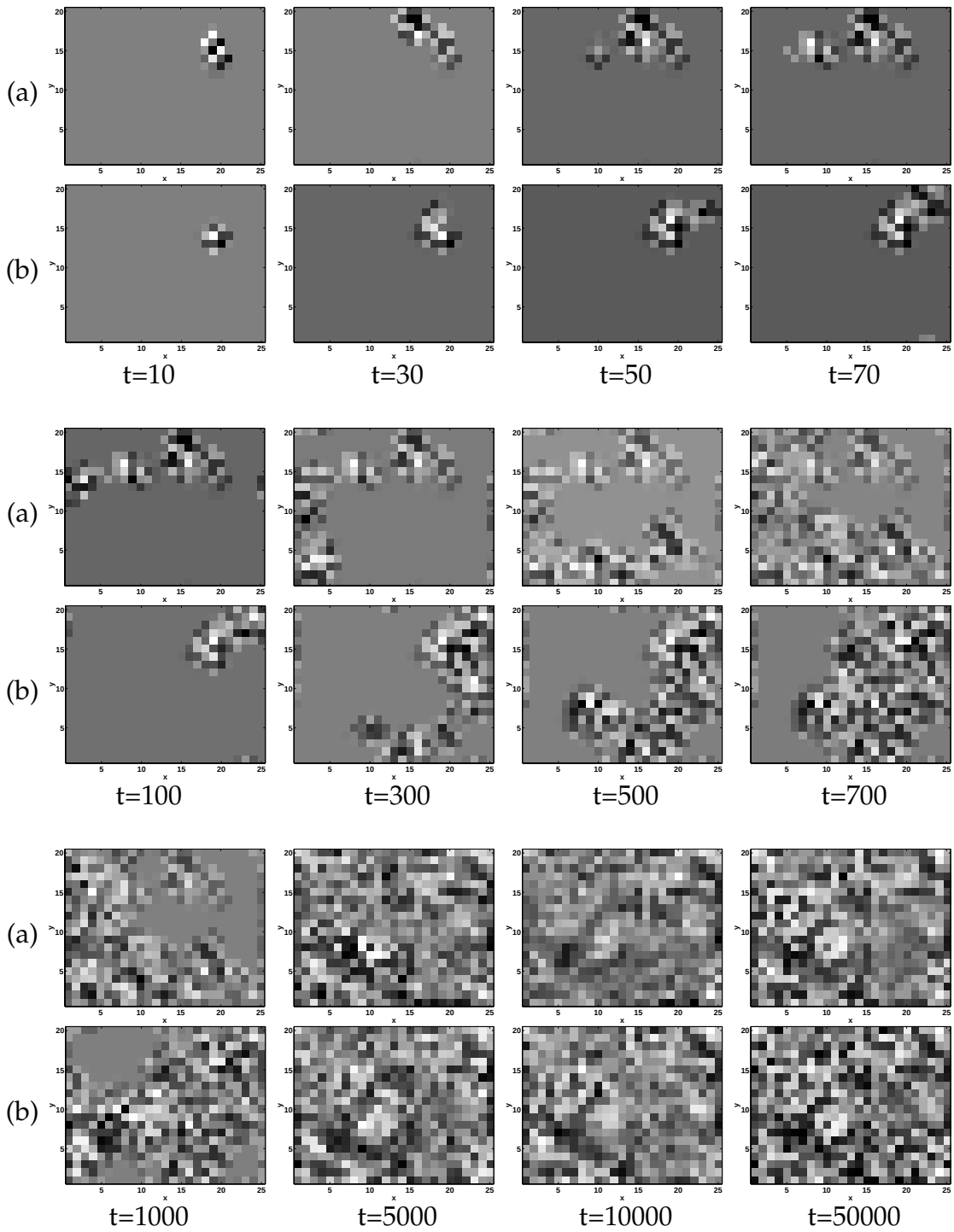


Figure 8.4: Exploration of environment a using the counting rule under large motion noises ($\sigma_M = 0.2$). The evolution of the constructed occupancy maps is shown in rows with respect to time. The maps are scaled to use the full color codes (not shown). (a) Random exploration; (b) Active exploration directed by local error reduction.

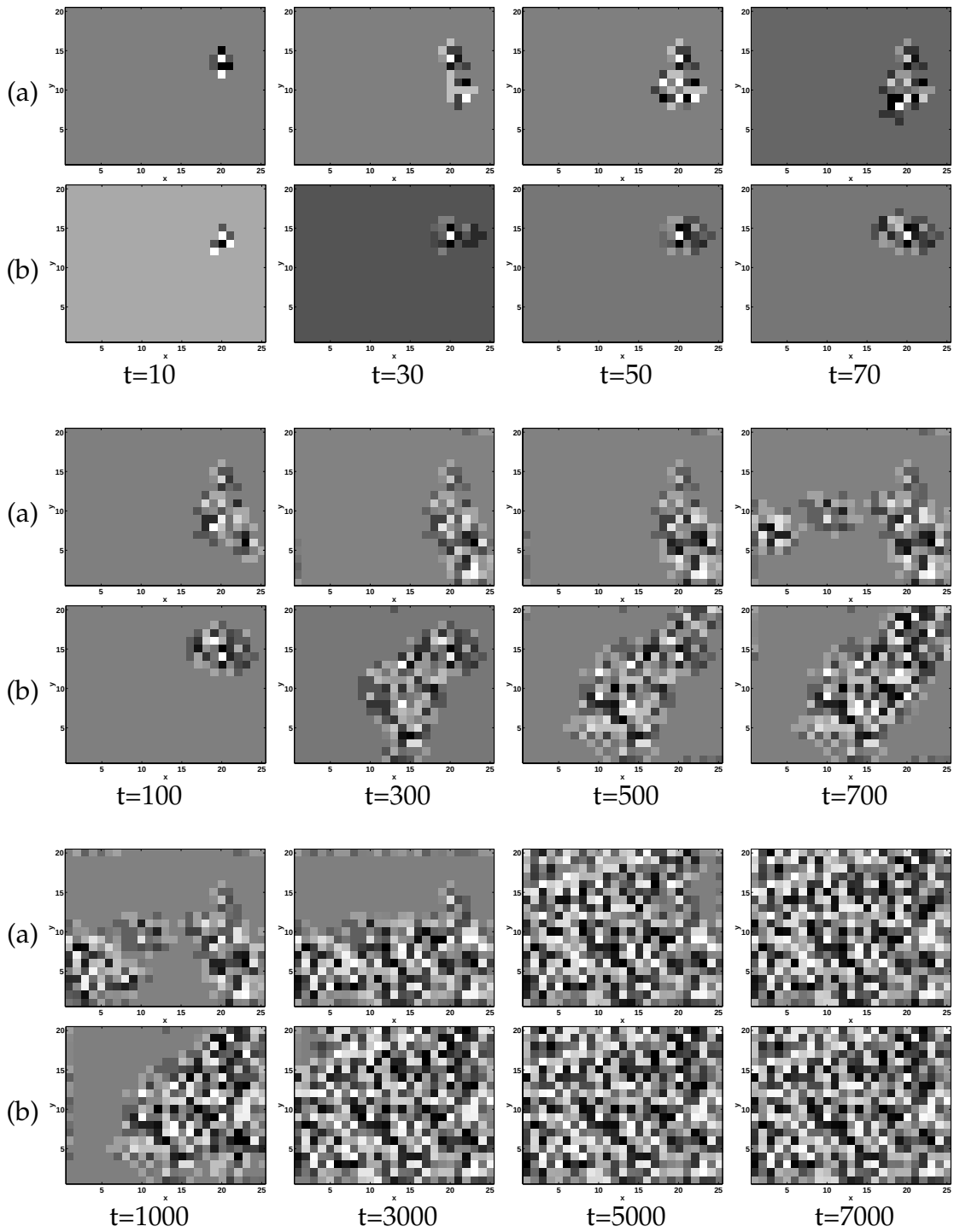


Figure 8.5: Exploration of environment b using the counting rule with accurate odometry ($\sigma_M = 0$). The evolution of the constructed occupancy maps is shown in rows with respect to time. The maps are scaled to use the full color codes (not shown). (a) Random exploration; (b) Active exploration directed by local error reduction.

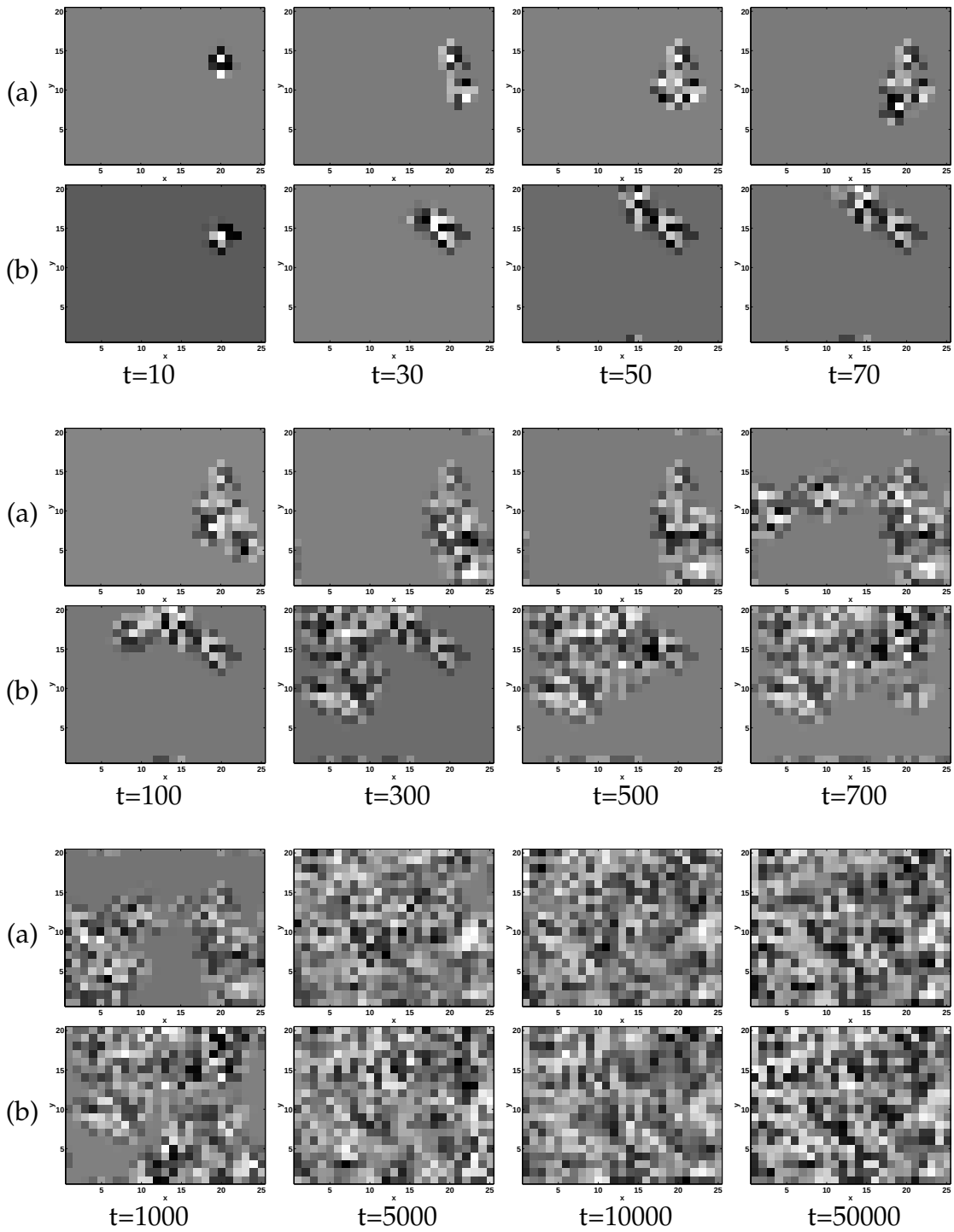


Figure 8.6: Exploration of environment b using the counting rule with large motion noises ($\sigma_M = 0.2$). The evolution of the constructed occupancy maps is shown in rows with respect to time. The maps are scaled to use the full color codes (not shown). (a) Random exploration; (b) Active exploration directed by local error reduction.

is about 20%, when the odometry is accurate (Fig. 8.7(a)). With increasing motion noses, the advantage of active explorations decreases to about 10% (Fig. 8.7(c)).

Compared with random exploration using the multiplicative rule (triangle marker), the active explorations with the adaptive multiplicative rule are more superior. Under large motion noises ($\sigma_M = 0.2$), the advantage is still kept about 20% (Fig. 8.7(c)).

The occupancy maps constructed from $q_t(s|z)$ at the end of exploration are shown in Fig. 8.8. All the maps are consistent with the true occupancy probabilities (Fig. 8.8(a)). The maps learned in the active exploration directed by the local learning rate (Fig. 8.8(b)) are better than the maps obtained in random exploration. There is less uncertainty in the maps, demonstrating better prediction performance.

With respect to exploration behavior, active exploration behaviors are different from random exploration behavior. For example, the exploration patterns shown in Fig. 8.9(b) and (c) are the active exploration directed by the local learning rate and by the global performance estimation respectively. Active explorations are systematic, faster, and produce better maps than random exploration (Fig. 8.9(a)). With gain-based exploration, the agent gradually expands its territory due to the fact that the gain in learning diminishes with respect to the visiting frequency. However, in random exploration, the exploration is not systematic, and the resulting exploration frontier is loosely connected and irregularly scattered[2]. The difference between exploration behaviors is evident even under large motion noises (Fig. 8.10).

For environment b, the exploration patterns of the agent are compared in Fig. 8.11 and 8.12. The difference of the exploration behaviors between the active explorations and random exploration is again observed.

In summary, in this chapter, three intrinsic reward functions are derived to direct action exploration in partially observable environments. The local error reduction is the gain in learning Partially Observable Markov Decision Process. The local learning rate is an empirical measure of learning efficiency. The global performance estimation is an subjective approximation of the objective measure. It models the competence motivation of the agent. Maximizing these intrinsic rewards produces curiosity-driven exploratory behavior which is observed in natural agents.

Intrinsic reward has been considered previously to produce curiosity-driven behaviors. In [Sch91; Her01] curiosity is modelled to guide the agent in

[2]videos are available from `http://www.neuro.uni-bremen.de/~bailu/video/index.html`

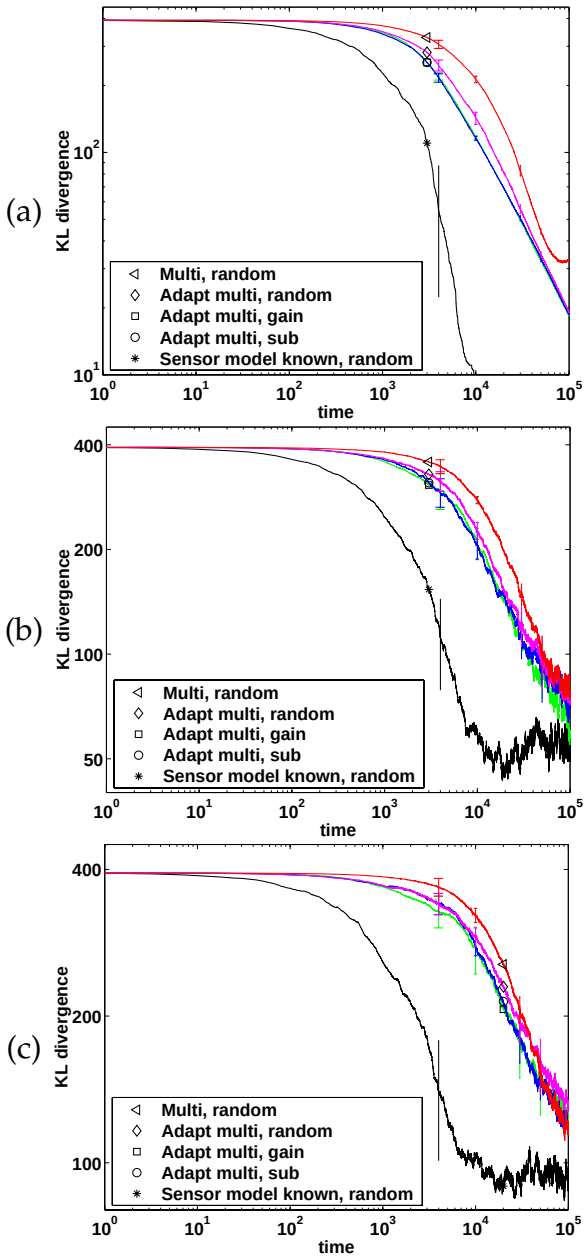


Figure 8.7: Active explorations using the adaptive multiplicative rule compared with random exploration under different locomotion noise levels. The error bars depict $\pm$ standard deviation. The learning rates η_0, μ, β and ϵ used for learning sensor model are the same as in chapter 4. (a) No motion noise ($\sigma_M = 0$). For active exploration directed by the local learning rate, $\theta = 0.1, \gamma = 0.1, \lambda = 0.95$. For active exploration directed by the global performance estimation, $\theta = 0.1, \gamma = 0.9, \lambda = 0.95, \xi = 0.001$; (b) Medium motion noises ($\sigma_M = 0.1$). For active exploration directed by the local learning rate, $\theta = 0.3, \gamma = 0.1, \lambda = 0.5$. For active exploration directed by the global performance estimation, $\theta = 0.1, \gamma = 0.9, \lambda = 0.95, \xi = 0.001$; (c) Large motion noises ($\sigma_M = 0.2$). For active exploration directed by the local learning rate, $\theta = 0.1, \gamma = 0.1, \lambda = 0.1$. For active exploration directed by the global performance estimation, $\theta = 0.1, \gamma = 0.9, \lambda = 0.1, \xi = 0.001$.

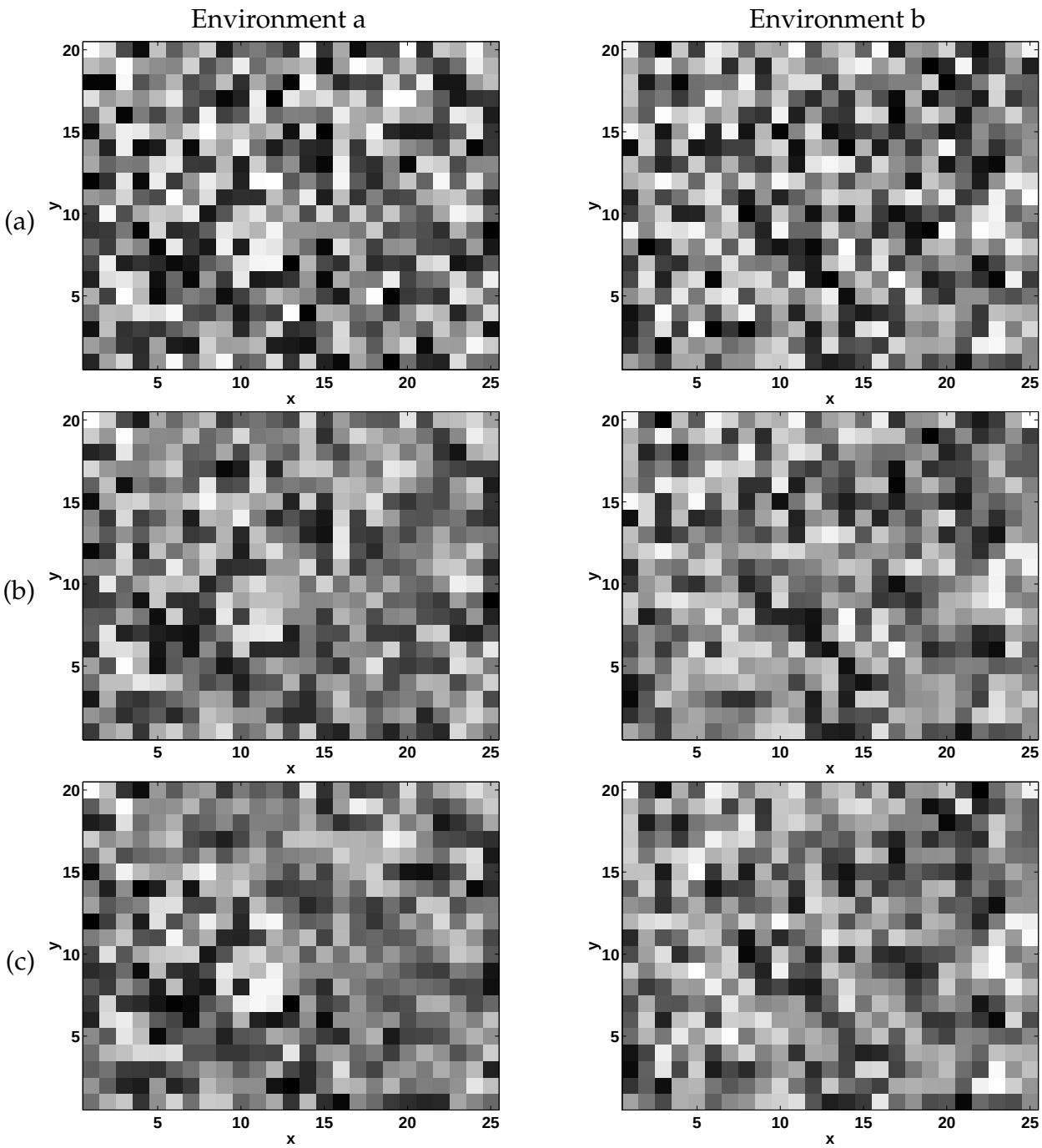


Figure 8.8: Maps constructed from prediction records $q_t(s|z)$ for two environments under different exploration schemes using the adaptive multiplicative rule ($\sigma_M = 0.2, t = 10^5$). (a) True maps of the environments; (b) The active exploration directed by the local learning rate; (c) Random exploration. The maps learned by active exploration have low uncertainty uniformly across the map. The uncertainty in maps learned by random exploration is scattered randomly.

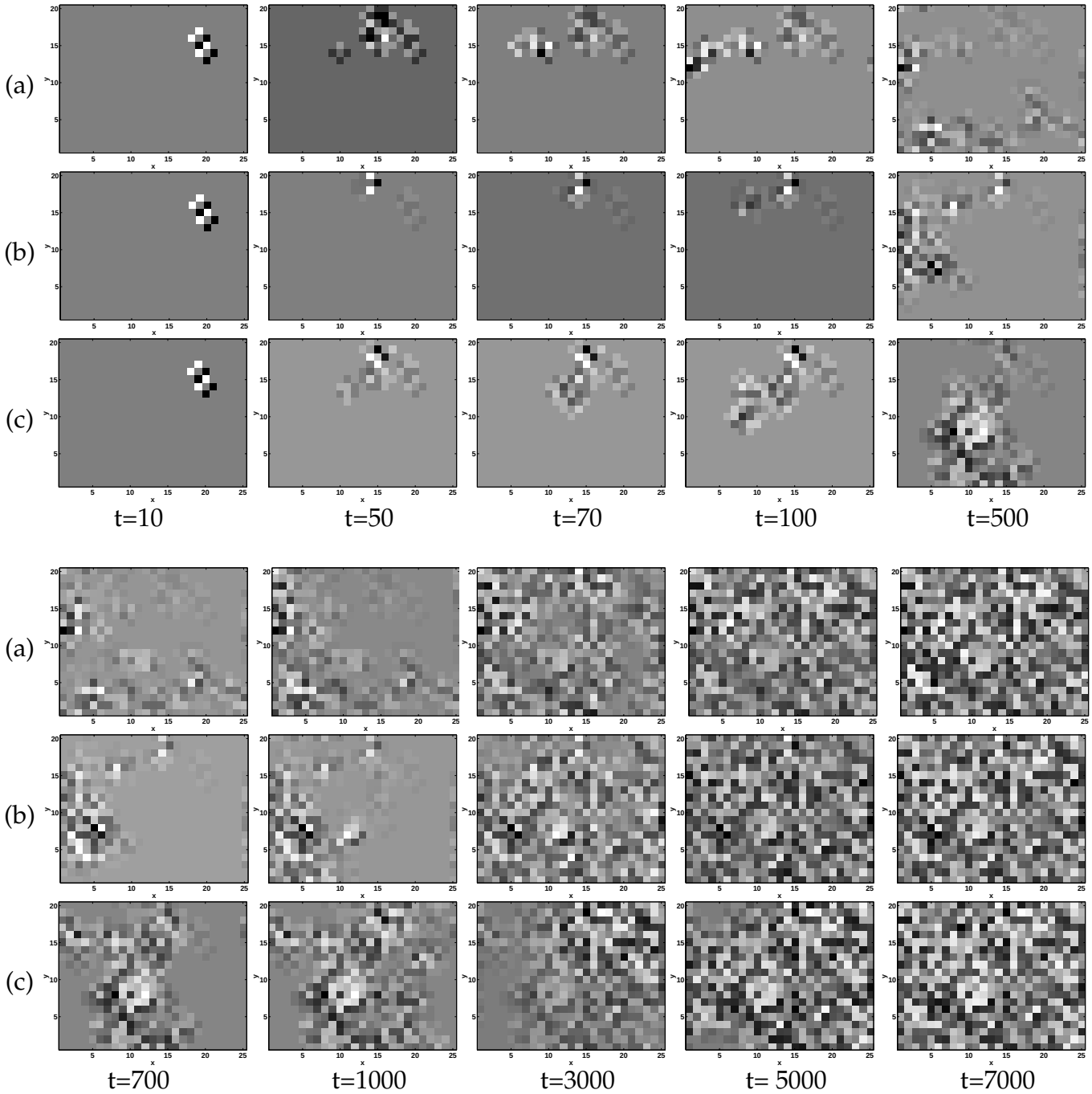


Figure 8.9: Exploration of environment a using the adaptive multiplicative rule with accurate odometry ($\sigma_M = 0$). Each row shows the occupancy maps constructed from $q_t(s|z)$ at different time. The maps are scaled to use the full color codes (not shown). (a) Random exploration; (b) Active exploration directed by the local learning rate; (c) Active exploration directed by the global performance estimation.

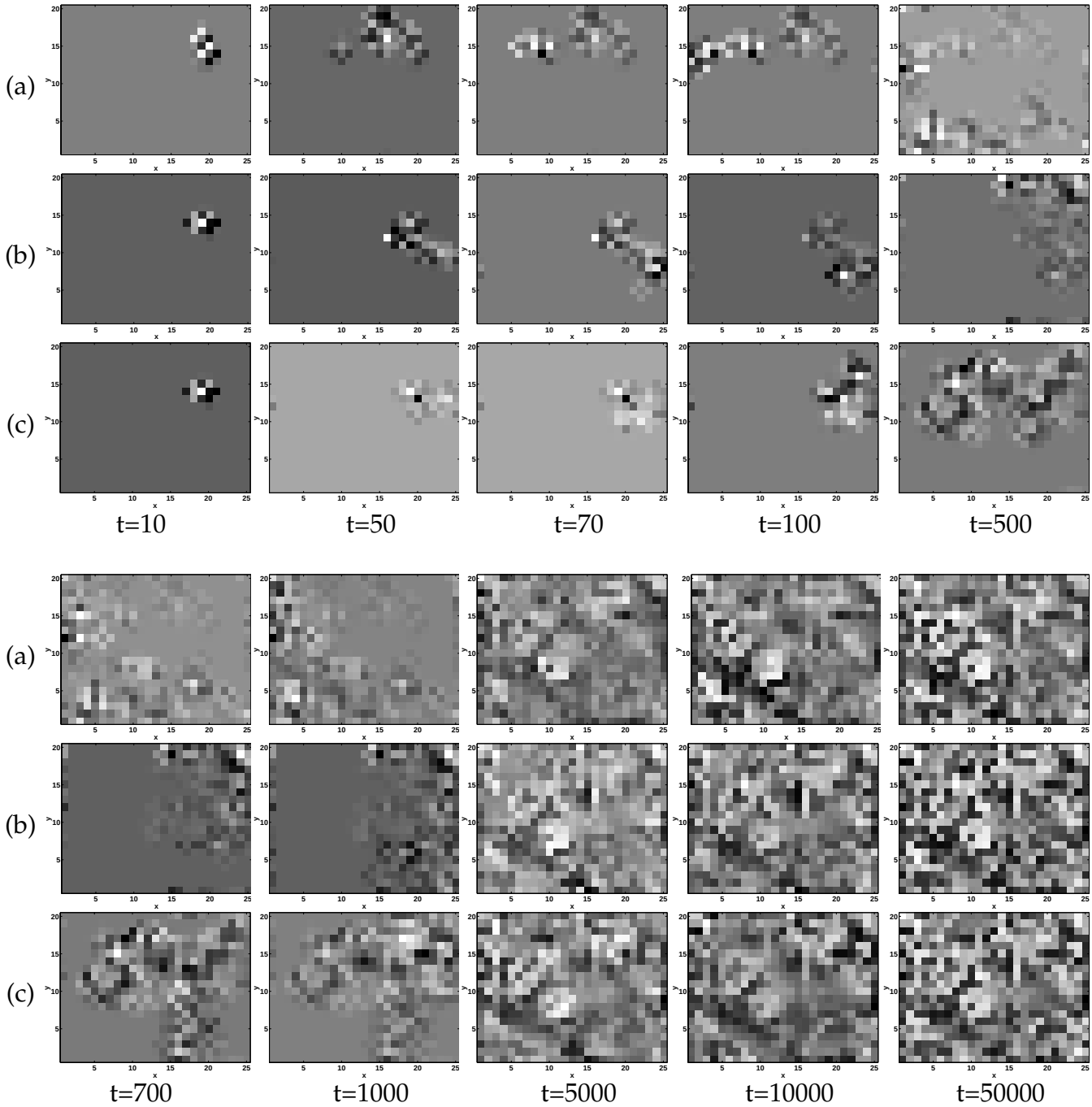


Figure 8.10: Exploration of environment a using the adaptive multiplicative rule under high motion uncertainty ($\sigma_M = 0.2$). Each row shows the occupancy maps constructed from $q_t(s|z)$ at different time. The maps are scaled to use the full color codes (not shown). (a) Random exploration; (b) Active exploration directed by the local learning rate; (c) Active exploration directed by the global performance estimation.

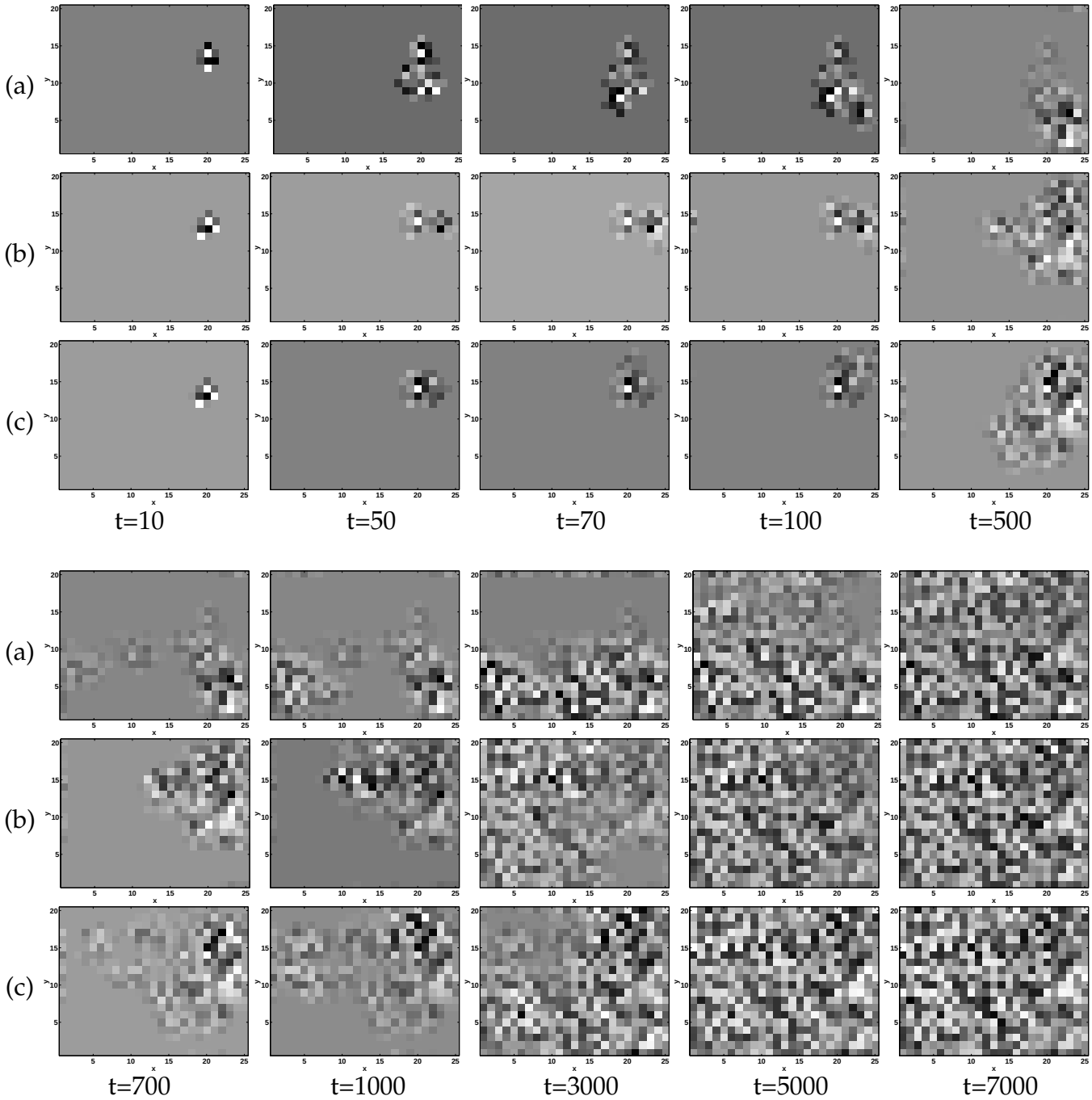


Figure 8.11: Exploration of environment b using the adaptive multiplicative rule with accurate odometry ($\sigma_M = 0$). Each row shows the occupancy maps constructed from $q_t(s|z)$ at different time. The maps are scaled to use the full color codes (not shown). (a) Random exploration; (b) Active exploration directed by the local learning rate; (c) Active exploration directed by the global performance estimation.

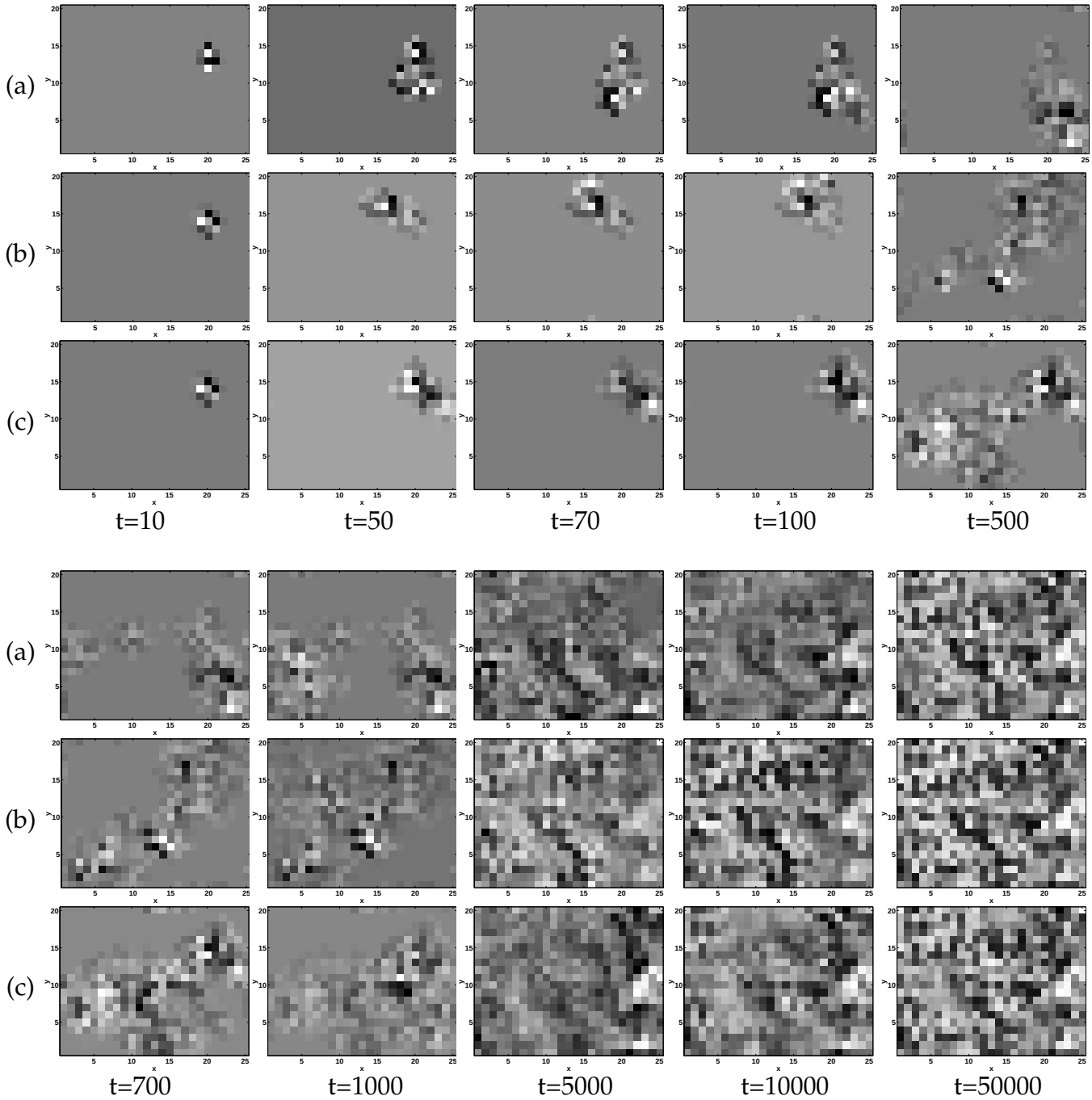


Figure 8.12: Exploration of environment b using the adaptive multiplicative rule under high motion uncertainty ($\sigma_M = 0.2$). Each row shows the occupancy maps constructed from $q_t(s|z)$ at different time. The maps are scaled to use the full color codes (not shown). (a) Random exploration; (b) Active exploration directed by the local learning rate; (c) Active exploration directed by the global performance estimation

data or action selection. Singh et al. combined intrinsic motivation with extrinsic rewards of the traditional reinforcement learning framework, and demonstrated that the agent improved the efficiency in developing hierarchical skills [SBC04]. The intrinsic reward defined here is the gain in local learning. The gain function is appropriate to model curiosity because it measures the efficiency in learning [Her01].

The work presented in [MB99] is also related to ours. The difference is that we use multi-armed bandits to approximate a POMDP, and the gain in learning local bandits is propagated. However, Meuleau et al. approximated a Markov Decision Process by local bandits, and used local estimation error to direct exploration, resulting in error-based exploration. Their work can be considered as a special case of the general approach presented here.

Chapter 9

Conclusions and Outlooks

Place learning is an essential step for an agent to be autonomous. It connects mind and behavior. As an attempt to provide this connection, this thesis gives some interesting results valuable for future research.

9.1 Conclusions

Model is an internal representation of the environment based on the experiences of the agent. It maps place to sensory stimuli, and explains the motor effects and the topology in the environment. On the one hand, model is a cognitive basis for intrinsic motivations. The evaluation on the fitness of the model produces feelings like anxiety, arousal and fear, and also motivations like curiosity and competence. On the other hand, model is a cognitive basis for learning, prediction, exploration, and exploitation. The agent's behavior is a natural consequence driven by its internal model and the evaluation of the model. Therefore, one could arrive that

$$\textit{adaptive behavior} = \textit{model} + \textit{self evaluations.}$$

Normally, learning is local. Organizing local learning through exploratory behavior, the agent achieves optimal learning in a global sense. A POMDP is defined to model local place learning and global optimal exploration. A Bayesian network is proposed as a representation of the POMDP.

In the local level, learning rate is critical for fast convergence. In chapter 3, the issue of learning rate adaptation in a linear network subject to the probability constraint is addressed by *the adaptive multiplicative rule* (Eq. 3.22-3.27). The probability constraint is obeyed by multiplicative gradient with explicit normalization. Learning rates are adapted by a meta-level gradient, which is the

gradient of the loss function with respect to the learning rates in the logarithmic domain. Thereby, the learning rates are kept positive, and at the same time follow the changes of prediction loss caused both by the base level gradient and by the normalization projection. The resulting base level multiplicative gradient together with the meta level learning rate adaptation is called adaptive multiplicative gradient descent . The method to derive learning rate adaption rules applies easily to general loss functions with exponentiated gradient descent rule. The proposed adaptive gradient decent is valuable for learning in slowly changing environment. In chapter 4, a *counting rule* is introduced with a fixed decaying scheme in learning rate (Eq.4.7).

Both the adaptive multiplicative rule and the counting rule are applied to learn the observation probability of the POMDP. The proposed adaptive rules achieve better convergence, and are also robust to noises (Fig. 4.3 and 4.6).

In the global level, exploration is critical for optimal global learning. This is analytically evident in the simplified model of multi-armed bandits studied in chapter 5-7. Global learning refers to information acquisition for all possible alternatives. The learning performance is evaluated by the sum of the errors in local learning (Eq.5.16). The gain of action selection is the negative derivative of the global error with respect to the sampling number (Eq.5.19). Gain maximization principle leads to an equalization of the action selection gain, and to the minimization of global error. This gain equality condition is equivalent to the marginal value in economic models, and is a natural consequence of the Lagrange multiplier principle. The proposed *ideal gain-maximization exploration policy* (Eq.5.20) is optimal in minimizing the total mean squared error in learning. For practical situations, the ideal optimal policy is approached by the *realistic gain-maximization exploration policy* (Eq.5.45) based on the confidence interval method.

The gain maximization principle is general enough to derive optimal exploration solutions for other learning problems with additional objectives like reward maximization. In chapter 7, it is demonstrated that exploration in selection problems is equivalent to learning the reward differences. The optimal sampling strategy, *the ideal gain-maximization selection policy* (Eq.7.7), is derived by maximizing the gain in learning these difference variables, and is achieved in practical situations by the *realistic gain-maximization selection policy* (Eq.7.12).

By generalizing objective functions (Eq.6.1 and 7.2), gain-maximization principle is shown to be the underlying mechanism of a general class of exploration strategies (Eq.6.3 and 7.4), including error-based, counter-based and gain-based strategies as special cases.

Although the gain-maximization policies are derived in the sense of total mean squared error. The analysis of the upper bounds on the sample complexities

of exploration policies proves that under probability measures, the gain based polices are also better than error-based and counter-based polices (Theorem 6.3.4 and 7.3.3).

Gain based exploration can be traded with exploitation, resulting *gain-based reward-maximization policies* (Eq.7.30 and 7.31). Numerical results show that for fixed data sets, the reward maximization performance is significantly improved (Fig.7.7 and 7.13) .

Both the analytical and the numerical results boil down to the observation that

$$\textit{optimal exploration} = \textit{local learning} + \textit{gain maximization.}$$

As an application of this observation, chapter 8 proposes active exploration methods directed by gain functions. The main assumption is that gain signal is an intrinsic reward embodying the agent's curiosity. The resemblances between the gain function and curiosity, i.e. transiency, reference point phenomenon, is the rationale of this assumption. Extending results from local gain adaptation and the multi-armed bandit, two gain functions are defined to model the agent's curiosity. The first gain function is the local error reduction (Eq.8.4), obtained by approximating the POMDP model as a set of local multi-armed bandits. The second gain function is the average local learning rate (Eq.8.7), which is correlated to the decrease in loss function. Both of the two gain functions are large when the learning is far from convergence, and decrease to zero during convergence. However, the first gain function implicitly assumes that the environment is static, and may has limited application for dynamic environments. While the second gain function is applicable for dynamic environments, since the local gain can increase when big prediction error happens. Trying to minimize the global exploration performance directly, a global performance estimation (Eq.8.22) is defined as another intrinsic reword function.

Maximizing intrinsic rewards, active exploration outperforms random exploration with robustness (Fig.8.1 and 8.7). The resulting curiosity-driven exploration behavior is very similar to the frontier-based exploration proposed previously (Fig. 8.3-8.6, and Fig. 8.9-8.12). This result supports the idea that intrinsic emotions are important for intelligent behaviors.

9.2 Outlooks

Other than the lessons learned, there are several provoking questions worth of attention.

First, how to trade-off exploration and exploitation? By the simple linear trade-off scheme presented in section 7.4, the gain-based exploration appears very attractive for reward maximization. However, this result is obtained by selecting appropriate trade-off parameter for fixed data set. The apparently worse UCB policy does not need free parameter and has theoretical performance guarantee. Therefore, it will be profitable to provide our policy with an adaptive tradeoff coefficient and with comparable performance for general data sets.

Second, is the gain-based exploration strategy also adopted by human beings? Exploration by gain maximization is the optimal behavior of a rational agent. Does human also follow similar rational behaviors when facing similar situations like multi-armed bandits? What is the possible objective function underlying human exploration behavior? Are there constraints that possibly prevent human from optimal exploration? These questions help find out the underlying principles of exploratory behaviors of human. Some related theoretical and experimental studies have been done for human decisions in reward maximization tasks [DOD+06; LS06]. It is also interesting to investigate the pure exploration situations.

Third, how to represent the map of the environment in a better way? Although probabilistic approaches for place learning fit well with experimental recordings, with more and more findings in Hippocampus and related cortex, it seems that animals have their own mathematics. What is the physiological and physical reasons for their representations? What computation are they undertaking? How are multiple maps represented in the same model [WM93; GKM04]? Maybe these are the most interesting questions these natural agents raise for us.

Appendix A

Calculation of Hessian-vector Product

In Eq. 3.27, the Hessian-vector product has to be calculated. This is done by introducing a differential operator $\mathcal{R}\{\cdot\}$ on any real valued function $f(\boldsymbol{w})$ parameterized by weight vector $\boldsymbol{w}$

$$\mathcal{R}\{f(\boldsymbol{w})\} \equiv \frac{\partial}{\partial r} f(\boldsymbol{w} + r\boldsymbol{v})|_{r=0}, \tag{A.1}$$

where r is a line parameter, and $\boldsymbol{v}$ is perturbation vector of the same dimension as $\boldsymbol{w}$.

According to the chain rule of partial derivative, the operator $\mathcal{R}\{\cdot\}$ maps the weight vector to the perturbation vector.

$$\mathcal{R}\{\boldsymbol{w}\} = \boldsymbol{v} \tag{A.2}$$

and

The operator $\mathcal{R}\{\cdot\}$ maps the gradient vector $\frac{\partial \ell(\boldsymbol{w})}{\partial \boldsymbol{w}}$ to the Hessian-vector product. This can be shown by Taylor expansion of the perturbed gradient $\frac{\partial \ell(\boldsymbol{w})}{\partial \boldsymbol{w}}|_{\boldsymbol{w}=\boldsymbol{w}+r\boldsymbol{v}}$

$$r\mathbf{H}\boldsymbol{v} = \frac{\partial \ell(\boldsymbol{w})}{\partial \boldsymbol{w}}\Big|_{\boldsymbol{w}=\boldsymbol{w}+r\boldsymbol{v}} - \frac{\partial \ell(\boldsymbol{w})}{\partial \boldsymbol{w}} + O(r^2), \tag{A.3}$$

where $\mathbf{H}$ is the Hessian matrix of the loss function $\ell(\boldsymbol{w}$ with respect to the weight vector $\boldsymbol{w}$. Thus, the product $\mathbf{H}\boldsymbol{v}$ satisfies the definition of the $\mathcal{R}\{\cdot\}$ operator

$$\begin{aligned} \mathbf{H}\boldsymbol{v} &= \lim_{r\to 0} \frac{\frac{\partial \ell(\boldsymbol{w})}{\partial \boldsymbol{w}}\Big|_{\boldsymbol{w}=\boldsymbol{w}+r\boldsymbol{v}} - \frac{\partial \ell(\boldsymbol{w})}{\partial \boldsymbol{w}}}{r} = \frac{\partial}{\partial r}\frac{\partial \ell(\boldsymbol{w})}{\partial \boldsymbol{w}}\Big|_{\boldsymbol{w}=\boldsymbol{w}+r\boldsymbol{v}, r=0} \\ &= \mathcal{R}\{\ell(\boldsymbol{w}\} \end{aligned} \tag{A.4}$$

Since the Hessian-vector product is a differentiation on the loss gradient, its calculation depends on loss function and the structure of the network. The network used in the adaptive multiplicative gradient descent method is a one layered feed-forward neural network with Kullback-Leibler divergence as the loss function. In the following, we summarize the calculation of the Hessian-vector product for this network.

The loss function is given by

$$\ell = \sum_k y_{t+1}(k) \log \frac{y_{t+1}(k)}{\hat{y}_{t+1}(k)},$$

where the prediction is

$$\hat{y}_{t+1}(k) = \sum_i x_{t+1}(i) w_t(k,i).$$

The gradient is calculated according to

$$\frac{\partial \ell}{\partial w_t(k,i)} = \frac{\partial \ell}{\partial \hat{y}_{t+1}(k)} \frac{\partial \hat{y}_{t+1}(k)}{\partial w_t(k,i)} = \ell' x_{t+1}(i),$$

where the first and second-order derivative of the loss function is defined by

$$\begin{aligned} \ell' &= -\frac{y_{t+1}(k)}{\hat{y}_{t+1}(k)} \\ \ell'' &= \frac{y_{t+1}(k)}{\hat{y}_{t+1}(k)^2} \end{aligned}$$

According to Eq. A.4, the Hessian-vector product can be calculated by applying the chain rule of differentiation

$$\begin{aligned} \mathcal{R}\{\frac{\partial \ell}{\partial w_{ki}}\} &= \mathcal{R}\{\ell' x_{t+1}(i)\} \\ &= \mathcal{R}\{\ell'\} x_{t+1}(i) + \ell' \mathcal{R}\{x_{t+1}(i)\} \\ &= \ell'' \mathcal{R}\{\hat{y}_{t+1}(k)\} x_{t+1}(i) \\ &= \frac{y_{t+1}(k) x_{t+1}(i)}{\hat{y}^2_{t+1}(k)} \sum_{j=1}^{N} v_t(k,j) x_{t+1}(i) \end{aligned}$$

Therefore, the elements of $\mathbf{H}_t(k)\boldsymbol{v}_t(k)$ indexed by i is:

$$\left[\mathbf{H}_t(k)\boldsymbol{v}_t(k)\right]_i = \frac{y_{t+1}(k) x_{t+1}(i)}{\hat{y}^2_{t+1}(k)} \sum_{j=1}^{N} v_t(k,j) x_{t+1}(j) \tag{A.5}$$

Putting into the probability notations, Equation (A.5) becomes:

$$\left[\mathbf{H}_t(k)\boldsymbol{v}_t(k)\right]_i = \frac{p_{t+1}(s_k)\hat{p}_{t+1}(i)}{\hat{p}^2_{t+1}(s_k)} \sum_j v_t(k,j)\hat{p}_{t+1}(j) \tag{A.6}$$

Appendix B

Mean Squared Error of Exploration Policies

B.1 Performance Ratio of Exploration Policies for Optimal Learning

The performance ratio $\frac{E_L^e(t)}{E_L^*(t)}$ depends on specific values of $\sigma(a)$. In order to see how large the performance ratio on average is, we consider $\sigma(a)$ as a random variable, and calculate the ratio of the expected total mean squared error $\frac{\langle E_L^e(t)\rangle}{\langle E_L^*(t)\rangle}$. The expectation $\langle\cdot\rangle$ is with respect to the probability distribution of $\sigma(a)$.

Here, we prove the following result. If the standard deviations $\sigma(a)$ of the reward distributions are independently and identically distributed, then the ratio of the expected total mean squared error $\frac{\langle E_L^e(t)\rangle}{\langle E_L^*(t)\rangle}$ is bounded as:

$$1 \leq \frac{\langle E_L^e(t)\rangle}{\langle E_L^*(t)\rangle} \leq n, \tag{B.1}$$

where n is the number of arms. When the variance $\mathrm{Var}[\sigma(a)] \to \infty$ the performance ratio $\frac{\langle E_L^e(t)\rangle}{\langle E_L^*(t)\rangle} = n$. When $\mathrm{Var}[\sigma(a)] = 0$, the performance ratio $\frac{\langle E_L^e(t)\rangle}{\langle E_L^*(t)\rangle}$ reaches its minimal value 1.

Proof

First, we show that $0 \leq \dfrac{\langle\sigma(a)\rangle^2}{\langle\sigma^2(a)\rangle} \leq 1$ for any random variable $\sigma(a)$ (assuming $\langle\sigma^2(a)\rangle \neq 0$).

Since the variance of $\sigma(a)$ is non-negative, and can be written as

$$\begin{aligned}
\mathrm{Var}[\sigma(a)] &= \Big\langle \Big(\sigma(a) - \langle\sigma(a)\rangle\Big)^2 \Big\rangle \\
&= \Big\langle \sigma^2(a) - 2\sigma(a)\langle\sigma(a)\rangle + \langle\sigma(a)\rangle^2 \Big\rangle \\
&= \langle\sigma^2(a)\rangle - \langle\sigma(a)\rangle^2 \\
&\geq 0,
\end{aligned}$$

it is easy to show that

$$0 \leq \frac{\langle\sigma(a)\rangle^2}{\langle\sigma^2(a)\rangle} = \frac{\langle\sigma(a)\rangle^2}{\mathrm{Var}[\sigma(a)] + \langle\sigma(a)\rangle^2} \leq 1. \tag{B.2}$$

Since $\sigma(a)$ is independently and identically distributed, then for all $a, b \in \mathcal{A}$ and $a \neq b$,

$$\langle\sigma(a)\sigma(b)\rangle = \langle\sigma(a)\rangle\langle\sigma(b)\rangle = \langle\sigma(a)\rangle^2 = \langle\sigma(b)\rangle^2,$$

$$\langle\sigma^2(a)\rangle = \langle\sigma^2(b)\rangle.$$

The expectation of $E_L^*(t)$ is equal to

$$\begin{aligned}
\langle E_L^*(t)\rangle &= \Big\langle \frac{1}{t}\Big(\sum_a \sigma(a)\Big)^2 \Big\rangle \\
&= \frac{1}{t}\left[\sum_a \langle\sigma^2(a)\rangle + \sum_a \left[\langle\sigma(a)\rangle \sum_{b\neq a} \langle\sigma(b)\rangle\right]\right] \\
&= \frac{1}{t}\Big[n\langle\sigma^2(a)\rangle + n(n-1)\langle\sigma(a)\rangle^2\Big].
\end{aligned}$$

Similarly, the expectation of $E_L^e(t)$ is given by

$$\begin{aligned}
\langle E_L^e(t)\rangle &= \Big\langle \frac{n}{t}\sum_a \sigma^2(a) \Big\rangle \\
&= \frac{n}{t}\sum_a \langle\sigma^2(a)\rangle \\
&= \frac{n^2}{t}\langle\sigma^2(a)\rangle.
\end{aligned}$$

Then the ratio of the expected total mean squared error is equal to

$$\frac{\langle E_L^e(t)\rangle}{\langle E_L^*(t)\rangle} = \frac{n}{1 + (n-1)\frac{\langle\sigma(a)\rangle^2}{\langle\sigma^2(a)\rangle}}. \tag{B.3}$$

From Equation (B.2), it is easy to show that the ratio of the expected total mean squared error (Eq. B.3) is bounded. When $\mathrm{Var}[\sigma(a)] \to \infty$, $\frac{\langle E_L^e(t)\rangle}{\langle E_L^*(t)\rangle}$ approaches the upper bound n. $\frac{\langle E_L^e(t)\rangle}{\langle E_L^*(t)\rangle}$ reaches the lower bound 1 when $\mathrm{Var}[\sigma(a)] = 0$. □

B.1.1 Performance Ratio with Power Law Distributed Reward Standard Deviation

As an example, we calculate the performance ratio in ranking task for two frequently occurring distributions. First, we consider the power law distribution of a continuous real variable x whose probability density function is defined as

$$p(x) = Cx^{-\gamma},\ x \in [l, u], \gamma > 1. \tag{B.4}$$

Here the constant γ is the exponent of the power law, and C is the normalization constant satisfying

$$C = \frac{1-\gamma}{u^{1-\gamma} - l^{1-\gamma}}.$$

We assume that x is bounded in the region $[l, u]$, where $l > 0$ is the lowest value of x, and u is the expectation of the largest value sampled from the power law distribution.

$$u = \langle x_{max} \rangle$$

In theory, u can be infinity. However, due to the limited number of samples, the largest sample observed is finite. Therefore, for practical calculations we assume that u is finite.

The mean value of x in power law distribution (B.4) is given by

$$\langle x \rangle = \frac{1-\gamma}{2-\gamma}\frac{u^{2-\gamma} - l^{2-\gamma}}{u^{1-\gamma} - l^{1-\gamma}}. \tag{B.5}$$

The second moment, the mean square, is given by

$$\langle x^2 \rangle = \frac{1-\gamma}{3-\gamma}\frac{u^{3-\gamma} - l^{3-\gamma}}{u^{1-\gamma} - l^{1-\gamma}}. \tag{B.6}$$

When $\gamma \in (2, 3]$, $\langle x^2 \rangle$ is infinity when $u \to \infty$, but the mean $\langle x \rangle$ is still finite. So the distribution has infinite variance. For $\gamma = 2.5$, the distribution is a power law distribution which has often been observed in nature [New05]. So we

use a power law distribution with exponent 2.5 in the first simulation (Section 5.6.1). The mean value of the largest sample is calculated following the method described in [New05].

$$\langle x_{max} \rangle = klB\left(k, \frac{\gamma - 2}{\gamma - 1}\right).$$

Here k is the number of samples, and $B(i, j)$ is Legendre's beta-function defined by

$$B(i, j) = \frac{\Gamma(i)\Gamma(j)}{\Gamma(i + j)},$$

with $\Gamma(i)$ being the standard Γ-function

$$\Gamma(i) = \int_0^\infty y^{i-1} e^{-y} dy, \qquad i > 0.$$

For the first simulation where we used the power law distribution, the exponent $\gamma = 2.5$, the number of arms $n = 100$, and the sample size for $\sigma(a)$ is given by $k = 100(\text{arms/bandit}) \times 100(\text{instances of bandit}) = 10^4$. The upper limit u of the distribution becomes

$$u = \langle x_{max} \rangle = 10^4 lB\left(10^4, \frac{1}{3}\right) = 1243.47l$$

Then the mean (Eq.B.5) and mean square (Eq.B.6) become

$$\langle x \rangle = 2.915l, \tag{B.7}$$

$$\langle x^2 \rangle = 102.791l^2. \tag{B.8}$$

Putting Eq.B.7 and Eq. B.8 into Eq. B.3, the performance ratio is given by

$$\frac{\langle E_L^e(t) \rangle}{\langle E_L^*(t) \rangle} = \frac{n}{1 + 0.0827(n - 1)} \approx 10.89$$

For the simulation where we used the power law distribution, $\frac{\langle E_L^e(t) \rangle}{\langle E_L^*(t) \rangle}$ was indeed close to 10. The numerical result confirms the calculation.

B.1.2 Performance Ratio with Uniformly Distributed Reward Standard Deviation

When the exponent γ is set to 0, Eq. B.4 becomes a uniform distribution. The value of u and l are constant numbers. The mean and the mean square are given by

$$\langle x \rangle = \int_l^u \frac{1}{u-l} x dx = \frac{u+l}{2} \tag{B.9}$$

$$\langle x^2 \rangle = \int_l^u \frac{1}{u-l} x^2 dx = \frac{u^2+ul+l^2}{3} \tag{B.10}$$

Combining Eq.B.9 and Eq. B.10 with Eq. B.3, the performance ratio becomes

$$\begin{aligned} \frac{\langle E_L^e(t) \rangle}{\langle E_L^*(t) \rangle} &= \frac{n}{1+(n-1)\frac{3}{4}\frac{(u+l)^2}{u^2+ul+l^2}} \\ &< \frac{n}{1+\frac{3}{4}(n-1)} \\ &< \frac{4}{3}. \end{aligned}$$

From the simulation with uniformly distributed $\sigma(a)$, it is observed that $\frac{\langle E_L^e(t) \rangle}{\langle E_L^*(t) \rangle} \approx 1.3$. The numerical result matches the theoretical analysis here.

Appendix C

Sample Complexity of Exploration Policies

C.1 Proof of Lemma 6.3.1

Proof. The variable $\zeta_{ik} \in [0, \epsilon_{ik}]$ divides the interval $\big[\mu(i), \mu(k)\big]$ into two consecutive intervals, i.e. $\big[\mu(i), \mu(i) + \zeta_{ik}]$ and $[\mu(k) - (\epsilon_{ik} - \zeta_{ik}), \mu(k)\big]$. According to the relative position to the point $\big(\mu(i) + \zeta_{ik}\big)$ or $\big(\mu(k) - (\epsilon_{ik} - \zeta_{ik})\big)$, the wrong pairwise ordering $\hat{\mu}_t(k) < \hat{\mu}_t(i)$ only have three exclusive configurations:

1. Both $\hat{\mu}_t(i)$ and $\hat{\mu}_t(k)$ are larger than $\mu(i) + \zeta_{ik}$:

$$\mu(i) + \zeta_{ik} = \mu(k) - (\epsilon_{ik} - \zeta_{ik}) \leq \hat{\mu}_t(k) < \hat{\mu}_t(i).$$

2. Only $\hat{\mu}_t(i)$ is larger than $\mu(i) + \zeta_{ik}$:

$$\hat{\mu}_t(k) < \mu(i) + \zeta_{ik} = \mu(k) - (\epsilon_{ik} - \zeta_{ik}) \leq \hat{\mu}_t(i).$$

3. Both $\hat{\mu}_t(i)$ and $\hat{\mu}_t(k)$ are smaller than $\mu(i) + \zeta_{ik}$:

$$\hat{\mu}_t(k) < \hat{\mu}_t(i) < \mu(i) + \zeta_{ik} = \mu(k) - (\epsilon_{ik} - \zeta_{ik}).$$

Since these three events can not happen at the same time, the probability of the wrong pairwise ordering $\hat{\mu}_t(k) < \hat{\mu}_t(i)$ is equal to the sum of the three

probabilities

$$
\begin{aligned}
&P\Big(\hat{\mu}_t(k) < \hat{\mu}_t(i)\Big) \\
&= P\Big(\mu(i) + \zeta_{ik} = \mu(k) - (\epsilon_{ik} - \zeta_{ik}) \leq \hat{\mu}_t(k) < \hat{\mu}_t(i)\Big) \\
&\quad + P\Big(\hat{\mu}_t(k) < \mu_t(i) + \zeta_{ik} = \mu(k) - (\epsilon_{ik} - \zeta_{ik}) \leq \hat{\mu}_t(i)\Big) \\
&\quad + P\Big(\hat{\mu}_t(k) < \hat{\mu}_t(i) < \mu(i) + \zeta_{ik} = \mu(k) - (\epsilon_{ik} - \zeta_{ik})\Big) \\
&\leq P\Big(\mu(i) + \zeta_{ik} \leq \hat{\mu}_t(k) \text{ and } \mu(i) + \zeta_{ik} \leq \hat{\mu}_t(i)\Big) \\
&\quad + P\Big(\hat{\mu}_t(k) < \mu(i) + \zeta_{ik} \text{ and } \mu(i) + \zeta_{ik} \leq \hat{\mu}_t(i)\Big) \\
&\quad + P\Big(\hat{\mu}_t(k) \leq \mu_t(k) - (\epsilon_{ik} - \zeta_{ik})\Big) && \text{(C.1)} \\
&= P\Big(\mu(i) + \zeta_{ik} \leq \hat{\mu}_t(i)\Big) + P\Big(\hat{\mu}_t(k) \leq \mu(k) - (\epsilon_{ik} - \zeta_{ik})\Big) && \text{(C.2)}
\end{aligned}
$$

Eq.C.1 is due to relaxing the first and third probabilities. □

C.2 Union Bound

The probability of event A or event B occurring is less than the sum of the probabilities the individual events will occur:

$$
\begin{aligned}
P(A \text{ or } B) &= P(A) + P(B) - P(A \text{ and } B) \\
&\leq P(A) + P(B) && \text{(C.3)}
\end{aligned}
$$

C.3 Chebyshev Inequality

If a random variable x has a finite mean μ and finite variance σ^2, then $\forall \varepsilon \geq 0$,

$$P(|x - \mu| \geq \varepsilon) \leq \frac{\sigma^2}{\varepsilon^2} \tag{C.4}$$

The variance of the sample mean $\hat{\mu}_\tau = \frac{1}{\tau}\sum_{j=1}^{\tau} r_j$ is

$$\mathrm{Var}[\hat{\mu}_\tau] = \langle(\hat{\mu}_\tau - \mu)^2\rangle = \frac{\sigma^2}{\tau}. \tag{C.5}$$

So the Chebyshev Inequality for random variable $\hat{\mu}_\tau$ is

$$P(|\hat{\mu}_\tau - \mu| \geq \varepsilon) \leq \frac{\sigma^2}{\varepsilon^2 \tau}. \tag{C.6}$$

Assuming that the probability density of $\hat{\mu}_\tau$ is symmetrical, we get

$$\begin{aligned} & P(\hat{\mu}_\tau \geq \mu + \varepsilon) \\ & = P(\hat{\mu}_\tau \leq \mu - \varepsilon) \\ & = \frac{1}{2} P(|\hat{\mu}_\tau - \mu| \geq \varepsilon) \\ & \leq \frac{\sigma^2}{2\varepsilon^2 \tau}. \end{aligned} \tag{C.7}$$

C.4 Optimal Section of the Mean Distance for Optimal Learning

The upper bound of the total probability of overestimating $\mu(i)$ and underestimating $\mu(k)$ is proportional to the quantity $\frac{\sigma^{\frac{2}{\beta+1}}(k)}{(\epsilon_{ik}-\zeta_{ik})^2} + \frac{\sigma^{\frac{2}{\beta+1}}(i)}{\zeta_{ik}^2}$, which is denoted by $f(\zeta_{ik})$

$$f(\zeta_{ik}) = \frac{\sigma^{\frac{2}{\beta+1}}(k)}{(\epsilon_{ik} - \zeta_{ik})^2} + \frac{\sigma^{\frac{2}{\beta+1}}(i)}{\zeta_{ik}^2} \tag{C.8}$$

Here we show that, the optimal value of ζ_{ik}, that sections the interval $[\mu(i), \mu(k)]$ and minimizes $f(\zeta_{ik})$, is

$$\zeta_{ik}^* = \epsilon_{ik} \frac{\sigma^{\frac{2}{3(\beta+1)}}(i)}{\sigma^{\frac{2}{3(\beta+1)}}(i) + \sigma^{\frac{2}{3(\beta+1)}}(k)}.$$

The first derivative of $f(\zeta_{ik})$ is equal to

$$\frac{df(\zeta_{ik})}{d\zeta_{ik}} = 2\frac{\sigma^{\frac{2}{\beta+1}}(k)}{(\epsilon_{ik} - \zeta_{ik})^3} - 2\frac{\sigma^{\frac{2}{\beta+1}}(i)}{\zeta_{ik}^3}. \tag{C.9}$$

The second derivative is given by

$$\frac{d^2 f(\zeta_{ik})}{(d\zeta_{ik})^2} = 6\frac{\sigma^{\frac{2}{\beta+1}}(k)}{(\epsilon_{ik} - \zeta_{ik})^4} + 6\frac{\sigma^{\frac{2}{\beta+1}}(i)}{\zeta_{ik}^4}. \tag{C.10}$$

Since the second derivative is positive, the zero point of the first derivative is the minimum of $f(\zeta_{ik})$. Putting Eq.C.9 into zero, we get the solution in Eq.6.20.

C.5 Proof of Theorem 6.3.4

Proof. From Eq.6.22, $C_L(\infty)$ is equal to

$$C_L(\infty) = 4n(n-1)\sum_{i=1}^{n}\sigma^2(i). \tag{C.11}$$

So its expectation reads

$$\langle C_L(\infty)\rangle = 4n^2(n-1)\langle\sigma^2\rangle. \tag{C.12}$$

$C_L(0)$ is given by

$$\begin{aligned} C_L(0) &= n\Big[(n-4)\sum_{i=1}^{n}\sigma^2(i) + 3\sum_{i=1}^{n}\sigma^{\frac{4}{3}}(i)\sum_{j=1}^{n}\sigma^{\frac{2}{3}}(j)\Big] \\ &= n\Big[(n-1)\sum_{i=1}^{n}\sigma^2(i) + 3\sum_{1\le i\neq j\le n}\sigma^{\frac{4}{3}}(i)\sigma^{\frac{2}{3}}(j)\Big], \end{aligned} \tag{C.13}$$

and its expectation value is

$$\langle C_L(0)\rangle = n^2(n-1)\langle\sigma^2\rangle + 3n^2(n-1)\langle\sigma^{\frac{4}{3}}\rangle\langle\sigma^{\frac{2}{3}}\rangle. \tag{C.14}$$

Similarly,

$$\begin{aligned} C_L(1) &= \sum_{j=1}^{n}\sigma(j)\Big[(n-4)\sum_{i=1}^{n}\sigma(i) + 3\sum_{i=1}^{n}\sigma^{\frac{2}{3}}(i)\sum_{j=1}^{n}\sigma^{\frac{1}{3}}(j)\Big] \\ &= (n-1)\sum_{i=1}^{n}\sigma^2(i) + (n-1)\sum_{1\le i\neq j\le n}\sigma(i)\sigma(j) + 3\sum_{1\le i\neq j\le n}\sigma^{\frac{2}{3}}(i)\sigma^{\frac{4}{3}}(j) \\ &\quad +3\sum_{1\le i\neq j\le n}\sigma^{\frac{1}{3}}(i)\sigma^{\frac{5}{3}}(j) + 3\sum_{1\le i\neq j\neq k\le n}\sigma(i)\sigma^{\frac{2}{3}}(j)\sigma^{\frac{1}{3}}(k). \end{aligned} \tag{C.15}$$

This leads to the expectation value $\langle C_L(1)\rangle$

$$\begin{aligned} \langle C_L(1)\rangle &= n(n-1)\langle\sigma^2\rangle + n(n-1)^2\langle\sigma\rangle^2 + 3n(n-1)\langle\sigma^{\frac{2}{3}}\rangle\langle\sigma^{\frac{4}{3}}\rangle \\ &\quad +3n(n-1)\langle\sigma^{\frac{1}{3}}\rangle\langle\sigma^{\frac{5}{3}}\rangle + 3n(n-1)(n-2)\langle\sigma\rangle\langle\sigma^{\frac{2}{3}}\rangle\langle\sigma^{\frac{1}{3}}\rangle. \end{aligned} \tag{C.16}$$

On the one hand, when the ratios of lower moments to the second moment vanish, i.e. $\frac{\langle\sigma^b\rangle}{\langle\sigma^2\rangle} = 0$, $b \in (0, \frac{5}{3}]$, the ratio of the expected complexities becomes

$$\begin{aligned} \frac{\langle T_L(0)\rangle}{\langle T_L(1)\rangle} &= \frac{\langle T_L(0)\rangle}{\langle T_L(1)\rangle} = \frac{n^2(n-1)}{n(n-1)} \\ &= n, \end{aligned} \tag{C.17}$$

$$\begin{aligned} \frac{\langle T_L(\infty)\rangle}{\langle T_L(1)\rangle} &= \frac{\langle T_R(\infty)\rangle}{\langle T_R(1)\rangle} = \frac{4n^2(n-1)}{n(n-1)} \\ &= 4n. \end{aligned} \tag{C.18}$$

On the other hand, when all the moments $\langle\sigma^b\rangle$,$b \in (0, 2]$ are finite, the ratio of the expected complexities depends on the dominating terms of n, and the asymptotic ratio is a finite number.

$$\lim_{n\to\infty} \frac{\langle T_L(0)\rangle}{\langle T_L(1)\rangle} = \lim_{n\to\infty} \frac{\langle C_L(0)\rangle}{\langle C_L(1)\rangle} = \frac{\langle\sigma^2\rangle + 3\langle\sigma^{\frac{4}{3}}\rangle\langle\sigma^{\frac{2}{3}}\rangle}{\langle\sigma\rangle^2 + 3\langle\sigma\rangle\langle\sigma^{\frac{2}{3}}\rangle\langle\sigma^{\frac{1}{3}}\rangle} \tag{C.19}$$

$$\lim_{n\to\infty} \frac{\langle T_L(\infty)\rangle}{\langle T_L(1)\rangle} = \lim_{n\to\infty} \frac{\langle T_L(\infty)\rangle}{\langle T_L(1)\rangle} = \frac{4\langle\sigma^2\rangle}{\langle\sigma\rangle^2 + 3\langle\sigma\rangle\langle\sigma^{\frac{2}{3}}\rangle\langle\sigma^{\frac{1}{3}}\rangle} \tag{C.20}$$

□

Appendix D

Sample Complexity of Selection Policies

D.1 Performance Ratio of Selection Polices

For independently and identically distributed standard deviations $\sigma(a)$, the ratio of $\frac{\langle E_E^e(t)\rangle}{\langle E_E^*(t)\rangle}$ is bounded as:

$$\frac{2n}{n+2(n-1)^{1/2}} \leq \frac{\langle E_E^e(t)\rangle}{\langle E_E^*(t)\rangle} \leq n, \tag{D.1}$$

where n is the number of arms. When the variance $\mathrm{Var}[\sigma(a)] \to \infty$ the performance ratio $\frac{\langle E_E^e(t)\rangle}{\langle E_E^*(t)\rangle} = n$. When $\mathrm{Var}[\sigma(a)] = 0$, the performance ratio $\frac{\langle E_E^e(t)\rangle}{\langle E_E^*(t)\rangle}$ reaches its minimal value $\frac{2n}{n+2(n-1)^{1/2}}$.

Proof

Since $\sigma(a)$ is independently and identically distributed, then for all $a, b \in \mathcal{A}$ and $a \neq b$,

$$\begin{aligned}
\langle\sigma(a)\sigma(b)\rangle &= \langle\sigma(a)\rangle\langle\sigma(b)\rangle = \langle\sigma(a)\rangle^2 = \langle\sigma(b)\rangle^2, \\
\langle\sigma^2(a)\rangle &= \langle\sigma^2(b)\rangle.
\end{aligned}$$

The expectation of $E^*_E(t)$ (Eq.7.9) is equal to

$$\begin{aligned}
\langle E^*_E(t)\rangle &= \Big\langle \frac{1}{t}\Big[\sum_{i=1}^{n-1}\sigma(i) + (n-1)^{\frac{1}{2}}\sigma(n)\Big]^2\Big\rangle \\
&= \frac{1}{t}\Bigg[\sum_{i=1}^{n-1}\langle\sigma^2(i)\rangle + (n-1)\langle\sigma^2(n)\rangle \\
&\quad + \sum_{1\le i\neq j\le n-1}\langle\sigma(i)\sigma(j)\rangle + 2(n-1)^{1/2}\langle\sigma(n)\sum_{i=1}^{n-1}\sigma(i)\rangle\Bigg] \\
&= \frac{1}{t}\Big[2(n-1)\langle\sigma^2(i)\rangle + (n-1)(n-2)\langle\sigma(i)\rangle^2 + 2(n-1)^{\frac{3}{2}}\langle\sigma(i)\rangle^2\Big].
\end{aligned}$$

Similarly, the expectation of $E^e(t)$ (Eq. 7.10)is given by

$$\begin{aligned}
\langle E^e_E(t)\rangle &= \Big\langle \frac{2(n-1)}{t}\sum_{i=1}^{n}\sigma^2(i)\Big\rangle \\
&= \frac{2n(n-1)}{t}\langle\sigma^2(i)\rangle.
\end{aligned}$$

Then the ratio $\frac{\langle E^e(t)\rangle}{\langle E^*(t)\rangle}$ is equal to

$$\frac{\langle E^e_E(t)\rangle}{\langle E^*_E(t)\rangle} = \frac{2n}{2 + (n-2)\frac{\langle\sigma(a)\rangle^2}{\langle\sigma^2(a)\rangle} + 2(n-1)^{1/2}\frac{\langle\sigma(a)\rangle^2}{\langle\sigma^2(a)\rangle}}. \tag{D.2}$$

According to Eq. B.2, when $\mathrm{Var}[\sigma(a)] \to \infty$, $\frac{\langle E^e_E(t)\rangle}{\langle E^*_E(t)\rangle}$ approaches the upper bound n. $\frac{\langle E^e_E(t)\rangle}{\langle E^*_E(t)\rangle}$ reaches the lower bound $\frac{2n}{n+2(n-1)^{1/2}}$ when $\mathrm{Var}[\sigma(a)] = 0$. □

D.1.1 Performance Ratio with Power Law Distributed Reward Standard Deviation

Putting Eq. B.7 and Eq. B.8 into Eq. D.2, yields the performance ratio for the selection task

$$\frac{\langle E^e_E(t)\rangle}{\langle E^*_E(t)\rangle} = \frac{2n}{2 + 0.0827(n-2) + 0.165(n-1)^{1/2}} \approx 17.03.$$

D.1.2 Performance Ratio with Uniformly Distributed Reward Standard Deviation

Combining Eq. B.9 and Eq. B.10 with Eq. D.2, the performance ratio for the selection problem is equal to

$$\begin{aligned}\frac{\langle E_E^e(t)\rangle}{\langle E_E^*(t)\rangle} &= \frac{2n}{2+(n-2)\frac{3}{4}\frac{(u+l)^2}{u^2+ul+l^2}+2(n-1)^{1/2}\frac{3}{4}\frac{(u+l)^2}{u^2+ul+l^2}}\\ &< \frac{2n}{2+\frac{3}{4}(n-2)+\frac{3}{2}(n-1)^{1/2}}\\ &\approx 2.21.\end{aligned}$$

D.2 Optimal Section of the Mean Distance for Exploitation

The upper bound of the total probability of overestimating $\mu(i)$ and underestimating $\mu(n)$ is proportional to the quantity $\frac{(n-1)^{-\frac{1}{\beta+1}}\sigma^{\frac{2}{\beta+1}}(n)}{(\epsilon_{in}-\zeta_{in})^2}+\frac{\sigma^{\frac{2}{\beta+1}}(i)}{\zeta_{in}^2}$, which is denoted by $f(\zeta_{in})$.

$$f(\zeta_{in}) = \frac{(n-1)^{-\frac{1}{\beta+1}}\sigma^{\frac{2}{\beta+1}}(n)}{(\epsilon_{in}-\zeta_{in})^2}+\frac{\sigma^{\frac{2}{\beta+1}}(i)}{\zeta_{in}^2} \tag{D.3}$$

Here we show that, the the optimal value of ζ_{in}, that sections the interval $[\mu(i),\mu(n)]$ and minimizes $f(\zeta_{in})$, is

$$\zeta_{in}^* = \epsilon_{in}\frac{\sigma^{\frac{2}{3(\beta+1)}}(i)}{(n-1)^{-\frac{1}{3(\beta+1)}}\sigma^{\frac{2}{3(\beta+1)}}(n)+\sigma^{\frac{2}{3(\beta+1)}}(i)}.$$

The first derivative of $f(\zeta_{in})$ is equal to

$$\frac{df(\zeta_{in})}{d\zeta_{in}} = 2\frac{(n-1)^{-\frac{1}{\beta+1}}\sigma^{\frac{2}{\beta+1}}(n)}{(\epsilon_{in}-\zeta_{in})^3}-2\frac{\sigma^{\frac{2}{\beta+1}}(i)}{\zeta_{in}^3}. \tag{D.4}$$

The second derivative is given by

$$\frac{d^2f(\zeta_{in})}{(d\zeta_{in})^2} = 6\frac{(n-1)^{-\frac{1}{\beta+1}}\sigma^{\frac{2}{\beta+1}}(n)}{(\epsilon_{in}-\zeta_{in})^4}+6\frac{\sigma^{\frac{2}{\beta+1}}(i)}{\zeta_{in}^4}. \tag{D.5}$$

Since the second derivative is positive, the zero point of the first derivative is the minimum of $f(\zeta_{in})$. Putting Eq. D.4 into zero, we get the solution in Eq. 7.22.

D.3 Proof of Theorem 7.3.3

Proof. From Eq. 7.24, $C_E(0)$ is equal to

$$\begin{aligned} C_E(0) &= 2(n-1)\Big[\sum_{i=1}^{n}\sigma^2(i) + 3(n-1)^{-\frac{1}{3}}\sigma^{\frac{2}{3}}(n)\sum_{i=1}^{n-1}\sigma^{\frac{4}{3}}(i) \\ &\quad +3(n-1)^{-\frac{2}{3}}\sigma^{\frac{4}{3}}(n)\sum_{i=1}^{n-1}\sigma^{\frac{2}{3}}(i)\Big] \end{aligned} \tag{D.6}$$

So the expectation of $C_E(0)$ is

$$\begin{aligned} \langle C_E(0)\rangle &= 2(n-1)\Big[n\langle\sigma^2\rangle + 3(n-1)^{\frac{2}{3}}\langle\sigma^{\frac{2}{3}}\rangle\langle\sigma^{\frac{4}{3}}\rangle + 3(n-1)^{\frac{1}{3}}\langle\sigma^{\frac{4}{3}}\rangle\langle\sigma^{\frac{2}{3}}\rangle\Big] \\ &= 2n(n-1)\langle\sigma^2\rangle + 3(n-1)^{\frac{4}{3}}\big[1+(n-1)^{\frac{1}{3}}\big]\langle\sigma^{\frac{2}{3}}\rangle\langle\sigma^{\frac{4}{3}}\rangle \end{aligned} \tag{D.7}$$

$C_E(\infty)$ is given by

$$C_E(\infty) = 8(n-1)\sum_{i=1}^{n}\sigma^2(i), \tag{D.8}$$

and $\langle C_E(\infty)\rangle$ is equal to

$$\langle C_E(\infty)\rangle = 8(n-1)n\langle\sigma^2\rangle. \tag{D.9}$$

Similarly, $C_E(1)$ can be written as

$$\begin{aligned} &C_E(1) \\ =\ &\Big[\sum_{j=1}^{n-1}\sigma(j) + (n-1)^{\frac{1}{2}}\sigma(n)\Big]\Big[(n-1)^{\frac{1}{2}}\sigma(n) + \sum_{i=1}^{n-1}\sigma(i) \\ &+3(n-1)^{-\frac{1}{6}}\sigma^{\frac{1}{3}}(n)\sum_{i=1}^{n-1}\sigma^{\frac{2}{3}}(i) + 3(n-1)^{-\frac{1}{3}}\sigma^{\frac{2}{3}}(n)\sum_{i=1}^{n-1}\sigma^{\frac{1}{3}}(i)\Big] \end{aligned}$$

$$
\begin{aligned}
&= (n-1)\sigma^2(n) + 2(n-1)^{\frac{1}{2}}\sigma(n)\sum_{i=1}^{n-1}\sigma(i) + \sum_{i=1}^{n-1}\sigma^2(i) + \sum_{1\le i\neq j\le n-1}\sigma(i)\sigma(j) \\
&\quad +3(n-1)^{\frac{1}{3}}\sigma^{\frac{4}{3}}(n)\sum_{i=1}^{n-1}\sigma^{\frac{2}{3}}(i) + 3(n-1)^{\frac{1}{6}}\sigma^{\frac{5}{3}}(n)\sum_{i=1}^{n-1}\sigma^{\frac{1}{3}}(i) \\
&\quad +3(n-1)^{-\frac{1}{6}}\sigma^{\frac{1}{3}}(n)\sum_{i=1}^{n-1}\sigma^{\frac{2}{3}}(i)\sum_{i=1}^{n-1}\sigma(i) + 3(n-1)^{-\frac{1}{3}}\sigma^{\frac{2}{3}}(n)\sum_{i=1}^{n-1}\sigma^{\frac{1}{3}}(i)\sum_{i=1}^{n-1}\sigma(i) \\
&= (n-1)\sigma^2(n) + 2(n-1)^{\frac{1}{2}}\sigma(n)\sum_{i=1}^{n-1}\sigma(i) + \sum_{i=1}^{n-1}\sigma^2(i) + \sum_{1\le i\neq j\le n-1}\sigma(i)\sigma(j) \\
&\quad +3(n-1)^{\frac{1}{3}}\sigma^{\frac{4}{3}}(n)\sum_{i=1}^{n-1}\sigma^{\frac{2}{3}}(i) + 3(n-1)^{\frac{1}{6}}\sigma^{\frac{5}{3}}(n)\sum_{i=1}^{n-1}\sigma^{\frac{1}{3}}(i) \\
&\quad +3(n-1)^{-\frac{1}{6}}\sigma^{\frac{1}{3}}(n)\sum_{i=1}^{n-1}\sigma^{\frac{5}{3}}(i) + 3(n-1)^{-\frac{1}{6}}\sigma^{\frac{1}{3}}(n)\sum_{1\le i\neq j\le n-1}\sigma^{\frac{2}{3}}(i)\sigma(j) \\
&\quad +3(n-1)^{-\frac{1}{3}}\sigma^{\frac{2}{3}}(n)\sum_{i=1}^{n-1}\sigma^{\frac{4}{3}}(i) + 3(n-1)^{-\frac{1}{3}}\sigma^{\frac{2}{3}}(n)\sum_{1\le i\neq j\le n-1}\sigma^{\frac{1}{3}}(i)\sigma(j), \qquad \text{(D.10)}
\end{aligned}
$$

and its expectation is

$$
\begin{aligned}
&\langle C_E(1)\rangle \\
&= 2(n-1)\langle\sigma^2\rangle + 2(n-1)^{\frac{3}{2}}\langle\sigma\rangle^2 + (n-1)(n-2)\langle\sigma\rangle^2 \\
&\quad +3(n-1)^{\frac{4}{3}}\langle\sigma^{\frac{4}{3}}\rangle\langle\sigma^{\frac{2}{3}}\rangle + 3(n-1)^{\frac{7}{6}}\langle\sigma^{\frac{5}{3}}\rangle\langle\sigma^{\frac{1}{3}}\rangle \\
&\quad +3(n-1)^{\frac{5}{6}}\langle\sigma^{\frac{1}{3}}\rangle\langle\sigma^{\frac{5}{3}}\rangle + 3(n-1)^{\frac{5}{6}}(n-2)\langle\sigma^{\frac{1}{3}}\rangle\langle\sigma^{\frac{2}{3}}\rangle\langle\sigma\rangle \\
&\quad +3(n-1)^{\frac{2}{3}}\langle\sigma^{\frac{2}{3}}\rangle\langle\sigma^{\frac{4}{3}}\rangle + 3(n-1)^{\frac{2}{3}}(n-2)\langle\sigma^{\frac{2}{3}}\rangle\langle\sigma^{\frac{1}{3}}\rangle\langle\sigma\rangle \\
&= 2(n-1)\langle\sigma^2\rangle + (n-1)\left[(n-2) + 2(n-1)^{\frac{1}{2}}\right]\langle\sigma\rangle^2 \\
&\quad +3(n-1)^{\frac{2}{3}}\left[1 + (n-1)^{\frac{2}{3}}\right]\langle\sigma^{\frac{4}{3}}\rangle\langle\sigma^{\frac{2}{3}}\rangle + 3(n-1)^{\frac{5}{6}}\left[1 + (n-1)^{\frac{2}{6}}\right]\langle\sigma^{\frac{1}{3}}\rangle\langle\sigma^{\frac{5}{3}}\rangle \\
&\quad +3(n-1)^{\frac{2}{3}}\left[1 + (n-1)^{\frac{1}{6}}\right]\langle\sigma^{\frac{1}{3}}\rangle\langle\sigma^{\frac{2}{3}}\rangle\langle\sigma\rangle. \qquad \text{(D.11)}
\end{aligned}
$$

The expected complexity depends both on the number of arms n and moments $\langle\sigma^b\rangle$, $b \in \{\frac{1}{3}, \frac{2}{3}, 1, \frac{4}{3}, \frac{5}{3}, 2\}$.

On the one hand, when the ratios of lower moments to the second moment

vanish, i.e. $\frac{\langle\sigma^b\rangle}{\langle\sigma^2\rangle} = 0, b \in (0, \frac{5}{3}]$, the ratio of the expected complexities becomes

$$\begin{aligned} \frac{\langle T_E(0)\rangle}{\langle T_E(1)\rangle} &= \frac{\langle C_E(0)\rangle}{\langle C_E(1)\rangle} = \frac{2n(n-1)}{2(n-1)} \\ &= n. \end{aligned} \tag{D.12}$$

$$\begin{aligned} \frac{\langle T_E(\infty)\rangle}{\langle T_E(1)\rangle} &= \frac{\langle C_E(\infty)\rangle}{\langle C_E(1)\rangle} = \frac{8(n-1)n}{2(n-1)} \\ &= 4n \end{aligned} \tag{D.13}$$

On the other hand, when all the moments $\langle\sigma^b\rangle$,$b \in (0, 2]$ are finite, the ratio depends on the dominating terms of n, and the complexity ratio is a finite number.

$$\lim_{n\to\infty} \frac{\langle T_E(0)\rangle}{\langle T_E(1)\rangle} = \lim_{n\to\infty} \frac{\langle C_E(0)\rangle}{\langle C_E(1)\rangle} = \frac{2\langle\sigma^2\rangle}{\langle\sigma\rangle^2} \tag{D.14}$$

$$\lim_{n\to\infty} \frac{\langle T_E(\infty)\rangle}{\langle T_E(1)\rangle} = \lim_{n\to\infty} \frac{\langle C_E(\infty)\rangle}{\langle C_E(1)\rangle} = \frac{8\langle\sigma^2\rangle}{\langle\sigma\rangle^2} \tag{D.15}$$

□

Bibliography

[ACBF02] P. Auer, N. Cesa-Bianchi, and P. Fischer. Finite-time analysis of the multiarmed bandit problem. *Machine Learning*, 47(2/3):235–256, 2002.

[ACBFS03] P. Auer, N. Cesa-Bianchi, Y. Freund, and R. E. Schapire. The nonstochastic multiarmed bandit problem. *SIAM J. Comput.*, 32(1):48–77, 2003.

[Ama98] S. Amari. Natural gradient works efficiently in learning. *Neural Computation*, 10(2):251–276, 1998.

[And98] S. Andradottir. A review of simulation optimization techniques. In *WSC '98: Proceedings of the 30th conference on Winter simulation*, pages 151–158, Los Alamitos, CA, USA, 1998. IEEE Computer Society Press.

[AP05] H. Ahn and R. W. Picard. Affective-cognitive learning and decision making: A motivational reward framework for affective agents. In *Affective Computing and Intelligent Interaction, First International Conference*, pages 866–873, 2005.

[Art91] W. B. Arthur. Designing economic agents that act like human agents: A behavioral approach to bounded rationality. *American Economic Review*, 81(2):353–59, May 1991.

[BA96] K. I. Blum and L. F. Abbott. A model of spatial map formation in the hippocampus of the rat. *Neural Computation*, 8(1):85–93, 1996.

[Bal95] P. Baldi. Gradient descent learning algorithm overview: A general dynamic systems perspective. *IEEE Transactions on Neural Networks*, 6(1):182–195, January 1995.

[BF89] D.A. Berry and B. Fristedt. *Bandit problems: Sequential allocation of experiments*. John Wiley and Sons, London, England, 1989.

[BFHS96] W. Burgard, D. Fox, D. Hennig, and T. Schmidt. Estimating the absolute position of a mobile robot using position probability grids. In *AAAI/IAAI, Vol. 2*, pages 896–901, 1996.

[Bis95] C. M. Bishop. *Neural Networks for Pattern Recognition*. Oxford University Press, Inc., New York, NY, USA, 1995.

[BMW+03] F. Bourgault, A. Makarenko, S. Williams, B. Grocholsky, and H. Durrant-Whytet. Information based adaptive robotic exploration. In *Proceedings of the 2003 IEEE/RSJ Intl. Conference on Intelligent Robots and Systems*, pages 540–545, 2003.

[BO96] N. Burgess and J. O'Keefe. Neuronal computations underlying the firing of place cells and their role in navigation. *Hippocampus*, 7(6):749–762, 1996.

[BS69] D.E. Berlyne and J. Slater. perceptual curiosity exploratory behavior, and maze learning. pages 27–37, 1969.

[BSG03] M. Butz, O. Sigaud, and P. Gérard, editors. *Anticipatory Behavior in Adaptive Learning Systems, Foundations, Theories, and Systems*, volume 2684 of *Lecture Notes in Computer Science*. Springer, 2003.

[BV06] D. Bergemann and J. Valimaki. Bandit problems. In S. Durlauf and L. Blume, editors, *The New Palgrave Dictionary of Economics, 2nd ed*. Macmillan Press, London, 2006.

[BW94] I. Bachelder and A. Waxman. Mobile robot visual mapping and localization: a view-based neurocomputational architecture that emulates hippocampal place learning. *Neural Netw.*, 7(6-7):1083–1099, 1994.

[Cas98] A. R. Cassandra. *Exact and approximate algorithms for partially observable markov decision processes*. PhD thesis, Brown University, 1998.

[CGJ96] D. A. Cohn, Z. Ghahramani, and M. I. Jordan. Active learning with statistical models. *Journal of Artificial Intelligence Research*, 4:129–145, 1996.

[CN01] H. Choset and K. Nagatani. Topological simultaneous localization and mapping (slam): toward exact localization without explicit localization. *IEEE Transactions on Robotics and Automation*, 17(2):125 – 137, April 2001.

[Coh96] D. A. Cohn. Neural network exploration using optimal experiment design. *Neural Network.*, 9(6):1071–1083, 1996.

[CT91] T. M. Cover and J. A. Thomas. *Elements of information theory*. Wiley-Interscience, 1991.

[CZZZ05] Y. Cao, W. Zheng, L. Zhao, and C. Zou. Expression recognition using elastic graph matching. In *Affective Computing and Intelligent Interaction, First International Conference*, pages 8–15, 2005.

[Dam94] A. R. Damasio. *Descartes' error: Emotion, reason and the human brain*. Gosset/Putnam Press, New York, USA, 1994.

[DNDW+01] G. Dissanayake, P. Newman, H. Durrant-Whyte, S. Clark, and M. Csorba. A solution to the simultaneous localization and map building (slam) problem. *IEEE Transactions on Robotic and Automation*, 17(3):229–241, 2001.

[DOD+06] N. D. Daw, J. P. O'Doherty, P. Dayan, B. Seymour, and R. J. Dolan. Cortical substrates for exploratory decisions in humans. *Nature*, 441(7095):876–879, 2006.

[DWB06a] H. Durrant-Whyte and T. Bailey. Simultaneous localization and mapping: part i. *IEEE Robotics and Automation Magazine*, 13(2):99–110, 2006.

[DWB06b] H. Durrant-Whyte and T. Bailey. Simultaneous localization and mapping: part ii. *IEEE Robotics and Automation Magazine*, 13(3):108–117, 2006.

[EDMM02] E. Even-Dar, S. Mannor, and Y. Mansour. Pac bounds for multi-armed bandit and markov decision processes. In *COLT '02: Proceedings of the 15th Annual Conference on Computational Learning Theory*, pages 255–270, London, UK, 2002. Springer-Verlag.

[EDMM06] E. Even-Dar, S. Mannor, and Y. Mansour. Action elimination and stopping conditions for the multi-armed bandit and reinforcement learning problems. *Journal of Machine Learning Research*, 7(2/3):1079–1105, 2006.

[Eli02] C. Eliasmith. The myth of the turing machine: the failings of functionalism and related theses. *J. Exp. Theor. Artif. Intell.*, 14(1):1–8, 2002.

[ENT05] C. Estrada, J. Neira, and J. Tardos. Hierarchical slam: real-time accurate mapping of large environments. *IEEE Transactions on Robotics*, 4:588–596, 2005.

[EP03] A. I. Eliazar and R. Parr. Dp-slam: Fast, robust simultaneous localization and mapping without predetermined landmarks. In *IJCAI*, pages 1135–1142, 2003.

[ERP07] U. Ernst, D. Rotermund, and K. Pawelzik. Efficient computation based on stochastic spikes. *Neural Computation*, 19 (5):1313–1343, 2007.

[FBT99a] D. Fox, W. Burgard, and S. Thrun. Markov localization for mobile robots in dynamic environments. *Journal of Artificial Intelligence Research*, 11, 1999.

[FBT99b] D. Fox, W. Burgard, and S. Thrun. Markov localization for mobile robots in dynamic environments. *Journal of Artificial Intelligence Research*, 11:391–427, 1999.

[Fed72] V. V. Fedorov. *Theory of optimal experiments*. Academic press, New York, 1972.

[FKM05] A. D. Flaxman, A. T. Kalai, and H. B. McMahan. Online convex optimization in the bandit setting: gradient descent without a gradient. In *SODA '05: Proceedings of the sixteenth annual ACM-SIAM symposium on Discrete algorithms*, pages 385–394, Philadelphia, PA, USA, 2005. Society for Industrial and Applied Mathematics.

[FMW+04] M. Fyhn, S. Molden, M. P. Witter, E. I. Moser, and M. B. Moser. Spatial representation in the entorhinal cortex. *Science*, 305(5688):1258–1264, August 2004.

[Fon95] P. W. L. Fong. A quantitative study of hypothesis selection. In *Proceedings of the Twelfth International Conference on Machine Learning*, pages 226–234, San Francisco, CA, 1995. Morgan Kaufmann.

[Fow65] H. Fowler. *Curiosity and Exploratory Behavior*. The Macmillan Company, New York, USA, 1965.

[FTBD01] D. Fox, S. Thrun, W. Burgard, and F. Dellaert. Particle filters for mobile robot localization. In Arnaud Doucet, Nando de Freitas, and Neil Gordon, editors, *Sequential Monte Carlo Methods in Practice*, New York, 2001. Springer.

[FW86] J. E. Freund and R. E. Walpole. *Mathematical statistics (4th ed.)*. Prentice-Hall, Inc., Upper Saddle River, NJ, USA, 1986.

[GBFK98] J.-S. Gutmann, W. Burgard, D. Fox, and K. Konolige. An experimental comparison of localization methods. In *Proc. of the IEEE/RSJ International Conference on Intelligent Robots and Systems*, 1998.

[GG07] A. Gorchetchnikov and S. Grossberg. Space, time, and learning in the hippocampus: How fine spatial and temporal scales are expanded into population codes for behavioral control. *to appear in Neural Networks*, 2007.

[Git89] J.C. Gittins. *Multi-armed bandit allocation indices*. John Wiley and Sons, New York, NY, U.S.A., 1989.

[GKM04] J. F. Guzowski, J. Knierim, and I. Moser. Ensemble dynamics of hippocampal regions ca3 and ca1. *Neuron*, 44(4):581–584, 2004.

[GN02] J. Guivant and E. M. Nebot. Improving computational and memory requirements of simultaneous localization and map building algorithms. In *ICRA*, pages 2731–2736, 2002.

[Gol95] D. Goleman. *Emotional Intelligence*. Bantam Books, New York, USA, 1995.

[GOS79] J. D. Gibbons, I. Olkin, and M. Sobel. An introduction to ranking and selection. *The American Statistician*, 33(4):185–195, 1979.

[GSB07] G. Grisetti, C. Stachniss, and W. Burgard. Improved techniques for grid mapping with rao-blackwellized particle filters. *IEEE Transactions on Robotics*, 2007. In press.

[Hay86] S. Haykin. *Adaptive filter theory*. Prentice-Hall, Inc., Upper Saddle River, NJ, USA, 1986.

[HB04] J. Hawkins and S. Blakeslee. *On Intelligence*. Times Books, 2004.

[Heb49] D. Hebb. *The Organization of Behavior: A Neuropsychological Theory*. Wiley, New York, June 1949.

[Her01] J. M. Herrmann. Dynamical systems for predictive control of autonomous robots. *Theory in Biosciences*, 120(3-4):241–251, 2001.

[HFBT03] D. Hähnel, D. Fox, W. Burgard, and S. Thrun. A highly efficient FastSLAM algorithm for generating cyclic maps of large-scale environments from raw laser range measurements. In *Proceedings of the Conference on Intelligent Robots and Systems (IROS)*, 2003.

[HFM+05] T. Hafting, M. Fyhn, S. Molden, M. Moser, and E. Moser. Microstructure of a spatial map in the entorhinal cortex. *Nature*, June 2005.

[Hop82] J. Hopfield. Neural networks and physical systems with emergent collective computational abilities. *PNAS*, 79(8):2554–2558, April 1982.

[HPG99] J. M. Herrmann, K. Pawelzik, and T. Geisel. Self-localization of autonomous robots by hidden representations. *Autonomous Robots*, 7 (1):31–40, 1999.

[JJNH91] R. A. Jacobs, M. I. Jordan, S. J. Nowlan, and G. E. Hinton. Adaptive mixtures of local experts. *Neural Computation*, 3(1):79–87, 1991.

[Kae93] L. P. Kaelbling. *Learning in embedded systems*. MIT Press, Cambridge, MA, USA, 1993.

[KLM96] L. P. Kaelbling, M. L. Littman, and A. P. Moore. Reinforcement learning: A survey. *Journal of Artificial Intelligence Research*, 4:237–285, 1996.

[Kot05] A. Kotov. Application of psychological characteristics to d-script model for emotional speech processing. In *Affective Computing and Intelligent Interaction, First International Conference*, pages 294–302, 2005.

[KT79] D. Kahneman and A. Tversky. Prospect theory: An analysis of decision under risk. *Econometrica*, 47(2):263–91, March 1979.

[KW97] J. Kivinen and M. K. Warmuth. Exponentiated gradient versus gradient descent for linear predictors. *Information and Computation*, (132):1–63, 1997.

[Les69] D. Lester. *Explorations in exploration: stimulation seeking; an enduring problem in psychology*. Von Nostrand Reinhold Company, New York, NY, USA, 1969.

[LF99] J. Leonard and H. Feder. A computationally efficient method for large-scale concurrent mapping and localization. In D. Koditschek J. Hollerbach, editor, *International Symposium on Robotics Research*, 1999.

[Lin88] R. Linsker. Self-organization in a perceptual network. *Computer*, 21(3):105–117, 1988.

[Lin92] R. Linsker. Local synaptic learning rules suffice to maximize mutual information in a linear network. *Neural Computation*, 4(5):691–702, 1992.

[LK02] C. H. Loch and S. Kavadias. Dynamic portfolio selection of npd programs using marginal returns. *Manage. Sci.*, 48(10):1227–1241, 2002.

[LM97] F. Lu and E. Milios. Globally consistent range scan alignment for environment mapping. *Auton. Robots*, 4(4):333–349, 1997.

[LMS+03] B. Lisien, D. Morales, D. Silver, G. Kantor, I. Rekleitis, and H. Choset. Hierarchical simultaneous localization and mapping. In *Proceedings of the 2003 IEEE/RSJ Intl. Conference on Intelligent Robots and Systems*, 2003.

[Low94] G. Lowenstein. The psychology of curiosity: a review and reinterpretation. *Psychological Bulletin*, 116(1):75–98, 1994.

[LR85] T.L. Lai and H. Robbins. Asymptotically effcient adaptive allocation rules. *Advances in Applied Mathematics*, 6:4–22, 1985.

[LS06] Y. Loewenstein and H. S. Seung. Operant matching is a generic outcome of synaptic plasticity based on the covariance between reward and neural activity. *Proc Natl Acad Sci U S A*, 103(41):15224–15229, October 2006.

[Mac92] D. MacKay. Information-based objective functions for active data selection. *Neural Computation*, 4(4):590–604, 1992.

[MB99] N. Meuleau and P. Bourgine. Exploration of multi-state environments: Local measures and back-propagation of uncertainty. *Machine Learning*, 35(2):117–154, 1999.

[MBG+96] B. L. McNaughton, C. A. Barnes, J. L. Gerrard, K. Gothard, M. W. Jung, J. J. Knierim, H. Kudrimoti, Y. Qin, W. E. Skaggs, M. Suster, and K. L. Weaver. Deciphering the hippocampal polyglot: the hippocampus as a path integration system path. *The Journal of Experimental Biology*, (199):173–185, 1996.

[MBJ+06] B. L. Mcnaughton, F. P. Battaglia, O. Jensen, E. I. Moser, and M. Moser. Path integration and the neural basis of the 'cognitive map'. *Nature Reviews Neuroscience*, 7(8):663–678, 2006.

[Min86] M. Minsky. *The society of mind*. Simon & Schuster, Inc., New York, NY, USA, 1986.

[Mor88] H. P. Moravec. Sensor fusion in certainty grids for mobile robots. *AI Magazine*, 9(2):61–74, 1988.

[MT03] M. Montemerlo and S. Thrun. Fastslam 2.0: An improved particle filtering algorithm for simultaneous localization and mapping that provably converges. In *Proc. IEEE Conf. Robotics Automation*, pages 1985–1991, 2003.

[MT04] S. Mannor and J. N. Tsitsiklis. The sample complexity of exploration in the multi-armed bandit problem. *Journal of Machine Learning Research*, 5:623–648, 2004.

[MTKW02] M. Montemerlo, S. Thrun, D. Koller, and B. Wegbreit. Fastslam: a factored solution to the simultaneous localization and mapping problem. In *Eighteenth national conference on Artificial intelligence*, pages 593–598, Menlo Park, CA, USA, 2002. American Association for Artificial Intelligence.

[Mur99] K. P. Murphy. Bayesian map learning in dynamic environments. In *NIPS*, pages 1015–1021, 1999.

[Mur05] K. Murphy. A survey of pomdp solution techniques. Technical report, University of British Columbia, November 2005.

[New05] M. E. J. Newman. Power laws, pareto distributions and zipf's law. *Contemporary Physics*, 46:323, 2005.

[NJ02] A. Ng and M. Jordan. On discriminative vs. generative classifiers: A comparison of logistic regression and naive bayes. In *Advances in Neural Information Processing Systems 14*, 2002.

[NPB95] I. Nourbakhsh, R. Powers, and S. Birchfield. Dervish: An office-navigating robot. *AI Magazine*, 16(2), 1995.

[NT01] J. Neira and J.D. Tardós. Data association in stochastic mapping using the joint compatibility test. *IEEE Transactions on Robotic and Automation*, 17(6):890–897, 2001.

[OB05] J. O'Keefe and N. Burgess. Dual phase and rate coding in hippocampal place cells: theoretical significance and relationship to entorhinal grid cells. *Hippocampus*, 15(7):853–66, 2005.

[OD71] J. O'Keefe and J. Dostrovsky. The hippocampus as a spatial map. preliminary evidence from unit activity in the freely-moving rat. *Brain Res*, 34(1):171–175, November 1971.

[ON78] J. O'Keefe and L. Nadel. *The Hippocampus as a Cognitive Map*. Clarendon Press, Oxford, U.K, 1978.

[ONPS05] M. Ochs, R. Niewiadomski, C. Pelachaud, and D. Sadek. Intelligent expressions of emotions. In *Affective Computing and Intelligent Interaction, First International Conference*, pages 707–714, 2005.

[OS98] J. Oh and H. Seung. Learning generative models with the up-propagation algorithm. In *NIPS '97: Proceedings of the 1997 conference on Advances in neural information processing systems 10*, pages 605–611, Cambridge, MA, USA, 1998. MIT Press.

[PAF00] H. Park, S. Amari, and K. Fukumizu. Adaptive natural gradient learning algorithms for various stochastic models. *Neural Networks*, 13(7):755–764, 2000.

[Pan98] J. Panksepp. *Affective Neuroscience*. Oxford university press, 1998.

[Pas03] M. A. Paskin. Thin junction tree filters for simultaneous localization and mapping. In Georg Gottlob and Toby Walsh, editors, *Proceedings of the Eighteenth International Joint Conference on Artificial Intelligence (IJCAI-03)*, pages 1157–1164, San Francisco, CA, 2003. Morgan Kaufmann Publishers.

[Pic97] R. W. Picard. *Affective Computing*. The MIT Press, 1997.

[PKM96] K. Pawelzik, J. Kohlmorgen, and K.-R. Müller. Annealed competition of experts for a segmentation and classification of switching dynamics. *Neural Computation*, 8:342–358, 1996.

[Put75] H. Putnam. The nature of mental states. *Mind, language and reality*, pages 429–440, 1975.

[RHW86] D. E. Rumelhart, G. E. Hinton, and R. J. Williams. Learning internal representations by error propagation. pages 318–362, 1986.

[Rob52] H. Robbins. Some aspects of the sequential design of experiments. *Bulletin American Mathematical Society*, 55:527–535, 1952.

[SB98a] R. S. Sutton and A. G. Barto. *Introduction to Reinforcement Learning*. MIT Press, Cambridge, MA, 1998.

[SB98b] R. S. Sutton and A. G. Barto. *Reinforcement Learning: An Introduction*. MIT Press, Cambridge, MA, 1998.

[SB03] C. Stachniss and W. Burgard. Exploring unknown environments with mobile robots using coverage maps. In *Proc. of the International Conference on Artificial Intelligence (IJCAI)*, 2003.

[SBC04] S. P. Singh, A. G. Barto, and N. Chentanez. Intrinsically motivated reinforcement learning. In *Advances in Neural Information Processing Systems 18*, 2004.

[SC87] R. Smith and P. Cheeseman. On the representation and estimation of spatial uncertainly. *International Journal of Robotics Research*, 5(4):56–68, 1987.

[Sch91] J. Schmidhuber. Curious model-building control systems. In *IEEE International Joint Conference on Neural Networks (5th IJCNN'91)*, volume 2, pages 1458–1463, Singapore, 1991. IEEE. CU Boulder.

[Sch99] N. N. Schraudolph. Local gain adaptation in stochastic gradient descent. Technical Report IDSIA-09-99, IDSIA, Corso Elvezia 36, 6900 Lugano, Switzerland, March 1999.

[SDM97] W. Schultz, P. Dayan, and P. R. Montague. A neural substrate of prediction and reward. *Science*, 275(5306):1593–1599, March 1997.

[SFL98] C. Smith, H. Feder, , and J. Leonard. Multiple target tracking with navigation uncertainty. In *Proc. IEEE Conference on Decision and Control*, pages 760–761, 1998.

[SGB05] C. Stachniss, G. Grisetti, and W. Burgard. Information gain-based exploration using rao-blackwellized particle filters. In *Proc. of Robotics: Science and Systems (RSS)*, Cambridge, MA, USA, 2005.

[SHP07] B. Si, J. M. Herrmann, and K. Pawelzik. Gain-based exploration: From multi-armed bandits to partially observable environments. In *Proceedings of the International Conference on Natual Computation*. 2007.

[SJY03] J. R. Swisher, S. H. Jacobson, and E. Yuecesan. Discrete-event simulation optimization using ranking, selection, and multiple comparison procedures: A survey. *ACM Transctions on Modeling and Computer Simulation*, 13(2):134–154, 2003.

[SK95] R. Simmons and S. Koenig. Probabilistic robot navigation in partially observable environments. In *Proceedings of the International*

Joint Conference on Artificial Intelligence (IJCAI '95), pages 1080 – 1087, July 1995.

[SK97] H. Shatkay and L. P. Kaelbling. Learning topological maps with weak local odometric information. In *IJCAI (2)*, pages 920–929, 1997.

[SMSM00] R. Sutton, D. McAllester, S. Singh, and Y. Mansour. Policy gradient methods for reinforcement learning with function approximation. In *Advances in Neural Information Processing Systems 12*, pages 1057–1063, 2000.

[SOS92] H. S. Seung, M. Opper, and H. Sompolinsky. Query by committee. In *Proceedings of the Fifth Annual ACM Conference on Computational Learning Theory (COLT 1992)*, pages 287–294, New York, NY, USA, 1992. ACM Press.

[SPH04] B. Si, K. Pawelzik, and J. M. Herrmann. Robot exploration by subjectively maximizing objective information gain. In *Proceedings of the IEEE International Conference on Robotics and Biomimetics*, pages 862–867. 2004.

[SSC90] R. Smith, M. Self, and P. Cheeseman. Estimating uncertain spatial relationships in robotics. In Ingemar J. Cox and Gordon T. Wilfong, editors, *Autonomous robot vehicles*. Springer-Verlag New York, Inc, 1990.

[Sut90] R. S. Sutton. Integrated architectures for learning, planning, and reacting based on approximating dynamic programming. In *Proceedings of the Seventh International Conference on Machine Learning*, pages 216–224, San Francisco, CA, 1990. Morgan Kaufmann.

[Sut92a] R. S. Sutton. Adapting bias by gradient descent: An incremental version of delta-bar-delta. In *National Conference on Artificial Intelligence*, pages 171–176, 1992.

[Sut92b] R. S. Sutton. Gain adaptation beats least squares? In *Proceedings of the 7th Yale Workshop on Adaptive and Learning Systems*, pages 161–166, 1992.

[TBF98] S. Thrun, W. Burgard, and D. Fox. A probabilistic approach to concurrent mapping and localization for mobile robots. *Machine Learning*, 31(1-3):29–53, 1998.

[TBF05] S. Thrun, W. Burgard, and D. Fox. *Probabilistic Robotics*. The MIT Press, September 2005.

[TFBD00] S. Thrun, D. Fox, W. Burgard, and F. Dellaert. Robust monte carlo localization for mobile robots. *Artificial Intelligence*, 128(1-2):99–141, 2000.

[Thr92] S. B. Thrun. Efficient exploration in reinforcement learning. Technical Report CMU-CS-92-102, School of Computer Science, Carnegie Mellon University, Pittsburgh, PA, USA, 1992.

[Thr98] S. Thrun. Bayesian landmark learning for mobile robot localization. *Machine Learning*, 33(1):41–76, 1998.

[Thr02] S. Thrun. Robotic mapping: A survey. In G. Lakemeyer and B. Nebel, editors, *Exploring Artificial Intelligence in the New Millenium*. Morgan Kaufmann, 2002.

[TK05] E. E. Thomson and W. B. Kristan. Quantifying stimulus discriminability: A comparison of information theory and ideal observer analysis. *Neural Comput.*, 17(4):741–778, 2005.

[TLK+04] S. Thrun, Y. Liu, D. Koller, A. Ng, Z. Ghahramani, and H. Durrant-Whyte. Simultaneous localization and mapping with sparse extended information filters. *International Journal of Robotics Research*, 23(7-8):693–716, 2004.

[TM92] S. B. Thrun and K. Möller. Active exploration in dynamic environments. In J. E. Moody, S. J. Hanson, and R. P. Lippmann, editors, *Advances in Neural Information Processing Systems 4*, pages 531–538, Sab Mateo, CA, 1992. Morgan Kaufmann Publishers.

[TMR90] J. S. Taube, R. U. Muller, and J. B Ranck. Head direction cells recorded from the postsubiculum in freely moving rats: I. description and quantitative analysis. *Journal of Neuroscience*, 10:420–435, 1990.

[TS95] M. Tsodyks and T. Sejnowski. Associative memory and hippocampal place cells. *Int. Journal of Neural Systems*, 6:S81–S86, 1995.

[Tur88] A. M. Turing. Computing machinery and intelligence. In A. Collins and E. E. Smith, editors, *Readings in Cognitive Science: A Perspective from Psychology and Artificial Intelligence*, pages 6–19. Kaufmann, San Mateo, CA, 1988.

[Val84] L. G. Valiant. A theory of the learnable. *Communications of the ACM*, 27(11):1134–1142, 1984.

[VK83] H.G. Voss and H Keller. *Curiosity and Exploration: Theories and Results*. Academic. Press, New York, USA, 1983.

[VSS06] S. V. N. Vishwanathan, N. N. Schraudolph, and A. J. Smola. Step size adaptation in reproducing kernel hilbert space. *Journal of Machine Learning Research*, 7:1107–1133, June 2006.

[Wat89] C. Watkins. *Learning from Delayed Rewards*. PhD thesis, Cambridge University, 1989.

[WD92] C. Watkins and P. Dayan. Q-learning. *Machine Learning*, 8(3-4):279–292, 1992.

[Whi59] R. White. Motivation reconsidered: The concept of competence. *Psychological Review*, 66(5):297–333, 1959.

[Wil92] R. J. Williams. Simple statistical gradient-following algorithms for connectionist reinforcement learning. *Machine Learning*, 8(3):229–256, 1992.

[WLC+05] T. J. Wills, C. Lever, F. Cacucci, N. Burgess, and J. O'Keefe. Attractor dynamics in the hippocampal representation of the local environment. *Science*, 308(5723):873–876, May 2005.

[WM93] M. A. Wilson and B. L. McNaughton. Dynamics of the hippocampal ensemble code for space. *Science*, 261(5124):1055–1058, August 1993.

[WTR94] H. Wan, D. Touretzky, and A. Redish. Towards a computational theory of rat navigation. In *Proceedings of the 1993 Connectionist Models Summer School*, pages 11–19. Lawrence Earlbaum Associates, 1994.

[Yam97] B. Yamauchi. A frontier-based approach for autonomous exploration. In *CIRA '97: Proceedings of the 1997 IEEE International Symposium on Computational Intelligence in Robotics and Automation*, page 146, Washington, DC, USA, 1997. IEEE Computer Society.

[YL96] B. Yamauchi and P. Langley. Place learning in dynamic real-world environments. In *Proceedings of RoboLearn 96: An International Workshop on Learning for Autonomous Robots*, pages 123–129, Key West, FL., 1996.

[ZGMS98] K. Zhang, I. Ginzburg, B. McNaughton, and T. Sejnowski. Interpreting neuronal population activity by reconstruction: A unified framework with application to hippocampal place cells. *J. Neurophysiol.*, 79:1017–1044, 1998.

Index

Q-learning, 12, 19, 119
0/1 loss, 71

action selection, 4, 118, 137
action value, 118
adaptive behavior, 136
adaptive multiplicative rule, 37, 136
additive rule, 35
additive step, 20
affective computing, 2
attention, 49
attractor network, 4

backpropagation, 19
batch learning, 19
Bayes filters, 31
Bayesian network, 7, 136
belief modulation, 36, 37
belief state, 7, 33

Chebyshev Inequality, 73, 74, 89
checkerboard maze, 9, 14
competence motivation, 112
confidence interval, 58
counting rule, 36, 137
curiosity, 1, 2, 49, 111

data association, 5, 31
    multihypothesis data association, 9, 36, 37
distribution
    Bernoulli distribution, 113
    Boltzmann distribution, 93
    Chi-square distribution, 58, 59
    normal distribution, 61
    power law distribution, 61, 85, 95
    uniform distribution, 61, 87, 95

eligibility trace, 119
emotion, 1
equilibrium condition, 55, 69
exploration, 3
    counter-based exploration, 49
    error-based exploration, 49
    gain-based exploration, 49
    random exploration, 52
exploration for exploitation, 78
exploration policy
    t-interval exploration policy, 53
    z-interval exploration policy, 53, 82, 90
    ideal gain-maximization exploration policy, 54, 137
    least-taken policy, 52
    realistic gain-maximization exploration policy, 59, 137

FastSLAM, 5
Fisher information matrix, 22
frontier-based exploration, 123, 138

gain adaptation, 21
gain maximization, 138
general gain-maximization exploration strategy, 69
generative model, 8
geometric step, 20
global performance estimation, 114, 138
gradient descent, 19
    adaptive multiplicative gradient descent, 24, 137
    additive gradient descent, 20
    BFGS method, 22

conjugate gradient descent, 22
exponentiated gradient descent, 20
meta gradient descent, 22
multiplicative gradient descent, 20
natural gradient descent, 22
Newton's method, 21
quasi-Newton method, 22
grid cells, 3

head direction cells, 3
Hessian matrix, 21, 26
hippocampus, 3
Hoeffding inequality, 73
homing, 3

information gain, 49
information gain-maximization, 6
intelligence, 1
artificial intelligence, 1
interval estimation method, 52

Jensen's inequality, 57, 84

Kalman filter, 19
extended Kalman filter (EKF), 5, 31
Kullback-Leibler divergence, 18, 25, 36

Lagrange multipliers, 55, 70
learning rate, 20, 136
learning rate adaptation, 21, 136
least squares, 19
local error reduction, 113, 138
local learning rate, 114, 138
localization, 4, 30

mapping, 4, 30
Markov Decision Process, 50
maximum likelihood estimation, 37
memory-prediction framework, 7
meta-exploration, 58, 61
mind, 2
motion model, 5, 33
motivation, 1
multi-armed bandit, 9, 50, 137
multiplicative rule, 37
mutual information-maximization, 6

navigation, 3

occupancy probability, 14, 31
on-line learning, 19
optimal exploration, 138

pairwise ordering, 71, 88
Partially Observable Markov Decision Process (POMDP), 7, 136
place cells, 3
place learning, 3, 30
prediction record, 17
Probably Approximately Correct (PAC), 72, 88

quadratic loss, 18, 35

Rao-Blackwellized particle filter, 5
reinforcement learning, 112
reward
external reward, 111
intrinsic reward, 33, 111, 138
reward maximization, 78
reward-maximization policy
ϵ-greedy policy, 12, 93
t-interval policy, 53, 93
z-interval policy, 52, 93
ideal gain-based reward-maximization policy, 94
realistic gain-based reward-maximization policy, 94
soft-max policy, 93
Upper Confidence Bound (UCB) policy, 12, 93

sample complexity, 71, 88
selection policy
general gain-maximization selection strategy, 82

ideal counter-based selection policy, 82, 90
ideal gain-maximization selection policy, 83, 137
realistic gain-maximization selection policy, 84, 137
selection task, 80
self evaluation, 136
sensor model, 5, 33
Simultaneous Localization and Mapping, 30
space exploration, 3
spatial behaviors, 3
state, 33

time difference error, 120
Turing test, 1

Wilson interval, 113