Share Market

Lucas Guesser Targino da Silva - RA: 203534

January 10, 2023

1 Basic Definitions

Definition 1 (Natural)**.** Given $v \in \mathbb{N}$, we define:

$$\mathbb{N}_v = \{n \in \mathbb{N} : n \geqslant v\} \tag{1}$$

Definition 2 (Range)**.** Given $n \in \mathbb{N}_2$, we define the **Range of** n as:

$$[[n]] = \{0, \dots, n-1\} = \{i \in \mathbb{N} : i < n\} \tag{2}$$

Definition 3 (Square)**.** Given $n \in \mathbb{N}_2$, we define the **Square of** n as:

$$sq(n) = [[n]] \times [[n]] \tag{3}$$

2 Problem Definitions

You are given an array in which the ith element is the price of a given stock on the day i. You are permitted to complete at most 1 transaction (i.e. buy once and sell once). What is the maximum profit you can gain?

Notice that you cannot sell a stock before buying it.

2.1 Definitions

Definition 4 (Price Tuple)**.** Given $n \in \mathbb{N}_2$, a **Price Tuple** p is a positive real tuple with n elements:

$$p = \langle p_0, \dots, p_{n-1} \rangle \in \mathbb{R}_+^n \tag{4}$$

Definition 5 (Operation Pair)**.** Given $n \in \mathbb{N}_2$, an **Operation Pair** ω is a pair:

$$\omega = \langle b(\omega), s(\omega) \rangle \in sq(n) \quad , b < s \tag{5}$$

when the context is clear enough, we will simply write $\omega = \langle b(\omega), s(\omega) \rangle = \langle b, s \rangle$.

Definition 6 (Ascending Operation Pair)**.** Given $n \in \mathbb{N}_2$ and a Price Tuple p, an Operation Pair $\omega = \langle b, s \rangle$ is said to be **Ascending** when:

$$p_b \leqslant p_{b+1} \leqslant \dots \leqslant p_{s-1} \leqslant p_s \tag{6}$$

Definition 7 (Maximal Operation Pair)**.** Given $n \in \mathbb{N}_2$ and a Price Tuple p, we say that an Ascending Operation Pair $\omega = \langle b, s \rangle$ is a **Maximal Operation Pair**, or simply that ω is **Maximal**, when it is Ascending and it satisfies the conditions:

$$p_s - p_b \geqslant p_{s+1} - p_b \quad \text{, if } s + 1 < n \tag{7}$$

$$p_s - p_b \geqslant p_s - p_{b-1} \quad \text{, if } b - 1 > 0 \tag{8}$$

Theorem 1. *Given $n \in \mathbb{N}_2$ and a Price Tuple p, an Operation Pair $\omega = \langle b, s \rangle$ is Maximal if and only if the following conditions are satisfied:*

$$p_s \geqslant p_{s+1} \quad \textit{, if } s + 1 < n \tag{9}$$

$$p_b \leqslant p_{b-1} \quad \textit{, if } b - 1 > 0 \tag{10}$$

Proof. It comes directly from the inequalities of the Definition 7. □

Definition 8 (Independent Operation Pairs)**.** Given two Operation Pairs ω, ω', we say that ω, ω' are **Independent** when:

$$s\left(\omega\right) < b\left(\omega'\right) \vee s\left(\omega'\right) < b\left(\omega\right) \tag{11}$$

moreover, given a set of Operation Pairs $S = \{\omega_0, \ldots, \omega_{m-1}\}$, we say that S is independent when it satisfies:

$$\forall \omega \forall \omega' \left(\omega \in S \wedge \omega' \in S \rightarrow \omega, \omega' \text{ are independent}\right) \tag{12}$$

i.e. all Operation Pairs $\omega, \omega' \in S$ are pairwise Independent.

Definition 9 (Operation Set)**.** Given $n \in \mathbb{N}_2$, an **Operation Set** $S \subseteq sq(n)$ is a set which satisfies:

1. $\forall \omega \left(\omega \in S \rightarrow \omega \text{ is an Operation Pair}\right)$
2. S is independent

Moreover, we denote the set of all possible Operations Set by:

$$\Omega_n = \{S \subseteq sq(n) : S \text{ is an Operation Set}\} \tag{13}$$

Definition 10. Given an $n \in \mathbb{N}_2$, a Price Tuple p, and an Operation Set $S \in \Omega_n$, the price $\rho\left(S\right)$ of S is:

$$\rho\left(S\right) = \sum_{\omega \in S} p_{s(\omega)} - p_{b(\omega)} \tag{14}$$

Lemma 1. *Given an $n \in \mathbb{N}_2$ and a Price Tuple p, if a Operation Set $S^* \in \Omega_n$ is has optimal price, then all Operation Pairs of S^* are Maximal, i.e.:*

$$\rho\left(S^*\right) = \max_{S \in \Omega_n} \left[\rho\left(S\right)\right] \rightarrow \forall \omega \left(\omega \in S^* \rightarrow \omega \textit{ is Maximal}\right) \tag{15}$$

Proof. Suppose $S^* \in \Omega_n$ is a Operation Set with optimal price. Suppose by contradiction that $\exists \omega \left(\omega \in S^* \wedge \omega \text{ is not Maximal}\right)$, and let $\omega = \langle b, s \rangle$ be such Operation Pair. By Theorem 1, one of the following cases must be true:

1. $p_s < p_{s+1}$, if $s + 1 < n$
2. $p_b > p_{b-1}$, if $b - 1 > 0$

Case 1

Suppose that $p_s < p_{s+1}$, $s+1 < n$.

Case 1.1

Suppose in addition that $\omega' = \langle b, s+1\rangle$ is independent of all $\omega \in S^* \setminus \{\omega\}$. Let $S' = (S^* \setminus \{\omega\}) \cup \{\omega'\}$. Notice that

$$\rho(\omega) = p_{s(\omega)} - p_{b(\omega)} < p_{s(\omega')} - p_{b(\omega')} = \rho(\omega') \tag{16}$$

Therefore $\rho(S^*) < \rho(S')$ (because S^* and S' differ only in the elements above), contradicting the optimality of S^*.

Case 1.2

Suppose in addition that $\exists \omega' (\omega' \in S^* \wedge b(\omega') = s+1)$, and let $\omega' = \langle s+1, s'\rangle$ be such Operation Pair. Let $\omega'' = \langle b, s'\rangle$ and $S'' = (S^* \cup \omega'') \setminus \{\omega, \omega'\}$. Notice that

$$\begin{aligned} \rho(\omega) + \rho(\omega') &= \\ (p_{s(\omega)} - p_{b(\omega)}) + (p_{s(\omega')} - p_{b(\omega')}) &= \\ (p_s - p_b) + (p_{s'} - p_{s+1}) &= \\ (p_{s'} - p_b) + (p_s - p_{s+1}) &< \\ p_{s'} - p_b &= \\ \rho(\omega'') & \end{aligned} \tag{17}$$

Therefore $\rho(S^*) < \rho(S'')$ (because S^* and S' differ only in the elements above), contradicting the optimality of S^*.

Case 1 - Conclusion

The hypothesis $p_s < p_{s+1}$, $s+1 < n$ leads to a contradiction. Therefore, if S^* is optimal, then $p_s \geqslant p_{s+1}$, $s+1 < n$, as we wanted to prove.

Case 2

Suppose that $p_b > p_{b-1}, b-1 > 0$. The proof of this case is similar to the proof of the Case 2.1, but this uses b and $b-1$ instead of s and $s-1$. $\square$

Definition 11. Given an $n \in \mathbb{N}_2$, let $\omega = \langle b, s\rangle \in sq(n)$ be an Operation Pair. We say that i is included in ω, and denote by $i \rhd \omega$, when $b \leqslant i \leqslant s$.

Definition 12. Given an $n \in \mathbb{N}_2$ and a Price Tuple p, we say that an Operation Set $S \in \Omega_n$ is Great when:

$$\forall i \big((i \in [[n-1]] \wedge p_i < p_{i+1}) \rightarrow (\exists \omega (\omega \in S \wedge i \rhd \omega \wedge (i+1) \rhd \omega)) \big) \tag{18}$$

i.e. the indices of all ascending pairs of p are included in ω.

Lemma 2. *Given an $n \in \mathbb{N}_2$ and a Price Tuple p, if a Operation Set $S^* \in \Omega_n$ is has optimal price, then S^* is Great.*

Proof. Suppose by contradiction that S^* is not Great, i.e.

$$\exists i \big((i \in [[n-1]] \wedge p_i < p_{i+1}) \wedge (\nexists \omega (\omega \in S \wedge i \rhd \omega \wedge (i+1) \rhd \omega)) \big) \tag{19}$$

and let i be such value.

Case 1

Suppose that $i \triangleright \omega \wedge \neg(i+1 \triangleright \omega)$ for some $\omega \in S^*$. If $i+1 \triangleright \omega'$ for some ω', then join ω and ω' to get an Operation Set with cost greater than S^*. If that is not the case, extend ω to include $i+1$ to get an Operation Set with cost greater than S^*. In all cases, the original hypothesis contradicts the optimality of S^*.

Case 2

Suppose that $\neg(i \triangleright \omega) \wedge i+1 \triangleright \omega$ for some $\omega \in S^*$. This case is similar to the previous one, except that the Operation Pair has to be extended backwards instead of forwards.

Case 3

Suppose that $\forall \omega(\omega \in S^* \rightarrow (\neg(i \triangleright \omega) \wedge \neg(i+1 \triangleright \omega)))$. Create a new Operation Set $S' = S^* \Cup \{\langle i, i+1 \rangle\}$, which has greater cost, contradicting the optimality of S^*.

Conclusion

The cases above cover all possible cases. Therefore, the lemma has been proven by contradiction.

□

Lemma 3. *Given an $n \in \mathbb{N}_2$ and a Price Tuple p, if an Operation Set $S^* \in \Omega_n$ satisfy:*

1. $\forall \omega \left(\omega \in S^* \rightarrow \omega \text{ is Maximal}\right)$
2. *S^* is Great*

then S has optimal price, i.e $\rho(S) = \max\limits_{S' \in \Omega_n} [\rho(S')]$.

It is tiresome to write this proof so I won't.

Theorem 2. *Given an $n \in \mathbb{N}_2$ and a Price Tuple p, an Operation Set $S^* \in \Omega_n$ has optimal price if and only if it satisfies:*

1. $\forall \omega \left(\omega \in S^* \rightarrow \omega \text{ is Maximal}\right)$
2. *S^* is Great*

Proof. Lemmas 1 and 2 prove the $\Rightarrow$ part. Lemma 3 proves the $\Leftarrow$. □

3 Problem Description

Input a Price Tuple p

Output a value $z \in \mathbb{R}$

Goal $\max z$

4 Solution - Dynamic Programming

Given that you buy on a day i, while the value does not decrease, you keep it. If it will drop the next day, you sell it.

4.1 Initial State

Find the first pair $\langle b, s\rangle$ for which the price increases, i.e. the first pair of consecutive indices for which $p_s - p_b > 0$.

4.2 Optimal Substructure

Let:

1. $\langle b_l, s_l\rangle$ be the last operation;
2. p_{s_l} be the price of the last sell;
3. i the index of the current day;
4. p_i the stock price of the current day;
5. p_{i-1} the stock price of the previous day;

Cases:

1. if $(s_l == i - 1) \wedge (p_{s_l} \leqslant p_i)$
   (a) replace $\langle b_l, s_l\rangle$ by $\langle b_l, i\rangle$
   (b) rationale: if the stock price is increasing, you keep it;
2. if $(s_l < i - 1) \wedge (p_{i-1} \leqslant p_i)$
   (a) add $\langle i - 1, i\rangle$
   (b) rationale: if you have no stock and it will increase, you buy and sell it;
3. the others are cases in which the stock price drops, and there is nothing to do;

Complexity: $\mathcal{O}(n)$

5 Solution - Simple

Algorithm 1 Simple-Algoruithm

```
1: Δp ← [⟨p_{i+1} − p_i⟩ for i ∈ {0, ..., n − 2}]
2: Δp_> ← filter(Δp, (>= 0))
3: r ← sum(Δp_>)
4: return r
```

The filter takes care of removing the drops on the price, while the sum of the differences computes the gains.

Complexity: $\mathcal{O}(n)$