Linear Programming Formulation of Markov Decision Process

Qianbo Yin

January 21, 2020

1 Introduction

In this report, I introduce how a Markov Decision Process can be formulated into a Linear Programming problem. With the natural property of duality in Linear Programming, we will see that Markov Decision Process can also be solved in the dual form.

2 Preliminary

2.1 Linear Programming Duality

Linear Programs have the following standard form:
standard form

$$\begin{aligned} \min \quad & c^T x \\ \text{s.t.} \quad & Ax = b \\ & x \geq 0 \end{aligned}$$

For every standard form linear program, the dual form associated with it is the following:

$$\begin{aligned} \max \quad & y^T b \\ \text{s.t.} \quad & y^T A \leq c^T \\ & y \geq 0 \end{aligned}$$

, where x is the primal variable and y is the dual variable

2.2 Lagrangian Duality

Even though we are primarily concerned with Linear Programming problem, the Lagrangian is applicable to any general optimization problem. To derive Lagrangian duality, we will introduce Lagrangian first. For any minimization problem (where $x \in R^n$):

$$\begin{aligned} \min \quad & f_0(x) \\ \text{s.t.} \quad & f_i(x) \leq 0, for\ i = 1, \ldots, m \end{aligned}$$

We can define the Lagrangian function as the following:

$$L(x, \lambda) = f_0(x) + \sum_{i=0}^{m} \lambda_i \cdot f_i(x)$$

Then, the original constrained optimization problem becomes the following unconstrained optimization problem:

$$\min_x \max_\lambda L(x, \lambda) = f_0(x) + \sum_{i=0}^{m} \lambda_i \cdot f_i(x)$$

, which is the primal formulation of Lagrangian optimization problem. Correspondingly, we have the dual formulation for Lagrangian optimization problem:

$$\max_\lambda \min_x L(x, \lambda) = f_0(x) + \sum_{i=0}^{m} \lambda_i \cdot f_i(x)$$

3 LP Formulation of MDP

3.1 Primal Form

The goal of Markov Decision Process is to find optimal policy and optimal value function. Specifically, if we have the optimal value function $v^*(s)$ for every $s \in S$, then optimal policy can be easily obtained by

$$\pi(a|s) = \arg\max_a \ q^*(s, a) = \arg\max_a \ r(s, a) + \gamma \sum_{s' \in S} p(s'|s, a) \cdot v^*(s')$$

Therefore, the Markov Decision Process is the problem of finding optimal value function for each state. We know that the optimality condition for value function is the Bellman optimality equation:

$$v^*(s) = \max_a r(s, a) + \gamma \sum_{s' \in S} p(s'|s, a) \cdot v^*(s')$$

Now we will try to transform the MDP problem into a Linear Programming problem. Since we want optimal value function, we can assume Bellman optimality equation hold. Moreover, we introduce more freedom to $v^*(s)$ by allowing it to be bigger than the maximum. Then we can change the Bellman optimality equation into inequality, by simply dropping the max:

$$v^*(s) \geq r(s, a) + \gamma \sum_{s' \in S} p(s'|s, a) \cdot v^*(s'), \text{ for all } a \in A$$

This expand one *max* equation into |A| inequalities. Notice that only in the case when equality hold in this equation, the optimality is achieved for the MDP. Therefore, any $v^*(s)$ should be minimized such that it satisfy Bellman optimality. Thus, we conclude that solving the MDP problem is equavelent to solving the following LP problem:

$$\begin{aligned} \min \quad & \sum_{s \in S} \alpha(s) \cdot v(s) \\ \text{subject to} \quad & v(s) - \gamma \sum_{s' \in S} p(s'|s, a) \cdot v(s') \geq r(s, a) \end{aligned}$$

for all $s \in S$ and $a \in A$.

And this is the Linear Programming primal formulation of MDP, which has $|S|$ primal variables (all the $v(s)$) and $|S| \cdot |A|$ constraints (one inequality for each (s, a) pair).

3.2 Dual Form

The dual form follows from the standard Linear Programming primal-dual pair. Given the primal problem (P), we have the dual problem:

$$
\begin{aligned}
\max \quad & \sum_{s\in S}\sum_{a\in A} r(s,a)\cdot x(s,a) \\
\text{subject to} \quad & \sum_{a\in A} x(s',a) - \sum_{s\in S}\sum_{a\in A} \gamma\cdot p(s'|s,a)\cdot x(s,a) = \alpha(s'),\ \text{for each } s\in S \\
& x(s,a)\geq 0
\end{aligned}
$$

4 Lagrangian of MDP-LP Formulation

4.1 Derive Lagrangian from Primal

In the Preliminary section (1.2), we have mentioned that for each constraint optimization problem, it is possible to turn it into a unconstrained optimization problem through Lagrange Multiplier. Here we shall transform the primal of MDP-LP problem using Lagrangian:
First we arrange the terms in primal form:

$$
\begin{aligned}
\min \quad & \sum_{s\in S}\alpha(s)\cdot v(s) \\
\text{subject to} \quad & -v(s) + \gamma\sum_{s'\in S} p(s'|s,a)\cdot v(s') + r(s,a)\leq 0,\ \text{for all } s\in S \text{ and } a\in A.
\end{aligned}
$$

Here the objective function $f_0(v(s)) = \sum_{s\in S}\alpha(s)\cdot v(s)$ and for each (s,a) pair, there is a constraint function $f_{(s,a)}(v(s))$ associated with it. Therefore, we can write down the Lagrangian:

$$
\begin{aligned}
L(v(s),x(s,a)) &= f_0(v(s)) + \sum_{s,a} x(s,a)\cdot f_{(s,a)}(v(s)) \\
&= \sum_{s\in S}\alpha(s)\cdot v(s) + \sum_{s\in S}\sum_{a\in A} x(s,a)\cdot(-v(s) + \gamma\sum_{s'\in S} p(s'|s,a)\cdot v(s') + r(s,a))
\end{aligned}
$$

Correspondingly, the original constrained optimization problem has been turned into a unconstrained optimization problem:

$$
\min_{v(s)}\max_{x(s,a)} L(v(s),x(s,a))
$$

This means we can also obtain the dual formulation easily by swapping the max and min:

$$
\max_{x(s,a)}\min_{v(s)} L(v(s),x(s,a))
$$

4.2 Derive Lagrangian from the Dual

Notice that it is also possible to obtain the dual Lagrangian directly from the dual of the linear programming problem. Here we directly consider the section 3.2 Dual Form with some arranging terms:

$$
\begin{aligned}
\max \quad & \sum_{s\in S}\sum_{a\in A} r(s,a)\cdot x(s,a) \\
\text{subject to} \quad & \sum_{a\in A} x(s',a) - \sum_{s\in S}\sum_{a\in A} \gamma\cdot p(s'|s,a)\cdot x(s,a) - \alpha(s') = 0,\ \text{for each } s\in S \\
& x(s,a)\geq 0
\end{aligned}
$$

Here, the dual objective function $f_0(x(s,a)) = \sum_{s\in S}\sum_{a\in A} r(s,a)\cdot x(s,a)$ and the constraint functions are $f_s(x(s,a))$ associated with each state. Therefore we can write out the Lagrangian:

$$\begin{aligned}
L(x(s,a), v(s) &= f_0(x(s,a)) + \sum_{s\in S} v(s)\cdot f_s(x(s,a)) \\
&= \sum_{s\in S}\sum_{a\in A} r(s,a)\cdot x(s,a) + \sum_{s\in S} v(s)\cdot(\sum_{a\in A} x(s',a) - \sum_{s\in S}\sum_{a\in A}\gamma\cdot p(s'|s,a)\cdot x(s,a) - \alpha(s')) \\
&= \sum_{s\in S}\alpha(s)\cdot v(s) + \sum_{s\in S}\sum_{a\in A} x(s,a)\cdot(-v(s) + \gamma\sum_{s'\in S} p(s'|s,a)\cdot v(s') + r(s,a))
\end{aligned}$$

This gives us the same Lagrangian as we derived before, which also provide the dual form of the Lagrangian unconstrained optimization problem:

$$\max_{x(s,a)}\min_{v(s)} L(v(s), x(s,a))$$

5 Solution Methods to Linear Programs

In this section, we will first examine a number of solution methods for linear programs. In this next and the following section, we will present the general framework of convex optimization problem and a number of general solution methods.

5.1 Simplex Method

Still consider the standard linear programming problem:

$$\begin{aligned}
\min \quad & c^T x \\
\text{s.t.} \quad & Ax = b \\
& x \geq 0
\end{aligned}$$

Let's denote the following notations first:
1. $P = \{x \in R^n | Ax = b, x \geq 0\}$ is the feasible set
2. $A \in R^{m\times n}$, the row vectors of A are linearly independent
3. A_i is i-th colomn vector of A and a_i^T is the i-th row vector of A

Then, let's define a basic feasible solution of a linear program:

Definition: Basic Feasible Solution
Consider a polyhedron feasible region P defined by linear equality and inequality constraints and let **x** be an element of R^n.
1)The vector **x** is a **basic solution** if All equality constraints are active, and out of the constraints that are active at **x**, there are n of them that are linearly independent.
2)If **x** is a basic solution and **x** ∈ P, then **x** is a **basic feasible solution**

Simplex Method (in general terms)
Step 0: Generate an initial basic feasible solution $x^{(0)}$. Let k = 0 and go to Step 1.
Step 1: Check optimality of $x^{(k)}$, If $x^{(0)}$ is optimal, STOP with optimal solution, else go to Step 2.
Step 2: Check whether LP is unbounded; if so STOP with no optimal solution, else go to Step 3.
Step 3: Generate another basic feasible solution $x^{(k+1)}$ such that $c^T x^{(k+1)} \leq c^T x^{(k)}$ Let k = k+1 and go to Step 1.

There are a few things in the Simplex algorithm to be treated rigorously. First, we consider the Step 3, moving to another basic feasible solution for improvement:
To obtain a new basic feasible solution, we consider the ones that are adjacent to the current basic feasible solution. Geometrically, this means the following:

Definition: Feasible Direction Let **x** be an element of a polyhedron P. A vector $d \in R^n$ is a **feasible direction** if there exist a $\theta > 0$ such that $\mathbf{x} + \theta d \in P$

Algebraically, it means the following:
Definition: Adjacent Basic Feasible Solutiony is a basic feasible solution. **y** is **adjacent** to **x** if they have exactly m - 1 basic variables in common.

This tells us what are the possible $x^{(k+1)}$ to move to in the Step 3. However, we also need to consider the condition of making improvement (i.e. $c^T x^{(k+1)} \leq c^T x^{(k)}$) while choosing the adjacent basic feasible solutions.

Next, we consider the optimality condition in Step 2:
We first write the A matrix into 2 parts:

$$A = [B\ N],$$

where B is m $\times$ m and form a basis of R^m, N is the remaining collomn vectors. Correspondingly, we write $\mathbf{x} = \begin{pmatrix} x_B \\ x_N \end{pmatrix}$, where x_B are the basic variables associated with B, and x_N are the non-basic variable. In the same way, we also partition $c = \begin{pmatrix} c_B \\ c_N \end{pmatrix}$.

Definition: reduced costs vector The reduced costs vector associated with $\mathbf{x} = \begin{pmatrix} x_B \\ x_N \end{pmatrix}$ is defined as

$$r_N = c_N^T - c_B^T B^{-1} N$$

Therefore we obtain our optimality condition:

1 ***Optimality Condition:***
If the reduced cost vector r_N corresponding to a basic feasible solution $\boldsymbol{x} = \begin{pmatrix} x_B \\ x_N \end{pmatrix}$ is non-negative, then $\boldsymbol{x}$ is an optimal solution.

With the new basic feasible solution to move into and the optimality condition, we are ready to specify Simplex Method in details:

Simplex Method (with details)
Step 0: Generate an initial basic feasible solution $x^{(0)}$. Let B be the basis matrix and N the non-basis matrix. Denote the set of basis indices as $\bar{B}$ and the set of non-basis indices as $\bar{N}$. Let k = 0 and go to Step 1.

—

Step 1: Check optimality of $x^{(k)}$. Compute $r_q = c_q - c_B^T B^{-1} N_q$ for all $q \in \bar{N}$. If $r_q > 0$ for all $q \in \bar{N}$, then **x** is optimal, STOP with optimal solution, else go to Step 2.

—

Step 2: Construct descent direction $d^q = \begin{pmatrix} -B^{-1}N_q \\ e_q \end{pmatrix}$. If $d^q \geq 0$, then linear program is unbounded, STOP with no optimal solution, else go to Step 3.

—

Step 3: Generate improved basic feasible solution. Find the biggest α such that $x^{(k+1)} = x^{(k)} + \alpha d^q$ is still feasible solution. Let $x^{(k+1)} = x^{(k)} + \alpha d^q$. Step 4: Update basis as following:

$$\bar{B} = \bar{B} - \{j^*\}\{q\}$$

, where j^* is leaving and q is entering.
Let k = k+1 and go to Step 1.

6 Theoretical Treatment for constraint/unconstrained optimization problem

6.1 Linear Programming Duality Theorems

Consider the following primal-dual pair:

$$\begin{aligned} \min \quad & c^T x \\ \text{s.t.} \quad & a_j^T x \geq b_j, j = 1, ..., r \end{aligned}$$

$$\begin{aligned} \max \quad & b^T u \\ \text{s.t.} \quad & \sum_{j=1}^{r} a_j u_j = c \\ & u \geq 0 \end{aligned}$$

, where x is primal variable and u is the dual variable.
Let's demote the optimal value of primal and dual as f^* and q^* respectively. First, we shall prove weak duality theorem, i.e. $q^* \leq f^*$:

$$b^T u = \sum_{j=1}^{r} b_j u_j + (c - \sum_{j=1}^{r} a_j u_j)^T x = c^T x + \sum_{j=1}^{r} u_j (b_j - a_j^T x \leq c^T x$$

Therefore, we have $q^* \leq f^*$.
Then, we shall consider the strong duality theorem:
Strong Duality Theorem
(1) If either f^* or q^* is finite, then $f^* = q^*$ and both the primal and dual have optimal solutions.
(2) If $f^* = -\infty$, then $q^* = -\infty$.
(3) If $q^* = \infty$, then $f^* = \infty$.

Using duality theorem (1), we obtain an optimality condition:
A pair of vectors (x^*, u^*) form a primal and dual optimal solution pair if and only if x^* is primal feasible, u^* is dual feasible and

$$u_j^* (b_j - a_j^T x^*) = 0, \forall j = 1, ..., r$$

6.2 Convex Duality Theorems

Consider the following convex optimization problem:

$$\begin{aligned} \min \quad & f(x) \\ \text{s.t.} \quad & x \in X, g(x) \leq 0 \end{aligned}$$

and the Lagrangian $L(x, u) = f(x) + u^T g(x), where\ x \in X\ and\ u \in R^r$.
The dual function q(u) is given by $q(u) = \begin{cases} inf_{x \in X} L(x, u), & \text{if } u \geq 0 \\ -\infty, & \text{otherwise} \end{cases}$
The dual problem is:

$$\begin{aligned} \max \quad & q(x) \\ \text{s.t.} \quad & u \in R^r \end{aligned}$$

Similarly to linear programming, we have strong duality theorem and optimality condition:
Convex Optimization Duality Theorem:
Assume that f^* is finite, and that there exist $\bar{x} \in X$ such that $g_j(\bar{x}) \leq 0$ for all j = 1,...,r
Then $q^* = f^*$ and the set of optimal solutions of the dual problem is nonempty.

Convex Optimization Optimality Condition:
There holds $q^* = f^*$, and (x^*, u^*) are a primal and dual optimal solution pair if and only if x^* is feasible, $u^* \geq 0$, and

$$x^* \in arg\min_{x \in X} L(x, u^*), \quad u_j^* g_j(x^*) = 0, \ for\ j = 1, ..., r$$

6.3 A Lagrangian Sensitivity Theorem

Notation: we use $S(x; \delta)$ to denote δ ball centered around x.
Proposition: There exist a scalar $\delta > 0$ and continuously differentiable function x: $S(0; \delta) \rightarrow R^n and \lambda : S(0; \delta) \rightarrow R^m$ such that $x(0) = x^*, \lambda(0) = \lambda^*$, and for all $u \in S(0; \delta)$, $x(u), \lambda(u)$ are a local minimum Lagrange multiplier pair for the problem

$$\begin{aligned} \min \quad & f(x) \\ \text{s.t.} \quad & h(x) = u \end{aligned}$$

Furthermore,

$$\nabla_u f[x(u)] = -\lambda(u), \forall u \in S(0; \delta)$$

7 General Solution Methods

In this section, we shall consider 3 different types of solution methods of optimization problem. The first type is iterative descent method, which iteratively improve a feasible solution within the feasible set. An example of this include Simplex Method. The second type of solution transform optimization problem into a set of equations that must be satisfied upon optimality. Examples of this type includes Lagrange Multiplier. The thrid type of solution methods eliminate the constraints by adding penalty term into the objective function. In this report we have not seen this type of solution method.

7.1 Iterative Descent Methods

This approach first initialize a feasible solution, x_0. At each iteration, we have a feasible solution x_k and a direction d_k such that
1) the direction itself is feasible, which means $\exists \alpha \ s.t. \ x_k + \alpha d_k \in P$
2) the direction improves the feasible solution, i.e. $\nabla f(x_k)' d_k < 0$ This can yield a new feasible solution $x_{k+1} = x_k + \alpha d_k$, which satisfy $f(x_{k+1}) < f(x_k)$

Many successful iterative solution methods fall into this category, including the Simplex Method.

7.2 Closed-form Solution

The second approach is based on the possibility of solving the system of equations and (possibly) inequalities which constitute necessary conditions for optimality for the optimization problem. In Lagrange Multiplier, these conditions are:

$$\nabla_x L(x, \lambda) = \nabla f(x) + \nabla h(x)\lambda = 0$$
$$\nabla_\lambda L(x, \lambda) = h(x) = 0$$

, where L is the Lagrangian function

$$L(x, \lambda) = f(x) + \lambda^T h(x)$$

7.3 Penalty Solution Methods

This approach eliminate the constraints through the use of penalty functions. For example, it is possible to use quadratic penalty to transform constraint optimization problem into

$$\begin{aligned} \min \quad & f(x) + \frac{1}{2} c_k |h(x)|^2 \\ \text{s.t.} \quad & x \in R^n \end{aligned}$$

Thus we also transform a constraint optimization problem into a unconstrained optimization problem. In this report we are primarily concerned with the first 2 types of solution methods.

8 Conclusion

To conclude, we have covered the followings in this report:
1) Formulating a Markov Decision Process into a Linear Programming problem
2) Theoretical treatment of LP problem and Convex Optimization problem
3) Three classes of solution methods to optimization problem and some particular solution methods to LP problem.

9 References

Markov Decision Process: Discrete Dynamic Processes (Puterman, 1994)

Introduction to Linear Optimization (Bertsimas and Tsitsiklis, 1997)

Convex Optimization Theory (Bertsekas, 2009)

Constrained Optimization and Lagrange Multiplier Methods (Bertsekas, 1996)