

# Optimalna strategija med črpanjem in skladiščenjem surove nafte

Univerza v Ljubljani  
Fakulteta za matematiko in fiziko,  
17. 1. 2022

Avtor:  
Pia Klemenc  
Tjaša Trafela

Mentor:  
Prof. dr. Sergio Cabello  
Doc. dr. Janoš Vidali

## Kazalo vsebine

1. Uvod .....	3
2. Opis problema .....	3
3. Spremenljivke in omejitve .....	3
4. Koda v programu Python .....	4
5. Uporaba programa in primeri .....	6

# 1. Uvod

V projektni nalogi bomo iskali optimalno strategijo med črpanjem, skladiščenjem in shranjevanjem nafte v daljšem časovnem obdobju pri določenih pogojih in omejitvah, z namenom dosege najvišjega dobička. Reševanja takšnega problema se bomo lotili s pomočjo programiranja in si z napisanim programom pogledali tudi nekaj primerov in rezultatov, jih predstavili v grafični obliki in si jih poskušali na čim bolj smiseln način interpretirati.

## 2. Opis problema

Proizvajalec nafte se mora odločiti, kolikšno količino sodov na dan bi izčrpali, prodali in kolikšno shranili, da bi prišli do maksimalnega dobička v  $n$  dneh. To optimalno operativno strategijo je potrebno izvajati dnevno, pri tem pa upoštevati fizične, operativne in finančne omejitve, kot so: zmogljivost skladiščenja in stroški shranjevanja, stroški črpanja in maksimalna dovoljena dnevna količina, ki jo lahko izčrpamo ter prodajna cena nafte.

## 3. Spremenljivke in omejitve

Naš nabor spremenljivk je naslednji:

- $Q_t^i$  ... količina sodov, izčrpana v času  $t$
- $Q_t^p$  ... količina sodov, prodana v času  $t$
- $X_t$  ... količina sodov zalog na začetku dneva
- $STR_t^s$  ... stroški skladiščenja enega soda
- $P_t$  ... prodajna cena enega soda
- $STR_t^c$  ... stroški črpanja enega soda
- $K$  ... maksimalna količina sodov, ki jo lahko izčrpamo v enem dnevu
- $S$  ... maksimalna količina sodov, ki jo lahko shranimo (kapaciteta skladišča)

Omejitve:

O1: Količina zalog na koncu dneva mora biti manjša od kapacitete skladišča:

$$0 \leq X_t + Q_t^i - Q_t^p \leq S$$

O2: Količina izčrpana v enem dnevu mora biti manjša od dovoljene dnevno izčrpane količine:  $0 \leq Q_t^i \leq K$

O3: Skupni stroški so končni in so enaki:

$$STR_t = STR_t^c * Q_t^i + STR_t^s * (X_t + Q_t^i - Q_t^p)$$

O4: V času  $t$  ne moremo prodati količine, ki je višja od naše trenutne zaloge:

$$Q_t^p \leq X_t + Q_t^i \text{ za vsak } t > 0 \text{ in}$$

$$Q_t^i, Q_t^p \geq 0 \text{ za vsak } t > 0$$

Predpostavljamo, da stroški skladiščenja niso konstantni in so odvisni od količine nafte, ki je bila v času  $t$  skladiščena. Kolikor več nafte je shranjene, toliko večji so stroški skladiščenja.

## 4. Koda v programu Python

```
def nafta(dni, K, S, P, STRc, STRs):
    inf = float('inf')
    d = {(0, 0): 0}
    p = {(0, 0): None}
    for i in range(dni):
        for x in range(S+1):
            kandidati = [(-inf, (0, 0, 0))]
            for Qi in range(K+1):
                for Qp in range(max(0, Qi - x), S + Qi - x + 1):
                    x_vceraj = x + Qp - Qi
                    ix_vceraj = (i, x_vceraj)
                    if ix_vceraj in d:
                        d_strategije = P[i] * Qp - STRc[i] * Qi - STRs[i] * x
                        di = d_strategije + d[ix_vceraj]
                        kandidati.append((di, (d_strategije, Qi, Qp, x_vceraj)))
            M, strategija = max(kandidati)
            d[i+1, x] = M
            p[i+1, x] = strategija
    koncna_zaloga = 0
    resitev = {}
    for i in range(dni, 0, -1):
        resitev[i, koncna_zaloga] = p[i, koncna_zaloga]
        koncna_zaloga = p[i, koncna_zaloga][3]
    resitev = dict(list(resitev.items())[::-1])
    return (d[dni, 0], resitev)
```

Slika 1: Koda programa

Program kot vhodne podatke sprejme spremenljivke število dni, v katerem išče optimalno strategijo, maksimalno dnevno izčrpano količino  $K$ , kapaciteto maksimalnih zalog  $S$ , prodajne cene  $P$ , stroške črpanja  $STR_c$  in stroške skladiščenja  $STR_s$ .

Predpostavimo, da prodajne cene, stroški črpanja in stroški skladiščenja niso konstantni in so lahko vsak dan različni, vendar so nam znani in podani v času  $t = 0$  in se v prihodnjih časih ne morejo več spremeniti. Zato so te spremenljivke podane s seznamom, v katerih za vsak dan podamo ceno in stroške.

Predpostavimo še, da je čas diskretna spremenljivka, saj iščemo optimalno strategijo ob vsakem koncu dneva, prav tako so diskretne spremenljivke tudi količine skladiščenja, črpanja, shranjevanja in prodajanja, saj govorimo o številu sodov in prodamo lahko le celoten sod.

V kodi najprej definiramo slovar  $d$ , katerega glavni namen je shranjevanje skupnega dobička  $i$ -tih dni, in slovar  $p$ , kamor bomo shranjevali vse potencialne optimalne strategije.

Program se sprehodi čez vse dni, vse možne končne zaloge, vse možne opcije izčrpane količine in vse možne opcije prodane količine. Za vse trojice  $(Q_i, Q_p, \text{končna zaloga v } i\text{-tem dnevu})$  v  $i$ -tem dnevu izračuna dobiček. Potem med seboj primerja elemente  $(i\text{-ti dan, končna zaloga } i\text{-tega dne})$ :  $(Q_i, Q_p, \text{začetna zaloga } i+1\text{-tega dne})$ , kjer sešteva dnevne dobičke z ujemajočima končno zalogo  $i$ -tega dne in začetno zalogo  $i+1$ -tega dne. Te skupne dobičke in pripadajoče pare  $= (i\text{-ti dan, končna zaloga } x)$  potem dodaja v slovar  $d$ , v slovar  $p$  pa dodaja pare  $= (i\text{-ti dan, končna zaloga } x)$  in pripadajočo četverico  $= (\text{dnevni dobiček}, Q_i, Q_p, \text{začetna zaloga})$ .

Rešitev v programu izgleda takole:

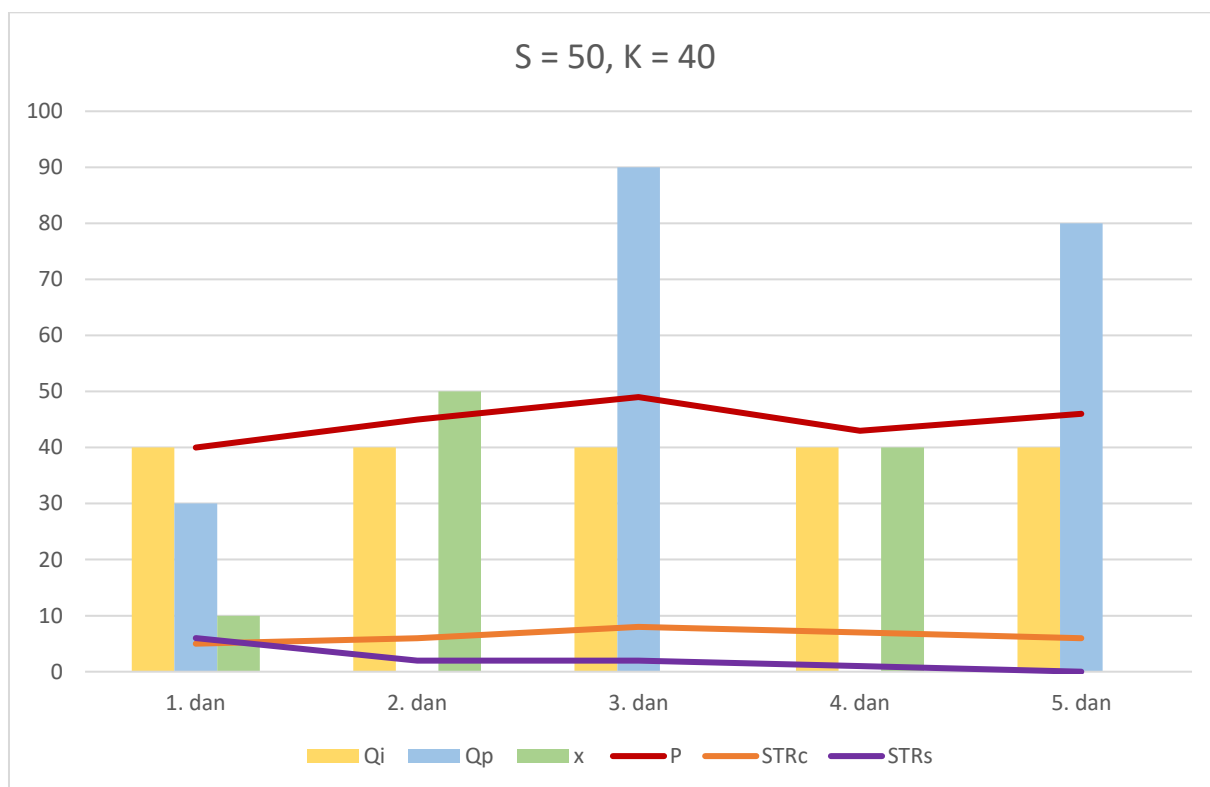
*(optimalen skupni dobiček, {(dan1, končna zaloga): (dnevni optimalni dobiček, izčrpana količina, prodana količina, začetna zaloga), (dan2, končna zaloga): (dnevni optimalni dobiček, izčrpana količina, prodana količina, začetna zaloga), ..., (zadnji dan, končna zaloga): (dnevni optimalni dobiček, izčrpana količina, prodana količina, začetna zaloga)}).*

Torej vrne optimalni dobiček na koncu gledanega obdobja, ob predpostavki, da na koncu vse zaloge prodamo ter dnevne optimalne strategije (optimalne izčrpane, prodane ter shranjene količine nafte).

## 5. Uporaba programa in primeri

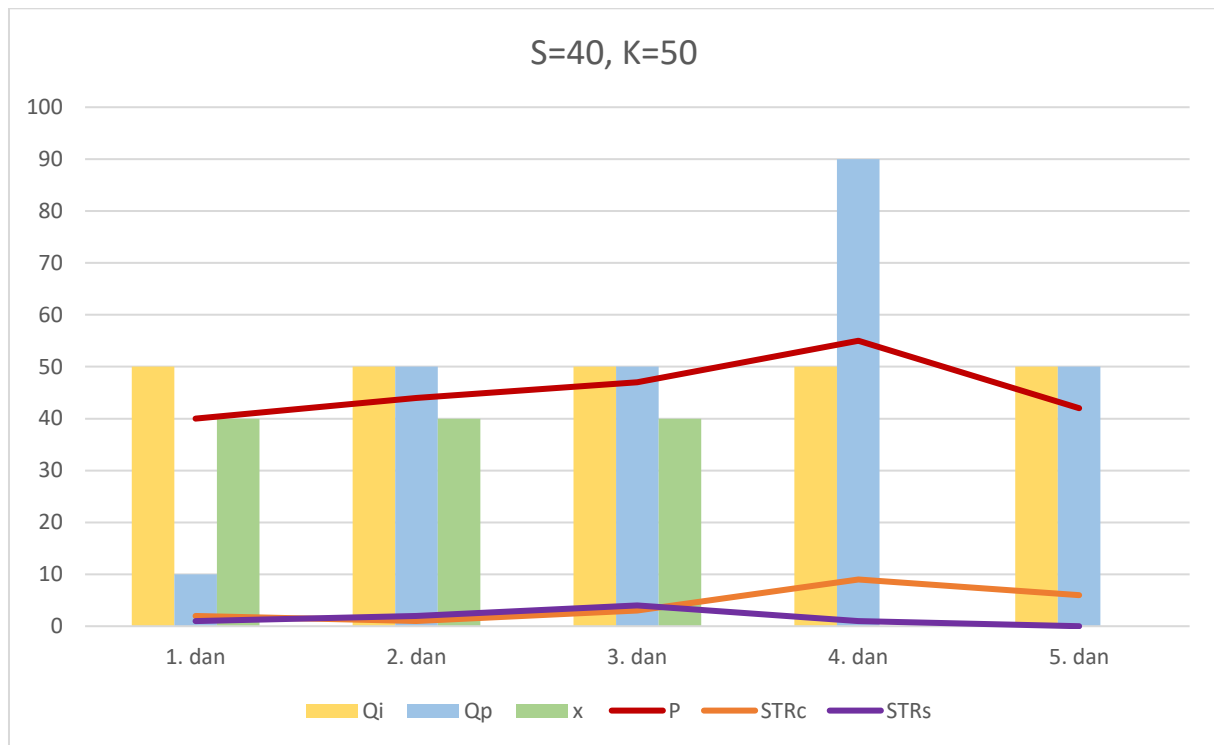
Pri interpretaciji rezultatov je pomembno, da upoštevamo, kako so se spreminjali stroški in prodajna cena skozi celotnih 5 dni, ne le na izbrani dan. Prav tako smo v vseh primerih vsak dan črpali maksimalno dovoljeno količino, saj stroški črpanja nikoli ne presežejo prodajne cene, v zadnjem dnevu pa vsakokrat prodamo vse zaloge, saj ni optimalno, da bi zaloge zadržali, zato so stroški shranjevanja zadnji dan kar enaki 0, saj ta podatek ni pomemben za naš program.

Graf 1: optimalna strategija za primer  
*nafta(5, 40, 50, [40, 45, 49, 43, 46], [5, 6, 8, 7, 6], [6, 2, 2, 1, 0])*



V prvem primeru je količina skladiščenja  $S$  večja od dovoljene količine za črpanje  $K$ . Vidimo, da je cena najvišja 3. dan, zato bi si po hitrem sklepanju takrat želeli prodati največ nafte, torej bi radi do takrat shranili čim več zalog, vendar le v takšni meri in pod pogoji, da bodo stroški skladiščenja in črpanja v 1. in 2. dnevu nižji od prodajne cene v 3. dnevu, da nam ne uničijo možnosti za večji dobiček. Zato smo 1. dan skladiščili le manjši del nakupa, 2. dan smo shranili vse kar smo izčrpali ter nato vse zaloge prodali 3. dan. V 4. dnevu so bili skupni stroški spet tako nizki, da smo celotno izčrpano količino shranili in jo prodali naslednji dan po višji ceni.

Graf 2: optimalna strategija za primer  
 nafta(5, 50, 40, [40, 44, 47, 55, 42], [2, 1, 3, 9, 6], [1, 2, 4, 1, 0])



V tem primeru pa je količina skladiščenja  $S$  manjša od količine dovoljenega črpanja  $K$ , kar pomeni, da bomo vsak dan morali prodati del izčrpanih zalog. Prodajna cena je najvišja 4. dan. Spet nas zanima, kolikšno količino dnevno naj shranimo in jo raje prodamo 4. dan, ne da bi pri tem znižali dobiček. Ker se iz 1. na 2. dan cena zviša za več kot se zvišajo skupni stroški, se nam bolj splača čim več shraniti. 2. in 3. dan ohranimo maksimalno količino zalog in prodamo kar ne moremo shraniti, vse zaloge pa prodamo 4. dan, ko je cena najvišja in višina preteklih stroškov shranjevanja ne zniža dobička. 5. dan prodamo vse kar izčrpamo, saj je prodajna cena višja od stroškov črpanja, zalog pa ne želimo imeti več.