Subchapter 1.1

Curse of optimality, and how do we break it

Xun Yu Zhou[1]

1 – Columbia University, Department of Industrial Engineering and Operations Research, and The Data Science Institute, New York, NY 10027, USA, Email: xz2574@columbia.edu

We strive to seek optimality, but often find ourselves trapped in bad "optimal" solutions that are either local optimizers, or are too rigid to leave any room for errors, or are simply based on wrong models or erroneously estimated parameters. A way to break this "curse of optimality" is to engage exploration through randomization. Exploration broadens search space, provides flexibility, and facilitates learning via trial and error. We review some of the latest development in this exploratory approach in the stochastic control setting with continuous time and spaces.

1.1.1 Introduction

Optimal solutions derived from various optimization theories are often bad traps that hinder practical use. An example immediately coming to mind is a local optimizer out of the first-order condition. Another example is the bang–bang control in optimal control theory: an optimal control takes only extreme values when the control variable appears linearly in the Hamiltonian. Such a control is too sensitive to estimation errors and thus tend to be very unstable and hardly usable.

Classical theories also often take the "separation principle" between estimation and optimization; see (Wonham, 1968) for example. One typically assumes a model, estimates model parameters based on past data, and then optimizes as if the underlying model was correct. Think of a gambler at an array of slot machines

("one-armed bandits") that have different but unknown probabilities of winning. He has to decide how many times to play each machine and in what order so as to maximize the expected gains. The classical estimation-and-optimization approach will tackle the problem in the following way: playing each machine for n rounds, where n is sufficiently and judiciously large, and observing the outcomes. If, say, Machine 1 has returned the most gains, then the gambler will believe it is indeed the best machine and he will henceforth play this machine *only*.

The flaw of this approach is evident: Machine 1 may well be a sub-optimal machine and sticking to it subsequently may then be a bad trap. This example is a precursor of what is now widely known as a *reinforcement learning* (RL) problem. The RL approach would take the bandits problem in a different way and formulate it as one that trades off near-term and long-term gains. Specifically, the gambler carefully balances between greedily exploiting what has been learned so far to choose the machine that yields near-term higher rewards, and continuously exploring the rest of the machines to acquire more information to potentially achieve long-term benefits. The so-called *ε-greedy strategy* (Sutton and Barto, 2018) exemplifies this idea: at nth play the gambler tosses a biased coin with head occurring with a probability $1 - \varepsilon_n$ and tail a probability ε_n. He then plays the *current* best machine if head appears and the other machines at random (with uniform probability) if tail appears. Here $\varepsilon_n > 0$ is a small number and ought to be smaller as n becomes larger.

The ε-greedy strategy is a *randomized* strategy: at each play, instead of deterministically and definitely playing a particular machine, the gambler lets a coin flip decide which machine to play. The problem now becomes how to design the scheme for $\{\varepsilon_n\}_{n\in\mathbb{N}}$ to achieve a good balance between exploration (learning) and exploitation (optimizing). A notable feature, and indeed one that is essentially different from the classical approach, is that the gambler is no longer interested in estimating the winning probabilities of the machines; rather he is focusing on learning the best sequence $\{\varepsilon_n\}_{n\in\mathbb{N}}$. In other words, *he learns his best strategies instead of learning a model*. This underpins the basic tenet in RL: An agent does not pre-assume a structural model nor attempt to estimate an exiting model's parameters but, instead, gradually learns the best (or near-best) strategies based on trial and error, through interactions with the black box environment and incorporation of the responses of these interactions.[1] This learning approach addresses to large extent the problem of "curse of optimality" due to engaging a wrong model.

The *exploration through randomization* approach may also be employed to break the curse of optimality even in problems where learning is not necessary. Take for example the non-convex optimization where the function to be minimized

[1]This sounds strikingly different from the model-based approach; but a careful reflection would reveal that it is exactly how people, especially babies and young children, learn things. Take learning a new language for example. Adults usually start with learning the grammar (the model) before actually speaking, whereas babies directly learn to speak (strategies) through interactions and trial-and-error. It is widely held that the latter learn a language much fast and effectively than the former.

is completely known. Still, the first-order condition and the associated algorithms such as the gradient descent (GD) give only local minima. *Simulated annealing*, independently proposed by (Kirkpatrick et al., 1983) and (Cerny, 1985), performs randomization at each iteration of the GD algorithm to get the iterates out of any possible trap of a local minimum. Specifically, at each iteration, the algorithm randomly samples a solution close to the current one and moves to it according to a probability distribution. This scheme facilitates a broader search or exploration for the global optima with the risk of moving to worse solutions at some iterations. The risk is however controlled by slowly cooling down over time the "temperature" which is used to characterize the level of exploration. Another example is to use randomization to smooth out an overly sensitive (and hence unstable) optimal bang-bang control that takes only extreme actions.

Randomization uses a probability distribution (measure) to replace a deterministic action. However, the latter can be embedded into the former as a Dirac measure. To avoid the situation in which the optimal distribution turns out to be a Dirac measure, one can force a minimal level of exploration. In the RL literature, entropy has been used to measure the level of exploration and *entropy-regularized* (also termed as "softmax") exploratory formulation has been proposed, mostly in the discrete-time and discrete-space Markov Decision Processes (MDPs) setting. In this formulation, exploration enters explicitly into the optimization objective as a regularization term, with a trade-off weight (the *temperature* parameter) imposed on the entropy of the exploration strategy; see (Nachum et al., 2017; Neu et al., 2017; Ziebart et al., 2008) and the references therein. (Wang et al., 2020) is the first to extend this formulation to the setting of stochastic control with continuous time and continuous state and action (control) spaces. They derive a stochastic relaxed control formulation to model the repetitive learning in RL, and use the differential entropy to regularize the exploration. They show that the optimal distribution for exploration is a *Gibbs measure* or a *Boltzmann distribution* of the form $\pi(u) \propto e^{\frac{1}{\lambda}H(u)}$ where λ is the temperature and H is the Hamiltonian. When the state depends on action u linearly and the reward is quadratic in u, the Hamiltonian is quadratic in u and hence the Gibbs measure specializes to a Gaussian distribution (under some technical assumptions), which in turn justifies the widely used *Gaussian exploration* (Haarnoja et al., 2017). (Wang and Zhou, 2020) further apply this result to a continuous-time Markowitz mean–variance portfolio selection problem, and devise an RL algorithm to learn the efficient investment strategies without any knowledge about the key parameters such as stocks' mean returns and volatility rates.

With a motivation other than RL, (Gao et al., 2020) apply the general framework and results of (Wang et al., 2020) to the temperature control problem for Langevin diffusions. A Langevin diffusion is a continuous-time version of a simulated annealing algorithm – the Langevin algorithm – to find the global minima of a non-convex function. The temperature process controls the level of random noises injected into the algorithm. The selection of this process can be formulated as a classical stochastic control problem, whose optimal solution is nevertheless

bang-bang and hence extremely prone to model mis-specifications. (Gao et al., 2020) take the entropy-regularized framework of (Wang et al., 2020) by randomizing this temperature process, and conclude that a truncated exponential distribution is optimal to sample temperatures and in turn sample the noises to be injected into the Langevin algorithm.

This subchapter reviews the approaches and main results in (Gao et al., 2020; Wang et al., 2020; Wang and Zhou, 2020), albeit in a finite time horizon instead of the infinite one, argues that exploration through randomization can effectively address the curse of optimality in settings including but not limited to RL, and suggests some open research questions.

The remainder of this subchapter proceeds as follows. In Section 1.1.2 we present the entropy-regularized exploratory stochastic control problem based on the notion of exploration through randomization. Section 1.1.3 derives the optimal distributions for sampling actions to control the dynamics. Section 1.1.4 gives a concrete application of the general theory to the sampling problem of the Langevin algorithm. In Section 1.1.5 we discuss the algorithmic aspects of the general theory in the RL context. Finally, Section 1.1.6 concludes.

1.1.2 Entropy-Regularized Exploratory Formulation

1.1.2.1 Classical stochastic control

Let $T > 0$, $b : [0, T] \times \mathbb{R}^d \times U \mapsto \mathbb{R}^d$ and $\sigma : [0, T] \times \mathbb{R}^d \times U \mapsto \mathbb{R}^{d\times n}$ be given. The classical stochastic control problem is to control the *state* (or *feature*) dynamics, a stochastic differential equation (SDE):

$$dx_s^u = b(s, x_s^u, u_s)ds + \sigma(s, x_s^u, u_s)dW_s,\ s \in [0, T]. \tag{1.1.1}$$

The process $u = \{u_s, 0 \le s \le T\}$, defined on a filtered probability space $(\Omega, \mathcal{F}, \mathbb{P}; \{\mathcal{F}_s\}_{s\ge0})$ along with a standard $\{\mathcal{F}_s\}_{s\ge0}$-adapted, n-dimensional Brownian motion $W = \{W_s,\ s \ge 0\}$, is an admissible (*open-loop*) control, denoted by $u \in \mathcal{A}^{\text{cl}}$, if (i) it is an $\{\mathcal{F}_s^W\}_{s\ge0}$-adapted measurable process taking values in U, where $\{\mathcal{F}_s^W\}_{s\ge0} \subset \{\mathcal{F}_s\}_{s\ge0}$ is the natural filtration generated by the Brownian motion, and $U \subset \mathbb{R}^m$ is the *action space* representing the constraints on an agent's decisions (*controls* or *actions*), and (ii) for any given initial condition $x_0^u = x_0 \in \mathbb{R}^d$, the SDE (1.1.1) admits solutions $x^u = \{x_s^u, 0 \le s \le T\}$ on the same filtered probability space, whose distributions are all identical.[2]

Given $x_0^u = x_0 \in \mathbb{R}^d$ at time $t = 0$, the objective of the control problem is to find $u \in \mathcal{A}^{\text{cl}}$ so that the total reward

$$J(u) := \mathbb{E}\left[\int_0^T r\left(s, x_s^u, u_s\right) ds + h(x_T^u)\right] \to \max \tag{1.1.2}$$

[2]Throughout this subchapter, admissible controls are defined in the *weak* sense, namely, the filtered probability space and the Brownian motion are also *part* of the control. This is to ensure, among others, that dynamic programming works; see (Yong and Zhou, 1999, Chapter 4). For simplicity, however, we will refer to, for example, only the process u as a control.

where $r : [0,T] \times \mathbb{R}^d \times U \mapsto \mathbb{R}$ and $h : \mathbb{R}^d \mapsto \mathbb{R}$ are the running and terminal reward functions respectively.

In the classical setting where the model is fully known (namely, when the functions b, σ, r and h are fully specified), one can solve this problem by Bellman's dynamic programming in the following manner; see e.g. (Yong and Zhou, 1999) for a systematic account of the method. Define the *optimal value function*

$$V^{\mathrm{cl}}(t,x) := \sup_{u \in \mathcal{A}^{\mathrm{cl}}} \mathbb{E}\left[\int_t^T r(s, x_s^u, u_s)\, ds + h(x_T^u) \middle| x_t^u = x\right], \quad (t,x) \in [0,T] \times \mathbb{R}^d, \tag{1.1.3}$$

where (and throughout this subcharpter) t and x are generic variables representing respectively the current time and state of the system dynamics.[3]

If $V^{\mathrm{cl}} \in C^{1,2}([0,T] \times \mathbb{R}^d)$, then it satisfies the *Hamilton–Jacobi–Bellman (HJB) equation*

$$\begin{cases} v_t(t,x) + \sup_{u \in U} H(t,x,u,v_x(t,x),v_{xx}(t,x)) = 0, \ (t,x) \in [0,T) \times \mathbb{R}^d; \\ v(T,x) = h(x) \end{cases} \tag{1.1.4}$$

where H is the (generalized) *Hamiltonian* (Yong and Zhou, 1999, Chapters 3 & 4)

$$\begin{aligned} H(t,x,u,p,P) &= \tfrac{1}{2}\mathrm{tr}\left[\sigma(t,x,u)' P \sigma(t,x,u)\right] + p \cdot b(t,x,u) + f(t,x,u), \\ &(t,x,u,p,P) \in [0,T] \times \mathbb{R}^d \times U \times \mathbb{R}^d \times \mathbb{R}^{d \times d}, \end{aligned} \tag{1.1.5}$$

where $\mathrm{tr}(A)$ denotes the trace of a square matrix A.

Let

$$\boldsymbol{u}^*(t,x) := \mathrm{argmax}_{u \in U} H(t,x,u,v_x(t,x),v_{xx}(t,x)), \ (t,x) \in [0,T) \times \mathbb{R}^d. \tag{1.1.6}$$

This is a *deterministic* mapping from the current time and state to the action space U, which is an instance of a *feedback policy* (or *feedback law*). It is important to understand the differences and relationship between an open-loop control and a feedback policy. The former is a stochastic process; so it is a function of the time t and the state of nature ω, and the latter is a deterministic function of the time t and the state of the system x. Throughout this subcharpter we call the former a *control* and the latter a *policy*. A policy $\boldsymbol{u}$ can *generate* a control by substituting $\boldsymbol{u}$ into the system dynamics (1.1.1) starting from any present time–state pair $(t,x) \in [0,T) \times \mathbb{R}^d$.

The verification theorem dictates that $\boldsymbol{u}^*$ is an optimal policy in the sense that it generates an optimal control for the problem (1.1.3) with *any* $(t,x) \in [0,T) \times \mathbb{R}^d$ via $u_s^* = \boldsymbol{u}^*(s, x_s^*)$ where x^* is the solution to (1.1.1) upon substituting u_s with $\boldsymbol{u}^*(s, x_s^*)$.

[3] In the classical control theory literature, V is termed simply the "value function". However, in the sequel, as customary in the RL literature, we will also use the term *value function* for any given feedback policy. Hence, to distinguish, we call V the "*optimal* value function".

Equation (1.1.6) stipulates that at any give time and state, the optimal action is guided by the Hamiltonian, *deterministically* and *rigidly*. Moreover, this action policy is derived off-line at $t = 0$ and *will* be carried out throughout, *assuming*, that is, the model is completely specified.

1.1.2.2 Exploratory formulation

As we have discussed in the introduction, there are various reasons the agent may be unable or unwilling to execute the "optimal" policy (1.1.6), and will instead need to explore through randomization. For example, in the case when the underlying model is not known, the agent is not able to maximize the unknown Hamiltonian in (1.1.6), and hence employs exploration to interact with and learn the best strategies through trial and error. The exploration is modelled by a *distribution* of controls $\pi = \{\pi_s(\cdot), s \geq 0\}$ over the control space U from which each trial is sampled. Here π is a density-function-valued stochastic process; i.e. $\pi_s(\cdot, \omega)$ is a probability density function on U for any $(s, \omega) \in [0, T] \times \Omega$. We therefore extend the notion of controls to distributions when exploration is called for. A classical control $u = \{u_s,\ s \geq 0\}$ can be regarded as a Dirac distribution $\pi_s(\cdot) = \delta_{u_s}(\cdot)$.

This subsection and the next subsection largely follow the formulation and analysis in Wang et al. (2020), except that we are in the setting of a finite time horizon while Wang et al. (2020) is for the infinite time horizon. However, all the results in the current setting can be derived analogously.

Given a distributional control π, the agent repeatedly sample *classical* controls from π for N rounds over the same time horizon to control the dynamics and observe the corresponding values of the total reward. As explained in Wang et al. (2020), when $N \to \infty$, by law of large numbers the limiting system dynamics under π becomes

$$dX_s^\pi = \tilde{b}(s, X_s^\pi, \pi_s)ds + \tilde{\sigma}(s, X_s^\pi, \pi_s)dW_s,\ s \in [0, T], \tag{1.1.7}$$

where the coefficients $\tilde{b}$ and $\tilde{\sigma}$ are defined as

$$\tilde{b}(s, y, \pi) := \int_U b\left(s, y, u\right) \pi(u)du,\ \ y \in \mathbb{R}^d,\ \pi \in \mathcal{P}\left(U\right), \tag{1.1.8}$$

and

$$\tilde{\sigma}(s, y, \pi) := \sqrt{\int_U \sigma^2\left(s, y, u\right) \pi(u)du},\ \ y \in \mathbb{R}^d,\ \pi \in \mathcal{P}\left(U\right), \tag{1.1.9}$$

with $\mathcal{P}\left(U\right)$ being the set of density functions of probability measures on U that are absolutely continuous with respect to the Lebesgue measure.

We call (1.1.7) the *exploratory formulation* of the controlled state dynamics, and $\tilde{b}(\cdot, \cdot)$ and $\tilde{\sigma}(\cdot, \cdot)$ in (1.1.8) and (1.1.9), respectively, the *exploratory drift* and the *exploratory volatility*.

Similarly, the reward function r in (1.1.2) is modified to the *exploratory reward*

$$\tilde{r}\left(s,y,\pi\right):=\int_{U}r\left(s,y,u\right)\pi(u)du,\ \ y\in\mathbb{R}^{d},\ \pi\in\mathcal{P}\left(U\right). \tag{1.1.10}$$

1.1.2.3 Entropy regularization

Given the exploratory formulation, it seems natural to set the objective to maximize

$$\mathbb{E}\left[\int_{0}^{T}\tilde{r}\left(s,X_{s}^{\pi},\pi_{s}\right)ds+h(X_{T}^{\pi})\right] \tag{1.1.11}$$

subject to (1.1.7) under $X_{0}^{\pi}=x_{0}$. However, it is entirely possible that the optimal distributional control for this problem is just Dirac, and hence we would then be in the realm of classical stochastic control. Indeed this happens when the so-called Roxin's condition is satisfied; see (Yong and Zhou, 1999, Chapter 2). Thus, in order to encourage a *genuine* exploration we need to regulate its level. We use Shanon's *differential entropy* to measure the level of exploration:

$$\mathcal{H}(\pi):=-\int_{U}\pi(u)\ln\pi(u)du,\ \ \pi\in\mathcal{P}\left(U\right),$$

and require the total expected entropy to maintain a minimum level

$$-\mathbb{E}\int_{0}^{T}\int_{U}\pi_{s}(u)\ln\pi_{s}(u)duds\geq a \tag{1.1.12}$$

where $a>0$ is given. Taking the Lagrange multiplier of this exploration constraint we arrive at the following new objective:

$$\mathbb{E}\left[\int_{0}^{T}\left(\tilde{r}\left(s,X_{s}^{\pi},\pi_{s}\right)-\lambda\int_{U}\pi_{s}(u)\ln\pi_{s}(u)du\right)ds+h(X_{T}^{\pi})\right]\rightarrow\max, \tag{1.1.13}$$

where $\lambda>0$ is the Lagrange multiplier, which can also be regarded as an exogenous exploration weighting parameter capturing the trade-off between exploitation (the original reward function) and exploration (the entropy). This constant is also known as the *temperature* parameter.

Denote by $\mathcal{B}(U)$ the Borel algebra on U. A density-function-valued process $\pi=\{\pi_{s}(\cdot),\,0\leq s\leq T\}$, defined on a filtered probability space $(\Omega,\mathcal{F},\mathbb{P};\{\mathcal{F}_{s}\}_{s\geq0})$ along with a standard $\{\mathcal{F}_{s}\}_{s\geq0}$-adapted, n-dimensional Brownian motion $W=\{W_{s},\,s\geq0\}$, is an admissible distributional control, denoted by $\pi\in\mathcal{A}$, if (i) for each $0\leq s\leq T$, $\pi_{s}(\cdot)\in\mathcal{P}(U)$ a.s.; (ii) for each $A\in\mathcal{B}(U)$, $\{\int_{A}\pi_{s}(u)du,0\leq s\leq T\}$ is $\{\mathcal{F}_{s}^{W}\}_{s\geq0}$-adapted measurable process; (iii) the SDE (1.1.7) with $X_{0}^{\pi}=x_{0}$ admits solutions $x^{\pi}=\{x_{s}^{\pi},0\leq s\leq T\}$ on the same filtered probability space, whose distributions are all identical.

1.1.3 Optimal Distributional Policies

To solve the entropy-regularized exploratory control problem (1.1.13), we again apply dynamic programming. Introduce the optimal value function

$$V(t,x) := \sup_{\pi\in\mathcal{A}} \mathbb{E}\left[\int_0^\infty \left(\int_U r(s, X_s^\pi, u)\,\pi_s(u)\,du - \lambda\int_U \pi_s(u)\ln\pi_s(u)du\right)ds + h(X_T^\pi)\middle| X_t^\pi = x\right]. \tag{1.1.14}$$

Using standard arguments, we deduce that V satisfies the HJB equation

$$v_t(t,x) + \sup_{\pi\in\mathcal{P}(U)}\int_U \left[H(t,x,u,v_x(t,x),v_{xx}(t,x)) - \lambda\ln\pi(u)\right]\pi(u)du, \quad (t,x)\in[0,T)\times\mathbb{R}^d, \tag{1.1.15}$$

with the terminal condition $v(T,x) = h(x)$.

Noting that $\pi\in\mathcal{P}(U)$ if and only if

$$\int_U \pi(u)du = 1 \quad \text{and} \quad \pi(u)\geq 0 \text{ a.e.} \quad \text{on } U, \tag{1.1.16}$$

we can solve the (constrained) maximization problem on the right hand side of (1.1.15) to get a *feedback* policy:

$$\boldsymbol{\pi}^*(u;t,x) = \frac{1}{Z(\lambda,t,x,v_x(t,x),v_{xx}(t,x))}\exp\left(\frac{1}{\lambda}H(t,x,u,v_x(t,x),v_{xx}(t,x))\right), \tag{1.1.17}$$

where $u\in U,\ (t,x)\in[0,T]\times\mathbb{R}^d$, and

$$Z(\lambda,t,x,v_x(t,x),v_{xx}(t,x)) := \int_U \exp\left(\frac{1}{\lambda}H(t,x,u,v_x(t,x),v_{xx}(t,x))\right)du \tag{1.1.18}$$

is the normalizing factor that makes $\boldsymbol{\pi}^*(\cdot;t,x)$ a density function.

The optimal policy (1.1.17) is a deterministic function of the variables u, t and x. For each given time–state pair (t,x), $\boldsymbol{\pi}^*(\cdot;t,x)$ is the density function of a Gibbs measure. When the temperature λ is very high, all the actions are chosen in largely equal probabilities. When the temperature cools down as $\lambda\to 0$, the distribution increasingly concentrates around the (global) maximizers of the Hamiltonian, giving rise to something resembling the ε-greedy policies in multi-armed bandit problems. When $\lambda = 0$, the distribution degenerates into the Dirac measure on the maximizers of the Hamiltonian which is the classical optimal control.

In the linear–quadratic (LQ) case when b,σ are linear in x and u and r,h quadratic in x and u, then the Hamiltonian is quadratic in u. In the infinite horizon case, Wang et al. (2020) prove that the Gibbs measure specializes to the Gaussian distribution under some technical assumptions. We expect the same to be true for the current case of a finite time horizon, although there may be some technical subtleties. Moreover, Wang and Zhou (2020) apply the LQ results to a continuous-time mean–variance portfolio selection problem and devise an

algorithm to solve it without needing to know the parameters of the underlying stocks.

In RL there is a widely used *heuristic* exploration strategy called the *Boltzmann exploration*, which assigns the following probability to action a when in state s_t at time t:

$$p(s_t, a) = \frac{e^{Q_t(s_t,a)/\lambda}}{\sum_{a=1}^{m} e^{Q_t(s_t,a)/\lambda}}, \quad a = 1, 2, \ldots, m, \tag{1.1.19}$$

where $Q_t(s, a)$ is the *Q-function* value of a state-action pair (s, a), and $\lambda > 0$ is a parameter that controls the level of exploration; see e.g. (Bridle, 1990; Cesa-Bianchi et al., 2017; Sutton and Barto, 2018). There is a clear resemblance between (1.1.17) and (1.1.19). This in turn suggests that the continuous counterpart of the Q-function is the Hamiltonian, given that the former is not well defined and can not be used to rank and select actions in the continuous setting (Tallec et al., 2019). The importance of this observation is twofold: the fact that we are able to derive a result that reconciles with an eminent heuristic strategy in the discrete setting verifies and justifies the entropy-regularized exploratory formulation for the continuous setting, and, more importantly, the formulation lays a *theoretical underpinning* of the Boltzmann exploration, thereby provides explainability of a largely heuristic approach.[4]

Putting (1.1.17) back to (1.1.15), we obtain the following (elegant) form of the HJB equation

$$v_t(t, x) + \lambda \ln Z(\lambda, t, x, v_x(t, x), v_{xx}(t, x)) = 0, \ (t, x) \in [0, T) \times \mathbb{R}^d; \ \ v(T, x) = h(x). \tag{1.1.20}$$

This equation appears to be a new type of parabolic partial differential equations (PDEs), which would provide a whole wealth of new research problems. For example, what about its well-posedness (existence and uniqueness) in both the classical and viscosity senses? How does its solution, along with its first- and second-order derivatives, depend on the temperature $\lambda > 0$? As a result, how does the optimal policy (1.1.17), along with its mean, variance and entropy, depend on λ? Does the solution converge when $\lambda \to 0$ and, if yes, what is the convergence rate? Some of these questions have been answered in Tang et al. (2021).

Another significant direction for research is the choice of the temperature λ. In this section, as in (Wang et al., 2020), λ is set to be an *exogenous* constant. However, the agent is supposed to learn more and hence need less exploration as time goes by. So it seems plausible that λ should depend on time and indeed decay over time. On the other hand, it seems also reasonable that λ should depend on the system state to optimize its use. In other words, λ ought to be *endogenous*. How can we then formulate the problem to optimize the temperature process?

[4] A type of the formula (1.1.17) was first derived in (Wang et al., 2020, eq. (17)), but the connection with Boltzmann exploration and Gibbs measure was not noted there.

1.1.4 Non-Convex Optimization and Langevin Diffusions

While the entropy-regularized exploratory formulation was originally motivated by RL in Wang et al. (2020), its use may go beyond RL, which this section will demonstrate. The presentation follows (Gao et al., 2020), although we take a finite horizon setup as opposed to that of the infinite horizon in (Gao et al., 2020).

Consider a finite-dimensional optimization problem:

$$\min_{x\in\mathbb{R}^d} f(x), \tag{1.1.21}$$

where $f\colon \mathbb{R}^d \to [0,\infty)$ is a *non-convex* function. The traditional gradient descent (GD) algorithm may be trapped in a local optimum. The Langevin algorithm injects noises into GD in order to get out of the trap:

$$X_{k+1} = X_k - \eta f_x(X_k) + \sqrt{2\eta\beta_k}\xi_k\,,\quad k=0,1,2,\cdots, \tag{1.1.22}$$

where f_x is the gradient of f, $\eta>0$ is the step size, $\{\xi_k\}$ is i.i.d Gaussian noise and $\{\beta_k\}$ is a sequence of the temperature parameters that typically decays over time to zero. The continuous-time version of this algorithm is the so-called *overdamped Langevin diffusion*:

$$dX_s = -f_x(X_s)dt + \sqrt{2\beta_s}dW_s,\quad X_0 = x_0 \tag{1.1.23}$$

where $x_0\in\mathbb{R}^d$ is an initialization, $W=\{W_s : s\geq 0\}$ is a standard d-dimensional Brownian motion with $W_0=0$, and $\beta=\{\beta_s : s\geq 0\}$ is an adapted, nonnegative stochastic process, which is also called the *temperature process* of the Langevin diffusion.

When $\beta_s\equiv\beta>0$, under some mild assumptions on f, the solution of (1.1.23) admits a unique stationary distribution which is the Gibbs measure with density $\pi(x)=\frac{1}{Z(\beta)}e^{-\frac{1}{\beta}f(x)}$ (Chiang et al., 1987). When β becomes small, this measure increasingly concentrates on the *global* minima of f. This provides a theoretical justification of using Langevin diffusion (1.1.23) to sample noises for the Langevin algorithm (1.1.22).

A natural problem is to control the temperature process $\{\beta_t : t\geq 0\}$ so that the performance of the continuous-time version of the Langevin algorithm (1.1.23) is optimized. Specifically, given an arbitrary initialization $X_0=x_0\in\mathbb{R}^d$, a computing budget $T>0$, and the range of the temperature $U=[a,b]$ where $0\leq a<b<\infty$, we aim to solve the following stochastic control problem where the temperature process is taken as the control:

$$\begin{array}{ll} \text{Minimize} & \mathbb{E}[f(X_T)], \\ \text{subject to} & \left\{\begin{array}{l} \text{equation (1.1.23)}, \\ \{\beta_s : 0\leq s\leq T\} \text{ is adapted}, \\ \beta_s\in U \text{ a.e.} s\in[0,T], \text{ a.s.} \end{array}\right. \end{array} \tag{1.1.24}$$

This is a classical control problem. Its HJB equation is:

$$v_t(t,x)+\min_{\beta\in[a,1]}[\beta\mathrm{tr}(v_{xx}(t,x))-f_x(x)\cdot v_x(t,x)]=0,\quad x\in\mathbb{R}^d;\ \ v(T,x)=f(x). \tag{1.1.25}$$

Then, the verification theorem yields that an optimal feedback policy is " bang-bang": $\beta^*(t,x)=b$ if $\mathrm{tr}(v_{xx}(x))<0$, and $\beta^*(t,x)=a$ otherwise. This policy stipulates that one should in some time–state pairs heat at the highest possible temperature, while in others cool down completely, depending on the sign of $\mathrm{tr}(v_{xx}(t,x))$. This policy, while *theoretically* optimal, is clearly too *rigid* to achieve good performance in practice as it concentrates on two extreme actions only, and a computational error of $v_{xx}(t,x)$ may cause drastic change from one extreme to the other. This motivates us to use exploratory formulation and entropy regularization in order to *smooth out* the temperature processes. Note that here the motivation is no longer from "learning" per se as we can perfectly assume that the function f is given and known.

We now present our entropy-regularized exploratory formulation of the problem. Instead of a classical control $\{\beta_s:0\le s\le T\}$ where $\beta_s\in U=[a,b]$ for $s\in[0,T]$, we consider a distributional control $\pi=\{\pi_s(\cdot):0\le s\le T\}$, which represents a randomization of classical controls over the control space U where a temperature $\beta_s\in U$ can be sampled from this distribution whose probability density function is $\pi_s(\cdot)$ at time s. The optimal value function of the exploratory problem is

$$V(t,x):=\inf_{\pi\in\mathcal{A}}\mathbb{E}\left[-\lambda\int_U\pi_s(u)\ln\pi_s(u)duds+f(X_T^\pi)\Big|X_t^\pi=x\right], \tag{1.1.26}$$

where the system dynamic is

$$dX_s^\pi=-f_x(X_s^\pi)dt+\tilde{\sigma}(\pi_s)dW_s, \tag{1.1.27}$$

with

$$\tilde{\sigma}(\pi):=\sqrt{\int_U 2u\pi(u)du}. \tag{1.1.28}$$

This problem is a special case of the general problem formulated in the previous section (except that we now have a minimization problem instead of a maximization one). Applying the general results there, we obtain the following optimal feedback policy:

$$\pi^*(u;t,x)\ =\ \frac{1}{Z(\lambda,v_{xx}(t,x))}\exp\left(-\frac{1}{\lambda}[\mathrm{tr}(v_{xx}(t,x))u]\right), \tag{1.1.29}$$

where $u\in U,\ \ (t,x)\in[0,T]\times\mathbb{R}^d$, and

$$Z(\lambda,v_{xx}(t,x)):=\int_U\exp\left(-\frac{1}{\lambda}[\mathrm{tr}(v_{xx}(t,x))u]\right)du>0.$$

This is a *truncated* (in U) *exponential distribution* with the (state-dependent) parameter $c(t,x) := \frac{\mathrm{tr}(v_{xx}(t,x))}{\lambda}$, and we do not require $\mathrm{tr}(v_{xx}(x)) > 0$ (i.e. v is in general non-convex) or $c(t,x) > 0$ here.

The HJB equation is

$$v_t(t,x) - f_x(x)\cdot v_x(x) - \lambda \ln(Z(\lambda, v_{xx}(t,x))) = 0, \quad (t,x) \in [0,T)\times\mathbb{R}^d, \qquad (1.1.30)$$

with $v(T,x) = f(x)$.

To apply the obtained results to sample the Langevin algorithm (1.1.22), we can follow the following steps. First, we solve the HJB equation (1.1.30) to get v. Second, with the initialization $X_0 = x_0$, and for each $k = 0, 1, 2, \cdots$, we sample β_k from $\pi^*(\cdot; \eta_k, X_k)$ where π^* is determined by (1.1.29), X_k is the current iterate and η_k is the cumulative step size. Finally we apply (1.1.22) to move to the next iterate where ξ_k is independently sampled from a standard Gaussian distribution. For a numerical experiment comparing the performance of this method (albeit based on the infinite horizon model) with other benchmarks, see (Gao et al., 2020).

1.1.5 Algorithmic Considerations for RL

The previous sections are more about the *theory* of an entropy-regularized exploratory formulation. We now discuss some aspects of the algorithm design in the RL context. Specifically, we need to design RL algorithms to *learn* the optimal solutions of the entropy-regularized problems and to output implementable policies, without assuming any knowledge about the underlying parameters or attempting to estimate these parameters.

First thing to note is that some of the theoretical results presented earlier already have algorithmic implications. For example, if we know Gaussian is optimal, then we will need to learn only two parameters (mean and variance). If an exponential distribution is optimal, then there is only one parameter to learn. Making use of this information could dramatically simplify the corresponding algorithms and speed up their convergence.

The following discussion, however, is more general without targeting for a particular distribution. It is a generalization of the algorithm developed in (Wang and Zhou, 2020) for the mean–variance portfolio selection problem. The two key steps involved in our algorithm are *policy evaluation* and *policy improvement*, as standard in RL for MDPs (Sutton and Barto, 2018).

First we define the *value function* of a given distributional policy $\boldsymbol{\pi}$. Note that $\boldsymbol{\pi}$ generates an open-loop distributional control through the exploratory dynamics (1.1.7) in the same way as in classical control. Specifically, for each given current time–state pair $(t,x) \in [0,T)\times\mathbb{R}^d$, $\boldsymbol{\pi}$ generates an open-loop control

$$\pi_s(u) := \boldsymbol{\pi}(u; s, X_s^\pi) \qquad (1.1.31)$$

where $\{X_s^\pi,\, t \le s \le T\}$ solves (1.1.7) with $X_t^\pi = x$ when the policy $\boldsymbol{\pi}$ is applied and assuming that $\{\pi_s(\cdot), t \le s \le T\} \in \mathcal{A}$. Now define the value function of $\boldsymbol{\pi}$:

$$V^{\boldsymbol{\pi}}(t,x) := \mathbb{E}\left[\int_t^T \left(\int_U r(s, X_s^\pi, u)\,\pi_s(u)\,du - \lambda\int_U \pi_s(u)\ln\pi_s(u)du\right)ds + h(X_T^\pi)\,\middle|\, X_t^\pi = x\right]. \qquad (1.1.32)$$

In an RL algorithm, one starts with an initial policy $\boldsymbol{\pi_0}$.[5] For each given $\boldsymbol{\pi_k}$, $k = 0, 1, 2, \cdots$, policy evaluation is carried out to obtain its value function $V^{\boldsymbol{\pi_k}}$. Then, a policy improvement theorem specifies the next policy $\boldsymbol{\pi_{k+1}}$, and the iterations go on. We now describe these steps.

For the policy evaluation, we follow Doya (2000) for learning the value function $V^{\boldsymbol{\pi}}$ under any arbitrarily given admissible policy $\boldsymbol{\pi}$. By Bellman's consistency, we have

$$V^{\boldsymbol{\pi}}(t,x) = \mathbb{E}\left[\int_t^{t'} \left(\int_U r\left(s, X_s^\pi, u\right) \pi_s\left(u\right) du - \lambda \int_U \pi_s(u) \ln \pi_s(u) du\right) ds + V^{\boldsymbol{\pi}}(t', X_{t'}^\pi) \Big| X_t^\pi = x\right], \tag{1.1.33}$$

for any $(t,x) \in [0,T) \times \mathbb{R}^d$ and $t' \in (t,T]$. This is actually analogous to the Bellman's principle of optimality for the *optimal* value function. Rearranging this equation and dividing both sides by $t'-t$, we obtain

$$\mathbb{E}\left[\frac{V^{\boldsymbol{\pi}}(t', X_{t'}^\pi) - V^{\boldsymbol{\pi}}(t, X_t^\pi)}{t'-t} + \frac{1}{t'-t}\int_t^{t'} \left(\int_U r\left(s, X_s^\pi, u\right) \pi_s\left(u\right) du - \lambda \int_U \pi_s(u) \ln \pi_s(u) du\right) ds \Big| X_t^\pi = x\right] = 0.$$

Letting $t' \to t$ in the left hand side motivates the definition of the *temproral difference* (TD) error

$$\delta_t := \dot{V}_t^{\boldsymbol{\pi}} + \int_U r\left(t, X_t^\pi, u\right) \pi_t\left(u\right) du - \lambda \int_U \pi_t(u) \ln \pi_t(u) du, \tag{1.1.34}$$

where $\dot{V}_t^{\boldsymbol{\pi}} := \frac{d}{dt} V^{\boldsymbol{\pi}}(t, X_t^\pi)$ is the sample-wise total derivative of $V^{\boldsymbol{\pi}}$ along (t, X_t^π).

The objective of the policy evaluation procedure is to minimize the expected total squared TD error in order to find the value function $V^{\boldsymbol{\pi}}$. In general, this can be done as follows. Denote by V^θ and $\boldsymbol{\pi}^\phi$ respectively the parameterized value function and policy (upon using regressions or neural networks, or taking advantage of any known parametric forms of them), with θ, ϕ being the vectors of suitable dimensions to be learned. We then minimize

$$\begin{aligned} C(\theta,\phi) :=& \ \tfrac{1}{2}\mathbb{E}\left[\int_0^T |\delta_t|^2 dt\right] \\ =& \ \tfrac{1}{2}\mathbb{E}\left[\int_0^T \left|\dot{V}_t^\theta + \textstyle\int_U r(t, X_t^\phi, u)\pi_t^\phi\left(u\right) du - \lambda \int_U \pi_t^\phi(u) \ln \pi_t^\phi(u) du\right|^2 dt\right], \end{aligned}$$

where $\pi^\phi = \{\pi_t^\phi(\cdot), 0 \le t \le T\}$ is generated from $\boldsymbol{\pi}^\phi$ with respect to a given initial state $X_0 = x_0$ at time 0. To approximate $C(\theta,\phi)$, we first discretize $[0,T]$ into small intervals $[t_i, t_{i+1}]$, $i = 0, 1, \cdots, l$, with a equal length Δt, where $t_0 = 0$ and $t_{l+1} = T$. Then we collect a set of samples $\mathcal{D} = \{(t_i, x_i), i = 0, 1, \cdots, l+1\}$ in the following way. The initial sample is $(0, x_0)$ for $i = 0$. Now, at each t_i, $i = 0, 1, \cdots, l$, we sample $\pi_{t_i}^\phi$ to obtain $u_i \in U$ and then use the *constant* control $u_t \equiv u_i$ to control the (classical) system dynamics (1.1.1) during $[t_i, t_{i+1})$. We

[5]The choice of this initialization can also be guided by the theory. For instance, if the theory stipulates that Gaussian is optimal, then we can choose $\boldsymbol{\pi_0}$ as Gaussian with some initial values of the mean and variance.

observe the state x_{i+1} at the next time instant t_{i+1} along with the reward r_i collected over $[t_i, t_{i+1})$. We then approximate $\dot{V}_t^\theta$ by

$$\dot{V}^\theta(t_i, x_i) := \frac{V^\theta(t_{i+1}, x_{i+1}) - V^\theta(t_i, x_i)}{\Delta t},$$

and approximate $C(\theta, \phi)$ by

$$C(\theta, \phi) = \frac{1}{2} \sum_{(t_i, x_i) \in \mathcal{D}} \left(\dot{V}^\theta(t_i, x_i) + r_i + \lambda \int_U \pi_{t_i}^\phi(u) \ln \pi_{t_i}^\phi(u) du \right)^2 \Delta t. \tag{1.1.35}$$

Finally, we search $(\theta^*, \phi^*)'$ that minimize $C(\theta, \phi)$ using stochastic gradient descent algorithms; see, for example, (Goodfellow et al., 2016, Chapter 8). This in turn leads to the value function V^{θ^*}, concluding the policy evaluation step.[6]

The policy improvement step is to update the next policy based on the current policy $\boldsymbol{\pi}$ along with the corresponding value function $V^{\boldsymbol{\pi}}$, the latter having been found by the policy evaluation. Assume that $V^{\boldsymbol{\pi}} \in C^{1,2}([0,T) \times \mathbb{R}^d) \cap C^0([0,T] \times \mathbb{R}^d)$, and that the policy $\tilde{\boldsymbol{\pi}}$ defined by

$$\tilde{\boldsymbol{\pi}}(u; t, x) = \frac{1}{Z(\lambda, t, x, V_x^{\boldsymbol{\pi}}(t,x), V_{xx}^{\boldsymbol{\pi}}(t,x))} \exp\left(\frac{1}{\lambda} H(t, x, u, V_x^{\boldsymbol{\pi}}(t,x), V_{xx}^{\boldsymbol{\pi}}(t,x)) \right) \tag{1.1.36}$$

generates admissible (open-loop) distributional controls for the exploratory dynamics (1.1.7). Then we can prove that $\tilde{\boldsymbol{\pi}}$ is better than $\boldsymbol{\pi}$ in that

$$V^{\tilde{\boldsymbol{\pi}}}(t,x) \geq V^{\boldsymbol{\pi}}(t,x), \quad (t,x) \in [0,T] \times \mathbb{R}^d. \tag{1.1.37}$$

There is an obvious resemblance between the updating rule (1.1.36) and the optimal policy (1.1.17). Their proofs are also similar: $\tilde{\boldsymbol{\pi}}$ achieves the supremum in (1.1.15) where v is replaced with $V^{\boldsymbol{\pi}}$. For a proof in the mean–variance setting, see Wang and Zhou (2020).

1.1.6 Conclusion

In this subchapter, we have put forth the notion of "curse of optimality" to capture the theoretical and empirical observations that traditional approaches to optimization often end with unfavorable solutions that are not globally optimal, or too extreme to be useful, or outright irrelevant practically. We find that an entropy-regularized exploratory reformulation of the problem, originally motivated by balancing exploration and exploitation for reinforcement learning, may

[6] In a recent paper, Jia and Zhou (2021) consider a general policy evaluation problem with continuous time and space. Applying a martingale approach, the authors find that the mean-square TD error method introduced here actually minimizes temporal variations rather than achieving accurate evaluation. They derive alternative policy evaluation methods based on the martingality, some of which correspond to well-studied TD algorithms such as TD(0) and TD(λ) for disctete-time MDPs.

provide viable solutions to *all* these setbacks. This is because the randomization involved in such a formulation helps escape from local traps, broadens search space and reduces the desire to be "perfect" (extreme) by allowing more flexibility and accommodation. In the realm of continuous time and state/action spaces, this is still a largely uncharted research area where open problems abound.

Bibliography

Bridle, J. S. (1990). Training stochastic model recognition algorithms as networks can lead to maximum mutual information estimation of parameters. In *Advances in neural information processing systems*, pages 211–217.

Cerny, V. (1985). Thermodynamical approach to the traveling salesman problem: an efficient simulation algorithm. *Journal of Optimization Theory and Applications*, 45(1):41–51.

Cesa-Bianchi, N., Gentile, C., Lugosi, G., and Neu, G. (2017). Boltzmann exploration done right. In *Advances in neural information processing systems*, pages 6284–6293.

Chiang, T.-S., Hwang, C.-R., and Sheu, S. J. (1987). Diffusion for global optimization in $\mathbb{R}^n$. *SIAM Journal on Control and Optimization*, 25(3):737–753.

Doya, K. (2000). Reinforcement learning in continuous time and space. *Neural Computation*, 12(1):219–245.

Gao, X., Xu, Z. Q., and Zhou, X. Y. (2020). State-dependent temperature control for langevin diffusions. *arXiv preprint arXiv:2011.07456;* to appear in *SIAM Journal on Control and Optimization*.

Goodfellow, I., Bengio, Y., and Courville, A. (2016). *Deep Learning*. MIT Press. `http://www.deeplearningbook.org`.

Haarnoja, T., Tang, H., Abbeel, P., and Levine, S. (2017). Reinforcement learning with deep energy-based policies. In *Proceedings of the 34th International Conference on Machine Learning*, pages 1352—-1361.

Jia, Y. and Zhou, X. Y. (2021). Policy evaluation and temporal-difference learning in continuous time and space: A martingale approach. *arXiv preprint arXiv:2108.06655*.

Kirkpatrick, S., Gelatt, J., and Vecchi, M. (1983). Optimization by simulated annealing. *Science*, 220(4598):671–680.

Nachum, O., Norouzi, M., Xu, K., and Schuurmans, D. (2017). Bridging the gap between value and policy based reinforcement learning. In *Advances in Neural Information Processing Systems*, pages 2775–2785.

Neu, G., Jonsson, A., and Gómez, V. (2017). A unified view of entropy-regularized markov decision processes. *arXiv preprint arXiv:1705.07798*.

Sutton, R. S. and Barto, A. G. (2018). *Reinforcement Learning: An Introduction*. MIT press.

Tallec, C., Blier, L., and Ollivier, Y. (2019). Making deep q-learning methods robust to time discretization. *arXiv preprint arXiv:1901.09732*.

Tang, W., Zhang, Y. P., and Zhou, X. Y. (2021). Exploratory HJB equations and their convergence. *arXiv preprint arXiv:2109.10269*.

Wang, H., Zariphopoulou, T., and Zhou, X. Y. (2020). Reinforcement learning in continuous time and space: A stochastic control approach. *Journal of Machine Learning Research*, 21:1–34.

Wang, H. and Zhou, X. Y. (2020). Continuous-time mean–variance portfolio selection: A reinforcement learning framework. *Mathematical Finance*, 30:1273–1308.

Wonham, M. (1968). On the separation theorem of stochastic control. *SIAM Journal on Control*, 6(2):312–326.

Yong, J. and Zhou, X. Y. (1999). *Stochastic Controls: Hamiltonian Systems and HJB Equations*, volume 43. Springer Science & Business Media.

Ziebart, B. D., Maas, A. L., Bagnell, J. A., and Dey, A. K. (2008). Maximum entropy inverse reinforcement learning. In *AAAI*, volume 8, pages 1433–1438. Chicago, IL, USA.