Capacity Control via Convex Optimization

Tianshu Chu

June 3, 2014

Abstract

This project proposes an infinite-horizon stochastic control model to solve the real-time capacity control problem. We build the model for a single facility, adopt a sequence of sequential convex programming, and prove that the solution converges to the optimal value from any initial points. Furthermore, we extend our model to capacity controls in a network of facilities, by introducing additional network constraints.

1 Introduction and Problem Statement

Capacity control is a very important problem in civil infrastructure operation. It manages the real-time capacity of a infrastructure facility such that the operation cost is minimized while the customer demands are satisfied. Examples include energy storage controls in a wind farm [KP11], pump and valve controls in a water distribution system [AS77], and traffic flow control [Hom05]. Although a similar capacity expansion problem is well studied [Lus82], no general mathematical model exists for the real-time capacity control problem. In this project, we model it as a stochastic control problem, and solve it with an iteration of sequential convex programming.

To begin with, we formulate the capacity control of a single facility. In particular, let $s_t$ denote the facility capacity at time $t$, and suppose $0 \leq s_t \leq S_{\text{max}}$, where $S_{\text{max}}$ is the maximum capacity. Let $u_t$ denote the control on the capacity of the facility during this period, and suppose $u_t$ satisfies $u_t \in [U_{\text{min}}, U_{\text{max}}]$, where $U_{\text{min}} \leq 0$, $U_{\text{max}} \geq 0$ are determined by the technical limitation of operations. Thus the system dynamics evolves according to

$$s_{t+1} = s_t + u_t, \quad t = 0, 1, \ldots, T,$$

where $T$ is the planning horizon. To achieve sustainable controls, we usually take $T \to \infty$. The demand at $t$ is denoted by $w_t$ and has a known probability distribution $W$ that is independent of $s_t$ or $u_t$. Also, we assume successive demands are independent and bounded. The stage cost at $t$ is defined as

$$g(s_t, u_t, w_t) = a(u_t)_+ + bs_t + p(s_t + u_t - w_t)_-,$$

where $(u_t)_- := \max(0, -u_t)$, $(u_t)_+ := \max(0, u_t)$, $a, b \in \mathbb{R}$ are the costs for capacity expansion and maintenance, respectively, and $p \in \mathbb{R}$ is the penalty for unsatisfied
demand. We assume $p > a + b$, since otherwise the problem is trivial. Now we can define the capacity management problem as:

$$\begin{align*}
\text{minimize} & \quad \lim_{T \to \infty} \mathbb{E} \left[ \sum_{t=0}^{T-1} \alpha^t g(s_t, u_t, w_t) \right] \\
\text{subject to} & \quad s_{t+1} = s_t + u_t, \quad t = 0, \ldots, T - 1 \\
& \quad 0 \leq s_t \leq S_{\text{max}}, \quad t = 0, \ldots, T - 1 \\
& \quad U_{\text{min}} \leq u_t \leq U_{\text{max}}, \quad t = 0, \ldots, T - 1
\end{align*}$$

over $u_t$, and any discount factor $0 < \alpha < 1$. Note when $\alpha \to 0$, this optimization finds the greedy controls. Otherwise, it finds a sequence of feasible controls such that the expectation of the total cost is minimized. This problem is convex but it cannot be directly solved because stochastic demand is involved and the time horizon is infinite. To overcome this difficulty, we estimate a policy function $\mu_t$ instead so that $u_t = \mu_t(s_t)$, at any $t$, and we can prove $\mu_t = \mu$ is stable. Furthermore, we show that the $(s, S)$ policy for inventory control [Sca60] can be adopted here, so we perform dynamic programming (DP) and at each iteration we use convex optimization to find the $(s, S)$ policy until convergence.

## 2 Dynamic Programming

If we define the cost-to-go function as $J_t = \min_{u_t, \ldots, u_{T-1}} \lim_{T \to \infty} \mathbb{E} \left[ \sum_{\tau=t}^{T-1} \alpha^{\tau-t} g(s_\tau, u_\tau, w_\tau) \right]$, for some $t \geq 0$, then $J_0$ is the optimal total cost. On the other hand, we can recursively compute $J_t(s)$, for all $s \in S$, $0 \leq t \leq T - 1$ by a DP operator $F$:

$$J_t(s) := F J_{t+1}(s) = \min_{u_t \in U(s)} \left[ \mathbb{E} g(s, u_t, w) + \alpha J_{t+1}(s + u_t) \right]$$

where $S = [0, S_{\text{max}}]$ is the state space, $U(s) = [\max(U_{\text{min}}, -s), \min(U_{\text{max}}, S_{\text{max}} - s)]$ is the control space at state $s$, $Y(s) = \{(s + U_{\text{min}})_+, \min(U_{\text{max}} + s, S_{\text{max}})\}$ is the domain of $G_t(y) = ay + \mathbb{E} p(y - w) + \alpha J_{t+1}(y)$, which is a convex function. The corresponding policy $\mu_t : S \to U(S)$ is

$$\mu_t(s) = y_t^*(s) - s$$

where $y_t^*(s) = \arg\min_{y \in Y(s)} G_t(y)$. This is slightly different with the step-wise $(s, S)$ policy because we already add the constraint of feasibility in the optimization of $G$. Since $F$ is a monotone and contractive operator, both $J_t(s)$ and $\mu_t(s)$ will converge to single fixed points $J^*(s)$ and $\mu^*(s)$ as $T \to \infty$, $\forall s \in S$, and these points are the optimal points (see Appendix for proofs). However, we can neither get an analytical expression of $\mu^*$ nor store it numerically since $\mu^*(s)$ depends on $s$ and $S$ is uncountable. Thus we estimate the optimal total cost $J^* : S \to \mathbb{R}$ offline (before the sequential observations and controls over time), then compute optimal controls $\mu^*(s_t)$ online.
3 Sequential Convex Programming

Theoretically, DP should work since both $G$ and $J$ are convex at each iteration. However, in practice we cannot easily get $J^+$ from $J$ since $S$ is a continuous space. Our idea is to discretize $S$ as a set of typical states $\tilde{S}$ and estimate $J^+$ for each of them, then perform piece-wise linear regression (PLR) to estimate $\tilde{J}^+$ over the original domain. It is worth pointing out that $J$ is a linear combination of $g$, which are piece-wise linear functions, so PLR can perfectly fit $J$ with the appropriate selection of knots.

There are several different ways to represent PLR. To make it acceptable in cvx, we express PLR as an inner product of a vector of basis functions $B(s)$ and a weight vector $\theta$. Here

$$B(s) = [1, s, (s - \xi_1)_+, \ldots, (s - \xi_{K-2})\Delta s)_+]$$

has the degree of freedom $K$, if there are $K - 2$ knots $\xi_1, \ldots, \xi_{K-2}$, and $\theta$ should be estimated from measured $J^+$ over $\tilde{S}$. However, $\tilde{J}^+(s; \theta) = B(s)^T \theta$ may not be convex since $\theta$ may have small negative elements due to regression errors. So we cannot directly perform convex optimization on $\tilde{G}$, which depends on $\tilde{J} = \tilde{J}^+$. To overcome this difficulty we rewrite $\tilde{J}^+$ as $\tilde{J}^+(s; \theta) = f(s; \theta) - g(s; \theta)$, where

$$f(s; \theta) = \sum_{i=1}^{K} B(s)_i (\theta)_i^+, \quad g(s; \theta) = \sum_{i=1}^{K} B(s)_i (\theta)_i^-$$

Then we can apply sequential convex programming (difference of convex programming) to find the optimal solution of $\tilde{G}$, if we start with $y^{(0)} = S_{\text{max}}/2$ and at each iteration $k$, we minimize

$$\tilde{G}(y; \theta) = ay + Ep(y - w)_+ + f(y, \theta) - g(y^{(k)}, \theta) - \nabla g(y^{(k)}, \theta)^T (y - y^{(k)})$$

Therefore, at each iteration inside DP, we should apply PLR and SCP instead of the original convex optimizations. The pseudo code of the whole procedure is shown in Algorithm 1.

4 Capacity Control in a Network

Now we consider a network of $N$ facilities, then $s_t := (s_{t1}, \ldots, s_{tN})$ becomes a $N$ dimensional vector so do $u_t$ and $w_t$. Thus the problem can be defined as

$$\text{minimize} \quad \lim_{T \to \infty} \sum_{t=0}^{T-1} \sum_{i=1}^{N} \alpha t g(s_{ti}, u_{ti}, w_{ti})$$

subject to

$$s_{t+1} = s_t + u_t, \quad t = 0, \ldots, T - 1$$

$$0 \leq s_t \leq S_{\text{max}}, \quad t = 0, \ldots, T - 1$$

$$U_{\text{min}} \leq u_t \leq U_{\text{max}}, \quad t = 0, \ldots, T - 1$$

$$s_t \in C_N, \quad t = 0, \ldots, T - 1,$$

where the last constraint denotes the possible network conditions. In the traffic flow control, $s_t$ could be the allowed amount of traffic entering the city, then we have

$$s_t \in C_N \iff Rs_t \leq c,$$
Algorithm 1 DP with \((s, S)\) policy.

**Offline Phase:**
Create discrete space \(\tilde{S} = 0 : \Delta_s : S_{\text{max}} \in \mathbb{R}^K\), with small increment \(\Delta_s\).
Create basis matrix \(B \in \mathbb{R}^{K \times K}\) for all \(s \in \tilde{S}\) with knots \(\Delta_s, \ldots, (K - 2)\Delta_s\).
Initialize \(\theta^+ = 0 \in \mathbb{R}^K\).
repeat
\(\theta \leftarrow \theta^+, \hat{J}^+ \leftarrow 0 \in \mathbb{R}^K\).
for each \(s \in \tilde{S}\) do
\(y^+ \leftarrow S_{\text{max}}/2\).
repeat
\(y \leftarrow y^+\).
\(y^+ \leftarrow \text{argmin}_{y' \in \mathcal{Y}(s)} \hat{G}(y'; \theta)\).
until \(\|y - y^+\|\) is smaller than the tolerance.
\(\hat{J}^+(s) \leftarrow \hat{G}(y) + (b - a)s\).
end for
\(\theta^+ \leftarrow (B^T B)^{-1} B^T \hat{J}^+\).
until \(\sup \|\theta - \theta^+\|\) is smaller than the tolerance.

**Online Phase:**
for each time period \(t\) do
Observe \(s_t \in S\) and compute \(\hat{J}(s_t; \theta) = B(s_t)^T \theta\).
Perform SCP on \(\hat{G}(y; \theta)\) over \(\mathcal{Y}(s_t)\) and obtain \(\mu(s_t)\).
end for

where \(R\) is the routing matrix, and \(c\) is the vector of intersection capacities. The problem can also be solved by Algorithm 1, but we replace all scalers with vectors and introduce the network constraint during the optimization of \(G\) at each iteration. However, when \(N\) is large, it is infeasible for us to find the global solution, so we should split the whole network into several blocks and apply ADMM.

## 5 Numerical Experiments

We estimate the performance of our algorithms with Matlab simulations. In particular, we set \(S_{\text{max}} = 1, -U_{\text{min}} = U_{\text{max}} = 0.1, a = 1, b = 0.8, p = 2, \alpha = 0.9\). Also, to simplify the problem, we set \(w \in \{0, 0.25, 0.5, 0.75, 1\}\) with equal probabilities, \(\tilde{S} = 0 : 0.05 : 1\).
We also simulate a simple network of 4 identical facilities with the same demands: \(w\), and the flow constraint \(R s_t \preceq c\), where \(R = [1, 0, 1, 0; 0, 1, 0, 1, 0, 1, 0, 1]\), \(c = [3; 3]\).

We first plot \(\mathcal{F}^k J\) vs. \(k\) (Figure 1) for some typical states to check the convergence speed of the total cost \(J\) in DP. We can see \(J\) converges within 40 iterations for all selected states in both the single facility and the facility network. Note the \(J\) in the network is estimated as the objective function given in Section 4.

In the following part, we focus on the total cost and policy of the single facility. The policy should be similar in the network since we just combine identical facilities.
Figure 1: Convergence of $J(s)$ with the initial function $J(s) = 0$. Left figure shows the single facility case, and right figure shows the 4-facility network case.

Figure 2: The total costs and corresponding policies at different iterations. Left figure plots $\hat{J}$ over $S$, and right figure plots $\mu$ over $\tilde{S}$.

with an additional linear network constraint. Figure 2 shows the regression curves $\hat{J}$ at typical iterations and the corresponding policies $\mu$. We can see the approximated total cost is non-convex, but this does not give influence on the optimum of our policy since the non-convexity appears at the right tail but our initial point is in the middle in SCP. Furthermore, the shape of $\hat{J}$ changes a lot during the first a few iterations. This demonstrates the advantage of our approach compared to traditional approximation on $J$: we do not give any (linear or quadratic) assumptions on $J$ so that it can converge closely to the true value in DP. Although 40 iterations are required for $J$ to converge, the shape of $\hat{J}$ does not change a lot after 10 iterations. The policy converges even faster, and the shape of $\mu$ does not change after just 4 iterations.

Finally, we compare our policy with a greedy policy ($\alpha = 0$) with the simulation. We choose $T = 50$, $s_0 = 0.5$ and sample 10 trajectories of demands from the given distribution, then we apply both policies at each time and the current state cost and the next state will be automatically generated. Then we compute the sample mean of discounted cumulative state costs over 10 trajectories for each policy and plot them in Figure 3. We find our policy works better than the greedy policy. Although the gap between the total costs is only 0.3, but it is significant, considering the scales of cost-related parameters are low.
Figure 3: Comparison to the greedy policy.

6 Conclusions

In this project, we propose a new model for real-time capacity control in civil infrastructure systems. A DP algorithm is developed to find the optimal controls in both a single facility and a infrastructure network. We also use numerical experiments to demonstrate the convergence and optimum of our algorithm. The main contributions of this project are:

- We propose a new mathematical model for real-time capacity control in civil infrastructure. No similar models exist in this domain.
- We introduce the supervised learning (regression) to overcome the difficulty of updating the cost function with continuous state space in DP, and implement SCP to find the solution if the regressed function is non-convex. Our approach is more accurate than approximate DP (in which the cost function is approximated as a certain type of functions) and more flexible and efficient than DP with discretized state space.
A Mathematical Proofs

Lemma 1. (Monotone Operator) For any two functions \( J : S \rightarrow \mathbb{R} \) and \( J' : S \rightarrow \mathbb{R} \), s.t. for all \( s \in S \), \( J(s) \leq J'(s) \), then we have

\[
(F^k J)(s) \leq (F^k J')(s), \forall s \in S, k = 1, 2, \ldots
\]

Proof. We can regard \( F^k J \) as the \( k \)-stage cost-to-go function in a finite horizon problem, with the terminal cost function \( \alpha^k J \). Obviously, as the terminal cost function increases uniformly so do the \( k \)-stage cost-to-go.

Lemma 2. (Contractive Operator) For any two bounded functions \( J : S \rightarrow \mathbb{R} \) and \( J' : S \rightarrow \mathbb{R} \), we have

\[
\|F J - F J'\|_{\infty} \leq \alpha \|J - J'\|_{\infty}.
\]

Proof.

\[
\|F J - F J'\|_{\infty} = \max_{s \in S} \left| \min_{u \in U(s)} [E g(s, u, w) + \alpha J(s + u)] - \min_{u \in U(s)} [E g(s, u, w) + \alpha J'(s + u)] \right|
\]

\[
\leq \max_{s \in S} \max_{u \in U(s)} \left| E g(s, u, w) - E g(s, u, w) + \alpha (J(s + u) - J'(s + u)) \right|
\]

\[
= \max_{y \in S} \alpha |J(y) - J'(y)| = \alpha \|J - J'\|_{\infty}.
\]

Proposition 1. (Convergence of DP) For any bounded function \( J : S \rightarrow \mathbb{R} \), we have

\[
J^*(s) = \lim_{k \rightarrow \infty} (F^k J)(s), \forall s \in S,
\]

where \( J^* \) is the unique solution of \( F J^* = J^* \).

Proof. First, we prove the sequence \( \{F^k J\} \) converges. It is sufficient to show it is a Cauchy sequence. From Lemma 2, we already know \( \|F^k J - F^{k+1} J\| \leq \alpha \|J - F J\| \), so we need to show for any \( \epsilon > 0 \), there exists some \( m > 0 \) such that \( \|F^l J - F^k J\| \leq \epsilon \) for all \( k > l \geq m \). This can be proved as

\[
\|F^l J - F^k J\| = \left| \sum_{i=l}^{k-1} (F^i J - F^{i+1} J) \right| \leq \sum_{i=l}^{k-1} \|F^i J - F^{i+1} J\| \leq \frac{\alpha^m}{1 - \alpha} \|J - F J\|.
\]

Thus for any \( \epsilon > 0 \), we can find \( m \) such that \( \alpha^m/(1 - \alpha) \|J - F J\| \leq \epsilon \). Second, we prove \( \{F^k J\} \) converges to a unique fixe point. Suppose \( J_1 = F J_1 \) and \( J_2 = F J_2 \) are both fixed points of \( F \), this implies

\[
\|F J_1 - F J_2\| = \|J_1 - J_2\|,
\]

which contradicts with Lemma 2, so it is unique and satisfies \( F J^* = J^* \).
Proposition 2. (Condition of Optimality) A stationary policy $\mu$ is optimal if and only if $FJ^* = F_\mu J^*$, where

$$F_\mu J(s) = E g(s, \mu(s), w) + \alpha J(s + \mu(s)), \forall s \in S.$$ 

Proof. If $\mu$ is optimal, let $J_\mu$ be the corresponding cost function. Since the policy is optimal, $J^* = J_\mu$. Also, the equation $J = F_\mu J$ has an unique solution $J_\mu$, so $J_\mu = F_\mu J_\mu \Rightarrow J^* = F_\mu J^* \Rightarrow FJ^* = F_\mu J^*$. 

If $FJ^* = F_\mu J$, we have $J^* = F_\mu J^*$. Since $J_\mu$ is the unique solution of $J = F_\mu J$, we have $J^* = J_\mu$, $\mu$ is the optimal policy.

References


[Sca60] HERBERT Scarf. The optimality of (s, s) policies in the dynamic inventory problem. mathematical methods in the social science, 1960.