Tutorial: Active Learning in Transportation and Logistics

Warren B Powell
Princeton University, wbpowell328@gmail.com, jungle.princeton.edu

---

Transportation and logistics poses a wide range of problem settings that require making decisions in the presence of uncertainty. Decisions might span routing and scheduling people and equipment, fleet planning, hiring driveres, planning inventories, consolidating orders, pricing, booking reservations, online advertising, equipment purchases, and technology investments. In addition to decisions that act directly on our transportation system, researchers face problems in the laboratory working with simulators, where decisions have to be made regarding tunable parameters in algorithms, or the choice of which experiment to run.

All of these decisions have to deal with different types of uncertainty. Uncertainty arises from customer demands, market response to pricing, the behavior of competitors, network congestion, weather, equipment problems, and people making decisions in the field. We also have to deal with the variability of computer simulations arising from any of a variety of factors.

A common theme in making decisions in transportation and logistics, whether it is in the field or the laboratory in a simulator, is the need to do some amount of experimentation, which means learning. We may have sources of uncertainty where we do not know the distributions of uncertain quantities, or we may be trying to learn how a system responds to different inputs. For example, we may need to learn how the market responds to price, how drivers respond to our ability to get them home, or how a simulator responds to assumptions about fleet mix or network configurations.

The richness of these problems can be seen in over 15 communities that contribute to decisions under uncertainty. We divide this literature between those that are motivated primarily by some sort of resource allocation problem under uncertainty, and those centered on active learning. The heart of the active learning literature is associated with the multiarmed bandit problem, a topic that has received almost no attention in the transportation science literature.

We have developed a single canonical framework (Powell 2019) that covers every sequential decision problem under uncertainty. This framework poses all sequential decision problems with an objective that involves optimizing over policies, which are methods for making decisions. We have identified four (meta)classes of policies that cover all the different strategies that have been used, either in the literature or industry practice.

The goal of this tutorial will be to focus on problems in transportation and logistics that involve active learning, which means that the decisions we make control or influence the observations we receive. We are going to argue that the same four classes apply to either stochastic resource allocation problems, pure learning problems, or hybrids. What changes when we introduce active learning is the design of the policies, and the choice of which policy works best.

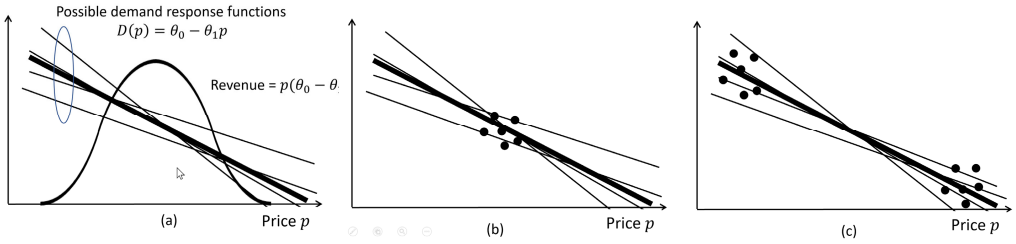


Figure 1 (a) Illustration of distribution of possible demand response functions. (b) Sampling prices that appear to maximize revenue. (c) Sampling extreme prices to improve estimates of the demand response functions.

We are going to begin in section 1 by introducing an active learning problem that arises in pricing (think about surge pricing for ride sharing, pricing for truckload markets or e-commerce), which we will present initially as a pure learning problem. We then transition to a hybrid problem by introducing inventory planning.

1. An active learning problem

Consider the problem of trying to learn the best price to maximize revenue from selling a product. Figure 1(a) illustrates our assumed linear relationship $D(p) = \theta_0 - \theta_1 p$ between demand and price, and depicts our uncertainty in this relationship. The bold line is the average, which we can use to create an estimated revenue curve.

To model the learning problem, assume we pick a price p_t and then observe demand $\hat{D}_{t+1}$ between t and $t+1$. Let $\bar{p}_t = (1 \quad p_t)^T$ be our input vector for the demand response function $D(p) = \theta_0 - \theta_1 p$ which we are trying to estimate. Let $\bar{\theta}_t$ be our estimate of θ given what we know as of time t. Standard equations for recursive least squares provides the following updating equations:

$$\varepsilon_{t+1} = (\bar{\theta}_{t0} - \bar{\theta}_{t1} p_t) - \hat{D}_{t+1}, \tag{1}$$

$$M_{t+1} = M_t - \frac{1}{\gamma_t} M_t (\bar{p}_t)(\bar{p}_t)^T M_t, \tag{2}$$

$$\gamma_t = 1 - (\bar{p}_t)^T M_t \bar{p}_t, \tag{3}$$

$$\bar{\theta}_{t+1} = \bar{\theta}_t + \frac{1}{\gamma_t} M_t \bar{p}_t \varepsilon_{t+1}, \tag{4}$$

$$\tag{5}$$

where M_t is a 2×2 matrix.

If someone was picking the prices p_t for us, then this would be a classical passive learning problem, but imagine now that we get to pick the prices. If we charge prices that maximize revenue, we will end up sampling points near the middle, which makes it hard to estimate the demand function (see figure 1(b)). The best way to learn the demand function is to test prices near the extremes (see figure 1(c)), but if we do this, we minimize revenue. The best policy would be to choose prices in-between. This is the classic tradeoff of exploration and exploitation.

This is a pure learning problem, but consider dynamic pricing for ride-sharing, truckload markets, and e-commerce, where learning the demand response is imbedded in a complex resource allocation problem. In fact, we would claim that there are many problems arising in transportation and logistics where we have to make decisions, then make an observation from which we learn more about the problem, which then guides future decisions. The challenge is first how to model this as an optimization problem, and then how to solve it.

2. A canonical framework for sequential decisions

There are a variety of frameworks for making decisions under uncertainty. Some of the most important are Markov decision processes (Puterman 2005), stochastic programming (Birge and Louveaux 2011), stochastic search (Spall 2003), "simulation optimization" (which is an expanding umbrella, as seen in Fu (2014)), optimal control (Lewis, Vrabie, and Syrmos 2012) and decision trees (Skinner 1999). A separate literature has evolved around decision problems focusing on learning. These include the multiarmed bandit problem as studied in applied probability (Gittins, Glazebrook, and Weber 2011) and computer science (Bubeck and Cesa-Bianchi 2012), "active learning" as studied within the machine learning community Settles (2010), and a body of literature from applied math, geosciences and operations research using terms such as expected improvement, knowledge gradient and kriging (see Powell and Ryzhov (2012)).

We use a simple framework that covers all of these fields, as described in Powell (2019). It consists of five elements:

- State variables S_t - This captures all the information we need to know at time t to model the system from time t onward. It consists of the state of physical resources R_t (inventories, the location of a vehicle), other information I_t, and a belief state B_t which captures probabilistic information about parameters we cannot observe directly (this is the focus of our work). The belief state in our learning problem is $B_t = (\bar{\theta}_t, M_t)$.
- Decision variables x_t. Decisions are determined by a policy $X^\pi(S_t)$ which depends only on the information in S_t. The decision in our pricing problem is $x_t = p_t$.
- Exogenous information variables, W_{t+1}, which is the information that arrives after we make decision x_t but before time $t+1$. This would be the demand $\hat{D}_t$ in our pricing problem.
- The transition function, where $S_{t+1} = S^M(S_t, x_t, W_{t+1})$. This describes how each state evolves over time. Equations (1) - (4) would be the transition equations in our pricing problem.
- The objective function. We begin by distinguishing two attributes of objectives:

  —Whether the problem is state-independent, or state-dependent. We let $\mathbb{E}_W F(x, W)$ represent state-independent objectives, where the function $F(x, W)$ does not depend on any dynamic information in S_t. We let $C(S_t, x_t)$ represent costs or contributions which might depend on the state S_t.

  —Whether we are optimizing the final-reward (how well we do after a period of training) or cumulative reward (where we want to optimize performance along the way).

The two most common objectives are

a) State-dependent, cumulative reward - This is given by

$$\max_\pi \mathbb{E}_{S_0} \mathbb{E}_{W_1,\ldots,W_T|S_0} \left\{ \sum_{t=0}^{T} C(S_t, X^\pi(S_t), W_{t+1}) | S_0 \right\}. \tag{6}$$

b) State-independent, final reward - This is given by

$$\max_\pi \mathbb{E}_{S^0} \mathbb{E}_{W^1,\ldots,W^N|S^0} \mathbb{E}_{\widehat{W}|S^0} F(x^{\pi,N}, \widehat{W}), \tag{7}$$

We would use (7) if we are optimizing using a simulator, but we would use (6) if we are learning in the field.

3. Designing policies

Note that both objectives (6) and (7) involve maximizing over policies. There are two general strategies for designing policies:

Policy search - Here we use either of the objective functions (6) - (7) to search within a family of functions to find the policy that works best. This means we have to a) find a class of function and b) tune any parameters. The challenge is finding the right family, and then performing the tuning (which can be hard). Policy search can be further divided into two classes:

- Policy function approximations (PFAs) - These are analytical functions that map states to actions (all remaining policies involve solving an optimization problem).
- Cost function approximations (CFAs) - Here we solve a (typically deterministic) optimization problem where the objective and/or constraints are parameterized to mitigate uncertainty.

Lookahead approximations - Alternatively, we can construct policies by approximating the impact of a decision now on the future. The challenge here is designing and computing the approximation of the future (this is also hard). Lookahead approximations can be further divided into two classes:

- Policies based on value function approximations (VFAs) - Here we use an approximation of the value of the state $S_{t+1} = S^M(S_t, x_t, W_{t+1})$ that we transition to from a decision x_t when we are in state S_t.
- Direct lookahead approximations (DLAs) - We optimize over a horizon $t, \ldots, t+H$ using an approximate lookahead model (which might be deterministic) to determine the best decision to make now.

All four classes of policies have been applied to both dynamic resource allocation problems, as well as to pure learning problems. While the same classes apply to both settings, the choice of which policies work best reflects the problem being solved. For example:

- With some exceptions (such as (s, S) policies for inventory planning), policies for managing physical resources require some sort of lookahead policy, possibly based on value function approximations (e.g. Simão et al. (2009) and Bouzaiene-Ayari et al. (2016)), but more typically a direct lookheads, which is the policy used by Google maps (which continuously updates a shortest path to the destination, or virtually any dynamic vehicle routing problem Pillac et al. (2013)).
- The most popular policies for pure learning problems use upper confidence bounding, which use a policy of the form

$$X^{UCB}(S^n|\theta^{UCB}) = \arg\max_{x\in\mathcal{X}}(\bar{\mu}_x^n + \theta^{UCB}\bar{\sigma}_x^n)$$

where $\mathcal{X}$ is a set of discrete choices, $\bar{\mu}_x^n$ is the estimate of the performance of alternative n, and $\bar{\sigma}_x^n$ is the standard deviation of $\bar{\mu}_x^n$. Also popular are one-period lookahead policies such as knowledge gradient or expected improvement (Powell and Ryzhov 2012).

Now consider problems in transportation and logistics, where we are managing a physical resource while collecting information: routing a drone to collect information about crops, moving, pricing a truckload movement to learn how the market responds (but requires moving the truck), and managing a utility truck to learn about power outages while fixing them (Al-Kanj, Powell, and Bouzaiene-ayari 2016). We will describe several strategies that combine the need for lookahead policies to handle the dynamic management of resources, with strategies to perform active learning to accelerate the learning process.

References

Al-Kanj L, Powell WB, Bouzaiene-ayari B, 2016 *The Information-Collecting Vehicle Routing Problem : Stochastic Optimization for Emergency Storm Response. arxiv* 1–32.

Birge JR, Louveaux F, 2011 *Introduction to Stochastic Programming* (New York: Springer), 2nd edition.

Bouzaiene-Ayari B, Cheng C, Das S, Fiorillo R, Powell WB, 2016 *From single commodity to multiattribute models for locomotive optimization: A comparison of optimal integer programming and approximate dynamic programming. Transportation Science* 50(2):1–24, URL `http://dx.doi.org/10.1287/trsc.2014.0536`.

Bubeck S, Cesa-Bianchi N, 2012 *Regret Analysis of Stochastic and Nonstochastic Multi-armed Bandit Problems. Foundations and Trends in Machine Learning* 5(1):1–122, URL `http://dx.doi.org/10.1561/2200000024`.

Fu MC, 2014 *Handbook of Simulation Optimization* (New York: Springer).

Gittins J, Glazebrook KD, Weber RR, 2011 *Multi-Armed Bandit Allocation Indices* (New York: John Wiley & Sons).

Lewis FL, Vrabie D, Syrmos VL, 2012 *Optimal Control* (Hoboken, NJ: John Wiley & Sons), 3rd edition.

Pillac V, Gendreau M, Guéret C, Medaglia AL, 2013 *A review of dynamic vehicle routing problems. European Journal of Operational Research* 225(1):1–11, URL `http://dx.doi.org/10.1016/j.ejor.2012.08.015`.

Powell WB, 2019 *A unified framework for stochastic optimization. European Journal of Operational Research* 275(3):795–821, URL `http://dx.doi.org/10.1016/j.ejor.2018.07.014`.

Powell WB, Ryzhov IO, 2012 *Optimal Learning* (Hoboken, NJ: John Wiley & Sons).

Puterman M, 2005 *Markov Decision Processes* (Hoboken, NJ: John Wiley & Sons Inc), 2nd edition.

Settles B, 2010 *Active Learning. Sciences-New York* .

Simão H, Day J, George AP, Gifford T, Nienow J, Powell WB, 2009 *An approximate dynamic programming algorithm for large-scale fleet management: A case application. Transportation Science* 43(2), URL `http://dx.doi.org/10.1287/trsc.1080.0238`.

Skinner DC, 1999 *Introduction to Decision Analysis* (Gainesville, Fl.: Probabilistic Publishing).

Spall JC, 2003 *Introduction to Stochastic Search and Optimization: Estimation, simulation and control* (Hoboken, NJ: John Wiley & Sons).