CJELOBROJNO PROGRAMIRANJE TE NJEGOVA PRIMJENA U EKONOMIJI

Sveučilište Josipa Jurja Strossmayera u Osijeku

Ekonomski fakultet u Osijeku

Preddiplomski studij smjer Poduzetništvo

Marko Glegj

CJELOBROJNO PROGRAMIRANJE TE NJEGOVA PRIMJENA U EKONOMIJI

Završni rad

Osijek, 2021

Sveučilište Josipa Jurja Strossmayera u Osijeku

Ekonomski fakultet u Osijeku

Preddiplomski studij smjer Poduzetništvo

Marko Glegj

CJELOBROJNO PROGRAMIRANJE TE NJEGOVA PRIMJENA U EKONOMIJI

Završni rad

Kolegij: Kvantitativne metode za poslovno odlučivanje

JMBAG: 0010217922

e-mail: mglegj@efos.hr

Mentor: Izv.prof.dr.sc. Martina Briš Alić

Osijek, 2021

Josip Juraj Strossmayer University of Osijek

Faculty of Economics

Undergraduate Study Entrepreneurship

Marko Glegj

INTEGER PROGRAMMING AND IT'S APPLICATION IN ECONOMY

Final paper

Osijek, 2021

IZJAVA

O AKADEMSKOJ ČESTITOSTI, PRAVU PRIJENOSA INTELEKTUALNOG VLASNIŠTVA, SUGLASNOSTI ZA OBJAVU U INSTITUCIJSKIM REPOZITORIJIMA I ISTOVJETNOSTI DIGITALNE I TISKANE VERZIJE RADA

1. Kojom izjavljujem i svojim potpisom potvrđujem da je završni rad isključivo rezultat osobnoga rada koji se temelji na mojim istraživanjima i oslanja se na objavljenu literaturu. Potvrđujem poštivanje nepovredivosti autorstva te točno citiranje radova drugih autora i referiranje na njih.
2. Kojom izjavljujem da je Ekonomski fakultet u Osijeku, bez naknade u vremenski i teritorijalno neograničenom opsegu, nositelj svih prava intelektualnoga vlasništva u odnosu na navedeni rad pod licencom *Creative Commons Imenovanje – Nekomercijalno – Dijeli pod istim uvjetima 3.0 Hrvatska*.
3. Kojom izjavljujem da sam suglasan/suglasna da se trajno pohrani i objavi moj rad u institucijskom digitalnom repozitoriju Ekonomskoga fakulteta u Osijeku, repozitoriju Sveučilišta Josipa Jurja Strossmayera u Osijeku te javno dostupnom repozitoriju Nacionalne i sveučilišne knjižnice u Zagrebu (u skladu s odredbama Zakona o znanstvenoj djelatnosti i visokom obrazovanju, NN br. 123/03, 198/03, 105/04, 174/04, 02/07, 46/07, 45/09, 63/11, 94/13, 139/13, 101/14, 60/15).
4. izjavljujem da sam autor/autorica predanog rada i da je sadržaj predane elektroničke datoteke u potpunosti istovjetan sa dovršenom tiskanom verzijom rada predanom u svrhu obrane istog.

Ime i prezime studenta/studentice: Marko Glegj

JMBAG: 0010217922

OIB: 32618077845

e-mail za kontakt: marko.glegj1997@gmail.com

Naziv studija: Preddiplomski sveučilišni studij, Smjer Poduzetništvo

Naslov rada: Cjelobrojno programiranje te njegova primjena u ekonomiji

Mentor/mentor za rad: Izv.prof.dr.sc. Martina Briš Alić

U Osijeku, 26.09.2022. godine

Potpis: Glegj

Cjelobrojno programiranje te njegova primjena u ekonomiji

SAŽETAK

Cjelobrojno programiranje koristi se sve više u poslovnom okruženju, kada je menadžmentu dostupna većina informacija, kriterija i ograničenja. Na temelju postavljenih uvjeta, cjelobrojnim programiranjem dolazi se do rezultata kao što su maksimalna dobiti ili minimalni trošak, a rezultat će biti u obliku cijelog broja, odnosno zanemaruju se decimalni brojevi. Cjelobrojno programiranje dio je većeg matematičkog programiranja i temelji se na uvjetima linearnog programiranja. Za rješavanje kompleksnih jednadžbi cjelobrojnog programiranja, potreban su alati Microsoft Excel programa bez kojeg se do funkcije cilja i konačna rješenja dolazi puno teže. Ovaj program je u stanju riješiti na tisuće optimalnih rješenja odjednom i pronaći ono idealni scenarij. Između ostalog, cjelobrojno programiranje unutar Excel alata koristi se za rješavanje različitih problema kao što su planiranje proizvodnje, logistički problemi, transportni problem i još mnogo drugih. Od kojih će nekoliko biti prikazano u daljnjem tekstu ovog završnog rada.

Ključne riječi: cjelobrojno programiranje, linearno programiranje, Excel, problemi cjelobrojnog programiranja

Integer programming and it's application in economy

ABSTRACT

Integer programming is increasingly used in the business environment when most information, criteria and constraints are available to management. Based on the set conditions, integer programming leads to results such as maximum profit or minimum cost, and the result will be in the form of an integer, decimal numbers are ignored. Integer programming is part of a larger mathematical programming and is based on linear programming conditions. To solve complex equations of integer programming, the tools of Microsoft Excel are needed, without which the goal function and the final solution are much harder to come by. This program is able to solve thousands of optimal solutions at once and find the ideal scenario. Among other things, integer programming within Excel tools is used to solve lots of different problems like planing production, logistical problems, transport problems and many others. From which some of them will be shown in the following text of this final paper.

Keywords: integer programming, linear programming, Excel, integer programing problems

SADRŽAJ

1. UVOD

Cjelobrojno programiranje je matematičko programiranje koje se smatra podvrstom linearnog programiranja, a temelji se na zaokruživanju rezultata i rješenja na cijeli broj. Kada god je to moguće, ako se radi o stvarima koje nisu djeljive, kao što su strojevi, tada se koristi cjelobrojno programiranje jer je moguće dobiti rezultat izražen u cijelom broju. Cilj koji se pokušava postići rješavanjem problema pomoću cjelobrojnog programiranja je ostvariti maksimalne ili minimalne rezultate. Pa se tako pomoću cjelobrojnog programiranja može odrediti maksimalan profit, ili pak minimalne troškove.

Cjelobrojno programiranje, također poznato i pod nazivom optimizacije, način je modeliranja vrlo širokog niza problema. Takvi problemi koji se mogu riješiti pomoću ovog programa javljaju se u mnogim područjima koja će biti i spomenuta u sklopu ovoga rada u poglavlju koje će se orijentirati na prikazivanje korištenja programiranja u praksi. Jedan od najčešćih modela koji se koriste za pronalazak optimiziranog rješenja je linearno programiranje. Cilj rada je prikazati kako se koristi model cjelobrojnog programiranja u matematičkom modelu, a zatim prikazati na konkretnim primjerima način pronalaska rješenja.

U posljednjem desetljeću, upotreba modela i softvera za cjelobrojno programiranje podataka dramatično se povećala. Do tog perioda, računala su mogla izračunati 50 do 100 cjelobrojnih varijabli. Danas je osobno računalo sposobno riješiti problem s tisućama cjelobrojnih varijabli, dobiti visokokvalitetno rješenje za svega nekoliko minuta.

1.1. Predmet istraživanja

Kroz ovaj završni rad obradit ćemo cjelobrojno programiranje kao podvrstu linearnog programiranja, razlika je u uklanjanju decimalnih brojeva i dobijanja rezultata u cijelom broju. Unutar rada prikazati će se teorijska podloga cjelobrojnog programiranja, u koju svrhu se koristi i kako može pomoći poduzećima.

1.2. Svrha i cilj istraživanja

Metoda za izračunavanje problema cjelobrojnog programiranja ima nekolicina, i većinom se koristi kod težih slučajeva, a u rijeđima za lake. U konkretnom poslovnom svijetu koristi se za minimiziranje troškova, maksimiziranje dobiti, a većinom je to u transportnim industrijama. Prema tome, unutar ovog rada prikazati će se na konkretnom primjeru kako se pomoću cjelobrojnog programiranja može utjecati na ostvarenje rezultata u transportnom problemu, Problemu asignacije i problemu raspoređivanja zaposlenika po smjenama. Ovi problemi rješavaju se u MS Excel programskom alatu čiji će princip rada i objašnjenje dobivenih rezultata biti objašnjeni u daljnjem tekstu.

1.3. Izvor podataka i metode prikupljanja

Prilikom pisanja završnog rada korišteno je više izvora literature od kojih je većina bila iz elektronske baze podataka, internet stranica, a nešto manje iz znanstvene i stručne literature. Prema tome, može se zaključiti kako tema cjelobrojnog programiranja nije toliko često obrađivana u znanstvenim publikacijama. Tijekom prikupljanja i pisanja rada najviše su korištene metoda dedukcije, povijesna metoda, metoda deskripcije, statistička metoda, te metoda usporedbe.

1.4. Sadržaj i struktura rada

Završni rad se dijeli na 5 poglavlja. Prvo poglavlje sadrži predmet i cilj istraživanja, također su navedene metode istraživanja i prikupljanja literature. Zatim drugo poglavlje teorijski obrađuje tema cjelobrojnog programiranja, prikazuje povezanost cjelobrojnog programiranja s linearnim programiranjem. Treće poglavlje navodi kojim se metodama mogu izračunati problemi cjelobrojnog programiranja i nabraja nekoliko osnovnih metoda i modela. Četvrto poglavlje je istraživačko poglavlje gdje je prikazan problem transporta, problem asignacije te problem raspodjele radnika. Posljednje poglavlje je zaključak, zatim sva korištena literatura i na kraju popis tablica i grafova.

2. POJAM CJELOBROJNOG PROGRAMIRANJA

2.1.Definiranje pojma cjelobrojnog programiranja

Kao što je već rečeno u uvodu, cjelobrojno programiranje je dio većeg matematičkog programiranja, i to je linearno programiranje. ˝Linearno programiranje je optimizacijska metoda kojom se određuje optimalna vrijednost (minimum ili maksimum) linearne funkcije cilja s određenim brojem strukturnih varijabli x1, x2,…,xn međusobno povezanih linearnim vezama, tj. ograničenjima u obliku linearnih jednadžbi ili nejednadžbi.˝[1]

Cjelobrojno programiranje uspješno je područje optimizacije. Ima nebrojne primjene u planiranju proizvodnje, rasporeda, logistici, upravljanju prihodima i mnogim drugim područjima. Zahvaljujući učinkovitom i pouzdanom softveru, široko se primjenjuje u industriji za poboljšanje donošenja odluka.

Prema Ardalu (2005) Cjelobrojno matematičko programiranje rabi se kada su funkcije cilja i funkcije ograničenja cjelobrojne varijable. Počinje se koristiti 1957. godine, na njima su radili Markowitz i Manne. Iako su radili na rješavanju problema koji će dati rezultate, zapravo nikako nisu stvorili jednu ustaljenu metodu dolaska do rezultata. Godinu dana kasnije, Gomory kreira metodu cjelobrojnog programiranja, kao standardnog solvera.

Modeli cjelobrojnog programiranja pojavljuju se u praktički svakom području primjene matematičkog programiranja. Kako bismo razvili preliminarnu procjenu važnosti ovih modela, u ovom odjeljku uvodimo tri područja u kojima je cjelobrojno programiranje igralo važnu ulogu u podržavanju menadžerskih odluka. Ne dajemo najsloženije dostupne formulacije za svaki slučaj, već dajemo osnovne modele i predlažemo moguća proširenja.

Cjelobrojno programiranje izražava optimizaciju linearne funkcije koja podliježe skupu linearnih ograničenja nad cjelobrojnim varijablama.

Primjer planiranja proizvodnje su svi modeli linearnog programiranja. Međutim, linearni programi s vrlo velikim brojem varijabli i ograničenja mogu se učinkovito riješiti. Nažalost, to više ne vrijedi kada se od varijabli traži da imaju cjelobrojne vrijednosti. Cjelobrojno programiranje je klasa problema koji se mogu izraziti kao optimizacija linearne funkcije koja podliježe skupu linearnih ograničenja nad cjelobrojnim varijablama. Zapravo je NP-tvrd.

[1] Linearno programiranje, https://sites.google.com/site/linearnoprogramiranje1/literatura, (02.07.2021.)

Možda je još važnija činjenica da su cjelobrojni programi koji se mogu riješiti do dokazane optimalnosti u razumnom vremenu mnogo manji po veličini od svojih pandana linearnog programiranja. Postoje iznimke, naravno, i ova dokumentacija opisuje nekoliko važnih klasa cjelobrojnih programa koji se mogu učinkovito riješiti, ali korisnike OPL-a treba upozoriti da je diskretne probleme općenito puno teže riješiti nego linearne programe.

U tipičnom problemu proračuna kapitala, odluke uključuju odabir niza potencijalnih ulaganja. Odluke o ulaganju mogu biti izbor između mogućih lokacija postrojenja, odabir konfiguracije kapitalne opreme ili odabir niza istraživačko-razvojnih projekata.

U mnogim primjenama ograničenja integralnosti odražavaju prirodnu nedjeljivost problema koji se proučava. Na primjer, kada se odlučuje koliko će nuklearnih nosača zrakoplova imati u američkoj mornarici, rješenja s razlomcima očito su besmislena, budući da je optimalan broj reda veličine jedan ili dva.

U tim situacijama, varijable odluke su inherentno integralne po prirodi problema odlučivanja. To nije nužno slučaj u svakoj aplikaciji za cjelobrojno programiranje, kao što je prikazano modelima proračuna kapitala i lokacije skladišta iz posljednjeg odjeljka.

„U ovim modelima, cjelobrojne varijable proizlaze iz (i) logičkih uvjeta, kao što je ako se razvije novi proizvod, tada se mora izgraditi novo postrojenje, i iz (ii) nelinearnosti kao što su fiksni troškovi za otvaranje skladišta. Razmatranja ove prirode toliko su važna za modeliranje da ovaj odjeljak posvećujemo analizi i konsolidaciji specifičnih tehnika formulacije cjelobrojnog programiranja, koje se mogu koristiti kao alati za širok raspon primjena (Ardal, 2005).“

Cjelobrojni linearni program je linearni program dodatno ograničen ograničenjima integralnosti. Prema tome, u problemu maksimizacije, vrijednost funkcije cilja, na optimumu linearnog programa, uvijek će biti gornja granica cilja optimalnog cjelobrojnog programiranja. Nadalje, svaka cjelobrojna izvediva točka uvijek je donja granica optimalne vrijednosti cilja linearnog programa.

Ideja grananja i povezivanja je korištenje ovih opažanja za sustavnu podjelu izvedive regije linearnog programiranja i izradu procjena problema cjelobrojnog programiranja na temelju tih podjela.

2.2. Problemi koje rješava metoda cjelobrojnog programiranja

Svaka organizacija svakodnevno razmišlja o načinu kako smanjiti troškove ili povećati prihode, a vrlo često to rade nesvjesno, obzirom da je to cijeli cilj poslovanja, stvaranje profita. Kako bi organizaciji bilo jednostavnije donijeti odluke koje se provode svakodnevno, a to može biti od samog rasporeda zaposlenika, korištenja prostora, proizvodnje i raspodjela oskudnih resursa po odjelima, u jednom trenutku će morati postaviti plan kojim će optimizirati ove radnje. Na sve se to može utjecati kroz unošenje problema u cjelobrojno programiranje.

Neki problemi koji se mogu formulirati kao problemi s cjelobrojnim programiranjem:

1) ″metode optimiziranja problema fiksnih troškova,
2) metode optimiziranja problema lokacije,
3) metode optimiziranja problema trgovačkog putnika,
4) metode optimiziranja problema ranca (upotrebljavaju se u optimizaciji kapaciteta prijevoznih sredstva – npr. brodova, zrakoplova, vagona, kamiona i slično),
5) metode optimiziranja investicijskih pothvata (elaborata),
6) metode optimiziranja problema grananja i ograđivanja,
7) metode „opreznog približavanja“,
8) metode cjelobrojnih formi.″ (Zelenika, 2016)

Algoritmi rješavanja problema cjelobrojnog programiranja:

1. Metoda odsijecanja ravnina,
2. Metode stabla odlučivanja,
3. Heurističke metode. (Milić, Kvesić, 2010).

Cjelobrojno programiranje obično se odnosi na cjelobrojno linearno programiranje koje je paradigma matematičkog modeliranja i rješenja. Odluke su modelirane kao vektor realnih brojeva, od kojih su neki dodatno ograničeni da uzimaju samo cjelobrojne vrijednosti.

Prema Barkoviću (2002) vektor odluke je ograničen da zadovolji sustav linearnih nejednadžbi. Treba minimizirati jednu funkciju cilja koja je opet linearna u vektoru odluke. Vrlo često su određene varijable odluke ograničene da uzimaju vrijednosti {0,1} za modeliranje logičkih ograničenja.

Modeli optimizacije linearnog cjelobrojnog programiranja rješavaju se iskorištavanjem prednosti nižih granica pronađenih rješavanjem problema linearnog programiranja u algoritmima grananja i vezanja i grananja i rezanja.

Opet su definirane varijable odluke i svaka je specificirana na domeni; domene koje se koriste u praksi slične su onima koje se koriste u modelima cjelobrojnog programiranja. Skup ograničenja definiran je na varijablama odluke i ta ograničenja mogu biti općenitija od onih koja se koriste u cjelobrojnom programiranju kako bi se omogućilo izravno modeliranje logičkih ograničenja.

„Primarni problem programiranja ograničenja je pronaći vektor odluke koji zadovoljava sva ograničenja. Za identifikaciju takvih rješenja (ako postoje) koriste se metode propagacije ograničenja. Neki alati za rješavanje programiranja ograničenja također dopuštaju specificiranje ciljnih funkcija, a nakon identificiranja izvedivih rješenja traže ona s boljim vrijednostima ciljnih funkcija (Kumar, 2010).“

Za mnoge važne probleme formulacija prirodnog cjelobrojnog programiranja može biti izvrstan model i korisna za pronalaženje optimalnih ili gotovo optimalnih rješenja. U drugim slučajevima, općenitost koju pruža modeliranje programiranja ograničenja može biti bolji izbor i može identificirati rješenja visoke kvalitete brže od ekvivalentnog IP modela.

Znati kada odabrati koji alat za određeni problem je inženjerska vještina, a pravi izbor može se promijeniti tijekom vremena kako se oba polja nastavljaju razvijati.

2.3.Područja primjene cjelobrojnog programiranja

Ova se razmatranja često pojavljuju u praksi i tako se linearno cjelobrojno programiranje može koristiti u mnogim područjima primjene, od kojih su neka ukratko opisana u nastavku.

Planiranje proizvodnje

„Jedan važan primjer koji se događa u planiranju poljoprivredne proizvodnje uključuje određivanje proizvodnog prinosa za nekoliko usjeva koji mogu dijeliti resurse (npr. zemljište, rad, kapital, sjeme, gnojivo, itd.). Mogući cilj je maksimiziranje ukupne proizvodnje, bez prekoračenja raspoloživih resursa (Kumar, 2010).“

U nekim slučajevima, to se može izraziti u terminima linearnog programa, ali varijable moraju biti ograničene na cijeli broj.

Zakazivanje rasporeda

„Ovi problemi uključuju usluge i raspoređivanje vozila u prometnim mrežama. Na primjer, problem može uključivati dodjelu autobusa ili podzemne željeznice pojedinačnim rutama kako bi se mogao ispuniti vozni red, a također ih opremiti vozačima. Ovdje binarne varijable odluke pokazuju je li autobus ili podzemna željeznica dodijeljen ruti i je li vozač dodijeljen određenom vlaku ili podzemnoj željeznici (Schultz i suradnici, 2006).“

Tehnika programiranja nula-jedan uspješno je primijenjena za rješavanje problema odabira projekata u kojima su projekti međusobno isključivi i/ili tehnološki međuovisni. Koristi se u posebnom slučaju cjelobrojnog programiranja, u kojem su sve varijable odluke cijeli brojevi. Može poprimiti vrijednosti kao nulu ili kao jedinicu.

Teritorijalna podjela

„Problem teritorijalne podjele ili distrikta sastoji se od podjele geografske regije na distrikte kako bi se planirale neke operacije uzimajući u obzir različite kriterije ili ograničenja. Neki od zahtjeva za ovaj problem su: susjedstvo, kompaktnost, ravnoteža ili jednakost, poštivanje prirodnih granica i socioekonomska homogenost (Boland i suradnici, 2006).“

Neke primjene za ovu vrstu problema uključuju: političko distriktiranje, distriktiranje škola, distriktiranje zdravstvenih usluga i distriktiranje gospodarenja otpadom.

Telekomunikacijske mreže

„Cilj ovih problema je dizajnirati mrežu linija za instaliranje tako da se ispuni unaprijed definirani skup komunikacijskih zahtjeva i da ukupni trošak mreže bude minimalan. To zahtijeva optimizaciju topologije mreže zajedno s postavljanjem kapaciteta različitih linija (Ardal, 2005).“

U mnogim slučajevima, kapaciteti su ograničeni na cjelobrojne količine. Obično postoje, ovisno o tehnologiji koja se koristi, dodatna ograničenja koja se mogu modelirati kao linearne nejednadžbe s cjelobrojnim ili binarnim varijablama.

Mobilne mreže

„Zadatak planiranja frekvencije u GSM mobilnim mrežama uključuje distribuciju dostupnih frekvencija preko antena tako da korisnici mogu biti opsluženi i da se interferencija između antena svede na najmanju moguću mjeru (Prabodanie, 2017).“

Ovaj problem se može formulirati kao cjelobrojni linearni program u kojem binarne varijable pokazuju je li frekvencija dodijeljena anteni.

2.4. Važnost cjelobrojnog programiranja

Cjelobrojno programiranje je grana matematičkog programiranja ili optimizacije koja uključuje stvaranje jednadžbi radi rješavanja problema. Pojam ″matematičkog programiranja″ odnosi se na činjenicu da je cilj rješavanja različitih problema povezan s pravilnim odabirom programa djelovanja. Čak i dodjela jednostavne odluke ″da ili ne″ može biti moćan način uspostavljanja linearnog okvira za rješavanje problema kako bi se identificirala učinkovitost.

Cjelobrojne probleme programiranja obično je puno teže riješiti nego linearne probleme programiranja. Isto tako, za razumijevanje rezultata linearnog programiranja postoje algoritmi poput simplex algoritma koji teorijski iščitava rezultate i pomaže u razumijevanju rezultata i rješavanju problema. Ovo nije situacija i kod cjelobrojnog programiranja, gdje se ljudi specilijaliziraju za razumjevanje modela.

Glavna područja u kojima se cjelobrojno programiranje koristi u praksi uključuje:

- Nametanje logičkih uvjeta u problemima linearnog programiranja,
- Gdje problem ne zna sve uvjete,
- Odabir skladišta,
- Određivanje radnih smjena zaposlenih,
- Balansiranje proizvodne linije,
- Raspored avionskih letova i rada zaposlenih,
- Vozni red, itd.[2]

Postoje trenuci kada moramo razmotriti problem optimizacije, posebno u slučajevima kada frakcijska rješenja nisu realna. Takav problem optimizacije naziva se problem cjelobrojnog

[2]Integer programming formulation examples, http://people.brunel.ac.uk/~mastjjb/jeb/or/moreip.html, (05.07.2021.)

programiranja. Postoje dvije osnovne kategorije cjelobrojnih programa, naime mješoviti cjelobrojni program i čisti cjelobrojni program.

Prema Kumaru i suradnicima (2010) kaže se da je problem mješoviti cjelobrojni program kada su neke od uključenih varijabli ograničene na cijeli brojevi i da je čisti cjelobrojni program ako sve varijable odluke moraju biti cijeli brojevi. U slučajevima kada su ograničenja mrežne prirode, tada se cjelobrojno rješenje može postići previđanjem granica integralnosti i rješavanjem rezultirajućeg linearnog programa.

„Općenito, međutim, varijable su frakcijske u rješenju linearnog programiranja i moraju se poduzeti dodatne mjere za definiranje razlučivosti cjelobrojnog programiranja. Također je vrijedno napomenuti da integer posebno traži rješenja temeljena na cijelim brojevima za optimizaciju problema (Prabodanie, 2017).“

S druge strane, linearno programiranje uglavnom traži najbolje vrijednosti za poboljšanje skupa uvjeta. Međutim, to je nemoguće uglavnom kada se radi o diskretnim vrijednostima.

Algoritmi za cjelobrojno programiranje minimiziraju ili maksimiziraju funkciju podložnu ograničenjima jednakosti, nejednakosti i cjelobrojnih ograničenja.

„Cjelobrojna ograničenja ograničavaju neke ili sve varijable u problemu optimizacije na samo cjelobrojne vrijednosti. To omogućuje precizno modeliranje problema koji uključuju diskretne količine (kao što su udjeli dionica ili ćelije u bateriji) ili odluke da ili ne. Kada postoje cjelobrojna ograničenja samo na nekim varijablama, problem se naziva program miješanog cijelog broja (MIP) (Crnjac-Milić i Kvesić, 2008).“

Primjeri problema cjelobrojnog programiranja uključuju optimizaciju portfelja u financijama, optimalno otpremanje proizvodnih jedinica (predanost jedinica) u proizvodnji energije, optimizaciju dizajna u inženjeringu te planiranje i rutiranje u aplikacijama transporta i opskrbnog lanca.

Tijekom posljednjih 20 godina, kombinacija bržih računala, pouzdanijih podataka i poboljšanih algoritama rezultirala je lakim rješenjima mnogih cjelobrojnih programa od praktičnog interesa. Modeli cjelobrojnog programiranja koriste se u širokom spektru aplikacija, uključujući planiranje, dodjelu resursa, planiranje proizvodnje, dizajn opskrbnog lanca, dizajn dražbi, telekomunikacijske mreže, mobilne mreže i mnoge druge.

Problem cjelobrojnog programiranja je matematička optimizacija ili program izvedivosti u kojem su neke ili sve varijable ograničene da budu cijeli brojevi. Termin se također odnosi na

Integer Linear Programming (ILP), u kojem su funkcija cilja i ograničenja linearni (Čerić, 2009).

Prema Masonu (2011) dva su glavna razloga za korištenje cjelobrojnih varijabli pri modeliranju problema kao linearnog programa:

- Cjelobrojne varijable predstavljaju veličine koje mogu biti samo cjelobrojne. Na primjer, nije moguće napraviti 3,7 automobila.
- Cjelobrojne varijable predstavljaju odluke i stoga bi trebale imati samo vrijednost 0 ili 1.

Cjelobrojno programiranje dodaje dodatna ograničenja linearnom programiranju. Cjelobrojni program počinje s linearnim programom i dodaje zahtjev da neke ili sve varijable imaju cjelobrojne vrijednosti.

2.5. Linearno programiranje

„U osnovi, linearno programiranje bavi se minimalizacijom ili maksimizacijom linearnih funkcija na temelju linearnih ograničenja. Neka područja u kojima je primijenjeno programiranje obloga uključuju probleme miješanja, mješavine proizvoda i probleme prehrane. Kemijska industrija, rafinerije nafte, prehrambena industrija i industrija čelika samo su neki od primjera industrija koje su zabilježile značajan uspjeh u korištenju linearnog programiranja. Problemi linerskog programiranja koji uključuju samo dvije varijable mogu se uspješno riješiti korištenjem grafičke metode koja nudi slikovni prikaz rješenja (Schultz i suradnici, 1996).“

„Temelj velikog dijela analitičkog odlučivanja je linearno programiranje. U linearnom programu postoje varijable, ograničenja i funkcija cilja. Varijable, odnosno odluke, poprimaju numeričke vrijednosti. Ograničenja se koriste za ograničavanje vrijednosti na izvedivo područje. Ova ograničenja moraju biti linearna u varijablama odluke. Funkcija cilja tada definira koja je posebna dodjela mogućih vrijednosti varijablama optimalna: to je ona koja maksimizira (ili minimizira, ovisno o vrsti cilja) funkciju cilja. Funkcija cilja također mora biti linearna u varijablama (Lukač i Neralić, 2012).“

Linearno programiranje može se primijeniti samo na probleme u kojima su funkcija cilja i ograničenja linearne prirode, tj. gdje se mogu izraziti u obliku jednadžbi koje simboliziraju ravne linije. U stvarnim situacijama, kada su funkcije cilja ili ograničenja nelinearne, linearno programiranje nikada se ne može koristiti.

Prvi ju je koristi Leonid Kantarović u kasnim 1930-im godinama kao dio rješenja problema proizvodnje. Kao i sve logističke metode, i linearno programiranje najviše se razvija i koristi tijekom II. svjetskog rata zbog raspoređivanja vojne logistike, opreme i vojnika. Kasnije, Kantaroviću se u istraživanju pridružuje Tjallinga Koompansa, te zajedničkim radom razvijaju linearno programiranje, za što su i 1975. godine dobili i Nobelovu nagradu za ekonomiju.

Linearno programiranje je grana zrelog inoperacijskog istraživanja. Maksimalni ili minimalni problemi u funkciji linearnog cilja pod linearnim uvjetom zajednički se nazivaju problemom linearnog programiranja. Rješenja za izvedivo su rješenja linearnih ograničenja, skup sastavljen od svih izvedivih rješenja naziva se izvedivo područje. Varijable odluke, funkcija cilja i uvjeti ograničenja tri su elementa problema linearnog programiranja. U problemu linearnog programiranja, optimalno rješenje može biti razlomak ili decimalni broj, ali za neke specifične probleme može uzeti samo cjelobrojne varijable, kao što je broj tvornice, zatim brojevi stroja koji radi. Stoga cjelobrojno programiranje za planiranje nazivamo problemom koji zahtijeva da dio ili sve varijable odluke budu cjelobrojne vrijednosti.

Cjelobrojno programiranje često se povezuje s linearnim programiranjem, ovisno o problemu koji se rješava, zapravo su to vrlo slični modeli programiranja Ako se na primjer radi o broju mašina ili proizvodnih programa, rezultati se mogu dobiti samo kao cjelobrojna vrijednost. Postoje i slučajevi kada se u problemu koriste relativno veliki brojevi, i u takvim situacijama se može koristiti zaokruživanje na najbližu cjelobrojnu vrijednost jer se smatra kako će pogreške zaokruživanja biti minimalne, takvo odstupanje je prihvatljivo i problem se može riješiti pomoću linearnog programiranja. No, postoje i problem ako će bilo kakvo zaokruživanje stvoriti pogreške. Prilikom zaokruživanja, ako se radi o malim rezultatima, tada je bolje ne zaokruživati.

3. METODE CJELOBROJNOG PROGRAMIRANJA

Postoji nekoliko metoda pomoću kojih je moguće dobiti rezultate, ali se problem linearnog programiranja, pa tako i cjelobrojnog linearnog programiranja, najčešće rješava pomoću Simplex metode.

3.1. Simplex metoda

″Simpleksna metoda, standardna tehnika u linearnom programiranju za rješavanje optimizacijskog problema, koji obično uključuje funkciju i nekoliko ograničenja izraženih kao nejednakosti. Nejednadžbe definiraju poligonalno područje, a rješenje je obično u jednom od vrhova. Simpleks metoda je sustavni postupak za testiranje vrhova kao mogućih rješenja..″[3]

Neki jednostavni problemi optimizacije mogu se riješiti crtanjem ograničenja na grafikonu. Međutim, ova metoda je korisna samo za sustave nejednakosti koje uključuju dvije varijable. U praksi problemi često uključuju stotine jednadžbi s tisućama varijabli, što može rezultirati astronomskim brojem ekstremnih točaka.

Dok je simpleks metoda učinkovita za rješavanje linearnih programa, ne postoji jedinstvena tehnika za rješavanje cjelobrojnih programa. Umjesto toga, razvijen je niz postupaka i čini se da izvedba bilo koje tehnike jako ovisi o problemu.

Dosadašnje metode mogu se općenito klasificirati kao slijedeći jedan od tri pristupa:

1. tehnike prebrojavanja, uključujući proceduru grananja i vezanja;
2. tehnike rezne ravnine; i
3. tehnike grupne teorije.

Osim toga, predloženo je nekoliko složenih postupaka koji kombiniraju tehnike koje koriste nekoliko ovih pristupa. U stvari, postoji trend u računalnim sustavima za cjelobrojno programiranje da uključuje niz pristupa i moguće ih sve koristi pri analizi određenog problema.

Godine 1947. George Dantzig, matematički savjetnik američkog ratnog zrakoplovstva, osmislio je simpleks metodu kako bi ograničio broj ekstremnih točaka koje je potrebno ispitati. Simpleksna metoda je jedan od najkorisnijih i najučinkovitijih algoritama ikada izumljenih, i

[3] Simplex metoda, https://sites.google.com/site/linearnoprogramiranje1/teorija-linearnog-programiranja/metode-rjesavanja-lp/simpleks-metoda, (05.07.2021.)

još uvijek je standardna metoda koja se koristi na računalima za rješavanje problema optimizacije

Koristi se za rješavanje linearnog problema, a obzirom da je cjelobrojno programiranje dio linearnog, korisno je i za cjelobrojno programiranje. Pronalazi idealno rješenje u konačnom broju koraka, od svih postavljenih koraka do pronalaska optimalnog rješenja.

Primjer.

Problem standardnog maksimuma je specijalna vrsta linearnog programa koji je jednostavan za analizu. To je linearni program kojime želimo maksimizirati funkciju cilja:

$$F = c_1x_1 + c_2x_2 + ... + c_nx_n$$

Ograničenja su u obliku:

$$a_{11}x_1 + a_{12}x_2 + ... + a_{1n}x_n \leq b_1$$
$$a_{21}x_1 + a_{22}x_2 + ... + a_{2n}x_n \leq b_2$$
$$. . . . .$$
$$a_{m1}x_1 + a_{m2}x_2 + ... + a_{mn}x_n \leq b_m$$
$$x_1, x_2, ... , x_n \geq 0$$

gdje je $b_j \geq 0$ za $j = 1, 2, ... , m$.

Kako bi se razumjelo daljnje metode programiranja, potrebno je interpretirati oznake koje će se koristiti:

- z Funkcija cilja
- X_j Varijabla odlučivanja
- a_{ij} Količina i-tog ograničenja potrebna za proizvodnju jedinice j-te varijable
- b_i Veličina ograničavajućeg faktora za i-to ograničenje
- c_j Koeficijent ili parametar funkcije cilja po jedinici j-te varijable

Opći problem matematičkog programiranja može se apstraktno iznijeti kao funkcija koja uvijek teži definirati maksimalne ili minimalne rezultate kao

$$min/\ max\ z\ (x_1, x_2, \dots , x_n)$$

uz ograničenja

$$g_i\ (x_1, x_2, \dots , x_n\ ) = b_i\ , \qquad i = 1,2, \dots , m$$

$$x_j \geq 0,\ j = 1,2,\ \dots\ ,n$$

pri čemu je x_j cijeli broj jer se problem rješava kroz cjelobrojno programiranje. Kada se problem definira na ovakav način, to znači da su ograničenja poznata i jasno postavljena, brojevi su izraženi cjelobrojno pa se i problem naziva problem čisto cjelobrojnog programiranja (eng. Pure integer linear programming problem).

Nadalje, nije nužno da svi brojevi u problemu budu izraženi cjelobrojno, ako je to slučaj, tada je riječ o problemu djelomičnog ili mješovitog cjelobrojnog programiranja (eng. Mixed integer programming problem), te se definira kao:

$$min/max\ z = \sum_{j=1}^{n} c_j x_j + \sum_{k=1}^{p} d_k \overline{x_k}$$

uz ograničenja

$$\sum_{j=1}^{n} a_{ij} x_j + \sum_{j=1}^{p} h_{ik} \overline{x_k} = b_i, \quad i = 1,2,\dots,m$$

$$x_j \geq 0, \quad j = 1,2,\dots,n \qquad x_k \geq 0, \quad k = 1,2,\dots,p$$

gdje je x_j cijeli broj. (Lukač, Nerelić, 2012:35)

3.2. Modeli cjelobrojnog programiranja

Kod cjelobrojnog programiranja može se još govoriti i o:

- binarnom programiranju gdje će vrijednosti biti 0 ili 1,
- mješovito cjelobrojno programiranje gdje su neke vrijednosti izražene cijelim brojem, a neke s decimalnim brojem,
- čisto cjelobrojno programiranje.[4]

Za bolje razumijevanje modela programiranja, prikazati će se i u matematičkom obliku. Model A prikazuje binarno programiranje gdje su vrijednosti 0 ili 1.

$$\text{Max } Z = x_1 + 2x_2$$

[4] Example of Integer Programming, http://web.pdx.edu/~stipakb/download/PA557/ReadingsPA557sec6.pdf, (07.07.2021.)

MODEL A gdje je : $2x_1 + x_2 < 2$

(x_1 će imati vrijednost 0 or I)

(x_2 će imati vrijednost 0 or I)

Model B prikazuje mješovito cjelobrojno programiranje.

Max Z = x 1 + 2x 2

MODEL B gdje je: $2x_1 + x_2 \leq 2$

(x_1 je cijeli broj)

Model C prikazuje čisto cjelobrojno programiranje.

Max z = xt + 2x2

MODEL C gdje je: $2x_1 + x_2 \leq 2$

(x_1 je cijeli broj)

(x_2 je cijeli broj)

4. PRIMJERI PRIMJENE CJELOBROJNOG PROGRAMIRANJA U EKONOMIJI

U nastavku ćemo obraditi nekoliko primjera primjene cijelobrojnog programiranja gdje ćemo rješavati neke probleme s kojima se poduzeća mogu sustresti u svom svakodnevnom poslovanju.

4.1.Transportni problem 1

Poduzeće ima dva proizvodna pogona A1 i A2, te tri skladišta W1,W2 i W3. Proizvodni pogoni mogu proizvesti 120 jedinica proizvoda tjedno. Respektivno, skladišta imaju kapacitete od 72, 72, 96, jedinica proizvoda tjedno. Cilj rješavanja ovog problema je da saznamo na koji način će se proizvodi prevesti od pogona do skladišta u što kraćem roku te uz minimalne troškove.

Vrijeme potrebno kamionima da prevezu proizvode od pogona do skladišta prikazano je u sljedećoj tablici:

Tablica 1. Vrijeme potrebno za prijevoz proizvoda od pogona do skladišta

	W_1	W_2	W_3	
A_1	180	360	120	120
A_2	240	60	90	120
	72	27	96	

Izvor: izrada autora prema vježbama s predavanja

Varijable odlučivanja:

Nakon postavljanja zadatka u tabličnom obliku, potrebno je definirati varijable odlučivanja.

Tablica 2. Varijable odlučivanja

X11	Broj jedinica proizvoda koji se prevozi iz pogona A1 do skladišta W1
X12	Broj jedinica proizvoda koji se prevozi iz pogona A1 do skladišta W2
X13	Broj jedinica proizvoda koji se prevozi iz pogona A1 do skladišta W3
X21	Broj jedinica proizvoda koji se prevozi iz pogona A2 do skladišta W1
X22	Broj jedinica proizvoda koji se prevozi iz pogona A2 do skladišta W2
X23	Broj jedinica proizvoda koji se prevozi iz pogona A2 do skladišta W3

Izvor: izrada autora prema vježbama s predavanja

Zatim je potrebno inerpretirati podatke iz tablice 1. kako bi se postavio problem koji će se zatim definirati tablično.

Tablica 3. Interpretacija problema

C_{11}	Cijena prijevoza proizvoda iz pogona A1 do skladišta W1 iznosi 180 NJ
C_{12}	Cijena prijevoza proizvoda iz pogona A1 do skladišta W2 iznosi 360 NJ
C_{13}	Cijena prijevoza proizvoda iz pogona A1 do skladišta W3 iznosi 120 NJ
C_{21}	Cijena prijevoza proizvoda iz pogona A2 do skladišta W1 iznosi 240 NJ
C_{22}	Cijena prijevoza proizvoda iz pogona A2 do skladišta W2 iznosi 60 NJ
C_{23}	Cijena prijevoza proizvoda iz pogona A2 do skladišta W3 iznosi 90 NJ

Izvor: izrada autora prema vježbama s predavanja

Sljedeći korak prilikom rješavanjam problema je prikazati transportni problem sa svim interpretiram podacima.

Tablica 4. Prikaz transportnog problema

x_{ij}	**W1**	**W2**	**W3**	a_i
A1	x_{11}	x_{12}	$x1_3$=120-x11-x12	120
A2	$x_{21}=72-x_1 1$	$x_{22}=72-x_{12}$	$x23=-24+x_{11}+x_{12}$	120
b_j	72	72	96	

Izvor: izrada autora prema vježbama s predavanja

X_{23} vrijednos se izračunala na sljedeći način:

$$x_{23}=96-x_{13}$$
$$x_{23}=96-(120-x_{11}-x_{12})$$
$$x_{23}=-24+x_{11}+x_{12}$$

Funkcija cilja:

″Funkcija cilja se određuje kao traženje minimalnog troška transporta koji se dobiva kao zbroj umnožaka jedinične cijene transporta između pojedinog ishodišta i odredišta te odgovarajuće količine proizvoda.″[5]

$$\min T = \sum_{i=1}^{m} \sum_{j=1}^{n} c_{ij} x_{ij}$$

[5] Bilješke s predavanja kolegija Kvantitativne metode za poslovno odlučivanje, (10.07.2021.)

Funkcija cilja je u svakom zadatku drugačija, a uvijek se odnosi na problem maksimiziranja dobiti ili minimaliziranja troškova. To je funkcija optimizacije dva navedena cilja uz sve postavljene uvjete i ograničenja. Postavlja se u obliku jednadžbi ili nejednadžbi. Obzirom da se u ovom primjeru govori o transportnom problemu, funkcija cilja biti će pronalazak rješenja koji ostvaruje minimalne troškove.

Funkcija cilja rješava se na način da se spajaju sve vrijednosti iz tablica 1 i 2 dok se ne dođe do konačne funkcije cilja, odnosno rješenja.
Znači spajaju se vrijednosti iz prve i druge tablice te se računaju dok se ne dođe do krajnjeg rješenja.

Tablica 5. Funkcija cilja

$minT=180x_{11}+360x_{12}+120x_{13}+240x_{21}+60x_{22}+90x2_{3}$
$minT=180x_{11}+360x_{12}+120*120-x_{11}-x_{12})+240*(72-x_{11})+60*(72-x_{12})+90*(-24+x_{11}+x_{12})$
$minT=180x_{11}+360x_{12}+14400-120x_{11}-120x_{12}+17280-240x_{11}+4320-60x_{12}-2160+90x_{11}+90x_{12}$

Izvor: izrada autora prema vježbama s predavanja

″ Ukupan broj varijabli odlučivanja xij je m x n, tj. kao umnožak broja redaka I broja stupaca. Ukupan broj ograničenja se izračunava kao zbroj broja stupaca I broja redaka, tj. iznosi m+n.″[6]

Konačna funkcija cilja glasi:

minT=	**-90**	**x_{11}**	**+**	**270**	**x_{12}**	**+**	**33840**

Definiranje ograničenja i crtanje grafa:

Definiranje ograničenja u problemu transporta postavlja se kako bi se dobila informacija koliko robe jedan kamion može prevesti od polazišta do odredišta, te koliko mu vremena treba. Na temelju izračuna vrijednosti ograničenja, dobivaju se točke za crtanje grafa.

Najprije se računaju vrijednosti ograničenja:

x11	≥	0
x12	≥	0
x13	≥	0
120-x11-x12	≥	0

[6] Bilješke s predavanja kolegija Kvantitativne metode za poslovno odlučivanje, (10.07.2021.)

-x11-x12	≥	-120
x11+x12	≤	120
x21	≥	0
72-x11	≥	0
-x11	≥	-72
x11	≤	72
x22	≥	0
72-x12	≥	0
-x12	≥	-72
x12	≤	72
x23	≥	0
-24+x11+x12	≥	0
x11+x12	≥	24

Nakon izračuna vrijednosti, u tablici 6. prikazani su konačni rezultati.

Tablica 6. Konačni rezultati ograničenja

x13	**1**	**x11**	**+**	**1**	**x12**	**≤**	120
x21	1	x11	+	0	x12	≤	**72**
x22	0	x11	+	1	x12	≤	**72**
x23	1	x11	+	1	x12	≥	**24**

Izvor: izrada autora prema vježbama s predavanja

Sljedeći korak koji se treba napraviti je ispoštovati uvjet nenegativnost, što znači da vrijednosti X_{11} i X_{12} ne mogu biti negativni brojevi. U ovom prikazanom problemu, na primjer, kamion ne može prevesti -90 komada robe, ali može prevesti 0, što znaći da skladište s takvim uvjetom ima dovoljnu količinu robe i tamo kamion neće ni voziti.

Uvjet neegativnosti postavlja:

$$X_{11} \geq 0$$

$$X_{12} \geq 0$$

S tim na umu, sada se u tablici 7. mogu izračunati vrijednosti x_{11} i x_{12} u funkciji cilja i ograničenja.

Tablica 7. Prikaz vrijednosti X_{11} i X_{12} u funkciji cilja i ograničenjima

	X11	**X12**		
trošak	-90	270	33840	min
X13	1	1	≤	120

X21	1	0	≤	72
X22	0	1	≤	72
X23	1	1	≥	24

Izvor: izrada autora prema vježbama s predavanja

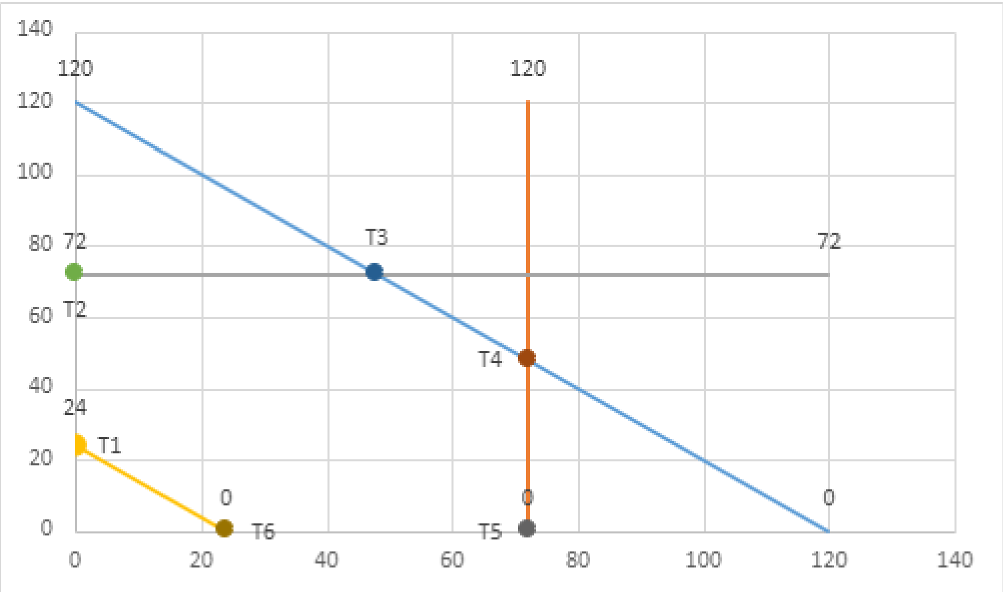
Nakon postavljanja ograničenja u tablici 7., sada se mogu izračunati vrijednosti varijabli X_{11} i X_{12}.

Tablica 8. Izračun vrijednosti varijabli X_{11} i X_{12}

	X11	**X12**	**X11**	**X12**
X13	0	120	120	0
X21	72	120	72	0
X22	0	72	120	72
X23	0	24	24	0

Izvor: izrada autora prema vježbama s predavanja

Na temelju tablice 8. sada se može praviti grafički prikaz vrijednosti.



Graf 1. Prikaz tablice 8. u grafičkom obliku.

Izvor: izrada autora prema vježbama s predavanja

Određivanje dopuštenog područja i moguće točke rješenja:

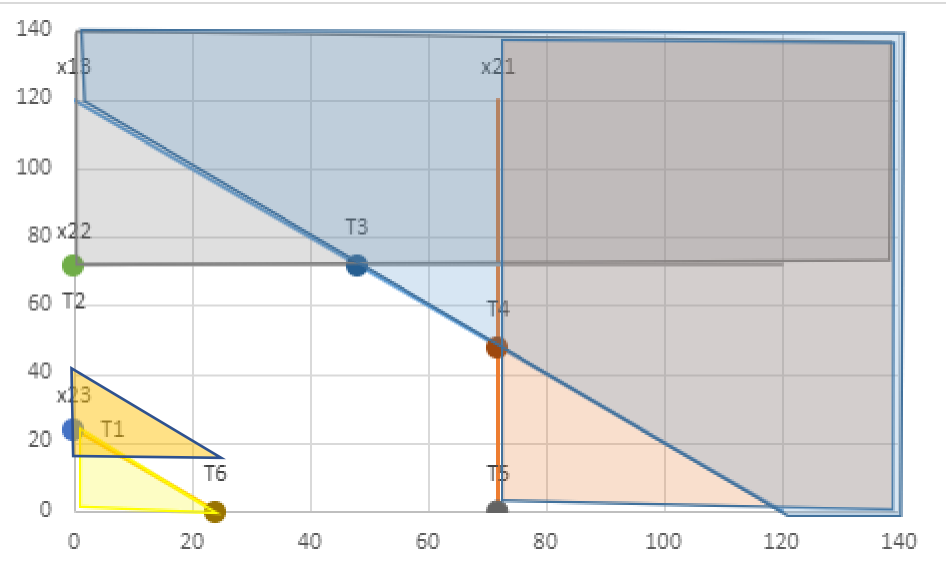
Grafom 1. prikazane su točke ograničenja, međutim, za točke dopuštenog područja, potrebno je prvo provjeriti jesu li ograničenja točna ili netočna. Rezultati mogućih točka kasnije služe za postavljanje konačnog rješenja minimalne vrijednosti funkcije cilja.

Tablica 9. Izračun dopuštenog područja

KONTROLNE TOČKE		**0**	**0**					
X13	1	X11	+	1	X12	≤	120	
			0			≤	120	TOČNO
X21	1	X11	+	0	X12	≤	72	
			0			≤	72	TOČNO
X22	0	X11	+	1	X12	≤	72	
			0			≤	72	TOČNO
X23	1	X11	+	1	X12	≥	24	
			0			≥	24	NETOČNO

Izvor: izrada autora prema vježbama s predavanja

Tablicom 9 dokazano je preklapanje lijeve i desne stranje jednadžbe čiji će se rezultati prikazati u grafu 2.



Graf 2. Preklapanje dopuštenog i nedopuštenog područja.
Izvor: izrada autora prema vježbama s predavanja

Iz grafa 2. sada se mogu iščitati točke rješenja, a to su točke koje se nalaze u području koje nije obojano.

Tablica 10. Moguće točke rješenja

T1	0	24
T2	0	72
T3	48	72
T4	72	48
T5	72	0

T6	24	0

Izvor: izrada autora prema vježbama s predavanja

Vrijednost funkcije cilja:

U ovom dijelu traži se konačno rješenje, odnosno minimalna vrijednost funkcije, a prepoznaje se nakon izračuna svih mogućig točaka rješenja iz tablice 10. koje se ubace u funkciju minimalnog troška. Pa kako bi se podsjetili, funkcija minimanlnog troška glasi:

$$minT = -90 \quad x11 \quad + \quad 270 \quad x12 \quad + \quad 33840$$

Kada se moguće točke rješenja ubace u funkciju minimalnog troška, vrijednosti funkcija su:

Tablica 11. Vrijednosti funkcije cilja uvrštene u funkciju minimalnog troška

T1	0	24	40320
T2	0	72	53280
T3	48	72	48960
T4	72	48	40320
T5	72	0	27360
T6	24	0	31680

Izvor: izrada autora prema vježbama s predavanja

Kao što je rečeno, rezultat koji prikazuje najmanju vrijednost funkcije cilja, koristi se za daljnji izračun konačne vrijednosti funkcije cilja, pa ako se pogleda tablica 11., najmanja vrijednost je .cilja minimalne vrijednosti, a dobivene vrijednosti se uvrštavaju u postavljena ograničenja. Pa je konačni rezultat na pitanje koliko robe treba transportirati iz kojeg pogona do potrebnog skladišta:

Tablica 12. Vrijednosti funkcije cilja u ostalim točkama i konačno rješenje

x_{11}	72
x_{12}	0
$x_{13}=120-x_{11}-x1_2$	48
$x_{21}=72-x_{11}$	0
$x_{22}=72-x_{12}$	72
$x_{23}=-24+x_{11}+x_{12}$	48

Izvor: izrada autora prema vježbama s predavanja

Interpretacija optimalnog rješenja glasi:

Minimalni ukupni transportni trošak i iznosu od 27360 NJ dobiva se na sljedeće načine:

- 72 tona proizvoda tjedno potrebno je prevoziti iz pogona A1 do skladišta W1 (x11 je 12)
- 48 tona proizvoda tjedno potrebno je prevoziti iz pogona A1 do skladišta W3 (x13=8)
- 72 tona proizvoda tjedno potrebno je prevoziti iz pogona A2 do skladišta W2 (x22=12)
- 48 tona proizvoda tjedno potrebno je prevoziti iz pogona A2 do skladišta W3 (x23=8)

Obzirom da su varijable x_{12} i x_{21} =0, znači da se iz pogona A1 i A2 ne treba prevoziti roba u skladišta W1 i W2.

4.2. Problem asignacije

Tvrtka Orbico u svom skladištu ima 4 zaposlenika, 3 stalno zaposlena i jednog studenta. Oni su podjeljeni na 4 različite funkcije. Viličar, Papirologija, Slaganje narudžbi i zaprimanje robe. Na temelju proučavanja kompetentnosti zaposlenika na pojedinim zadacima došli smo do sljedećih ocjena kompetentnosti:

Tablica 13. Ocjene kompetentnosi zaposlenika na pojedinim pozicijama.

Ocjena kompetentnosti na radu (1-10)	Radnik 1	Radnik 2	Radnik 3	Radnik 4
Viličar	9	6	3	0
Papirologija	7	2	10	2
Slaganje narudžbi	2	4	1	10
Zaprimanje robe	5	9	3	3

Izvor: Vlastita izrada

Problem asignacije nam pomaže odlučiti kojeg radnika staviti na koju poziciju, tako da je svaki radnik dobije posao za koji je najkompetentniji.

Kada ovaj problem postavimo kao problem cjelobrojnog programiranja on bi izgledao ovako:

$$maxZ=9x_{11}+6x_{12}+3x_{12}+0x_{14}+7x_{21}+2x_{22}+10x_{23}+2x_{24}+2x_{31}+4x_{32}+1x_{33}+10x_{34}+5x_{41}+9x_{42}+3x_{43}+3x_{44}$$

$$x_{11}+x_{12}+x_{13}+x_{14}=1$$

$$x_{21}+x_{22}+x_{23}+x_{24}=1$$

$$x_{31}+x_{32}+x_{33}+x_{34}=1$$

$$x_{41}+x_{42}+x_{43}+x_{44}=1$$

$$x_{11}+x_{21}+x_{31}+x_{41}=1$$

$$x_{12}+x_{22}+x_{32}+x_{42}=1$$

$$x_{13}+x_{23}+x_{33}+x_{43}=1$$

$$x_{14}+x_{24}+x_{34}+x_{44}=1$$

$$x_{ij} \in \{0,1\} \qquad (i=1,2,3,4;\ j=1,2,3,4)$$

Varijable odlučivanja možemo definirati kao:

Pojedinačna xij varijabla (i=1,2,3,4 ; j=1,2,3,4) predstavlja binaran broj i-tog radnog mjesta na kojem će se nalaziti pojedini j-ti radnik (za i=1,2,3,4; j=1,2,3,4).

Optimalno rješenje problema:

Do optimalnog rješenja sam došao koristeći se alatom Excel solver.

Tablica 14. Optimalno rješenje raspodjele radnika po poslovima

0= nije najbolji za određeni posao 1= najbolji za određeni posao	Radnik 1	Radnik 2	Radnik 3	Radnik 4
Viličar	1	0	0	0
Papirologija	0	0	1	0
Slaganje narudžbi	0	0	0	1
Zaprimanje robe	0	1	0	0

Izvor: Vlastita izrada

Iz optimalnog rješenja vidimo koji radnik je najbolji za koji posao.

- Radnik 1 je najbolji za odrađivanje posla viličara

- Radnik 2 je najbolji za zaprimanja robe
- Radnik 3 je najbolji za odrađivanje poslova papirologije
- Radnik 4 je najbolji za slaganje narudžbi

Ukupan koeficijent kompetitivnosti prilikom ovakvog rasporeda je 38 jedinica od ukupnih 40.

4.3.Transportni problem 2

Malo poduzeće bavi se proizvodnjom pletenih košara. Imaju 2 proizvodna pogona, jedan je u Bilju, a drugi u Tenju. Pogon u Bilju je u sposobnosti proizvesti 300 košara mjesečno, dok je pogon u Tenju u mogućnosti proizvesti 500 košara mjesečno.

Poduzeće također ima 3 trgovine koje prodaju te košare, one se nalaze u Osijeku, Zagrebu i Splitu. Respektivno, trgovine zahtjevaju 450, 150, 200 košara mjesečno.

Trošak transporta napravljenih košara od određenog pogona do određene trgovine izražen je NJ po košari.

Tablica 15. Troškovi transporta košara.

Cijena u NJ po transportiranoj košari	Trgovina u Osijeku	Trgovina u Zagrebu	Trgovina u Splitu	PONUDA
Proizvodni pogon u Bilju	10	40	90	300
Proizvodni pogon u Tenju	20	60	120	500
POTRAŽNJA	450	150	200	

Izvor: Vlastita izrada

Kada ovaj problem postavimo kao problem cijelobrojnog programiranja, on bi izgledao ovako:

$$minz=10x_{11}+40x_{12}+90x_{13}+20x_{21}+60x_{22}+120x_{23}$$

$$x_{11}+x_{12}+x_{13}=300$$

$$x_{21}+x_{22}+x_{23}=500$$

$$x_{11}+x_{21}=450$$

$$x_{12}+x_{22}=150$$

$$x_{13}+x_{23}=200$$

$$x_{ij} \geq 0,\ (i=1,2;\ j=1,2,3)$$

$$x_{ij} \in Z \; (i=1,2; \; j=1,2,3)$$

Varijable odlučivanja definiramo:

- x_{11} – Broj košara koji se prevozi iz proizvodnog pogona u Bilju do trgovine u Osijeku
- x_{12} – Broj košara koji se prevozi iz proizvodnog pogona u Bilju do trgovine u Zagrebu
- x_{13} – Broj košara koji se prevozi iz proizvodnog pogona u Bilju do trgovine u Splitu
- x_{21} – Broj košara koji se prevozi iz proizvodnog pogona u Tenju do trgovine u Osijeku
- x_{22} – Broj košara koji se prevozi iz proizvodnog pogona u Tenju do trgovine u Zagrebu
- x_{23} – Broj košara koji se prevozi iz proizvodnog pogona u Tenju do trgovine u Splitu
-

Nakon što sve varijable, sva ograničenja i funkciju cilja unesemo u Excelov solver dobijemo optimalno rješenje koje glasi:

Tablica 16. Količina prevezenih košara uz minimalni trošak prijevoza.

Količina transportiranih košara	**Trgovina u Osijeku**	**Trgovina u Zagrebu**	**Trgovina u Splitu**	**PONUDA**
Proizvodni pogon u Bilju	0	100	200	300
Proizvodni pogon u Tenju	450	50	0	500
POTRAŽNJA	450	150	200	

Izvor: Vlastita izrada

Iz optimalnog rješenja vidimo da minimalni trošak transporta košara iznosi 34000NJ, a dobili bi ga transportom:

100 košara iz pogona u Bilju do trgovine u Zagrebu

200 košara iz pogona u Bilju do trgovine u Splitu

450 košara iz pogona u Tenju do trgovine u Osijeku

50 košara iz pogona u Tenju do trgovine u Zagrebu

4.4.Problem rasporeda redara na Ultra festivalu 2022.

Globalni festival Ultra koji se svake godine održava u Splitu na Poljudu ima potrebu zaposliti redare koji će raditi u periodu od 19:00 do 07:00. Redari imaju 6. satno radno vrijeme organizirano po smjenama koje mogu početi u 19:00, 22:00 i 01:00. Na taj način se pokriva vrijeme održavanja festivala od 19:00 do 07:00. U tom periodu mogu postojati manji i veći zahtjevi za redarima. U tablici su zadani podaci:

Tablica 17. Broj potrebnih redara po smjeni

Redni broj smjene	**Smjena**	**Broj potrebnih redara**
1	19:00 – 22:00	60
2	22:00 – 01:00	75
3	01:00 – 04:00	100
4	04:00 – 07:00	90

Izvor: Vlastita izrada.

Radno vrijeme treba popuniti tako da je minimalan broj redara po smjenama ostvaren, ali da se pritom zaposli minimalan broj redara.

Varijable odlučivanja:

x_1 – Broj redara koji na posao stižu u 19:00h.
x_2 – Broj redara koji na posao stižu u 22:00h.
x_3 – Broj redara koji na posao stižu u 01:00h.

Funkcija cilja i ograničenja:

$$minZ=x_1+x_2+x_4$$
$$x_1 \geq 60, \text{ (1. Period od 19:00 do 22:00)}$$
$$x_1+x_2 \geq 75 \text{ (2. Period od 22:00 do 01:00)}$$
$$x_2+x_3 \geq 100 \text{ (3. Period od 01:00 do 04:00)}$$
$$x_3 \geq 90 \text{ (4. Period od 04:00 do 07:00)}$$
$$x_i \in Z, \text{ za } i=1,2,3$$

Interpretacija funkcije cilja:
$minZ=x_1+x_2+x_4$

Funkcija cilja nam pomaže odrediti minimalan broj redara koje je potrebno zaposliti kako bi se zadovoljili uvjeti potrebnog broja redara za određeni vremenski period.

Interpretacija ograničenja:

$x_1 \geq 60$, (1. Period od 19:00 do 22:00)

Ako se zaposli x_1 redara koji na posao dolaze u 19:00, x_2 redara koji na posao dolaze u 22:00, x_3 redara koji na posao dolaze u 01:00, na festivalu će u periodu od 19:00 do 22:00 nalaziti x_1 redara, a potrebno je minimalno 60 redara.

$x_1+x_2 \geq 75$ (2. Period od 22:00 do 01:00)

Ako se zaposli x_1 redara koji na posao dolaze u 19:00, x_2 redara koji na posao dolaze u 22:00, x_3 redara koji na posao dolaze u 01:00, na festivalu će u periodu od 22:00 do 01:00 nalaziti x_1+x_2 redara, a potrebno je minimalno 75 redara.

$x_2+x_3 \geq 100$ (3. Period od 01:00 do 04:00)

Ako se zaposli x_1 redara koji na posao dolaze u 19:00, x_2 redara koji na posao dolaze u 22:00, x_3 redara koji na posao dolaze u 01:00, na festivalu će u periodu od 01:00 do 04:00 nalaziti x_2+x_3 redara, a potrebno je minimalno 100 redara.

$x_3 \geq 90$ (4. Period od 04:00 do 07:00)

Ako se zaposli x_1 redara koji na posao dolaze u 19:00, x_2 redara koji na posao dolaze u 22:00, x_3 redara koji na posao dolaze u 01:00, na festivalu će u periodu od 04:00 do 07:00 nalaziti x_3 redara, a potrebno je minimalno 90 redara.

Uvjet cjelobrojnosti i nenegativnosti:

$x_i \in Z$, za i=1,2,3

Varijable x_1, x_2, x_3 moraju biti cijeli brojevi i ne smiju biti negativni jer to nema smisla, ne možemo u konačnom rezultatu imati negativan broj policajaca ili broj koji nije cijeli.

5. ZAKLJUČAK

Cjelobrojno programiranje metoda je linearnog programiranja, ali se razlikuje u rezultatu rješenja. Naime, linearnom programiranju nije važan oblik broja rezultata, dok kod cjelobrojnog programiranja inzistira se na dobivanju cijelih brojeva. Postoje metode u kojima je moguće ostaviti decimalne brojeve kao uvjet računanja, tada se govori o djelomičnom cjelobrojnom programiranju, ako su sva ograničenja i rezultat prikazani u cijelom broju, tada je riječ o čistom cjelobrojnom programiranju, a može se pojaviti i binarno cjelobrojnom programiranju gdje su rezultati prikazani kao 0 ili 1.

Korištenje modela cjelobrojnog programiranja sve češće je u praksi, a posebno u velikim organizacijama gdje je potrebno rasporediti veći broj ljudi, strojeva i oskudnih resursa. Kako bi se napravio najbolji raspored, prvo je potrebno znati s čim sve organizacija raspolaže, sva ograničenja i uvjete i zatim se odabire jedna od mogućih metoda za rješavanje problema.

Unutar rada predstavljena su dva rješenja problema transporta kao i rješenje problema asignacije te problem raspoređivanja radnika. Alati koji su korišteni u rješavanju ovih problema su program microsoft Excel i Excelov solver. Pomoću tih alata došli smo optimalnih rješenja i definirane funkcije cilja problema.

LITERATURA

Popis knjiga i stručnih članaka

1. Bilješke s predavanja kolegija Kvantitativne metode za poslovno odlučivanje, (10.07.2021.)
2. Aardal K., Nemhauser G., Weismantel R., (Eds.) (2005), Handbooks in Operations and Management Science.
3. Barković, D. (2002) Operacijska istraživanja', Ekonomski fakultet Osijek, Osijek,
4. Boland, N., Hughes, B. D., Merlot, L. T. G., Stuckey, P. J. (2006), New integer linear programming approaches for course timetabling, Computers & Operation Research 35(7), 2209-2233.
5. Crnjac Milić, M., Kvesić, Lj. (2008) Heurističke metode cjelobrojnog programiranja i njihove aplikacije u ekonomiji
6. Čerić V. (2009) Ostale metode linearne optimizacije, Sveučilište u Zagrebu, Zagreb
7. Kumar, S., Monga, K., Jones, B. C. (2010) Fifty Years of Integer Programming: A Review of the Solution Approaches, Volume 6, Issue 3
8. Lukač, Z., Neralić, L. (2012) Operacijska istraživanja, Element, Zagreb,
9. Mason, A. J. (2011) OpenSolver - An open source add-in to solve linear and integer programmes in Excel. In: Klatte D, Luthi HJ, Schmedders K (eds) Operations Research Proceedings, Springer, Berlin Heidelberg, str. 401-406.
10. Prabodanie, R. (2017) An Integer Programming Model for a Complex University Timetabling Problem: A Case Study, Industrial Engineering and Management Systems Vol 16, No 1, str.141-153
11. Schultz, R., Stougie, L., Van der Vlerk, H. M. (1996) Two-stage stochastic integer programming: a survey, Wiley Libriary,Milić, C.D., Kvesić, Lj.: Heurističke metode cjelobrojnog programiranja i njihove aplikacije u ekonomiji, Ekonomski vjesnik: Review of Contemporary Entrepreneurship, Business, and Economic Issues, Vol. 23 No. 2, 2010., str. 431-437
12. Zelenika, R.: Metode operacijskih istraživanja u kaleidoskopu obrazovnih i znanstvenih industrija, NAŠE MORE : znanstveni časopis za more i pomorstvo, Vol. 63 No. 1 Supplement, 2016., str. 44-52

Popis internet izvora:

13. Cjelobrojno programiranje, https://www9.scribd.com/document/77251846/Cjelobrojno-programiranje, (05.07.2021.)
14. Example of Integer Programming, http://web.pdx.edu/~stipakb/download/PA557/ReadingsPA557sec6.pdf, (07.07.2021.)
15. Integer programming formulation examples, http://people.brunel.ac.uk/~mastjjb/jeb/or/moreip.html, (05.07.2021.)
16. Kuzmanović, I., Sabo, K.: Linearno programiranje, http://www.mathos.unios.hr/lp/Materijali/predavanje16_lp.pdf, (05.07.2021.)
17. Linearno programiranje, https://sites.google.com/site/linearnoprogramiranje1/literatura, (02.07.2021.)
18. Matematičko programiranje, https://enciklopedija.hr/natuknica.aspx?ID=70150, (02.07.2021.)
19. Problem transporta i distribucije, https://ff.unze.ba/nabokov/operacionaIstraz/Sedmica01.pdf?parameter=01121980, (10.07.2021.)
20. Simplex metoda, https://sites.google.com/site/linearnoprogramiranje1/teorija-linearnog-programiranja/metode-rjesavanja-lp/simpleks-metoda, (05.07.2021.)
21. Što je Microsoft Excel, (2011), http://www.oblakznanja.com/2011/12/sto-je-microsoft-excel/, (10.07.2021.)

POPIS TABLICA

POPIS GRAFIKONA