Optimal Ordering Policies for Two-Product Inventory Problems

Dynamic Programming and Reinforcement Learning (DPRL) – Assignment 2

November 24, 2024

Group 57

Sara Abesová (2735679)
Vrije Universiteit Amsterdam
Amsterdam, Netherlands
s.abesova@student.vu.nl

Matúš Halák (2724858)
Vrije Universiteit Amsterdam
Amsterdam, Netherlands
m.halak@student.vu.nl

1 INTRODUCTION

A fundamental problem tackled within the field of inventory management and supply chain is finding the balance between holding costs and order costs. These are additional to production costs (e.g., warehouse and shipping fees). These costs respond to order size in opposite ways: larger, less frequent orders reduce order costs but increase holding costs. This creates a trade-off that needs to be optimized when looking to minimize overall costs.

This report addresses this order policy optimization problem, in the context of an infinite time horizon and two products with limited inventory capacity for each product. The objective is to make a program that can find a policy, which specifies an optimally-sized order that minimizes long-run costs for each inventory level of each item.

To achieve this goal, we define the problem as a Markov decision process (MDP). First, we tackle a simplified version of the problem with a fixed heuristic policy and use several methods to obtain the long-run average costs. Finally, we solve the optimal policy problem by value iteration on the Bellman equation and obtain the optimal long-run average costs using the optimal policy while adhering to constraints on storage capacity and avoiding lost sales.

2 METHODS

2.1 Part A: State Space and Action Space

2.1.1 *State Space.*

We define the state space $\mathcal{X}$ as the set of all possible combinations of inventory levels of the two products at any given time. Each state (x_1, x_2) represents the inventory levels of product 1 and product 2, respectively. Given limited storage capacities $c_1 = 20, c_2 = 20$ for products 1 and 2, the state space is defined as:

$$x \in \mathcal{X} = \{(x_1, x_2) \mid 1 \leq x_1 \leq c_1,\ 1 \leq x_2 \leq c_2\}$$

The constraints $1 \leq x_1 \leq c_1$ and $1 \leq x_2 \leq c_1$ ensure that inventory levels are never zero at the beginning of a day (to prevent lost sales) and do not exceed the storage capacity for each product. With 20 possible inventory levels for each of the products, the state space contains $c_1 \times c_2 = 20 \times 20 = 400$ possible states: $|\mathcal{X}| = 400$.

2.1.2 *Action Space.*

We define the action space $\mathcal{A}$ as the set of all possible order sizes for the two products. At each state (x_1, x_2), an action (a_1, a_2) specifies the quantities ordered for product 1 and product 2, respectively.

$$a \in \mathcal{A}_x = \{(a_1, a_2) \mid 0 \leq a_1 \leq c_1 - x_1,\ 0 \leq a_2 \leq c_2 - x_2\}$$

The action space is state-dependent and changes dynamically based on the current inventory levels of the two products. The $a_i \leq c_i - x_i$ constraint ensures that actions that would bring product's inventory over the capacity for that product are not taken.

2.2 Fixed Order Policy

The fixed order policy $\alpha_{\text{fix}} \in \mathbb{R}^{|\mathcal{X}|}$ is to make no order unless either item's inventory reaches 1 at the beginning of a timestep, and in that case, to order up to level 5 for both items.

$$\alpha_{\text{fix}}(x) = ((5 - x_1) \cdot \mathbb{I}(x_1 \leq 5), (5 - x_2) \cdot \mathbb{I}(x_2 \leq 5) \mid x_1 = 1 \vee x_2 = 1)$$

2.2.1 *Part B: State Transitions under fixed policy.*

To capture all possible state transitions and their corresponding probabilities under the fixed policy, we constructed a transition probability matrix $\mathcal{P} \in \mathbb{R}^{|\mathcal{X}| \times |\mathcal{X}|}$. This matrix represents how the inventory system transitions from one state x, representing the current inventory levels of the two products, to a new state x', after accounting for orders according to fixed policy and stochastic demand on any given day. When both inventory levels are higher than 1, the next state is determined solely by stochastic demand. The demands for each product (d_1, d_2) are independent events, each with a 50% chance of experiencing demand of 1 and a 50% chance no demand at any given time-step. As a result, the demand d on any given day can be:

$$d \in \mathcal{D} = \{(d_1', d_2'), (d_1', d_2), (d_1, d_2'), (d_1, d_2) \mid d_1, d_2 = 1 \wedge d_1', d_2' = 0\}$$

Because demands for each product are independent and both 50% likely, each combination occurs with probability 0.25:

$$P(d) = P(d_1 \cap d_2) = P(d_1' \cap d_2') = 0.5 \cdot 0.5 = 0.25$$

While the demand probabilities are the same for every state, the resulting state transitions depend on the current inventory levels x and the fixed order policy. These probabilities are then directly used to populate the matrix $\mathcal{P}$ (which represents all possible state transitions under the fixed policy).

$$\mathcal{P}_{x,x'} = \begin{cases} P(d) & \text{if } x' \in (x - d) \quad \forall d \in \mathcal{D} \\ 0 & \text{otherwise} \end{cases}$$

2.2.2 *Part C: Simulation under fixed policy.*

To estimate the long-run average-costs under the fixed policy, we simulated the system for a long period of $T = 10^5$ time steps. Since we look at long-run behavior, initial state does not matter and thus we initialized the inventory levels x randomly within the range $\{1, \ldots, c_i\}$ for each of the products. At each time interval, the holding costs h were calculated as: $h_x = h_1 \cdot x_1 + h_2 \cdot x_2 \mid h_1 = 1, h_2 = 2$, where h_1 and h_2 are the per-unit holding costs. Orders were made according to the fixed policy α_{fix} at start of each time-step, with fixed order cost $o = 5$. Therefore, at each time-step, Total Cost increased by $h_x + o \cdot \mathbb{I}(1 \in x)$. Stochastic demand was then applied independently to each product by sampling from $U(0, 1)$ and if sampled value was lower than $P(d)$ one item was subtracted from that product's inventory. Finally, the long-run average cost was the total cumulative cost divided by the number of time steps:

$$\text{Long-run Average Cost}(\Phi_*) = \frac{\text{Total Cost}}{T}$$

2.2.3 *Part D: Limiting Distribution (π_∞) under fixed policy.*

Next, we computed the limiting distribution π_∞, which represents the long-term probabilities of the system being in each possible inventory state (x_1, x_2) and thus $\pi_\infty \in \mathbb{R}^{|\mathcal{X}|}$. We obtained π_∞ by iterating under α_{fix} for $T = 10^5$ time-steps, and calculating the proportion of time-steps in which each possible state (x_1, x_2) was visited.[1] The

[1] We also obtained the exact stationary distribution π_* analytically. For this approach, we noted that $\pi_*^T \mathcal{P} = \pi_*^T$, which means π_* is a left eigenvector of $\mathcal{P}$, which can also be expressed as $\mathcal{P}^T \pi_* = \lambda \pi_*$ with $\lambda = 1$. This expression has the traditional eigenvalue problem form, and π_* can easily be found by identifying the eigenvector v of the transposed transition probability matrix $\mathcal{P}^T$ corresponding to λ_1 using `np.eig` and normalizing v_{λ_1} to ensure it sums to 1 to represent probabilities $\pi_* = \frac{v_{\lambda_1}}{\sum v_{\lambda_1}}$.

limiting distribution π_∞ was then used to compute the long-run average cost by taking the inner product with $C \in \mathbb{R}^{|\mathcal{X}|}$:

$$C_x = h_x + o \cdot \mathbb{I}(1 \in x) \quad \forall x \in \mathcal{X}$$

$$\text{Long-run average cost}(\Phi_*) = \langle \pi_\infty, C \rangle = \sum_{x \in \mathcal{X}} \pi_{\infty_x} C_x,$$

where C_x represents the total immediate cost (holding and ordering) associated with state x under α_{fix}.

2.2.4 Part E: Poisson Equation value iteration under fixed policy. As a final method to determine the long-run average cost under the fixed policy, we defined and solved the Poisson equation. We defined the Poisson equation as:

$$V_*(x) + \Phi_* = C_x + \sum_{x' \in \mathcal{X}} \mathcal{P}_{x,x'} V_*(x')$$

where $V_*(x)$ represents the expected difference in cumulative cost between starting at inventory levels x relative to the long-run average cost Φ_*, C_x represents the immediate costs of inventory levels x and the $\sum_{x'} \mathcal{P}_{x,x'} V_*(x')$ term represents the expected future costs of transitioning to other inventory levels x'. We recursively solved the Poisson equation using value iteration until convergence criterion $\epsilon = 10^{-8}$ and $\delta = \max(V_{t+1} - V_t) - \min(V_{t+1} - V_t)$:

$$V_{t+1} = C_x + \mathcal{P} V_t, \text{while } \delta > \epsilon$$

Upon convergence ($\delta \leq \epsilon$), the long-run average cost was computed as the average cost difference between two-successive time-steps:

$$\text{Long-run average cost}(\Phi_*) = \frac{\sum V_{t+1} - V_t}{|\mathcal{X}|}$$

2.3 Optimal Policy

2.3.1 Part F: Bellman Equation and Value Iteration. To determine the optimal policy α_*, we considered every possible action for each state, we chose to do this by constructing a transition probability tensor with a transition probability matrix for every possible action $\mathcal{P} \in \mathbb{R}^{|\mathcal{X}| \times |\mathcal{X}| \times |\mathcal{A}_*|}$ where we considered $\mathcal{A}_* = \{(a_1, a_2) \mid 0 \leq a_i \leq c_i - 1 \text{ for } i = 1, 2\}$. For each x, a combination, we consider only valid states $x' \in S_{x,a}$ that can be reached when taking action a and taking into account stochastic demand. When no valid state can be reached by taking action a in state x, only valid states possibly reached by stochastic demand are considered:

$$S_{x,a} = \left\{ x' \in \mathcal{X} \mid \exists d \in \mathcal{D} : x' = x - d + a \cdot \mathbb{I}\{x - d + a \in \mathcal{X}\} \right\}$$

$$\mathcal{P}_{x,x',a} = \begin{cases} 0.25 & \text{if } x' \in S_{x,a} \land |S_{x,a}| = 4, \\ 0.5 & \text{if } x' \in S_{x,a} \land |S_{x,a}| = 2, \\ 1 & \text{if } x' \in S_{x,a} \land |S_{x,a}| = 1, \\ 0 & \text{otherwise.} \end{cases}$$

Using this we defined the Bellman equation as:

$$V + \Phi_* e = \min_{a \in \mathcal{A}} \left\{ C_a + \mathcal{P}_{:,:,a} V \right\}$$

where V represents the expected cumulative cost of starting at a given state and following the optimal policy, Φ_* is the long-run average cost, $e = 1_{|\mathcal{X}|}$ and C_a represents the immediate costs taking a given action for all states. Since $o = 5$ regardless of order size, $C_a(x) = h_x + o \cdot \mathbb{I}(a \neq (0,0)) \forall x \in \mathcal{X}, \forall a \in \mathcal{A}$. $\mathcal{P}_{:,:,a}$ represents the probability transition matrix for a given action. The minimum over the actions is taken with respect to each state x.

We solved the Bellman equation to obtain the optimal policy α_* and the long-run average cost Φ_* under α_* using value iteration with the same δ and ϵ convergence criteria as in section 2.2.4, also defining Φ_* as the average cost difference between two successive time-steps as in section 2.2.4:

$$V_{t+1} = \min_{a \in \mathcal{A}} \left\{ C_a + \mathcal{P}_{:,:,a} V_t \right\}, \text{while } \delta > \epsilon$$

$$\alpha_*(x) = \arg\min_{a \in \mathcal{A}} \left\{ C_a + \mathcal{P}_{:,:,a} V_t \right\}, \text{once } \delta \leq \epsilon$$

3 RESULTS & DISCUSSION

3.0.1 Fixed Policy α_{fix}. Table 1 shows the Φ_* estimates obtained using all described methods. As expected, the Simulation, Limiting distribution, and Poisson value iteration methods converged to the same Φ_* estimate of around 10.445. This is because in the long-run α_{fix} effectively reduces the state space to $\mathcal{X}_{\text{fix}} = \{(x_1, x_2) \mid 1 \leq x \leq 5, \forall x\}$ which is a communicating and aperiodic Markov chain under α_{fix} see Figure 1, with a unique stationary distribution π_*. Because all 3 methods operate within $\mathcal{X}_{\text{fix}}$, they share the same $\mathcal{P}$, C, and π_* and must therefore converge to the same Φ_*. We confirmed this by obtaining results with $c_1, c_2 = 5$ equal to those inÏ Table 1.

3.0.2 Optimal Policy α_*. As can be seen in Table 1, the optimal policy α_* resulting from Bellman value iteration is an improvement upon α_{fix}, since the resulting Φ_* is lower (7.988) using α_*[2]. This makes sense when we investigate α_* in more detail in Figure 2. We see, that α_* uses a similar strategy to α_{fix}, of ordering only once either item's inventory reaches 1 at the beginning of the time-step. This is optimal because demand for each product on any given day can be at most 1 and thus waiting until inventory reaches level 1 minimizes order costs, while preventing lost sales as demand can still be met at inventory 1. The difference between α_{fix} and α_* is that α_* does not order up the item with sufficient inventory to minimize holding costs. Also, α_* minimizes holding costs by only ordering up to level 3 for both items instead of 5 and thus effectively restricts the state space to $\mathcal{X}_* = \{(x_1, x_2) \mid 1 \leq x \leq 3, \forall x\}$, which again, is a communicating, aperiodic Markov Chain under α_* see Figure 3. We confirmed this intuition by running the script with $c_1, c_2 = 3$ and c,d,e parts with filling up to 3, and obtained the same $\Phi_* = 7.988$ for all parts of the assignment.

4 CONCLUSION

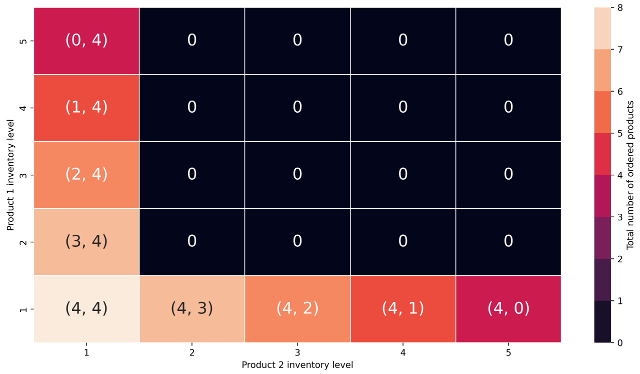
In this report we tackled the holding-order cost trade-off for a two-product inventory management problem, using MDP and infinite-horizon Dynamic Programming techniques. The results confirmed that Simulation, Limiting Distribution, and Poisson value iteration converged to the same estimate of Φ_* under the fixed policy. Using Bellman value iteration we identified and characterized the optimal policy leading to a lower Φ_*. Our results highlight the importance and utility of modeling real-world supply-chain problems as MDPs. The holding-order cost trade-off can nicely be investigated and visualized using our script with different holding and order costs.

[2] Our script solves the optimal order policy problem for any 2 items with parameters specified with command-line arguments

APPENDIX

Method	Φ_* estimate
c. Simulation	10.448
d. $\langle \pi_{\infty \text{iteration}}, C \rangle$	10.443[3]
e. Poisson Value iteration	10.445
f. Bellman Value iteration α_*	7.988

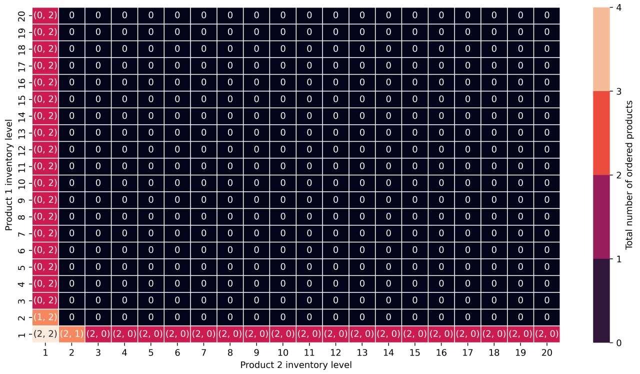
Table 1: Long-run average cost (Φ_*) estimates using different methods (Φ_* under α_{fix}: c-e, Φ_* under α_*: f) for two products with capacities $c_1 = c_2 = 20$



	1	2	3	4	5
5	(0, 4)	0	0	0	0
4	(1, 4)	0	0	0	0
3	(2, 4)	0	0	0	0
2	(3, 4)	0	0	0	0
1	(4, 4)	(4, 3)	(4, 2)	(4, 1)	(4, 0)

Figure 1: Fixed Policy α_{fix} in effectively reduced state space $\mathcal{X}_{\text{fix}}$

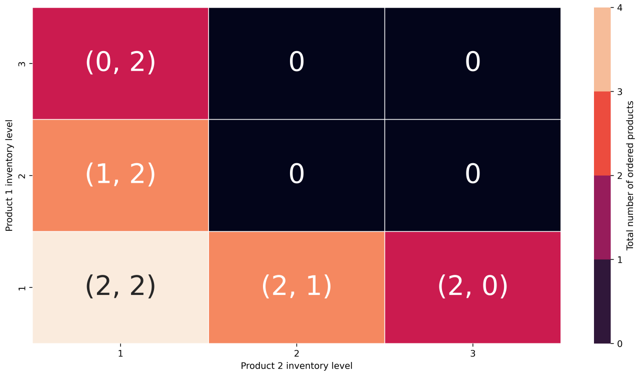
Fixed policy α_{fix} specifying order sizes for each inventory level in the effectively reduced state space under the fixed policy $\mathcal{X}_{\text{fix}}$. $\alpha_{\text{fix}}(x_1, x_2) = (a_1, a_2)$ tuple in each field specifies the order sizes for each product if an order is made for at least one of the products. For states where the fixed action is not to order either product, 0 is used. Color indicates total number of ordered items ($\sum(a_1, a_2)$).



	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
20	(0, 2)	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
19	(0, 2)	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
18	(0, 2)	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
17	(0, 2)	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
16	(0, 2)	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
15	(0, 2)	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
14	(0, 2)	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
13	(0, 2)	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
12	(0, 2)	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
11	(0, 2)	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
10	(0, 2)	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
9	(0, 2)	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
8	(0, 2)	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
7	(0, 2)	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
6	(0, 2)	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
5	(0, 2)	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
4	(0, 2)	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
3	(0, 2)	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
2	(1, 2)	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1	(2, 2)	(2, 1)	(2, 0)	(2, 0)	(2, 0)	(2, 0)	(2, 0)	(2, 0)	(2, 0)	(2, 0)	(2, 0)	(2, 0)	(2, 0)	(2, 0)	(2, 0)	(2, 0)	(2, 0)	(2, 0)	(2, 0)	(2, 0)

Figure 2: Optimal Policy α_*

Optimal policy α_* specifying order sizes for each inventory level. $\alpha_*(x_1, x_2) = (a_1, a_2)$ tuple in each field specifies the order sizes for each product if an order is made for at least one of the products. For states where the optimal action is not to order either product, 0 is used. Color indicates total number of ordered items ($\sum(a_1, a_2)$).



	1	2	3
3	(0, 2)	0	0
2	(1, 2)	0	0
1	(2, 2)	(2, 1)	(2, 0)

Figure 3: Optimal Policy α_* in effectively reduced state space $\mathcal{X}_*$

Optimal policy α_* specifying order sizes for each inventory level in the effectively reduced state space under the optimal policy $\mathcal{X}_*$. $\alpha_*(x_1, x_2) = (a_1, a_2)$ tuple in each field specifies the order sizes for each product if an order is made for at least one of the products. For states where the optimal action is not to order either product, 0 is used. Color indicates total number of ordered items ($\sum(a_1, a_2)$).

[3] The Φ_* estimate obtained using the exact, analytical method, $\langle \pi_{*\text{exact}}, C \rangle = 10.445$, matches the Φ_* estimate from Poisson value iteration up to 9 decimal places since it does not depend on iteration with random initiation.