Homework 7

Devashish C. Thakar

20 Nov 2019

In part 1, I describe the problem, simple optimization formulation, and the necessary conditions underlying the problem. In part 2, I describe the dynamic progrmming algorithm and a numerical example. In part 3, I formulate the actual dynamic program (Bellman Equation)

Part I: Description of the problem and optimization formulation

Lot-sizing is one of the most common problem faced by Operations Researchers in different settings. Common examples include managers in various service industries trying to understand stiaffing requirements, retail managers trying to decide between whether to produce or store inventory given a demand forecast, production managers trying to balance the costs of setting up production every time period versus producing in some time periods and storing for rest, etc. The overarching objective for lot-sizing can be summarized as follows:

- *Determine the optimal order quantity* ***lot size*** *in each period so that the demand in each period is met while the sum of ordering,purchasing, and inventory holding costs are minimized*

In the present instance, I consider the case of a **production manager** trying to balance the tradeoff between producing in a given time period (and incurring a fixed cost of production each time she does so), and carrying inventory between time periods (and incurring a per unit holding cost). **The production manager is thus the agent under consideration**.

I start with a simple optimization formulation of the problem at hand, and then build up to a dynamic programming formulation. With the following notations and assumptions in mind,

- D_t: demand in perio $t, t = 1, 2, 3, ..., T$
- c: unit production cost
- K: fixed production cost associated with producing in period t
- Q_t: lot-size in period t; a decision variable
- Y_t: indicator variable if production hapens in period t
- I_t: inventory left at the end of period t
- order cost in a period: $K + cy_t$ if y_t is ordered, 0 otherwise

- h: per unit holding cost of holding inventory for one time period
- ordering and demand occur at the start of period, holding cost charged at the end of a period

The formulation of the problem is therefore: Determine y_t, $\forall t$:

$$\min \sum_{t=1}^{T} KY_t + cQ_t + hI_t \tag{1a}$$

$$I_t = I_{t-1} + Q_t - D_t \ \forall t \tag{1b}$$

$$Q_t \leq Y_t(\sum_{t'=t}^{T} D_{t'}) \ \forall t \tag{1c}$$

$$I_0 = 0 \tag{1d}$$

$$Y_t \in \{0, 1\} \tag{1e}$$

$$I_t \geq 0 \ \forall t \tag{1f}$$

$$Q_t \geq 0 \ \forall t \tag{1g}$$

The most famous structural property for solving the general lot-sizing problem was introduced by Wagner and Whiten (1958). For parsimonious reresentation, I skip the proof of the necessary condition. But essentially, the Wagner-Whiten theorem implies that for production over $t = 1, 2, ..., T$ the optimal producion policy follows $y_t I_{t-1} = 0$. This essentially translates to saying that *under an optimal lot-sizing policy, inventory carried to period t from* $t-1$ *will be zero or the production quantity in period t will be zero*- **one cannot carry inventory into a period and produce at the same time**. Therefore, this also constitutes the **productive technology** of the problem.

In terms of the **information** available with the agent (production manager), she knows that the deterministic demand over the entire time horizon. This is therefore a deterministic lot-sizing formulation. There are no **enforecement technologies** and **matching mechanisms** for this problem.For the present lot-sizing problem of fulfiling demand in each time period, the decision is whether to produce in a current time period or not, and if so, for how many future time periods to produce. Thus, **produce or not to produce** is the control variable (vector of decisions across each time periods). The state variable is the **demand in each time period**.

Part II: Numerical Example to motivate the DP

Using the Wagner-Whiten property, the algorithm boils down to the following:

- Decision is whether or not to order in a given period. If we order, then the order quantity should be just enough to cover demand until next period.
- Solve a series of sub-problems(one-period problem, two-period problem, ...N period problem), and use the solution of each sub-problem to solve the next subproblem.

Consider a simple four-period problem of c=7, K=100, h=1, and demand at $t_0 = 70, t_1 = 90, t_2 = 140$ and $t_3 = 150$. Let F(t) be the cost of the cheaperst strategy that satisfies the demand in period 0,1,2...T. The best strategy is represented by the asterisk after the calculation of all possible costs in each time periods.

$$F(0) = \begin{cases} 100 + 7 * 70 = 590(\star) \end{cases}$$

F(1)=$min \begin{cases} 590 + 100 + 7 * 90 = 1320 \\ 100 + 7 * 160 + 1 * 90 = 1310(\star) \end{cases}$

F(2)=$min \begin{cases} \text{start at 2, best way to 1} \\ \text{start at 1, best way to 0} \\ \text{start at 0} \end{cases}$

F(2)=$min \begin{cases} 1310 + 100 + 7 * 140 = 2390(\star) \\ 590 + 100 + 7 * 230 + 1 * 140 = 2440 \\ 100 + 7 * 300 + 1 * 90 + 2 * 140 = 2570 \end{cases}$

F(3)=$min \begin{cases} \text{start at 3, best way to 2} \\ \text{start at 2, best way to 1} \\ \text{start at 1, best way to 0} \\ \text{start at 0} \end{cases}$

F(3)= $min \begin{cases} 2390 + 100 + 7 * 150 = 3540(\star) \\ 1310 + 100 + 7 * 290 + 1 * 150 = 3590 \\ 590 + 100 + 7 * 380 + 1 * 140 + 2 * 150 = 3790 \\ 100 + 7 * 450 + 1 * 90 + 2 * 140 + 3 * 150 = 4070 \end{cases}$

Therefore, the optimal ordering policy is:

t	0	1	2	3	4
# of periods supplied	2	0	1	1	1

Part III: Dynamic Programming Formulation

The numerical example above can also be thought of as a network, where each node t represetns a period and an arc from node $t^{'}$ to $t$represents the fact that we produce in both periods $t^{'}$ and t but not in periods in between

The cost $c_{t',t}$of reaching node t from $t^{'}$, is the cost of ordering in $t^{'}$ but not in $t^{'} + 1, t^{'} + 2, t^{'} + 3, ..., t - 1$. Going to the above example, at t=2, there were three possible options of reaching from t=0. The least cost path was to start at 3, best way to 2. This is what was done for all time periods- finding out the least costly path to reach a given time period. The cost of reaching a node can generally be represented as:
$c_{t',t}$= K + $c * (D_{t'} + ... + D_{t-1})$ + $h * (D_{t'+1} + ... + D_{t-1})$+...... $+h * (D_{t-1})$
The pseudo-code for the formulation is therefore:

- Step 1: t=0, z_t^*=0
- Step 2: t=t+1. If t >T+1, stop. Otherwise go to step 3.
- Step 3: For all $t^{'}$=1,2,....,t-1. compute:
  $c_{t',t} = K + c*(D_{t'} + ... + D_{t-1}) + h*(D_{t'+1} + ... + D_{t-1}) + ... + h*(D_{t-1})$
- Step 4: Compute: $z_t^* = \min_{t'=1,2,...,t-1}\{z_{t'}^* + c_{t',t}\}$
- Step 5: Compute: $p_t^* = \arg min_{t'=1,2,...,t-1}\{z_{t'}^* + c_{t',t}\}$
- Step 6: Go to step 2

The optimal cost is given by z_{T+1}^* and the optimal set of periods to produce can be obtained by backtracking from p_{T+1}^*. The final recursice solution is thus as follows:

$$F(t)=min\begin{cases}K_t + cD_t + F(t-1)\\ \min_{0\leq j\leq t}[K_j + c\sum_{k=j}^{t} D_k + \sum_{k=j}^{t-1}\sum_{l=j+1}^{t} h_j d_l]\end{cases}$$

$$F(0) = K_1 + cD_1$$

I also plan to characterize the total optimal cost of production as a function of various other state parameters such as per unit holding costs, fixed costs of production etc. I also plan to look at the total optimal cost considering some initial inventory and assuming some kind of an increasing cost function for every unit of inventory that is held from one time period to the next. This is more suitable for producing products that could be perishable in nature and holding them for longer periods of time could lead to not just holding costs, but lower value of sale due to reduced quality over time. The lower value of sales could be translated to higher costs of holding since I am not modeling the sales in this algorithm.