Taylor & Francis
Taylor & Francis Group

Ameriški matematični mesečnik

ISSN: (tiskano) (spletno) Domača stran revije: https://www.tandfonline.com/loi/uamm20

Džip čez puščavo neznane širine

Richard E. Korf

Navajanje tega članka: Korf (2022) A Jeep Crossing a Desert of Unknown Width, The American Mathematical Monthly, 129:5, 435-444, DOI: 10.1080/00029890.2022.2051404

Povezava na ta članek: https: //doi.org/10.1080/00029890.2022.2051404

Džip čez puščavo neznane širine

Richard E. Korf

Povzetek. Klasični problem jeepov se nanaša na prečkanje puščave, ki je širša od dosega jeepa, s pomočjo vnaprej postavljenih zalog goriva. S tem problemom in njegovimi različicami se je že veliko ukvarjalo, optimalna strategija pa je dobro znana, vendar vsa prejšnja dela predpostavljajo, da poznamo širino puščave. Mi obravnavamo primer, ko razdalje ne poznamo vnaprej. Strategijo ocenimo s konkurenčnim razmerjem, ki je razmerje med stroški strategije v najslabšem primeru in stroški optimalne rešitve, če bi razdaljo poznali vnaprej. Pokažemo, da nobena strategija s fiksnim zaporedjem predpomnilnikov ne more doseči končnega konkurenčnega razmerja. Optimalna strategija je iterativna strategija, ki za doseganje zaporedja ciljnih razdalj uporablja optimalno strategijo znane razdalje, pri čemer med iteracijami izprazni vse predpomnilnike. Optimalna iterativna strategija podvoji stroške vsake naslednje iteracije in doseže konkurenčno razmerje štiri.

1. UVOD. Imamo džip z določeno zmogljivostjo goriva in dosegom, ki ga lahko prevozi s polnim polnjenjem goriva. Brez izgube splošnosti predpostavimo, da lahko džip prevaža eno galono goriva in da lahko prevozi eno miljo na galono, vendar želi prečkati puščavo, ki je široka več kot eno miljo. Na začetku imamo neomejeno količino goriva, na poti pa lahko shranimo neomejeno količino goriva. Želimo čim bolj zmanjšati stroške potovanja na določeno razdaljo, ki jih merimo kot količino goriva, porabljenega od začetka. To je zgornja meja celotne prevožene razdalje ali porabljenega goriva, če v skrivališčih ostane gorivo.

V enosmerni različici problema moramo samo prečkati puščavo, v dvosmerni različici pa se moramo tudi vrniti na začetek. Sprejemamo dvosmerno različico, ker ne poznamo razdalje in se moramo vedno znati vrniti na začetek.

2. PREDHODNO DELO. Prvi rešitvi tega problema sta ponudili [3] in [7] leta 1947. Veliko lažje je določiti največjo razdaljo, ki jo lahko prevozimo z določeno količino goriva, kot najmanjšo količino goriva, ki je potrebna za prevoz določene razdalje. [3] je rešil enosmerni problem, [7] pa povratni problem. V [7] je bila predstavljena tudi alternativna različica s karavano džipov, ki si lahko delijo gorivo in od katerih mora pot opraviti le eden. Druge različice tega problema so v [1, 2, 4-6].

Z eno galono goriva lahko prevozimo 1/2 milje in se vrnemo. Z dvema galonama lahko prevozimo 1/4 milje, shranimo 1/2 galone goriva in se vrnemo. Nato gremo 1/4 milje, poberemo 1/4 galone, gremo 1/2 milje naprej in nazaj, poberemo preostalo 1/4 galone in se vrnemo na začetek. S tremi galonami lahko prehodimo 1/6 + 1/4 + 1/2 milje, pri čemer so skrivališča na 1/6 in 1/6 + 1/4 kilometrov od začetka. Na splošno lahko z n litri prevozimo 1/2 + 1/4 + 1/6 + --- + $1/2n$ milj, z $n - 1$ kešami, ki so razporejene na teh razdaljah druga od druge, od konca nazaj. Nikoli ne nosimo nazaj več goriva, kot ga potrebujemo za dosego prejšnje cache ali start, vsak skok naprej pa se začne s polno galono goriva.

Da bi našli najmanjšo količino goriva, ki je potrebna za premagovanje določene razdalje, poiščemo največjo vsoto tega niza, ki je manjša ali enaka določeni razdalji, in na ta mesta postavimo keše, ki se vračajo od konca. Nato od začetka opravimo toliko voženj, kolikor jih je potrebnih, da prenesemo potrebno količino goriva v prvo shrambo.

Količina goriva, ki je potrebna za prevoz d milj in vrnitev, je $O\ e^{2d}$ galon [7]. Tako je strošek goriva sorazmeren s približno $7{,}389^d$ galonami, v mejah velikega d.

doi.org/10.1080/00029890.2022.2051404
MSC: primarni 00A08, sekundarni 90B06; 68W40

3. NEZNANA RAZDALJA IN KONKURENČNO RAZMERJE. Vse prejšnje delo na tem problemu predpostavlja, da vemo, kako daleč moramo iti. Tu obravnavamo optimalno strategijo, kadar vnaprej ne vemo, kako daleč moramo potovati.

Za oceno takšne strategije uporabimo njeno konkurenčno razmerje [8]. Ta tehnika se pogosto uporablja za analizo algoritma, kadar informacije, ki so ključne za njegovo delovanje, niso vnaprej znane. Konkurenčno razmerje algoritma za ta problem je strošek doseganja določene razdalje z algoritmom, deljen s stroškom optimalne rešitve za to razdaljo, če bi razdaljo poznali vnaprej. Najslabše konkurenčno razmerje je največja vrednost tega razmerja za vse možne razdalje, optimalni algoritem pa je algoritem z najmanjšo tako največjo vrednostjo.

4. PREGLED. Najprej obravnavamo primer fiksnih, enakomerno razporejenih predpomnilnikov in pokažemo, da nobena taka strategija ne more doseči končnega konkurenčnega razmerja. Nato pokažemo, da lahko vsako rešitev s fiksnim zaporedjem predpomnilnikov pretvorimo v rešitev z enakomerno razporejenimi predpomnilniki brez izgube učinkovitosti. Nato uvedemo iterativno strategijo, pri kateri uporabimo zaporedje vedno daljših razdalj, pri čemer za vsako razdaljo uporabimo znano optimalno strategijo, med iteracijami pa izpraznimo vse predpomnilnike. Nazadnje pokažemo, da eno optimalno zaporedje iteracij podvoji stroške vsake predhodne iteracije in z vsako iteracijo poveča razdaljo za nekaj več kot tretjino milje.

5. STRATEGIJE S FIKSNIM ZAPOREDJEM PREDPOMNILNIKOV. Najpreprostejša Strategija bi večkrat uporabila določeno zaporedje predpomnilnikov, dokler ne bi dosegli cilja.

Da bi čim prej dosegli cilj, si vedno želimo iti čim dlje naprej, ne da bi pri tem žrtvovali učinkovitost. Za dano zaporedje predpomnilnikov opredelimo razpored za *naprej na* naslednji način. Predpostavimo, da smo pri skrinjici, da je razdalja do neposredno predhodne skrinjice p milj in da je skupno gorivo pri skrinjici, vključno z gorivom v džipu,
je f galon. Če je $f - p \geq 1$, pojdite naprej s polnim galonom. V nasprotnem primeru se vrnite nazaj s p galonami. Če smo na prvem odlagališču, začetek postane prejšnje odlagališče.
Ta urnik nikoli ne odnese nazaj več goriva, kot je najmanj potrebno za dosego prejšnje shrambe, in se vedno odpravi naprej s polnim polnjenjem goriva.

Pri časovnem razporedu naprej se džip ob vsakem obratu vrne nazaj na začetek. *Skok* opredelimo kot premik med dvema sosednjima kešema, *potovanje* pa kot začetek na začetku, premik naprej do neke keše in nato vrnitev na začetek.

Če so na primer med seboj oddaljene 1/3 milje, bo pot do prve skrinje v obe smeri trajala 1/3 galone, skrinja pa 1/3 galone. Pri dveh takšnih potovanjih ostane v prvi skrinji 2/3 galone. Pri tretjem potovanju se porabi 1/3 galona za pot do prve skrinje, tam se pobere 1/3 galona, nadaljuje do druge skrinje, tam se odloži 1/3 galona, vrne do prve skrinje, kjer se porabi še 1/3 galona, pobere 1/3 galona, ki ostane v prvi skrinji, in se vrne na začetek.

Naj bo $f(n)$ začetna količina goriva, ki je potrebna za dostavo 1/3 galone do n-*tega* predpomnilnika, če so vsi prejšnji predpomnilniki prazni. Potem je $f(1) = 1$, $f(2) = 2f(1) + 1 = 3$, $f(3) = 2f(2) + 2f(1) + 1 = 9$. Na splošno je $f(n) = 1 + \sum_{i=1}^{n-1} 2f(i) = 3^{n-1}$. Na spletni strani
tri keše na miljo, stroški doseganja d milj in vrnitev je $3^{3d-1} = O(27^d)$.

Če so predpomnilniki oddaljeni 1/4 milje, je strošek doseganja d milj $O(16^d)$. To je veliko bolje kot $O(27^d)$, vendar veliko slabše kot $O(7{,}389^d)$ za optimalno strategijo znane razdalje. Ker je eksponentna stopnja rasti teh strategij večja kot pri optimalni strategiji znane razdalje, so njihova konkurenčna razmerja neomejena.

Pravzaprav bomo v več korakih pokazali, da nobena strategija s fiksnim zaporedjem predpomnilnikov ne more doseči končnega konkurenčnega razmerja. Najprej bomo pokazali, da ima vnaprejšnji razpored s poljubnimi enakomerno razporejenimi predpomnilniki neomejeno konkurenčno razmerje. Nato bomo pokazali da je vnaprejšnji razpored enako učinkovit kot kateri koli drug razpored. Nazadnje bomo pokazali, da je pri poljubnem številu predpomnilnikov njihova enakomerna razporeditev najbolj učinkovita.

Napredni razpored z enakomerno razporejenimi predpomnilniki. Analiza keša, ki sta med seboj oddaljeni 1/3 ali 1/4 milje, je enostavna, saj se pri vsakem potovanju izpraznijo vse keše pred zadnjo obiskano kešo. Pri bolj oddaljenih skrinjicah pa se pri vsakem obhodu do skrinjice v nekaterih prejšnjih skrinjicah pusti gorivo. To znatno otežuje analizo.

Razmislite na primer o kešah, ki so med seboj oddaljene 1/5 milje. Pri potovanju do prve skrinjice boste porabili $1/5$ galone v vsako smer, pri skrinjici pa $3/5$ galone. Drugo potovanje bo prispelo do prve skrinje s $4/5$ galone, pobralo $1/5$, potovalo do druge skrinje, tam pustilo 3/5 galone, se vrnilo do prve skrinje, pobralo še $1/5$ in se vrnilo na začetek ter pustilo 1/5 v prvi skrinji. Naslednjič je doseči drugo skrinjico zaradi ostanka goriva v prvi skrinjici lažje kot prvič. Ključ do analize tega algoritma je razdelitev stroškov potovanja med več ciljev, ki jim služi. V zgornjem primeru drugo potovanje prispeva k doseganju drugega odlagališča tako prvič kot drugič. Medtem ko dejanski algoritem vsako potovanje vedno začne s polnim galonom, zato sta porabljeno gorivo in število potovanj vedno celo število, sprostimo prvotni problem, da omogočimo razdelitev potovanja na več potovanj, ki se začnejo z delnimi obremenitvami goriva, katerih vsota je ena. Neprekinjeno različico problema je pogosto veliko lažje analizirati kot diskretno različico, optimalni stroški neprekinjene relaksacije problema pa so spodnja meja stroškov diskretne različice, saj je rešitev diskretne različice tudi rešitev zvezne različice.

Izrek 1. *Za vnaprejšnji urnik z enakomerno razporejenimi predali, ki so med seboj oddaljeni $1/k$ milje, asimptotični stroški doseganja določene razdalje eksponentno rastejo z razdaljo,*

osnova eksponent pa je od spodaj omejena z $\left(\frac{k}{k-2}\right)^k$.

Dokaz. Prvo potovanje se začne s polno galono na začetku, na poti do prve skrinje in nazaj na razdalji $1/k$ milj porabi po 1/k galon in v skrinji pusti $1 - 2/k = \frac{k-2}{k}$ galon. Za prenos $\frac{k-2}{k}$ galon na razdalji $1/k$ milj je torej potrebna ena galona.

Ker bo ena galona prenesla $\frac{k-2}{k}$ galon $1/k$ milj, je potrebno $1/\frac{k-2}{k} = \frac{k}{k-2}$ galon.

za prenos polne galone v prvi predpomnilnik. Prenos polne galone v drugo shrambo, potrebujemo $\frac{k}{k-2}$ galon za prvo shrambo. Ker je za prenos ene galone potrebno $\frac{k}{k-2}$ galon do prvega predala, je za prenos $\frac{k}{k-2}$ galon do prvega predala potrebno $\left(\frac{k}{k-2}\right)^2$ galon in

hence a full gallon to the second cache. Ker je na miljo k predalov, je za prenos polne galone za eno miljo potrebnih $\left(\frac{k}{k-2}\right)^k$ galon, za prenos polne galone pa d milj potrebuje $\left(\frac{k}{k-2}\right)^{kd}$ galone.

Zaradi enostavnosti smo za mejnik vzeli prenos polne galone goriva v določen predalčnik, vendar je rezultat enak tudi pri doseganju naslednjega predala, saj se to zgodi na istem potovanju, ko se uporablja vnaprejšnji razpored. Tako stroški za začetno doseganje vsakega naslednjega kilometra rastejo eksponentno, osnova eksponentnega števila pa je $\left(\frac{k}{k-2}\right)^k$. To je le spodnja meja dejanskih stroškov, saj smo predpostavili delne obremenitve z gorivom in vožnje, medtem ko so v prak∎ dovoljene le polne obremenitve in celo število voženj.

Računalniške simulacije terminskega načrta dejansko kažejo, da asimptotska rast hitro konvergira k tej natančni vrednosti. Razlog za to je, da se preostalo gorivo, ki ostane pri vsakem potovanju, nikoli ne zapravi, temveč se uporabi pri naslednjem potovanju.

Za $k = 5$ je $\left(\frac{k}{k-2}\right)^k \approx 12{,}86$, za $k = 6$ je $\left(\frac{k}{k-2}\right)^k \approx 11{,}39$ in za $k = 7$ je $\left(\frac{k}{k-2}\right)^k \approx$

k-2 k-2 k-2

10.54. Ko gre k v neskončnost, se ta količina približuje $e^2 \approx 7{,}389$. Tako za vsako končno vrednost k stroški te strategije rastejo eksponentno hitreje kot stroški optimalne strategije, zato je konkurenčno razmerje v najslabšem primeru neomejeno.

Pri tej strategiji je treba obravnavati še dva vidika: vnaprejšnje načrtovanje skokov in predpostavko o enakomerno razporejenih predpomnilnikih.

Naprejšnji razpored je optimalen način načrtovanja skokov. Nikoli ni optimalno prenesti nazaj več goriva, kot je najmanj potrebno za dosego prejšnjega predpomnilnika, in skoraj nikoli ni optimalno začeti katerega koli skoka z manj kot polnim polnjenjem. Edina izjema je zadnji skok do prvega predpomnilnika za primer znane razdalje, če dejanska razdalja ni enaka eni od delnih vsot vrste. V okviru teh omejitev je mogoče posamezne skoke načrtovati na različne načine. V primeru znane razdalje lahko na primer najprej prenesemo vse gorivo, ki ga potrebujemo med začetkom in prvim predpomnilnikom, nato skoraj vse to gorivo prenesemo v drugi predpomnilnik itd. in pustimo le toliko, da se vrnemo na začetek. Časovni razpored za naprej je na drugem koncu spektra in se vedno premakne naprej, ko je v shrambi dovolj goriva, da se prenese poln tovor naprej in se še vedno vrne do prejšnje shrambe. Obstaja tudi vrsta vmesnih urnikov med tema dvema urnikoma . Pravzaprav imajo vsi razporedi, ki upoštevajo zgornji dve omejitvi, enake stroške.

Lemma 2. *Naprejšnje razporejanje skokov je optimalno.*

Dokaz. Upoštevajte katere koli tri sosednje predale, x, y in z, po vrstnem redu, pri čemer je x najbližje začetku. V vsaki rešitvi danega primera problema bo opravljeno določeno število skokov od x do y in od y do z. V optimalni rešitvi se bo vsak tak skok začel s polnim polnjenjem goriva. Število takih skokov je določeno s skupno količino goriva, potrebnega v točki z. Ne glede na vrstni red teh skokov je skupno število skokov od x do y in od y do z enako, če je v točki y vedno dovolj goriva za vsak skok od y do z s polnim polnjenjem. Ker je nadaljnji razpored v skladu s to omejitvijo, je to optimalni razpored. ■

Vmesni predpomnilniki izboljšajo učinkovitost. Učinkovitost prenosa goriva je lastnost enega ali več krožnih skokov. Opredeljena je kot neto količina goriva, dodanega v ciljni predpomnilnik, pomnožena z razdaljo od vira do cilja in deljena s količino goriva, odstranjenega iz izvornega predpomnilnika. Enote so prenesene galone, pomnožene s kilometri in deljene s skupnimi galoni. Zaradi enostavnosti uporabljamo količino goriva, odstranjenega iz izvornega predala, in ne količino goriva, ki je bila porabljena. Za en skok je to le gorivo, dodano v ciljni predpomnilnik, pomnoženo s prevoženo razdaljo, saj vedno začnemo s polnimi galoni. Na primer, en skok med dvema predpomnilnikoma, ki sta oddaljena 1/3 milje, doda 1/3 galone

v drugi predpomnilnik, kar pomeni učinkovitost $(1/3) \cdot (1/3) = 1/9$. Podobno je učinkovitost enega skoka med dvema predpomnilnikoma, ki sta oddaljena 1/4 milje, $(1/2) \cdot (1/4) = 1/8$. To majhno povečanje učinkovitosti posameznih skokov pomeni veliko zmanjšanje

pri stroških potovanja d milj, in sicer z $O(27^d)$ na $O(16^d)$. Za merjenje učinkovitosti potovanja , ki vključuje več skokov, je treba odstraniti vse gorivo v vseh vmesnih predalih.

Lemma 3. *Pri sprotnem razporedu lahko glede na kateri koli par sosednjih predpomnilnikov vedno izboljšamo učinkovitost prenosa goriva med njima z dodajanjem vmesnega predpomnilnika, pri čemer je najučinkovitejša postavitev na sredini med obema predpomnilnikoma.*

Intuicija je, da se z zmanjšanjem količine goriva na krovu zmanjša tudi učinkovitost prevoza, saj je prevožena razdalja konstantna. Z dodajanjem vmesnega predala se poveča povprečna količina goriva na krovu na celotni dolžini potovanja.

Dokaz. Učinkovitost bomo izračunali z vmesnim predpomnilnikom in brez njega. Ker je skupna razdalja v obeh primerih enaka, lahko za primerjavo učinkovitost opredelimo kot delež dostavljenih galon in odstranjenih galon na začetku.

Predpostavimo, da imamo tri predpomnilnike x, y in z v vrsti, pri čemer je x najbližje začetku. Naj bo razdalja med x in y a, razdalja med y in z pa b. Brez predpomnilnika pri y gremo neposredno od x do z in nazaj. Če začnemo z enim galonom pri x, bomo do z *dostavili* $1 - 2(a + b)$ galonov, kar pomeni učinkovitost $1 - 2a - 2b$.

Sedaj dodamo vmesni predpomnilnik na lokaciji y. Najprej opravimo krožno pot od x do y, začnemo z eno galono in dostavimo od 1 do $2a$ galon do y. Nato opravimo več krožnih voženj od x do z in nazaj do x, dokler ne porabimo vsega goriva v y. Za vsako takšno potovanje je treba porabiti 1 galono iz y, da se nadomesti gorivo, ki je bilo porabljeno od x do y, in še eno galono iz y. a galon za potovanje od y nazaj do x, kar pomeni neto zmanjšanje za $2a$ galonov v y, $\frac{1-2a}{2a}$ takšna potovanja bodo porabila gorivo v točki y. $\frac{1-2a}{2a}$ ni celo število, povečamo število galonov, s katerimi začnemo pri x, na $1 + m$, tako da je $m \frac{1-2a}{2a}$ celo število, pri čemer se te količine linearno povečajo, ne da bi to vplivalo na učinkovitost. Vsak takšen obhod bo prinesel $1 - 2b$ galon goriva v z, skupaj $\frac{(1-2a)(1-2b)}{2a}$ galonov. Skupno gorivo, s katerim se začne, je število takšnih potovanj plus galon za začetno krožno potovanje od x v y ali $\frac{1-2a}{2a} + 1 = \frac{1}{2a}$. Učinkovitost je razmerje med dobavljenim gorivom in gorivom, ki smo ga ali $\frac{(1-2a)(1-2b)}{2a} \cdot 2a = 1 - 2a - 2b + 4ab$. Če primerjamo učinkovitost z vmesnim predpomnilnikom in brez njega, dobimo $1 - 2a - 2b + 4ab$ proti $1 - 2a - 2b$ ali $4ab \geq 0$. Vmesni predpomnilnik torej poveča učinkovitost. Poleg tega, ker se največji produkt dveh vrednosti, ki sta enaki konstanti, pojavi pri enakih vrednostih, se največja učinkovitost pojavi, ko je $a = b$ in je y na sredini med x in z. ∎

Zakaj ta rezultat ne velja tudi za primer znane razdalje? V tem primeru se med vsakim parom sosednjih predpomnilnikov prenese fiksna količina goriva, razdalje med predpomnilniki pa so izbrane tako, da se vsak skok začne s polno obremenitvijo. Če dodamo vmesni predpomnilnik, se učinkovitost večine skokov izboljša, vendar se zadnji skok iz vmesnega predpomnilnika začne brez polne obremenitve, kar izniči predhodne koristi.

Lemma 4. *V primeru neznane razdalje morajo biti v optimalni rešitvi za fiksno zaporedje predalov predali enakomerno razporejeni.*

Dokaz. Predpostavimo, da imamo optimalno rešitev, pri kateri predpomnilniki niso enakomerno razporejeni. Potem morajo obstajati trije sosednji predali x, y do z, pri čemer razdalja od x do y ni enaka razdalji od y do z. Lemma 3 nam pove, da lahko izboljšamo učinkovitost prenosov v tem segmentu tako, da y prestavimo enako daleč med x in z, kar nasprotuje naši predpostavki, da je obstoječa rešitev optimalna. ∎

Zdaj lahko navedemo in dokažemo naš splošni rezultat za fiksne predpomnilnike.

Izrek 5. *Nobena strategija za primer neznane razdalje s fiksnim zaporedjem predpomnilnikov ne more doseči končnega konkurenčnega razmerja v najslabšem primeru.*

Dokaz. Predpostavimo, da imamo optimalno strategijo za primer neznane razdalje s fiksnim zaporedjem predpomnilnikov. Lemma 2 nam pove, da lahko posamezne skoke razporedimo z uporabo vnaprejšnjega razporeda brez izgube učinkovitosti. Lemma 4 nam pove, da morajo biti predpomnilniki enakomerno razporejeni. Nazadnje nam izrek 1 pove, da stroški vnaprejšnjega razporeda s predpomnilniki, ki so med seboj oddaljeni $1/k$ milje, eksponentno rastejo z razdaljo in da osnova eksponent je vsaj $\left(\frac{k}{k-2}\right)^k$. Ker je osnova eksponent za optimalno strategijo je za znano razdaljo e^2, ki je manjša od te vrednosti za vsak končni k, je konkurenčno razmerje vsake take strategije neskončno. ■

Edini način, da se doseže hitrost rasti e^2 za primer neznane razdalje s fiksnim zaporedjem predpomnilnikov, je, da se interval med sosednjimi predpomnilniki zmanjša na nič, kar pomeni neskončno število predpomnilnikov v vsakem končnem intervalu. Ker je za napredovanje po vnaprejšnjem načrtu potrebno več kot polno polnjenje goriva v katerem koli predpomnilniku, bi bilo za premik na poljubno razdaljo od začetka potrebno shraniti neskončno količino goriva v predpomnilnik.

Ta rezultat se mi je zdel zelo presenetljiv. Ko sem se lotil tega dela, sem bil prepričan, da obstaja optimalna strategija s fiksnim zaporedjem predpomnilnikov, in sem jo želel najti.

6. ITERATIVNE STRATEGIJE. Če zaporedje predpomnilnikov ni fiksno, se mora v določenem trenutku spremeniti. Dodajanje ali odstranjevanje predpomnilnikov za najbolj oddaljeno točko, ki je bila takrat dosežena, nima nobenega učinka in je enako, kot če bi imeli te predpomnilnike od začetka. Če dodamo nove predpomnilnike v regiji, ki je že bila prehojena, potem nam Lemma 3 pove, da je bolj učinkovito vključiti nove predpomnilnike z začetka. Podobno se učinkovitost zmanjša, če odstranimo predpomnilnike v regiji, ki je že bila prehojena.

Edina druga možnost je, da odstranite gorivo iz vseh skrinjic in začnete znova z novim zaporedjem skrinjic, saj bi bilo potratno puščati gorivo. Vsakič, ko zamenjamo keše, bomo napredovali za nekaj razdalje od začetka in edino, kar se bomo naučili, bo, da je cilj bolj oddaljen. Tako lahko te točke določimo vnaprej in katero koli strategijo, ki smo jo uporabljali do te točke, lahko nadomestimo s strategijo, ki uporablja optimalno zaporedje odlagališč za to znano razdaljo.

Tako nam ostane strategija, ki izbere zaporedje ciljnih razdalj in za vsako razdaljo uporabi znano optimalno strategijo, vmes pa izprazni predpomnilnike. To imenujemo iterativna strategija. Intuicija za tem je, da ker stroški doseganja določene razdalje z optimalno strategijo rastejo eksponentno z razdaljo, stroški neuspešnih iteracij ne bodo vplivali na asimptotične stroške algoritma, zaradi česar je konkurenčno razmerje končno. Preostali izziv je poiskati zaporedje ciljnih razdalj, ki zmanjšuje konkurenčno razmerje v najslabšem primeru.

Izračun konkurenčnega razmerja v najslabšem primeru. Cilj se vedno najde v zadnji iteraciji iterativne strategije, saj se algoritem konča, ko najde cilj. Zato mora biti območje zadnje iteracije enako ali večje od razdalje do cilja, območje prejšnje iteracije pa mora biti manjše od razdalje do cilja. Da bi cilj dosegli čim prej, uporabimo vnaprejšnji razpored.

V praksi bo to, kje je cilj najden v zadnji iteraciji, le malo vplivalo na stroške te iteracije. Zaradi enostavnosti merimo stroške iteracije s količino goriva, porabljenega od začetka, ali enakovredno s skupno prevoženo razdaljo, če dosežemo ciljno razdaljo iteracije. Ne glede na to, kje je bil v zadnji iteraciji najden cilj, bo to zgornja meja prevožene razdalje, kar nam daje zgornjo mejo konkurenčnega razmerja. Tako uporabimo celoten strošek zadnje iteracije kot strošek te iteracije, ne glede na to, kje v tej iteraciji je bil najden cilj.

Vendar pa to, kje je cilj najden v zadnji iteraciji iterativne strategije, bistveno vpliva na optimalne stroške iskanja cilja, če bi vnaprej poznali njegovo oddaljenost, saj so stroški takšne optimalne rešitve $O(e^{2d})$.

Spomnite se, da je konkurenčno razmerje strategije za primer neznane razdalje dejanski strošek iskanja cilja deljen z optimalnim stroškom iskanja cilja, če bi vnaprej poznali njegovo razdaljo. "Konkurenčno razmerje iteracije" imenujemo konkurenčno razmerje, če je cilj najden v tej iteraciji. Dejanski stroški iskanja cilja v iteraciji so vsota stroškov vseh iteracij do vključno te iteracije. Ker je treba cilj najti za območjem predzadnje iteracije, je optimalni strošek minimalen in s tem konkurenčno razmerje iteracije maksimalno, če je cilj najden tik za območjem predzadnje iteracije, zato uporabimo strošek te iteracije kot strošek optimalne rešitve. S tem dobimo tudi zgornjo mejo konkurenčnega razmerja. Tako je najslabše konkurenčno razmerje iteracije vsota stroškov vseh iteracij do vključno te iteracije, deljena s stroški predzadnja ponovitev. Najslabše konkurenčno razmerje je največje konkurenčno razmerje vseh iteracij v neskončnem zaporedju iteracij.

Katero zaporedje ciljnih razdalj bo v najslabšem primeru zmanjšalo konkurenčno razmerje? Našo analizo bistveno poenostavimo, če stroškov iteracij ne predstavimo z njihovimi dejanskimi stroški, temveč z zaporedjem množiteljev ali razmerij stroškov dveh zaporednih iteracij, kot sledi: Naj bo m_1 strošek prve iteracije. Naj bo m_2 strošek druge iteracije, deljen s stroškom prve iteracije, tako da je strošek druge iteracije $m\,m_{12}$. Podobno je strošek tretje ponovitve $m\,m\,m_{123}$, itd.

Konkurenčno razmerje n-*te* iteracije je vsota stroškov vseh n iteracij, deljena s stroški $(n-1)$-*te* iteracije, ali

$$\frac{m_1 + m\,m_{12} + m\,m\,m_{123} + \cdots + m\,m_{12}\cdots m_{n-1} + m\,m_{12}\cdots m_n}{m\,m_{12}\cdots m_{n-1}}$$

$$= \frac{1}{m\,m_{23}\cdots m_{n-1}} + \frac{1}{m\,m_{34}\cdots m_{n-1}} + \cdots + \frac{1}{\mathrm{mn\text{-}2mn\text{-}1}} + \frac{1}{\mathrm{mn\text{-}1}} + 1 + m_n .$$

Kako izračunati konkurenčno razmerje prve iteracije, bomo obravnavali na koncu tega poglavja pod naslovom "Optimalne začetne iteracije".

Lemma 6. *Obstaja optimalna iteracijska strategija, pri kateri so konkurenčna razmerja, povezana z iskanjem cilja v vsaki iteraciji, enaka.*

Dokaz. Dano zaporedje iteracij bomo preoblikovali v zaporedje z enakimi kompetitivnimi razmerji, ne da bi povečali najslabše razmerje. Če je konkurenčno razmerje prve iteracije manjše od najslabšega konkurenčnega razmerja celotnega zaporedja, potem povečujemo m_1 , dokler se njegovo konkurenčno razmerje ne izenači s to vrednostjo. Ostali množitelji ostanejo nespremenjeni, učinek tega pa je povečanje stroškov vseh naslednjih iteracij, vendar ne njihovih konkurenčnih razmerij, ki so neodvisna od m_1 . Nato za vsako iteracijo n, katere konkurenčno razmerje je manjše od konkurenčnega razmerja prve iteracije, povečamo m_n , dokler njeno konkurenčno razmerje ni enako konkurenčnemu razmerju prve iteracije. Povečanje m_n ne bo vplivalo na konkurenčno razmerje nobene prejšnje iteracije, povečalo bo konkurenčno razmerje n-te iteracije in zmanjšalo konkurenčno razmerje vsake naslednje iteracije, saj se m_n pojavi le v imenovalcih konkurenčnih razmerij teh iteracij. S ponavljanjem tega postopka za vsako iteracijo dobimo novo strategijo ■ vsemi konkurenčnimi razmerji, ki so enaka največjemu konkurenčnemu razmerju prvotne strategije.

Pri strategiji z neenakimi konkurenčnimi razmerji lahko dejansko zmanjšamo njihovo maksimalno vrednost, če jih vse izenačimo, vendar tega močnejšega rezultata ne potrebujemo. Zgoraj smo predpostavili, da so stroški iteracij realna števila, v resnici pa so cela števila, saj se vsako potovanje začne s polno galono goriva. Ker stroški goriva naraščajo eksponentno z razdaljo, relativna razlika med realnimi vrednostmi in najbližjimi celimi števili s povečevanjem iteracijskih razdalj pade na nič.

Lemma 7. *V vsaki iterativni strategiji, v kateri so vsa konkurenčna razmerja enaka, tvorijo množitelji, ki se začnejo z* m_2 *, padajoče zaporedje.*

Dokaz. Dokaz je z indukcijo glede na dolžino zaporedja.

Osnovni primer: Najprej pokažemo, da je $m_3 < m_2$.

je $\frac{m_1+\mathrm{m1m2}}{m_1} = 1 + m_2$, konkurenčno razmerje tretje iteracije pa je

$\frac{\mathrm{m1+m1m2+m1m2m3}}{\mathrm{m1m2}} = \frac{1}{m_2} + 1 + m_3$. Če sta ti razmerji enaki, dobimo m1m2 m2

$$1 + m_2 = \frac{1}{m_2} + 1 + m_3 \implies m_2 = \frac{1}{m_2} + m_3 \implies m_3 < m_2 .$$

Stopnja indukcije: Predpostavimo, da je za vse m med m_2 in vključno m_n $m_i < m_{i-1}$. Dokazati želimo, da je $m_{n+1} < m_n$. Konkurenčno razmerje n-te iteracije je

$$\frac{1}{m_2 m_3 \cdots m_{n-1}} + \frac{1}{m_3 m_4 \cdots m_{n-1}} + \cdots + \frac{1}{m_{n-2}m_{n-1}} + \frac{1}{m_{n-1}} + 1 + m_n .$$

Podobno je konkurenčno razmerje $(n+1)$ prve iteracije

$$\frac{1}{m_2 m_3 \cdots m_n} + \frac{1}{m_3 m_4 \cdots m_n} + \cdots + \frac{1}{m_{n-1}m_n} + \frac{1}{m_n} + 1 + m_{n+1} .$$

Če te vrednosti med seboj izenačimo in od obeh strani odštejemo 1, dobimo

$$\frac{1}{m_2 m_3 \cdots m_{n-1}} + \frac{1}{m_3 m_4 \cdots m_{n-1}} + \cdots + \frac{1}{m_{n-2}m_{n-1}} + \frac{1}{m_{n-1}} + m_n$$
$$= \frac{1}{m_2 m_3 \cdots m_n} + \frac{1}{m_3 m_4 \cdots m_n} + \cdots + \frac{1}{m_{n-1}m_n} + \frac{1}{m_n} + m_{n+1} .$$

Z indukcijsko hipotezo vemo, da $m_n < m_{n-1}$, zato $\frac{1}{m_n} > \frac{1}{m_{n-1}}$, in lahko prepišemo $\frac{1}{m_n}$ kot $\frac{1}{m_{n-1}} + x_{n-1}$, za nek pozitiven x_{n-1}. Podobno, ker $m_{n-1} < m_{n-2}$ in $m_n < m_{n-1}$, imamo $m_{n-1} m_n < m_{n-2} m_{n-1}$ in $\frac{1}{m_{n-1}m_n} > \frac{1}{m_{n-2}m_{n-1}}$, in lahko prepisati $\frac{1}{m_{n-1}m_n}$ kot $\frac{1}{m_{n-2}m_{n-1}} + x_{n-2}$ za nekaj pozitivnih x_{n-2}. Na splošno lahko prepišemo $\frac{1}{m_i m_{i+1} \cdots m_n}$ kot $\frac{1}{m_{i-1} m_i \cdots m_{n-1}} + x_{i-1}$ za nekaj pozitivnih x_{i-1}. Tako lahko zgornjo enačbo prepišemo enačba kot

$$\frac{1}{m_2 m_3 \cdots m_{n-1}} + \frac{1}{m_3 m_4 \cdots m_{n-1}} + \cdots + \frac{1}{m_{n-1}} + m_n$$
$$= \frac{1}{m_2 m_3 \cdots m_n} + \frac{1}{m_2 m_3 \cdots m_{n-1}} + x_2 + \cdots + \frac{1}{m_{n-1}} + x_{n-1} + m_{n+1} .$$

Če od obeh strani enačbe odštejemo skupne člene, dobimo

$$m_n = \frac{1}{m_2 m_3 \cdots m_n} + x_2 + x_3 + \cdots + x_{n-1} + m_{n+1} \Longrightarrow m_{n+1} < m_n$$

kar smo skušali dokazati. Tako imamo z indukcijo za vse $i \geq 2$ $m_{i+1} < m_i$, zaporedje množiteljev pa je padajoče zaporedje. ■

Na koncu lahko navedemo in dokažemo glavni rezultat našega članka.

Izrek 8. *Optimalna strategija za primer neznane razdalje, merjena z najslabšim konkurenčnim razmerjem, je iterativna strategija, ki podvoji stroške vsake naslednje iteracije in doseže najslabše konkurenčno razmerje štiri.*

Dokaz. Pokazali smo, da je edina optimalna alternativa fiksnemu zaporedju predalov iterativna strategija, ki uporablja optimalno strategijo znane razdalje za doseganje vsake od zaporedja vedno večjih ciljnih razdalj. Lemma 6 nam pove, da ob taki strategiji obstaja enako učinkovita strategija, pri kateri so vsa konkurenčna razmerja enaka. Lemma 7 nam pove, da v taki strategiji množitelji vsake zaporedne iteracije tvorijo padajoče zaporedje. Vsi množitelji morajo biti večji od ena, saj mora vsaka iteracija iterirati dlje kot prejšnja. Zato morajo konvergirati k meji, ki je večja ali enaka ena. Tekmovalno razmerje za določeno iteracijo lahko zapišemo kot vsoto členov, vključno z množiteljem za to iteracijo, ena, in zaporedjem členov, vsak od njih je recipročni produkt vedno daljšega zaporedja množiteljev. Obravnavamo tri primere: (1) množitelji konvergirajo k ena, njihovi zmnožki pa konvergirajo h končni vrednosti; (2) množitelji konvergirajo k ena, njihovi zmnožki pa so neomejeni; in (3) množitelji konvergirajo k vrednosti, večji od ena, zato morajo biti njihovi zmnožki neomejeni. V primeru 1 se recipročne vrednosti produktov približajo vrednosti, ki ni ničelna. Konkurenčna razmerja so vsote teh recipročnih vrednosti, in ker je lahko iteracij neskončno veliko, bo konkurenčno razmerje v najslabšem primeru neomejeno. To ni optimalno, saj lahko dosežemo končno konkurenčno razmerje. V primeru 2 ali 3 produkti naraščajo neomejeno, njihovi vzajemni koeficienti pa so enaki nič. Tako gredo izrazi, ki vključujejo zgodnje množitelje, na nič, konkurenčno razmerje pa je določeno z mejno vrednostjo m množiteljev.

Katera vrednost m daje najnižje konkurenčno razmerje? Konkurenčno razmerje je neskončna vsota $m + 1 + \frac{1}{m} + \frac{1}{m^2} + \frac{1}{m^3} + \cdots$. Ker je $\sum_{i=0}^{\infty} m^{-i} = \frac{m}{m-1}$, je konkurenčna je $m + \frac{m}{m-1} = \frac{m^2}{m-1}$. Prva izpeljanka tega izraza je $\frac{m^2-2m}{(m-1)^2}$. Če določimo to enaka nič, dobimo $m = 2$ in najslabše konkurenčno razmerje 4. ■

Tako mora vsaka naslednja iteracija podvojiti stroške prejšnje iteracije. Kolikšna je dodatna razdalja x? Ker je doseganje razdalje d stane $O(e^{2d})$, doseganje $d + x$ pa $O(e^{2(d+x)})$. Če to določimo kot dvakratni strošek prejšnje iteracije, dobimo enačbo $2e^{2d} = e^{2(d+x)}$. Reševanje te enačbe za x nam da $x = (\ln 2)/2 \approx .3465736$. Tako je ena od optimalnih rešitev za primer neznane razdalje serija iteracij, od katerih je vsaka za nekaj več kot tretjino milje daljša od prejšnjega in med iteracijami počisti vse predpomnilnike.

Optimalne začetne iteracije. Dejansko lahko izboljšamo preprosto strategijo podvajanja in še vedno dosežemo najslabše konkurenčno razmerje 4. Za izračun konkurenčnih razmerij tukaj ocenimo stroške iteracije kot skupno razdaljo do cilja in nazaj ali enakovredno količino porabljenega goriva v nasprotju s količino goriva, odstranjenega z začetka. Najdaljša prva iteracija s konkurenčnim razmerjem 4 ali manj porabi 8 galon za 1,359 milje. Najslabše razmerje se pojavi, ko je cilj najden tik za petim zavetiščem, 0,44226 milje od začetka. Druga iteracija se začne s 25 galoni za premagovanje razdalje 1,9 milje. Ker je to več kot 1/2 milje dlje kot prejšnja iteracija, ta iteracija nikoli ne doseže zadnjega predpomnilnika, s čimer prihrani nekaj več kot eno galono goriva. Ker je $(8 + 24)/8 = 4$, je konkurenčno razmerje te iteracije manjše od 4.

Pri vsaki naslednji iteraciji je v najslabšem primeru cilj najden tik za zadnjim predpomnilnika, s čimer prihranite manj kot eno galono. Na primer, skupni stroški prvih treh iteracij so skoraj $8 + 25 + 67 = 100$, cilj pa je najden tik za območjem druge iteracije, pri čemer je konkurenčno razmerje malo manjše od $(100/25) = 4$. Preglednica 1 prikazuje prvih deset takšnih optimalnih iteracij s porabljenim gorivom, prevoženo razdaljo, dodatno razdaljo nad prejšnjo iteracijo, konkurenčno razmerje in razmerje goriva glede na prejšnjo iteracijo. Z razširitvijo zaporedja se razmerje goriva približuje 2, konkurenčno razmerje pa 4. S tem prihranimo več iteracij v primerjavi s čisto strategijo podvajanja, ki se začne pri 1. Na primer, 7. iteracija v tem zaporedju gre dlje kot 12. iteracija v strategiji podvajanja, število prihranjenih iteracij pa še naprej narašča.

7. SKLEPNE UGOTOVITVE. Obravnavali smo klasični problem, ko džip prečka puščavo, ki je večja od njegovega dosega goriva, vendar razdalje ne poznamo vnaprej. Učinkovitost strategije smo ocenili z njenim konkurenčnim razmerjem v najslabšem primeru, ki je največje razmerje med dejanskimi stroški iskanja cilja in optimalnimi stroški, če bi razdaljo poznali vnaprej. Presenetljivo smo ugotovili, da nobena strategija, ki temelji na fiksnem zaporedju predalov, ne more doseči nobenega končnega konkurenčnega razmerja. Namesto tega je

Preglednica 1. Prvih 10 iteracij v optimalni rešitvi.

Iter.	Gorivo	Razdalja	Dodajte. Dist.	Comp. Razmerje	Razmerje goriva
1	8	1.358929		3.652759	
2	25	1.907979	.549051	3.987737	3.125000
3	67	2.394676	.486697	3.961064	2.680000
4	168	2.852076	.457400	3.986346	2.507463
5	404	3.289934	.437858	3.994787	2.404762
6	944	3.713936	.424002	3.997901	2.336634
7	2160	4.127655	.413719	3.999123	2.288136
8	4864	4.533467	.405812	3.999624	2.251852
9	10816	4.933022	.399555	3.999836	2.223684
10	23808	5.327507	.394485	3.999927	2.201183

optimalna rešitev je iterativna rešitev, pri kateri uporabimo optimalno strategijo znane razdalje za doseganje vsake od zaporedja večjih ciljnih razdalj. Ena od optimalnih iterativnih strategij podvoji stroške vsake zaporedne iteracije in doseže najslabše konkurenčno razmerje štiri. Dejansko lahko stroške vsake iteracije več kot podvojimo, vendar se razmerje stroškov zaporednih iteracij v mejah velikih razdalj približa dvema.

ZAHVALA. Zahvaljujem se Michaelu Christu in Jamesu Ralstonu za koristne razprave o tem problemu. Zahvaljujem se tudi dvema anonimnima recenzentoma, ki sta s svojimi pripombami močno izboljšala članek.

ODNOSI

[1] Bailey, H. (2009). Džipi v sovražni puščavi. *Coll. Math. J.* 40(3): 182-187.
[2] Dewdney, A. K. (1987). Računalniška rekreacija. *Sci. Am.* 256(6): 128-131.
[3] Fine, N. J. (1947). Problem jeepov. *Amer. Math. Monthly.* 54(1): 24-31.
[4] Franklin, J. N. (1960). Doseg flote letal. *J. Soc. Ind. Appl. Math.* 8(3): 541-548.
[5] Gale,D. (1970). The jeep once again, or jeeper by the ducat. *Amer. Math. Monthly.* 77(5): 493-501.
[6] Haurath, A., Jackson, B., Mitchem, J., Schmeichel, E. (1995). Galeov problem z džipom za vožnjo na izlet. *Amer. Math. Monthly.* 102(4): 299-309.
[7] Phipps, C. G. (1947). The jeep problem: a more general solution. *Amer. Math. Monthly.* 54(8): 458-462.
[8] Sleator, D. D., Tarjan, R. E. (1985). Amortizirana učinkovitost pravil za posodabljanje seznamov in paging. *Comm. ACM.*
28(2): 202-208.

RICHARD KORF je leta 1983 doktoriral iz računalništva na univerzi Carnegie-Mellon. Je profesor računalništva na univerzi UCLA. Običajno se ukvarja s hevrističnim iskanjem v umetni inteligenci. V prostem času se ukvarja z alpinizmom.
Oddelek za računalništvo, University of California, Los Angeles, Los Angeles, CA 90095
korf@cs.ucla.edu