Određivanje optimalne visine leta jednomotornog klipnog zrakoplova na odabranoj ruti

Zorić, Marko

Undergraduate thesis / Završni rad

2021

Degree Grantor / Ustanova koja je dodijelila akademski / stručni stupanj: **University of Zagreb, Faculty of Transport and Traffic Sciences / Sveučilište u Zagrebu, Fakultet prometnih znanosti**

Permanent link / Trajna poveznica: https://urn.nsk.hr/urn:nbn:hr:119:495464

Rights / Prava: In copyright / Zaštićeno autorskim pravom.

Download date / Datum preuzimanja: **2024-10-13**

Repository / Repozitorij:

Faculty of Transport and Traffic Sciences - Institutional Repository

SVEUČILIŠTE U ZAGREBU

FAKULTET PROMETNIH ZNANOSTI

ZAVRŠNI RAD

ODREĐIVANJE OPTIMALNE VISINE LETA JEDNOMOTORNOG KLIPNOG ZRAKOPLOVA NA ODABRANOJ RUTI

Mentor: dr. sc. Petar Andraši

Student: Marko Zorić

JMBAG: 0135245662

ZAGREB, rujan 2021.

Sveučilište u Zagrebu

Fakultet prometnih znanosti

ZAVRŠNI RAD

ODREĐIVANJE OPTIMALNE VISINE LETA JEDNOMOTORNOG KLIPNOG ZRAKOPLOVA NA ODABRANOJ RUTI

DETERMINING THE OPTIMUM FLIGHT ALTITUDE OF A SINGLE-ENGINE PISTON AIRCRAFT ON THE SELECTED ROUTE

Mentor: dr. sc. Petar Andraši

Student: Marko Zorić

JMBAG: 0135245662

ZAGREB, rujan 2021.

SVEUČILIŠTE U ZAGREBU
FAKULTET PROMETNIH ZNANOSTI
ODBOR ZA ZAVRŠNI RAD

Zagreb, 27. travnja 2021.

Zavod: **Zavod za aeronautiku**
Predmet: **Planiranje letenja i performanse II**

ZAVRŠNI ZADATAK br. 6432

Pristupnik: **Marko Zorić (0135245662)**
Studij: Aeronautika
Smjer: Pilot
Usmjerenje: Civilni pilot

Zadatak: **Određivanje optimalne visine leta jednomotornog klipnog zrakoplova na odabranoj ruti**

Opis zadatka:

Opisati performanse jednomotornog klipnog zrakoplova
Odrediti horizontalnog profila zrakoplova
Odrediti vertikalnog profila zrakoplova
Analiza rezultata izračuna performansi zrakoplova i odabir optimalne visine leta

Mentor:

Petar Andraši, mag. ing. aeronaut.

Predsjednik povjerenstva za završni ispit:

SAŽETAK

Jednomotorni klipni zrakoplov služi za kratke rute te se njegov let sastoji od faze polijetanja te faze krstarenja i faze slijetanja. Za svaku od navedenih faza se rade proračuni vremena leta te potrošnje goriva, osim za fazu slijetanja u kojem je potrošnja goriva jednaka kao i u horizontalnom letu.

Zapovjednik zrakoplova odabire visinu krstarenja na temelju performansi zrakoplova i na temelju podataka o smjeru i jačini vjetra na raznim visinama na ruti. U sadržaju ovog završnog rada objašnjen je odabir optimalne visine jednomotornog klipnog zrakoplova koja se temelji na izračunatim podatcima o potrošenom gorivu te vremenom leta na odabranoj ruti. Različite postavke snage utječu na brzinu zrakoplova što može uzrokovati manju ili veću potrošnju goriva na ruti te vrijeme leta.

KLJUČNE RIJEČI: performanse jednomotornog klipnog zrakoplova, odabir optimalne visine, krstarenje.

SUMMARY

Single engine piston aircraft is used for short routes and flight is composed of three phases which are climb, cruise and landing. For each of the phases, calculations must be made for flight time and fuel consumption, except for the landing phase because fuel consumption is the same as in horizontal flight.

Commander of aircraft selects the cruise altitude based on the performance of aircraft and based on wind direction and velocity at various altitudes on the route. The content of this final paper explains the optimal height selection of a single piston engine aircraft based on calculated data of fuel consumption and flight time on the seleted route. Different power settings affect the speed of the aircraft which can cause less or more fuel consumption on the route and flight time.

KEY WORDS: single engine piston aircraft performance, optimum height selection, cruise.

SADRŽAJ

1. Uvod ... 1
2. Opis performansi jednomotornog klipnog zrakoplova ... 3
   2.1. Polijetanje zrakoplova ... 3
   2.2. Penjanje zrakoplova ... 4
   2.3. Horizontalni let zrakoplova ... 7
   2.4. Istrajnost ... 9
   2.5. Dolet ... 10
3. Određivanje horizontalnog profila zrakoplova ... 12
4. Određivanje vertikalnog profila zrakoplova ... 13
   4.1. Python kodovi vertikalnog profila ... 14
      4.1.1. Python kodovi za fazu krstarenja ... 15
      4.1.2. Python kodovi za fazu penjanja ... 17
      4.1.3. Python kodovi za provjeru udaljenosti TOC ... 20
   4.2. Grafički prikaz vertikalnog profila ... 22
5. Rezultati proračuna ... 26
6. Analiza rezultata ... 30
7. Zaključak ... 32

LITERATURA ... 33

Popis tablica ... 35

Popis slika ... 36

Prilog 1 ... 37

1. Uvod

Jednomotorni klipni zrakoplovi odličan su alat za školovanje civilnih pilota diljem svijeta zbog svoje jednostavnosti te niskih troškova održavanja. Raspon mogućih visina nije velik kao kod zrakoplova za prijevoz putnika isto kao i performanse, ali potrošnja goriva je znatno manja. Najpoznatiji primjerak jednomotornog klipnog zrakoplova je Cessna 172 čija povijest seže daleko u 1956. godinu kada je izrađen prvi primjerak u Sjedinjenim Američkim Državama. U Hrvatskoj C172 koristi se za školovanje u Hrvatskom zrakoplovnom nastavnom središtu te su zbog toga korištene performanse navedenog zrakoplova prilikom računanja podataka u završnom radu.

Optimalna visina leta ovisi o željenom ishodu, među rasponom mogućih visina leta ona bi bila visina gdje će zrakoplov potrošiti najmanje goriva za let do odredišta ili gdje će vrijeme leta biti najmanje. Prije leta rade se izračuni kako bi se detaljno dobili podaci o vremenu leta te potrošnji goriva za let na odabranoj ruti. Za izračune koristi se operativni priručnik zrakoplova uz meteorološke informacije za dan leta s podatcima o vjetru na ruti i rasponu temperature na mogućim visinama.

Izračuni za optimalnu visinu leta zahtijevaju mnogo vremena ako se rade na papiru zbog velike količine informacija. U završnom radu izračuni su rađeni u programskom jeziku opće namjene Python u kojemu je moguće raditi izračune u malom vremenu. Korištenja programskog jezika je tablica s visinama leta i detaljnim izračunima o potrošnji goriva te vremenu leta na odabranoj ruti. Završni rad podijeljen je u 7 cjelina:

1. Uvod
2. Opis performansi jednomotornog klipnog zrakoplova
3. Određivanje horizontalnog profila zrakoplova
4. Određivanje vertikalnog profila zrakoplova
5. Rezultati
6. Analiza
7. Zaključak

U drugom poglavlju naveden je tip zrakoplova korišten pri izračunu podataka za završni rad te su opisane performanse jednomotornog klipnog zrakoplova te vanjski utjecaji koji utječu na njih u različitim fazama leta.

Treće poglavlje ukratko opisuje horizontalni profil zrakoplova s vanjskim utjecajima. Odabrana ruta leta prikazana je u tablici te ukratko opisana.

U četvrtom poglavlju objašnjen je vertikalni profil leta, od čega se sastoji i kako nastaje. Navedene su visine koje su prikazane u izračunima u programskom jeziku Python te je ukratko objašnjen programski jezik. Većina kodova korištenih za izračun potrošnje goriva i vremena leta prikazani su uz dodatna objašnjenja ispod koda. Vrijednosti uzete iz operativnog priručnika zrakoplova te korištene prilikom izračuna u programu prikazane su tablično uz grafički prikaz vertikalnog profila leta.

U petom poglavlju objašnjen je primarni cilj izračuna u programu Python uz tablični prikaz rezultata.

U šestom poglavlju su dobiveni rezultati analizirani te objašnjeni uz glavne razlike između postavka snage zrakoplovnog motora te dobivene vrijednosti različitih visina leta.

2. Opis performansi jednomotornog klipnog zrakoplova

„Pojam performanse zrakoplova koristi se kako bi se opisala sposobnost zrakoplova za ostvarenje određenih ciljeva koji ga čine korisnim u svrhu letenja“ [1].

Performanse zrakoplova prikazuju se karakteristikama potrošnje goriva i udaljenosti u penjanju te horizontalnom letu te se dodatno izračunavaju istrajnost i dolet. Za višemotorni zrakoplov se dodatno računa i izračun za snižavanje.

Za izradu završnog rada korišteni su podatci za jednomotorni klipni zrakoplov Cessna 172 koji je certificiran pod klasom CS23. „Prema performansama pripada klasi B laganih zrakoplova s konfiguracijom sjedišta od 9 ili manje koja ne uključuje pilotsko sjedište i sa MTOM (eng. Maximum Take Off Mass, hrv. maksimalna masa pri polijetanju) do 5700 kilograma“ [2].

2.1. Polijetanje zrakoplova

U fazi polijetanja zrakoplova nakon odobrenja za polijetanje, zapovjednik ručicu snage postavlja u krajnju prednju poziciju te zrakoplov razvija najveću moguću snagu pri polijetanju. Postoji nekoliko elemenata koje utječu na performanse zrakoplova pri polijetanju koje će se detaljnije objasniti u daljnjem tekstu.

Brzina zrakoplova pri polijetanju je, u pravilu, za 15% veća od brzine sloma uzgona [1]. Povećana masa zrakoplova će utjecati na duljinu uzlijetanja zrakoplova. Poznato je da 10% povećanja mase zrakoplova pri polijetanju kao posljedicu ima 20% povećanja potrebne duljine zatrčavanja zrakoplova za uzlijetanje [1]. Nadmorska visina aerodroma također je jedan od čimbenika koji utječe na performanse zrakoplova pri polijetanju. Aerodromi koji se nalaze na velikim nadmorskim visinama utječu na polijetanje tako da će zrakoplov morati imati povećanu brzinu pri polijetanju jer će motor zrakoplova i efikasnost propelera biti umanjeni. Duljina zatrčavanja se povećava za 1% svakih 100 stopa nadmorske visine aerodroma iznad razine mora. Također se duljina zatrčavanja povećava za 1% svakih 1 stupanj Celzija iznad standardne temperature za elevaciju aerodroma polijetanja [1]. Brzina uzdizanja zrakoplova, kao i kut penjanja zrakoplova, bit će umanjeni zbog navedenih utjecaja kod polijetanja.

Čeona i leđna komponenta vjetra pri polijetanju također utječu na duljinu zatrčavanja zrakoplova. Čeoni vjetar ima pozitivan utjecaj. Udaljenost u zatrčavanju pri polijetanju umanjena je za 1,5% za svaki čvor čeone komponente vjetra do 20 čvorova [1]. Penjanje nakon uzlijetanja u uvjetima čeone komponente vjetra kao posljedicu uzrokuje manju putnu brzinu zrakoplova u odnosu na Zemlju te strmiji kut penjanja nakon polijetanja.

Leđni vjetar ima negativan utjecaj na parametre pri polijetanju zrakoplova. Za leđnu komponentu vjetra od 5 čvorova udaljenost u zatrčavanju treba biti pomnožena s vrijednosti 1,25, a za jačinu leđnog vjetra od 10 čvorova s vrijednosti 1,55 [1]. U uvjetima leđnog vjetra, duljina zatrčavanja zrakoplova bit će povećana, a kut penjanja bit će umanjen. Polijetanje s leđnim vjetrom treba izbjegavati ako nije ustanovljeno da je duljina uzletno sletne staze dovoljno dugačka za sigurno polijetanje u takvim uvjetima.

Nagib uzletno sletne staze također ima utjecaj na parametre pri polijetanju. Nagib uzbrdo uzletno sletne staze ima kao posljedicu veću duljinu zatrčavanja zrakoplova. Uzbrdica od 2% povećava duljinu zatrčavanja za 15% [1].

Površina uzletno sletne staze također ima utjecaj na zrakoplov pri polijetanju. Travnata površina uzrokuje povećanje duljine zatrčavanja do 15% [1].

2.2. Penjanje zrakoplova

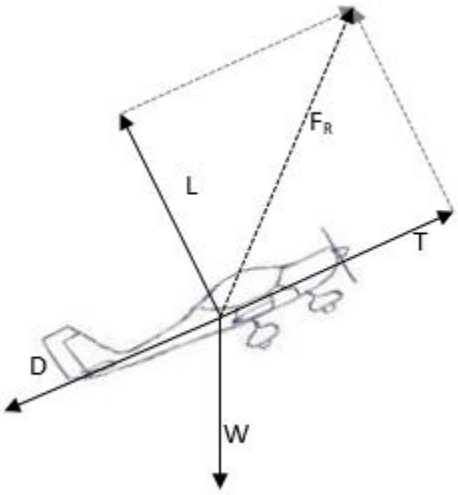
Kretanje i let zrakoplova zahtijevaju određen rad koji se dobiva mehaničkom energijom zrakoplovnog motora. Mehanička energija dolazi u dva oblika, kinetička energija (energija brzine) i potencijalna energija (energija pozicije).

Performanse pozitivnog penjanja javljaju se kada zrakoplov stekne potencijalnu energiju povećanjem visine. Čimbenici koji pridonose pozitivnim performansama penjanja kod većine zrakoplova su:

- Zrakoplov penje (dobiva potencijalnu energiju) koristeći višak snage iznad onoga potrebnog za održavanje horizontalnog leta.

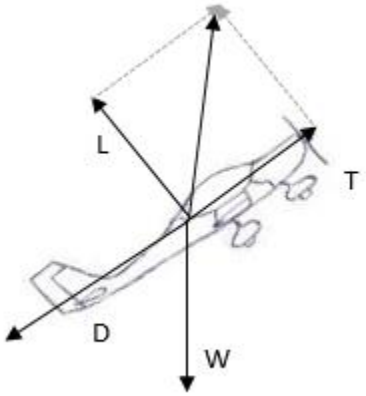
- Zrakoplov penje pretvarajući zračnu brzinu (kinetička energija) u visinu leta (potencijalna energija).

Primjer prvog čimbenika bio bi kada zrakoplov sa snagom motora od ukupno 200 konjskih snaga koristi samo 130 konjskih snaga kako bi održao zrakoplov u horizontalnom letu na određenoj visini. U tom slučaju ostaje 70 konjskih snaga dostupnih za penjanje na veću visinu. Pilot zrakoplova održava jednaku brzinu leta i povećava snagu kako bi ostvario penjanje (slika 1).



Slika 1: Penjanje zrakoplova koristeći višak snage

Primjer drugog faktora bio bi kada zrakoplov leti brzinom od 150 čvorova na određenoj visini i postavke snage su konstantne, ali pilot koristeći kontrole zrakoplova krene penjati na veću visinu i brzina zrakoplova pada za određenu vrijednost ispod 150 čvorova (slika 2).



Slika 2: Penjanje zrakoplova pretvarajući zračnu brzinu

Performanse penjanja zrakoplova bitne su radi nadvisivanja prepreka te penjanja na veću visinu leta što može imati prednosti boljih vremenskih prilika i manje potrošnje goriva. Povećanje visine leta povećava potrebnu snagu te smanjuje dostupnu snagu. Performanse penjanja zrakoplova smanjuju se s povećanjem visine leta.

Prilikom penjanja na visinu odabire se brzina najboljeg kuta penjanja V_X koja će pružiti nadvisivanje prepreka ili brzina najbržeg uzdizanja V_Y koja pruža najveći porast visine u vremenu.

Postoje dvije visine koje ograničavaju visinu leta na koju zrakoplov može popeti, a to su apsolutni plafon leta i servisni plafon leta. Apsolutni plafon leta označava visinu na kojoj ne postoji višak snage te je to maksimalna visina na koju zrakoplov može popeti. Servisni plafon leta je visina na kojoj zrakoplov može penjati s maksimalnom brzinom uspona od 100 stopa u minuti [3].

Performanse zrakoplova mogu se naći u operativnom priručniku zrakoplova (eng. Pilot Operating Handbook – POH). Prikaz performansi može biti naveden grafički ili tablično. Navedene performanse zrakoplova su: potrošnja goriva prema različitim visinama i postavkama snage, potrošnja goriva prema fazama leta, vrijeme leta, brzina leta, brzina najboljeg kuta penjanja, najbolja brzina uspona.

2.3. Horizontalni let zrakoplova

U fazi krstarenja, zrakoplov je u horizontalnom letu kada su kut penjanja i kut nagiba zrakoplova jednaki nuli. Na slici 3 vidljivo je kako bi zrakoplov letio horizontalno na jednakoj visini zahtjeva se jednakost sile uzgona i sile težine te potisne sile zrakoplova sa silom otpora.

Slika 3: Sile koje djeluju na zrakoplov u horizontalnom letu [4]

Sila otpora zrakoplova definira potrebnu potisnu silu kako bi zrakoplov održavao konstantan horizontalan let na jednakoj visini. Sila otpora prikazana je formulom 1:

$$F_D = \frac{1}{2} \times \varphi \times V^2 \times S_W \times C_D \tag{1}$$

- φ - gustoća zraka
- V - brzina zrakoplova
- S_W - Površina zrakoplova
- C_D - koeficijent otpora

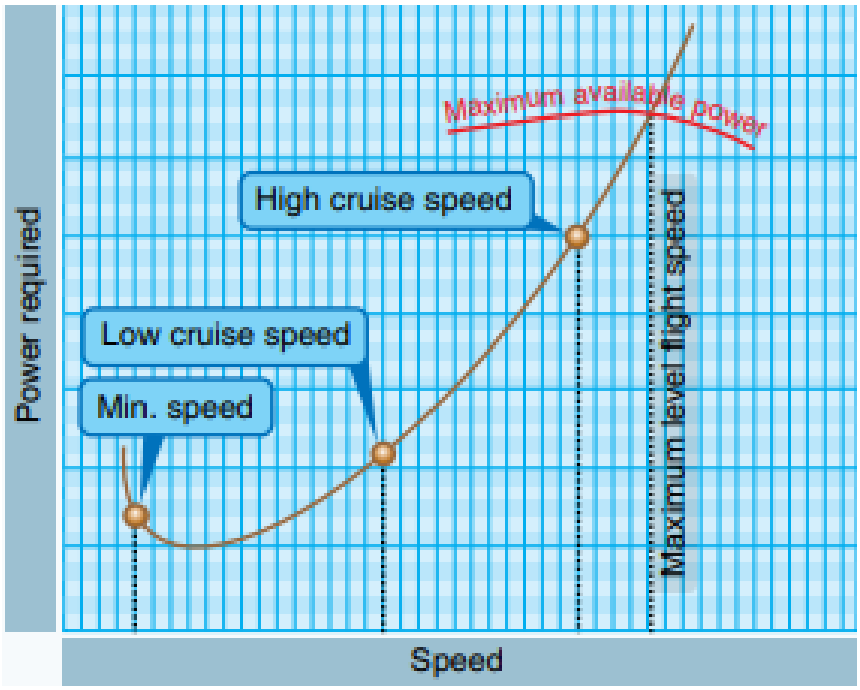
Svaki dio zrakoplova doprinosi otporu zrakoplova te se ukupni otpor dijeli na inducirani otpor koji nastaje zbog površina koji generiraju silu uzgona i na štetni otpor koji nastaje kretanjem površine zrakoplova zrakom. Inducirani otpor prikazan je formulom 2:

$$F_{D,i} = \frac{1}{V^2} \tag{2}$$

Štetni otpor prevladava prilikom velike brzine leta, a inducirani otpor malom brzinom. Štetni otpor prikazan je formulom 3:

$$F_{D,0} = \frac{1}{2}\,\varphi \times V^2 \times S_W \times C_{L.0} \tag{3}$$

Za primjer, ako zrakoplov leti horizontalno brzinom 100 čvorova te ubrza na brzinu leta od 200 čvorova kao posljedicu izazvat će četverostruko povećanje iznosa štetnog otpora. Potrebna snaga za savladavanje povećanja iznosa štetnog otpora zbog ubrzanja zrakoplova bit će osam puta veća od originalne veličine. Obrnuto, kada zrakoplov u horizontalnom letu leti s dvostruko većom brzinom inducirani otpor iznosi jednu četvrtinu originalne vrijednosti i potrebna snaga potrebna za savladavanje novog iznosa otpora iznosi jednu polovinu originalne vrijednosti. Maksimalna brzina u horizontalnom letu postiže se kada se potrebna snaga izjednači s maksimalnom dostupnom snagom zrakoplovnog motora i prikazano je na slici 4.



Slika 4: Dijagram ovisnosti potrebne snage s brzinom zrakoplova [5]

Minimalna brzina leta nije definirana snagom nego stabilnošću i gubitkom uzgona zrakoplova. Kretanje zrakoplova (kinetička energija, KE) opisano je brzinom leta, a vertikalna pozicija (potencijalna energija, PE) zrakoplova njegovom trenutnom visinom leta i može se

prikazati formulama 4 i 5. Kretanje i pozicija povezani su s masom zrakoplova, a kretanje je direktno proporcionalno kvadratu brzine zrakoplova dok je pozicija direktno proporcionalna visini leta [3].

$$KE = {}^{1}/_{2} \times m \times V^2 \tag{4}$$

$$PE = m \times g \times h \tag{5}$$

- m - masa zrakoplova
- V - brzina zrakoplova
- g - gravitacijska akceleracija=9,18 ${}^{m}/_{s^2}$
- h - visina zrakoplova

Performanse jednomotornog klipnog zrakoplova u horizontalnom letu nalaze se u operativnom priručniku zrakoplova (eng. Pilot Operating Handbook – POH) te se dobivaju informacije o potrošnji goriva na različitim visinama u horizontalnom letu, brzini zrakoplova i vremenu leta. Podatci o performansama mogu se prikazati tablično ili grafički, ali najčešće je tablično. Performanse ovise o atmosferskim uvjetima i o zadanim postavkama snage motora zrakoplova. Smanjenje potrošnje goriva u letu najznačajnije je u fazi krstarenja. Navedene informacije detaljnije su objašnjenje u naknadnim poglavljima.

2.4. Istrajnost

Istrajnost je pojam koji opisuje određenu količinu vremena koju zrakoplov može provesti u zraku s punim spremnikom goriva. Parametri za istrajnost se razlikuju ovisno o tipu zrakoplova, tj. je li zrakoplov na mlazni ili na propelerski pogon.

Specifična potrošnja goriva kod klipnih zrakoplova dana je u smislu snage, a ne potiska. Prilikom teoretskog izračuna istrajnosti za klipne zrakoplove prikazana je zavisnost formulom 6:

$$E = \frac{\eta_{pr}}{c} \sqrt{2\rho_\infty S} \frac{C_L^{3/2}}{C_D} \left(W_1^{-1/2} - W_0^{-1/2}\right) \tag{6}$$

- η_{pr} - učinkovitost propelera
- c - specifična potrošnja goriva
- ρ_∞ - gustoća zraka

- S – površina krila zrakoplova
- C_L - koeficijent uzgona
- C_D - koeficijent otpora
- W_1 - težina količine goriva umanjena za potrošenu količinu
- W_0 - početna težina količine goriva

Prema zavisnosti za istrajnost vidljivo je da maksimalna istrajnost kod klipnih zrakoplova zavisi od određenih uvjeta. Uvjeti za maksimalnu istrajnost su let pri maksimalnom omjeru $\frac{C_L^{3/2}}{C_D}$, maksimalna moguća efikasnost propelera, najmanja moguća specifična potrošnja goriva, najveća moguća razlika između W_0 i W_1 te let na razini mora gdje gustoća zraka ima najveću vrijednost.

Teoretska maksimalna istrajnost za klipni zrakoplov postiže se letom pri brzini gdje je koeficijent otpora nultog uzgona jednak jednoj trećini otpora uzrokovanog silom uzgona [6].

2.5. Dolet

Dolet zrakoplova je ukupna udaljenost, mjerena u odnosu na Zemlju, prijeđena sa zrakoplovom spremnika punih gorivom. Dolet zrakoplova je zbog specifične potrošnje goriva usko povezan s performansama zrakoplovnog motora.

Za prikaz doleta zrakoplova koristi se Berquetova jednadžba doleta i prikazana je formulom 7:

$$R = \frac{V_\infty}{c_t}\,\frac{L}{D}\ln\frac{W_0}{W_1} \tag{7}$$

- V_∞ - brzina zrakoplova
- c_t - specifična potrošnja goriva potiska
- L - sila uzgona
- D - sila otpora

Kako bi zrakoplov postigao maksimalan dolet potreban je let gdje je iznos $V_\infty(L/D)$ maksimalan.

Izraz za dolet klipnog zrakoplova izražava se kroz specifičnu potrošnju goriva c umjesto koristeći specifičnu potrošnju goriva potiska c_t kao što je to prikazano u Berquetovoj jednadžbi doleta zrakoplova. Umjesto izraza za brzinu zrakoplova V_∞, na mjestu brojnika u jednadžbi pojavljuje se učinkovitost propelera η_{pr}. Za klipne zrakoplove, Berquetova jednadžba doleta može se prikazati jednadžbom 8:

$$R = \frac{\eta_{pr}}{c} \frac{L}{D} \ln \frac{W_0}{W_1} \tag{8}$$

- η_{pr} - učinkovitost propelera
- c - specifična potrošnja goriva
- W_1 - masa količine goriva umanjena za potrošenu količinu
- W_0 - početna masa količine goriva

Kako bi klipni zrakoplovi ostvarili maksimalan dolet prema Berquetovoj jednadžbi za dolet klipnih zrakoplova, vidljivo je da je potreban let pri maksimalnom omjeru L/D, najveća moguća učinkovitost propelera, najmanja moguća potrošnja goriva i najveći moguća razlika između W_0 i W_1.

Teoretski, maksimalni dolet za klipni zrakoplov postiže se letom pri brzini gdje je koeficijent otpora nultog uzgona jednak iznosu otpora uzrokovanim silom uzgona [6].

3. Određivanje horizontalnog profila zrakoplova

Plan leta zrakoplova sadrži putanju leta prema rutnim točkama u prostoru, točnije geografskih točaka, koje zrakoplov nadlijeće prilikom leta od aerodroma polijetanja do aerodroma slijetanja. Svaka od tih točaka sadrži određene geografske koordinate te visinu nadlijetanja zadane točke. Plan leta se najčešće sastoji od tri faze: faza polijetanja, faza krstarenja te faza slijetanja. Za svaku od tih faza leta postoji određeni horizontalni profil leta zrakoplova [7].

Postojeći vjetar na visini leta je vrlo bitan prilikom izrade rute za let do odredišta. Smanjena potrošnja goriva te ušteda goriva prilikom leta je najznačajnija prilikom faze krstarenja. Utjecaj smjera i jačine vjetra na ruti u fazi krstarenja može također umanjiti vrijeme leta na principu korištenja prednosti leđnog vjetra ili izbjegavanja čeonog vjetra [8].

Za potrebe ovog završnog rada ruta koja je korištena u programu „Python“ je ruta od Zagreba do Osijeka pri čemu je aerodrom polijetanja LDZA, a aerodrom slijetanja LDOS. Navedena ruta nije najkraća ruta od Zagreba do Osijeka već je to ruta s 4 rutne točke, a razlog tome je testiranje algoritma na više etapa rute Zagreb-Osijek. Točke na ruti su odabrane na osnovu VFR (eng. Visual Flight Rules, hrv. pravila vizualnog letenja) karte.

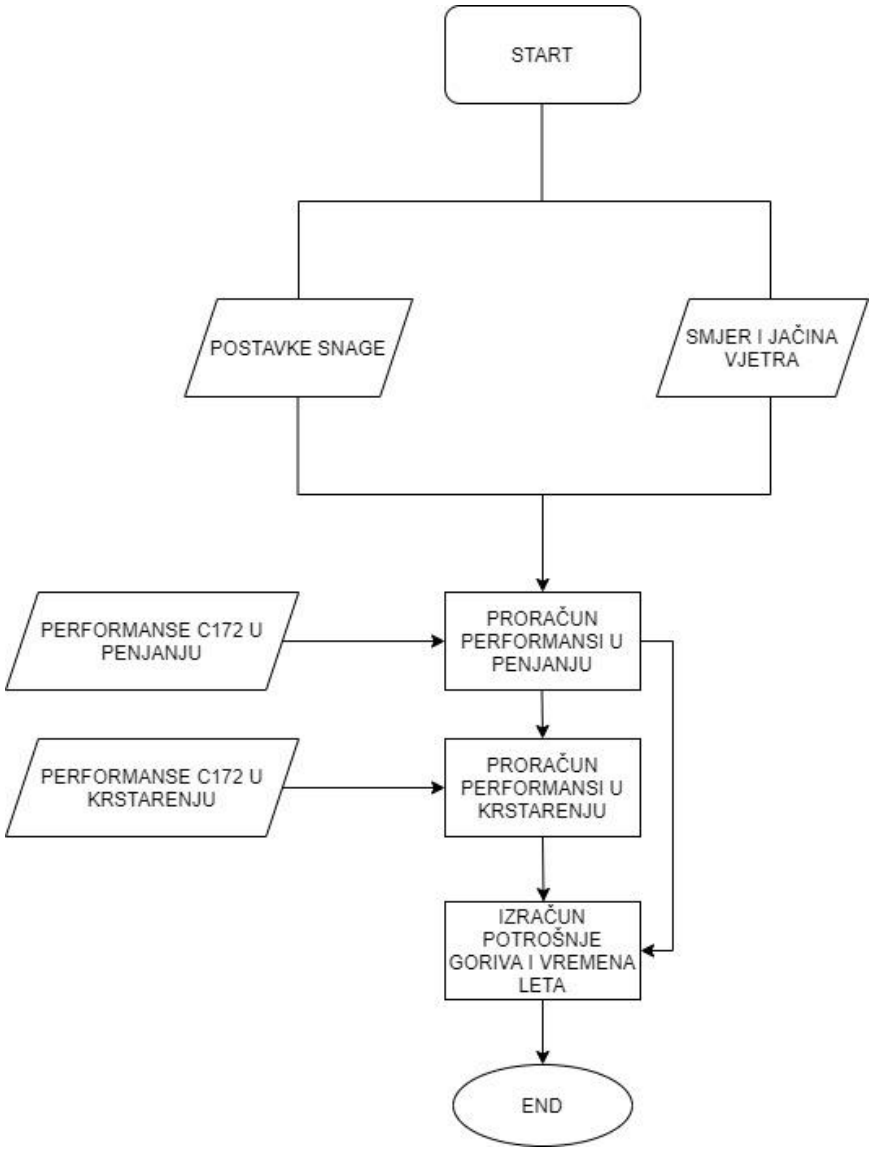
U tablici 1 navedene su rutne točke s njihovim međusobnim udaljenostima i magnetskim smjerom leta te su podatci iz tablice također korišteni prilikom izračuna u programu.

Tablica 1: Rutne točke

Rutne točke	Udaljenost (NM)	Magnetski smjer leta (°)
LDZA – S1	15,8	188
S1 - SISAK	16,3	091
SISAK - VBA	36,7	058
VBA – A1	49,5	100
A1 – LDOS	22,7	098

4. Određivanje vertikalnog profila zrakoplova

Na slici 5 prikazan je dijagram toka izračuna performansi. U programskom jeziku Python dobiveni su izračuni potrošnje goriva i vremena zrakoplova u fazi polijetanja i fazi krstarenja. U programu su navedeni podatci o tri različite postavke snage zrakoplova koje utječu na konačne rezultate uz navedene performanse zrakoplova u penjanju i krstarenju iz operativnog priručnika zrakoplova. Također na konačan rezultat utječe smjer i jačina vjetra na ruti.



Slika 5: Dijagram toka

Visina leta je okomita linija koja predstavlja udaljenost zrakoplova od neke odabrane horizontalne površine na kojoj se temelje mjerenja. Visina leta je vrlo važan prostorni i

sigurnosni podatak te je od velike važnosti prilikom određivanja rute raspolagati s informacijom na kojoj će se visini letjeti prilikom krstarenja [9].

Plan leta zrakoplova sastoji se od nekoliko faza. Prva je polijetanje, druga krstarenje i treća faza je slijetanje. Za svaku od faza, putanja leta je definirana s horizontalnim profilom koji prati geografske točke na ruti te s vertikalnim profilom leta kojeg definiraju određene visine leta na zadanoj ruti [5].

Vertikalni plan leta je konstruiran kao kombinacija zrakoplovnih performansi te zrakoplovne konfiguracije. Izračuni se odrađuju sistematski, počevši s izračunima za penjanje na željenu visinu krstarenja te za fazu krstarenja. Izračuni se vrše segment po segment od jedne rutne točke do druge rutne točke te tako nastaje profil leta od aerodroma polijetanja do aerodroma slijetanja.

Za ovaj završni rad izračuni za vertikalni profil leta su računati u programu „Python“. Izračuni koji su odrađivani temelje se na poznatim podatcima iz POH-a zrakoplova C172 N te o poznatim podatcima o smjeru i jačini vjetra na svakoj etapi zrakoplovne rute. Prilikom leta u fazi krstarenja korištene su 3 različite postavke snage, a to su 2100 RPM, 2300 RPM i 2500 RPM. (eng. Rotation Per Minute, hrv. okretaji po minuti). Vertikalni profil leta proteže se od visine aerodroma polijetanja do konačne visine 10 000 ft (eng. feet, hrv. stopa).

Python je programski jezik raznih namjena te se može primijeniti svugdje gdje se koriste podatci koji se temelje na matematičkim izračunima. Sličnost s većinom programskih jezika je ta da Python radi u paru s tumačem koji izvršava finalne redove kodova. Python se sa svojom osnovom u sintaksi na engleskom jeziku smatra jednim od najjednostavnijih kodirajućih jezika za učenje i čitanje. [10]

4.1. Python kodovi vertikalnog profila

U nastavku su prikazani te objašnjeni kodovi korišteni u programu za izračun optimalne visine leta jednomotornog klipnog zrakoplova.

```
import numpy as np
```

```
import math
from matplotlib import pyplot as plt

def interpoliraj(fa, a, fb, b, x):
    if x == a:
        return fa
    if x == b:
        return fb
    else:
        return fa + ((fb-fa)*(x-a)/(b-a))
```

Naredba „import“ služi za dodatne biblioteke, „numpy“ i „math“ služe za matematičke operacije te „mathplotlib“ je za izradu grafova. Linija koda „def interpoliraj (fa, a, fb, b, x)“ je definicija funkcije za interpolaciju gdje su fa, a, fb, b nama poznati podatci te nas zanima vrijednost funkcije u točki x.

4.1.1. Python kodovi za fazu krstarenja

```
def kruzanje(D, trueTrack, wind, windVelocity, visina):
visinaIspod = math.floor(visina/2000)*2000
    visinaIznad = math.ceil(visina/2000)*2000
```

„Def kruzanje“ je funkcija za krstarenje te su u zagradi ulazne varijable. „Math.floor“ i „math.ceil“ su funkcije za zaokruživanje. U tablici „cruise performance“ zadane su nam vrijednosti svakih 2000 ft te se traži broj koji je visina zaokružena na 2000 te tražimo visinu prvu vrijednost iznad i prvu vrijednost ispod.

```
TASispod = cruisePerformance[str(visinaIspod)]["2500"][1]
    TASiznad = cruisePerformance[str(visinaIznad)]["2500"][1]
    TAS = interpoliraj(TASispod, visinaIspod, TASiznad, visinaIznad, visina)

GPHispod = cruisePerformance[str(visinaIspod)]["2500"][2]
    GPHiznad = cruisePerformance[str(visinaIznad)]["2500"][2]
```

```
GPH = interpoliraj(GPHispod, visinaIspod, GPHiznad, visinaIznad, visina)
```

Koristi se funkcija za interpoliranje te se određuje brzina TAS (eng. True Air Speed, hrv. stvarna brzina leta) među poznatim vrijednostima brzine TAS iz tablice „cruise performance“ te se također odrađuje ista funkcija za interpoliranje gdje se traži vrijednost potrošnje goriva u GPH (eng. Gallon Per Hour, hrv. galoni po satu).

„True track” tj. pravi smjer zrakoplova je uzet iz VFR karte te je unesen u program za svaku etapu rute. Od vrijednosti pravog smjera leta se oduzima vrijednost 4 jer je varijacija na ruti 4E što znači da se iznos od 4 stupnjeva treba oduzeti od iznosa pravog smjera kako bi dobili magnetski smjer leta.

```
if(abs(trueTrack-wind)<=90):
        alpha = abs(trueTrack-wind)
    else:
        alpha = abs(trueTrack-(wind+180))

    CWC = windVelocity * math.sin(math.radians(alpha))
    TWC = windVelocity * math.cos(math.radians(alpha))
```

Naredbom „if“ zadan je uvjet „trueTrack-wind<=90“ te „alpha=abs(truetrack-wind)“ jer prilikom računanja ispravke kuta upada vjetra nam je bitan podatak o smjeru od kuda dolazi vjetar te kut upada vjetra na naš smjer leta koji se zove „Alpha”. Naredbom „if“ tražimo podatak je li nam je vjetar čeoni ili leđni te uz to postoji uvjet „else” koji se naknadno ispunjava ako uvjet „if“ nije zadovoljen. Komponente „cwc“ i „twc“ su formule za izračun vrijednosti jačine bočnog vjetra (eng. Cross Wind Component, hrv. komponenta bočnog vjetra) te leđnog (eng. Tail Wind Component, hrv. komponenta leđnog vjetra) ili čeonog vjetra (eng. Head Wind Component, hrv. komponenta čeonog vjetra).

```
GS = TAS + TWC
```

```
    DA = CWC*60/TAS
    WCA = 0-DA
    tU = 60*D/GS
    FuelRequired = GPH * tU / 60

return [round(FuelRequired,2), round(tU,2)]
```

Nadalje, „GS“ je formula za izračun putne brzine zrakoplova (eng. Ground Speed) u odnosu na Zemlju. „DA“ je formula za izračun kuta zanosa smjera leta (eng. Drift Angle) uzrokovan trenutnim vjetrom na etapi. „WCA“ (eng. Wind Correction Angle, hrv. kut ispravke vjetra) je vrijednost za koju se mora ispraviti smjer leta da bi se spriječilo zanošenje zrakoplova uzrokovanog vjetrom te kako bi se mogao nastaviti let po zadanoj ruti. „tU“ je formula za izračun vremena leta na etapi rute između dvije točke prilikom krstarenja. „fuel required“ je formula za izračun potrošenog goriva prilikom krstarenja.

Naredbom „return“ dobivaju se podatci o vrijednosti potrošenog goriva te vremena leta prilikom krstarenja zaokruženo na dvije decimale.

4.1.2. Python kodovi za fazu penjanja

```
def penjanje(visinaAerodroma, visinaPenjanja, trueTrack, wind, windVelocity):
visinaIspod = math.floor(visinaAerodroma/1000)*1000
    visinaIznad = math.ceil(visinaAerodroma/1000)*1000
penjanjeIspod = math.floor(visinaPenjanja/1000)*1000
    penjanjeIznad = math.ceil(visinaPenjanja/1000)*1000
```

„Def penjanje“ je funkcija za penjanje na željenu visinu krstarenja te su u zagradi ulazne varijable. Prvo se koristi podatak o nadmorskoj visini aerodroma, nakon toga se koristi visina penjanja na željenu visinu krstarenja. Tablice iz operativnog priručnika zrakoplova C172 su prikazane s poznatim podatcima svakih 1000 ft te se traži prva tisućica iznad i ispod vrijednosti te se odrađuje operacija interpoliranja ako željena visina nije navedena u tablici. Nakon toga se unose vrijednosti „true track“, „Wind direction“, te „wind velocity“.

```
f1 = timeFuelDistance[str(visinaIspod)][4]
    f2 = timeFuelDistance[str(visinaIznad)][4]
    fA = interpoliraj(f1, visinaIspod, f2, visinaIznad, visinaAerodroma)
f1 = timeFuelDistance[str(penjanjeIspod)][4]
    f2 = timeFuelDistance[str(penjanjeIznad)][4]
    fP = interpoliraj(f1, penjanjeIspod, f2, penjanjeIznad, visinaPenjanja)
 fU = fP - fA

t1 = timeFuelDistance[str(visinaIspod)][3]
    t2 = timeFuelDistance[str(visinaIznad)][3]
    tA = interpoliraj(t1, visinaIspod, t2, visinaIznad, visinaAerodroma)

    t1 = timeFuelDistance[str(penjanjeIspod)][3]
    t2 = timeFuelDistance[str(penjanjeIznad)][3]
    tP = interpoliraj(t1, penjanjeIspod, t2, penjanjeIznad, visinaPenjanja)

    tU = tP - tA

D1 = timeFuelDistance[str(visinaIspod)][5]
    D2 = timeFuelDistance[str(visinaIznad)][5]
    DA = interpoliraj(D1, visinaIspod, D2, visinaIznad, visinaAerodroma)

    D1 = timeFuelDistance[str(penjanjeIspod)][5]
    D2 = timeFuelDistance[str(penjanjeIznad)][5]
    DP = interpoliraj(D1, penjanjeIspod, D2, penjanjeIznad, visinaPenjanja)

    DU = DP - DA
```

Računa se potrošnja goriva prema podatcima iz tablice te se koristi operacija interpolacije ako je potrebno. Računa se i vrijeme leta prema podatcima iz tablice te prema podatcima o smjeru i jačini vjetra i također se koristi operacija interpolacije ako je potrebno. Računa se i prijeđena udaljenost prilikom penjanja do željene visine krstarenja te se također u obzir uzima smjer i jačina vjetra.

```
if(abs(trueTrack-wind)<=90):
```

```
        alpha = abs(trueTrack-wind)
    else:
        alpha = abs(trueTrack-(wind+180))

    CWC = windVelocity * math.sin(math.radians(alpha))
    HWC = (-1) * windVelocity * math.cos(math.radians(alpha))
```

Naredbom „if“ zadan je uvjet „trueTrack-wind<=90“ te „alpha=abs(truetrack-wind)“ jer prilikom računanja ispravke kuta upada vjetra bitan je podatak o smjeru od kuda dolazi vjetar te kut upada vjetra na zadani smjer leta koji se zove „Alpha“. Naredbom „if“ traži se podatak je li vjetar čeoni ili leđni te uz to postoji uvjet „else“ koji se naknadno ispunjava ako uvjet „if“ nije zadovoljen. Komponente „cwc“ i „twc“ su formule za izračun vrijednosti jačine bočnog vjetra te leđnog ili čeonog vjetra.

```
tas1 = timeFuelDistance[str(visinaIspod)][1]
    tas2 = timeFuelDistance[str(visinaIznad)][1]
    TAS = interpoliraj(tas1, visinaIspod, tas2, visinaIznad, visinaPenjanja)
    WCA = 60 * CWC / TAS
```

Računa se potrošnja goriva prema podatcima iz tablice te prema podatcima o smjeru i jačini vjetra.

```
if wind - trueTrack == 0 or wind-trueTrack == 180:
        trueHDG = trueTrack
    elif wind - trueTrack > 0:
        trueHDG = trueTrack + WCA
    elif wind - trueTrack < 0:
        trueHDG = trueTrack - WCA

magHDG = trueHDG - 4
```

Računa se podatak o pravom smjeru leta koristeći „true track“ te „wca“ te se računa magnetski smjer leta.

```
if abs(wind-trueTrack) <= 90:
        GS = TAS + HWC
    else:
        GS = TAS - HWC

FF = fU * 60 / tU

return [round(fU,2), round(tU,2), round(D,2)]
```

Računa se vrijednost putne brzine zrakoplova u odnosu na Zemlju „GS“ u ovisnosti imamo li leđnu ili čeonu komponentu vjetra. Ako je čeoni vjetar, iznos jačine čeonog vjetra se oduzima od vrijednosti stvarne brzine leta „TAS“, a ako je leđni vjetar, zbraja se s brzinom „TAS“. Leđna komponenta vjetra je u programu označena s „+HWC“.

Računa se potrošnja goriva prilikom penjanja na visinu krstarenja. Naredba “return” vraća podatke o potrošenom gorivu “fu”, vremenu “tu” te prijeđenoj udaljenosti “D”.

```
for visinaPenjanja in [item for item in np.linspace(2000,10000,81)]:
p = penjanje(325, visinaPenjanja, 192, 10, 5)
```

Funkcija „for“ označava petlju, a ona nam vrti sve visine penjanja od 2000 ft do 10000 ft te tih visina ima onoliko koliko ih se postavi. Brojke unutar zagrade označavaju početnu visinu 2000 ft te konačnu visinu 10000 ft te zadnja brojka označava broj koliko vertikalnih profila leta želimo prikazati na grafu između početne i konačne visine.

U varijablu „p“ se spremaju podatci koje daje funkcija za penjanje te ona vraća podatak o potrošenom gorivu, vremenu leta te prijeđenoj udaljenost do „toc“ (eng. top of climb, hrv. točka vrhunca penjanja).

4.1.3. Python kodovi za provjeru udaljenosti TOC

```
# fuel<
```

```
    fU = p[0]
    # ukupno vrijeme
    tt = p[1]
    # udaljenost do TOC
    toc = p[2]

    if(p[2]<15.8):
        print('prije smo dosli')
        trenutanDistance = p[2]
```

Naredbom „if“ provjerava se je li zrakoplov prilikom penjanja na željenu visinu leta preletio prvu točku rute „S1“ koja se nalazi na udaljenosti 15,8 NM te taj podatak ima značenje je li režim krstarenja započinje prije prve rutne točke, na prvoj rutnoj točki ili nakon prve rutne točke.

```
# wind

        if(visinaPenjanja < 5000):
            # interpoliramo smjer vjetra
            windispod = flyingConditions['2000'][0][1][0]
            windiznad = flyingConditions['5000'][0][1][0]
            wind = interpoliraj(windispod, 2000, windiznad, 5000,
visinaPenjanja)
```

Koristi se operacija interpolacije prema poznatim podatcima o smjeru i jačini vjetra na 2000, 5000 te 10 000 ft. Interpolacija se odrađuje u dijelu između 2000 te 5000 ft te u dijelu između 5000 te 10 000 ft.

```
# interpoliramo brzinu vjetra
            windispod = flyingConditions['2000'][0][1][1]
            windiznad = flyingConditions['5000'][0][1][1]
            windVelocity = interpoliraj(windispod, 2000, windiznad, 5000,
visinaPenjanja)
```

```
        elif(visinaPenjanja == 5000):
# interpoliramo smjer vjetra
            wind = flyingConditions['5000'][0][1][0]
            windVelocity = flyingConditions['5000'][0][1][1]
        else:
# interpoliramo smjer vjetra
            windispod = flyingConditions['5000'][0][1][0]
            windiznad = flyingConditions['10000'][0][1][0]
            wind = interpoliraj(windispod, 5000, windiznad, 10000,
visinaPenjanja)

# interpoliramo brzinu vjetra
            windispod = flyingConditions['5000'][0][1][1]
            windiznad = flyingConditions['10000'][0][1][1]
            windVelocity = interpoliraj(windispod, 5000, windiznad, 10000,
visinaPenjanja)


        var = kruzanje(15.8-trenutanDistance,192,wind,windVelocity,
visinaPenjanja)
        fU += var[0]
        tt += var[1]
```

Odrađuje se operacija interpolacije prema zadanim podatcima o smjeru i jačini vjetra prema podatcima iz tablice gdje se nalaze poznati podatci o vjetru na 2000, 5000 i 1000 ft. U jednom slučaju odrađuje se operacija interpolacije ako je visina leta između 2000 i 5000 ft te u drugom slučaju ako je visina leta između 5000 i 1000 ft.

4.2. Grafički prikaz vertikalnog profila

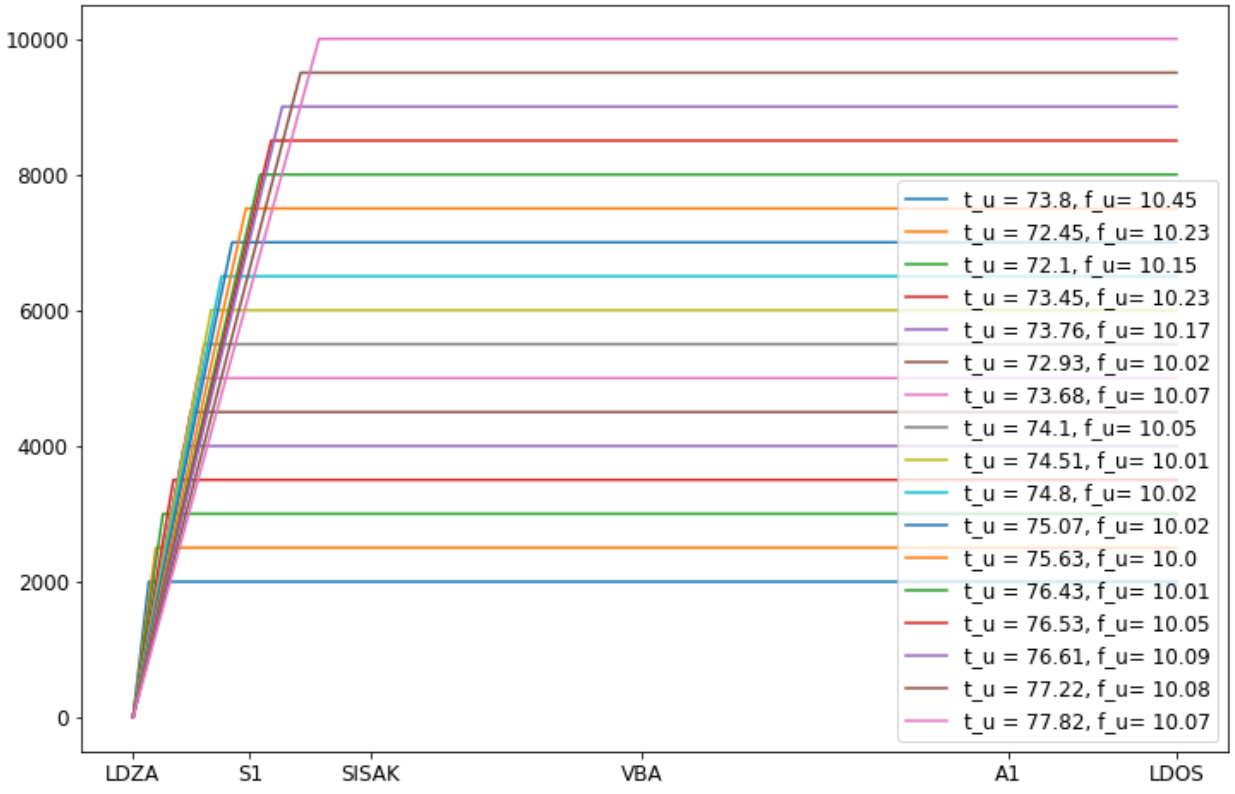
Visina zrakoplova je podatak koji određuje kako će izgledati vertikalni profil zrakoplova. Linije na grafikonu koji je prikazan na slici 7 prikazuju udaljenost koju ćemo prijeći u odnosu na Zemlju penjući na željenu visinu krstarenja te krstarenje na željenoj visini.

Vertikalni profil zrakoplova prikazan je raznobojnim linijama koje su podijeljene svakih 500 ft. Grafikon može biti prikazan s manjim i s većim razmakom. Razmak je svakih 500 ft radi

ravnomjernog i jasnog prikaza podataka, ali u rezultatima je korišten razuman razmak od 100 ft što povećava preciznost dobivenih rezultata.

U desnom dijelu grafikona prikazani su podatci o vremenu leta i potrošenom gorivu za taj let te su podatci napravljeni u raznim bojama da bi se mogle vidjeti različitosti među njima.

Grafikon vertikalnog profila zrakoplova započinje u točki aerodroma polijetanja „LDZA“ te kreće iz točke polijetanja zrakoplova penjući na željenu visinu krstarenja. Linije na grafikonu se kreću do aerodroma slijetanja „LDOS“ te u navedenom primjeru ne postoji segment spuštanja jer se radi o jednomotornom klipnom zrakoplovu gdje je zanemariva razlika između potrošnje u spuštanju i horizontalnom letu pa se računa potrošnja u horizontalnom letu skroz do odredišnog aerodroma. Ovisno na koju visinu krstarenja zrakoplov penje, grafikon će za svaku od ponuđenih visina biti drugačiji. Iz grafikona je jasno vidljivo da se na visinu krstarenja može doći prije, poslije ili na prvoj rutnoj točki „S1“.



Slika 6: Vertikalni profil leta [11]

Izračuni koji su bili potrebni za stvaranje grafikona vertikalnog profila leta temelje se na podatcima iz operativnog priručnika zrakoplova. Podatci iz operativnog priručnika zrakoplova Cessna 172 N prikazani su u tablicama 2 i 3.

Tablica 2: Performanse krstarenja (eng. cruise performance) [12]

Visina leta (ft)	RPM	%BHP	KTAS	GPH
2000	2500	75	116	8, 4
	2300	60	105	6,7
	2100	47	94	5, 6
4000	2500	71	115	8
	2300	57	105	6,4
	2100	46	93	5, 5
6000	2500	67	115	7, 6
	2300	54	104	6,2
	2100	44	92	5, 4
8000	2500	64	114	7, 2
	2300	52	103	6
	2200	47	97	5, 5
10 000	2500	61	114	6, 8
	2300	50	102	5,8
	2200	45	96	5, 4

Tablica 3: Performanse penjanja (eng. cruise performance) [12]

Visina leta (ft)	Temp. (° Cel.)	Brzina penjanja (kias)	Brzina uzdizanja (fpm)	Vrijeme (min)	Potrošeno gorivo (gallons)	Distance (NM)
0 (razina mora)	15	73	770	0	0	0
1000	13	73	725	1	0,3	2
2000	11	72	675	3	0,6	3
3000	9	72	630	4	0,9	5
4000	7	71	580	6	1,2	8

5000	5	71	535	8	1,6	10
6000	3	70	485	10	1,9	12
7000	1	69	440	12	2,3	15
8000	-1	69	390	15	2,7	19
9000	-3	68	345	17	3,2	22
10 000	-5	68	295	21	3,7	27

Uz prethodno navedene podatke potrebne za izradu grafikona vertikalnog profila leta, također su bitni podatci o smjeru i jačini vjetra na ruti. Podatci o temperaturi, smjeru i jačini vjetra uzeti su s Internet stranica meteoroloških podataka „Hrvatske Kontrole Zračne Plovidbe“ te su prikazani su tablici 4.

Tablica 4: Vjetar (eng. wind) [13]

Visina (ft)	**t (°C)** **w/v (°/kts)**	**S1-SISAK**	**SISAK-VBA**	**VBA-A1**	**A1-LDOS**
2000	t	19	20	20	20
	w/v	010/5	350/5	355/5	020/10
5000	t	12	12	13	13
	w/v	050/10	045/10	040/5	045/5
10 000	t	1	0	0	0
	w/v	060/10	040/10	050/13	045/15

5. Rezultati proračuna

Rezultati izračuna u programu „Python“ nastali su izračunom podataka o potrošenom gorivu te vremenu leta od aerodroma polijetanja do aerodroma slijetanja. Program je izračunao podatke o potrošnji goriva te vremenu leta svakih 100 ft visine te su u konačnici nastali podatci od 2000 ft do 10 000 ft.

Cilj prikupljanja podataka te izračuna koji su nastali na temelju performansi zrakoplova te utjecaja vjetra na ruti su određivanje optimalne visine leta za let na ruti od Zagreba do Osijeka. Optimalna visina može biti visina leta gdje će zrakoplov potrošiti najmanju količinu goriva ili visina leta na kojoj će zrakoplov najbrže stići do odredišta ovisno o željenom ishodu. Traži se visina leta za najekonomičniji i za najbrži let do odredišta.

Prilikom računanja željenih podataka u programu korištene su tri različite postavke snage motora zrakoplova.

Postavke snage su različite kako bi se što bolje pokazale performanse zrakoplova pri različitim režimima rada motora za let od Zagreba do Osijeka. Najveća postavka snage zrakoplova je 2500 RPM te je u navedenom režimu potrošnja goriva najveća, ali je i brzina zrakoplova najveća što će uzrokovati najkraće vrijeme leta. Srednja postavka snage je 2300 RPM te je to režim leta između najveće i najmanje potrošnje goriva. Postavka snage motora od 2100 RPM je postavka gdje je potrošnja goriva najmanja što će kao rezultat dati najduže vrijeme leta do odredišnog aerodroma. Najmanja postavka snage prema priručniku zrakoplova je 2200 RPM za visinu 8000 ft te 10 000 ft. Različite postavke snage se koriste prilikom leta ovisno je li cilj stići na odredište s najmanjom potrošnjom goriva što će uzrokovati najduže vrijeme leta ili je cilj što prije stići na odredište što će kao posljedicu imati veću potrošnju goriva.

Nakon prikupljanja podataka te izračuna uz pomoć svih navedenih tablica i različitih postavka snage motora zrakoplova, dobiveni su željeni rezultati o potrošnji goriva te vremenu leta koji su prikazani u tablici 5.

Tablica 5: Rezultati [11]

	2500 RPM		2300 RPM		2100 RPM	
Visina leta (ft)	Vrijeme leta (min)	Potrošeno gorivo (gph)	Vrijeme leta (min)	Potrošeno gorivo (gph)	Vrijeme leta (min)	Potrošeno gorivo (gph)
2000	73,8	10,45	81,17	9,27	90,27	8,67
2100	73,51	10,4	80,76	9,21	89,79	8,63
2200	73,2	10,35	80,36	9,18	89,31	8,61
2300	72,91	10,31	79,96	9,13	88,86	8,59
2400	72,67	10,28	79,61	9,09	88,44	8,55
2500	72,45	10,23	79,3	9,07	88,08	8,54
2600	72,28	10,21	79,05	9,02	87,82	8,53
2700	72,16	10,17	78,85	9,01	87,6	**8,51**
2800	72,07	10,16	78,72	8,99	87,45	8,52
2900	**72,06**	10,15	**78,66**	8, 99	87,41	8,53
3000	72,1	10,15	78,67	8,99	87,45	8,54
3100	72,27	10,16	78,79	8,98	87,63	8,56
3200	72,47	10,17	79	9	87,91	8,6
3300	72,74	10,18	79,26	9,02	88,24	8,63
3400	73,07	10,21	79,59	9,04	88,68	8,69
3500	73,45	10,23	79,98	9,08	89,19	8,73
3600	73,62	10,24	80,09	9,08	89,34	8,74
3700	73,65	10,22	80,1	9,07	89,34	8,76
3800	73,73	10,22	80,14	9,06	89,41	8,77
3900	73,86	10,22	80,23	9,07	89,55	8,78
4000	73,76	10,17	80,06	9,03	89,34	8,76
4100	73,51	10,14	79,76	9,02	88,92	8,75
4200	73,28	10,09	79,51	8,98	88,55	8,72
4300	73,1	10,06	79,28	8,97	88,24	8,71
4400	72,99	10,04	79,15	8,97	88,04	8,71
4500	72,93	10,02	79,1	8,97	87,92	8,72
4600	72,95	10,01	79,1	**8,96**	87,9	8,72
4700	73,02	10,01	79,2	8,99	87,99	8,75
4800	73,18	10,03	79,37	9,02	88,19	8,79
4900	73,4	10,06	79,64	9,04	88,49	8,84

	2500 RPM		2300 RPM		2100 RPM	
Visina leta (ft)	**Vrijeme leta (min)**	**Potrošeno gorivo (gph)**	**Vrijeme leta (min)**	**Potrošeno gorivo (gph)**	**Vrijeme leta (min)**	**Potrošeno gorivo (gph)**
5000	73,68	10,07	80,01	9,09	88,87	8,88
5100	73,76	10,07	80,11	9,1	88,97	8,9
5200	73,86	10,07	80,21	9,11	89,08	8,91
5300	73,94	10,06	80,31	9,13	89,17	8,92
5400	74,03	10,04	80,4	9,13	89,25	8,93
5500	74,1	10,05	80,52	9,13	89,33	8,95
5600	74,18	10,03	80,61	9,14	89,44	8,97
5700	74,27	10,04	80,71	9,14	89,51	8,97
5800	74,33	10,02	80,8	9,16	89,59	8,98
5900	74,42	10,03	80,9	9,17	89,69	8,99
6000	74,51	10,01	81,01	9,18	89,78	9,01
6100	74,56	10,02	81,06	9,18	89,55	9,01
6200	74,62	10,01	81,1	9,2	89,33	9,03
6300	74,69	10,03	81,14	9,21	89,13	9,03
6400	74,74	10,01	81,18	9,21	88,91	9,05
6500	74,8	10,02	81,24	9,23	88,68	9,05
6600	74,86	10,01	81,28	9,24	88,49	9,08
6700	74,91	10,03	81,31	9,26	88,27	9,07
6800	74,97	10,02	81,36	9,27	88,08	9,08
6900	75,02	10,02	81,39	9,28	87,87	9,09
7000	75,07	10,02	81,43	9,3	87,66	9,1
7100	75,2	10,02	81,53	9,29	87,51	9,11
7200	75,31	10,01	81,61	9,3	87,36	9,12
7300	75,42	10,01	81,7	9,32	87,23	9,12
7400	75,52	10	81,8	9,32	87,07	9,12
7500	75,63	10	81,89	9,33	86,95	9,13
7600	75,73	**9,99**	81,97	9,34	86,79	9,13
7700	76,15	10,03	82,4	9,38	87,0	9,18
7800	76,24	10,02	82,46	9,39	86,84	9,18
7900	76,34	10,02	82,54	9,38	86,69	9,19
8000	76,43	10,01	82,62	9,4	**86,54**	9,18

	2500 RPM		2300 RPM		2100 RPM	
Visina leta (ft)	**Vrijeme leta (min)**	**Potrošeno gorivo (gph)**	**Vrijeme leta (min)**	**Potrošeno gorivo (gph)**	**Vrijeme leta (min)**	**Potrošeno gorivo (gph)**
8100	76,45	10,03	82,64	9,42	86,56	9,22
8200	76,47	10,03	82,67	9,43	86,58	9,25
8300	76,49	10,04	82,7	9,46	86,59	9,27
8400	76,51	10,05	82,72	9,48	86,6	9,3
8500	76,53	10,05	82,75	9,5	86,62	9,31
8600	76,54	10,06	82,78	9,53	86,63	9,35
8700	76,56	10,07	82,8	9,54	86,64	9,38
8800	76,58	10,07	82,82	9,57	86,66	9,41
8900	76,6	10,09	82,86	9,59	86,68	9,44
9000	76,61	10,09	82,88	9,61	86,7	9,45
9100	76,73	10,1	82,99	9,62	86,79	9,48
9200	76,85	10,09	83,11	9,63	86,89	9,49
9300	76,98	10,1	83,22	9,64	87	9,51
9400	77,1	10,08	83,33	9,66	87,09	9,54
9500	77,22	10,08	83,46	9,66	87,19	9,55
9600	77,34	10,08	83,58	9,68	87,28	9,57
9700	77,45	10,08	83,69	9,69	87,38	9,57
9800	77,57	10,08	83,81	9,7	87,48	9,59
9900	77,69	10,08	83,92	9,71	87,58	9,61
10 000	77,82	10,07	84,03	9,73	87,68	9,63

6. Analiza rezultata

Podaci u tablici 5 prikazuju željene rezultate testiranja algoritama na više etapa rute leta od Zagreba do Osijeka. Tablica sadrži podatke o vremenu leta i potrošnji goriva na visinama od 2000 do 10 000 ft s tri različite postavke snage motora zrakoplova. Vjetar na ruti nije znatno utjecao na razlike u rezultatima. Smjer vjetra na ruti bio je sjeverni prema sjeverno istočnom s maksimalnom komponentom jačine vjetra od 15 čvorova te je taj maksimum jedino vidljiv u zadnjoj etapi A1-LDOS na visini 10 000 ft. Većim dijelom rute jačina vjetra bila je 5-10 čvorova te je većinom vjetar bio lijevi bočni te leđni prilikom leta prema odredištu.

Vrijeme leta najkraće je na visini od 2900 stopa te kod postavke snage motora zrakoplova od 2500 RPM i iznosi 72,06 minute. U tome slučaju vrijeme leta je najkraće jer motor zrakoplova radi u režimu najveće snage gdje su okretaji najveći, a sukladno tome brzina zrakoplova je najveća, i s time zrakoplov u najkraćem vremenu stigne do odredišta. Sukladno postavci najveće snage, potrošnja goriva prikazana u tablici u režimu 2500 RPM je veća od vrijednosti potrošnje goriva u ostala dva režima rada motora zbog sagorijevanja veće količine goriva. Na visini 7600 stopa najmanja je potrošnja goriva u usporedbi s ostalom potrošnjom u dijelu režima 2500 RPM i iznosi 9,99 GPH, a vrijeme leta na toj visini iznosi 75,73 minute i to vrijeme leta je za 3,67 minute duže od najkraćeg vremena leta na visini 2900 stopa. Kako bi zrakoplov na odredište stigao relativno brzo i s najmanjom mogućom potrošnjom goriva za taj režim rada motora visina za ekonomski dolet biti će 7600 stopa.

Za srednju postavku snage od 2300 RPM rezultati se razlikuju od prethodnih. Zadana postavka snage daje srednju vrijednost rezultata koji se nalaze unutar dobivenih rezultata postavka 2500 RPM i 2100 RPM. Najkraće vrijeme leta pri navedenoj postavci iznosi 78,66 minute na visini leta od 2900 stopa, a potrošnja goriva iznosi 8,99 galona po satu. Zanimljiv podatak je da je za 2500 RPM i 2300 RPM rezultat najkraćeg vremena leta na istoj visini od 2900 stopa. Za razliku od 2500 RPM pri srednjoj postavci snage vrijeme leta na toj visini biti će duže za 6,6 minuta. Potrošnja goriva na visini najkraćeg vremena leta pri srednjoj postavci snage iznosi 8,99 galona po satu te je potrošnja goriva za iznos 1 galona po satu leta zrakoplova veća od rezultata za istu visinu pri postavci snage 2500 RPM. Za srednju postavku snage najmanja

potrošnja goriva dobivena je za visinu 4600 stopa te iznosi 8,96 galona po satu, a vrijeme leta iznosi 79,1 minutu. Optimalna visina leta za srednju postavku snage može biti 4600 stopa jer je na toj visini leta potrošnja goriva najmanja, a vrijeme leta se od rezultata najkraćeg vremena leta razlikuje za 0,44 minute što je zanemarivo. U konačnici mogu se uzeti dvije visine kao optimalne visine leta, a razlika u potrošnji goriva između visine ekonomskog i najbržeg doleta iznosi 0,03 galona po satu što je također zanemarivo. Visina od 2900 ft osigurava nadvisivanje terena na odabranoj ruti, ali radi sigurnosti bolje bi bilo odabrati visinu od 4600 stopa kao optimalnu visinu jer su razlike u rezultatima između dviju visina zanemarive.

Najmanja postavka snage zrakoplova u fazi krstarenja je 2100 RPM. Najkraće vrijeme leta u navedenoj postavci snage iznosi 86,54 minute na visini leta od 8000 ft, a potrošnja goriva iznosi 9,18 galona po satu. U podatcima iz tablica za krstarenje vidljivo je da na 8000 ft zrakoplov prelazi u veću postavku snage od 2200 RPM što uzrokuje veću indiciranu brzinu leta (eng. indicated airspeed, IAS) stoga podatci od 8000-10000 ft za RPM 2100 ne bi trebali biti važeći. Vrijeme leta je kraće u usporedbi s rezultatima dobivenima koristeći ostale dvije postavke snage zbog manjih okretaja u radu motora te manje brzine zrakoplova prilikom krstarenja, ali pozitivan ishod je manja potrošnja goriva. Najmanja potrošnja goriva pri navedenoj postavci iznosi 8,51 galona po satu na visini leta od 2700 stopa, a vrijeme leta na toj visini iznosi 87,6 minuta. Optimalna visina leta za navedenu postavku snage može biti 2700 stopa jer su rezultati nakon 8000 stopa nevažeći. Konačni rezultati optimalnih visina prikazani su tablicom 6.

Tablica 6: Rezultati optimalne visine leta

	Postavke snage motora								
	2500			**2300**			**2100**		
	Visina leta (ft)	Potrošnja goriva (gal)	Vrijeme leta (min)	Visina leta (ft)	Potrošnja goriva (gal)	Vrijeme leta (min)	Visina leta (ft)	Potrošnja goriva (gal)	Vrijeme leta (min)
Ekonomičan let	7600	9,99	75,73	4600	8,96	79,1	2700	8,51	87,6
Najbrži let	2900	10,15	72,06	2900	8,99	78,66	7900	9,19	86,69

7. Zaključak

Optimalna visina leta ovisi o željenom ishodu, je li cilj na odredišni aerodrom stići s ekonomičnim ili najbržim doletom. Atmosferski uvjeti i meteorološka situacija imaju utjecaj na rezultate vrijednosti potrošnje goriva i vremena u zrakoplovstvu. Podatci o potrošnji goriva te vremenu leta mogu detaljno biti izračunati prije leta te tako pomoći u pripremi leta te mogućoj uštedi na vremenu i gorivu odabirom optimalne visine leta na ruti. Zrakoplov ima mogućnost koristiti različite postavke snage motora koje uzrokuju različitu potrošnju goriva i vrijeme leta do odredišnog aerodroma. Vjetar utječe na izračuna koji se računaju prije leta i njegov utjecaj na zrakoplov počinje prilikom zatrčavanja za polijetanje i prestaje tek nakon slijetanja. S visinom leta potrošnja goriva se smanjuje te je najčešće veća visina optimalna visina leta no vjetar na ruti i ostali atmosferski uvjeti mogu utjecati tako da je niža visina ipak optimalnija za let ovisno o meteorološkoj situaciji na drugim visinama.

Završnom radu prilikom određivanja optimalne visine leta korištene su 3 različite postavke snage. Postavke 2500 RPM i 2300 RPM dali su konačne rezultate koji potvrđuju da je manja potrošnja goriva na većoj visini te da će najbrži let biti na manjim visinama jer nećemo puno vremena izgubiti penjanjem na veću visinu te ćemo stići na odredište brže. Postavka snage 2100 RPM kao rezultat dala je obrnute rezultate. Dobiveno je da je s najmanjom postavkom snage na nižoj visini manja potrošnja goriva, a na većoj visini najbrži let do odredišta. Dobiveni rezultati mogu se smatrati nevažećima jer je vidljivo iz podataka operativnog priručnika zrakoplova da zrakoplov na visini od 8000 do 10000 ft prelazi u veću postavku snage od 2200 RPM te je za navedene visine to najmanja postavka snage.

Proračuni koji su prikupljani za ovaj završni rad rađeni su u programskom jeziku Python te je program uvelike olakšao dobivanje potrebnih rezultata za let na odabranoj ruti.

U zrakoplovstvu izračuni te njihove vrijednosti imaju velik utjecaj na odabir optimalne visine leta stoga je potrebno prije leta pribaviti točne podatke o vjetru na ruti i imati točne podatke o performansama zrakoplova za taj let kako bi se uspješno odabrala optimalna visina leta.

LITERATURA

[1] Take off and landing performance, Civil Aviation Authority of New Zealand, 2020., https://www.aviation.govt.nz/assets/publications/gaps/Take-off-and-landing-performance.pdf (20.8.2021.)

[2] https://www.easa.europa.eu/sites/default/files/dfu/CS-23%20Initial%20issue.pdf

[3] Pilot's Handbook of Aeronautical Knowledge, Aircraft performance, FAA, 2021., https://www.faa.gov/regulations_policies/handbooks_manuals/aviation/phak/media/13_phak_ch11.pdf (22.8.2021.)

[4] Forces Acting on the Aircraft – Aerodynamics of Flight, Aeronautics Guide, 2017., https://www.aircraftsystemstech.com/2017/05/forces-acting-on-aircraft.html (4.9.2021.)

[5] B.D. Dancila, R.M. Botez, Vertical flight path segments sets for aircraft flight plan prediction and optimisation, The Aeronautical Journal, Volume 122, Issue 1255, 2018., https://www.cambridge.org/core/journals/aeronautical-journal/article/abs/vertical-flight-path-segments-sets-for-aircraft-flight-plan-prediction-and-optimisation/4424B963EB6038779EFD8F821C415F60 (3.8.2021.)

[6] John D. Anderson Jr., Aircraft performance and design, 2010., https://www.academia.edu/40606141/AIRCRAFT_PERFORMANCE_AND_DESIGN (20.8.2021.)

[7] B. D. Dancila, R. M. Botez, Vertical flight segments sets for aircraft flight plan prediction and optimisation, Aeronautical Journal – New Series, 2018., https://www.researchgate.net/publication/326210471_Vertical_flight_path_segments_sets_for_aircraft_flight_plan_prediction_and_optimisation/link/5c6437a592851c48a9d23e81/download (2.8.2021.)

[8] R. S. F. Patron, R.M. Botez, A. Kessaci, Horizontal Flight Trajectories Optimization for Commercial Aircraft through a Flight Management System, Aeronautical Journal – New Series, 2014., (PDF) Horizontal Flight Trajectories Optimization for Commercial Aircraft through a Flight Management System (researchgate.net) (4.8.2021.)

[9] Novak D.: Planiranje letenja i performanse II, Fakultet prometnih znanosti, Zagreb, 2015.

[10] E. Witman, What is Phyton? The popular, scalable programming language, explained, Insider, 2021., https://www.businessinsider.com/what-is-python (10.8.2021.)

[11] Python program

[12] Cessna Aircraft Company Wichita, Pilot's Operating Handbook Cessna, Kansas; 1977.

[13]Hrvatska kontrola zračne plovidbe, Služba zrakoplovne meterologije, https://met.crocontrol.hr/hr/web/guest/23 (15.8.2021.)

Popis tablica

Tablica 1: Rutne točke 12
Tablica 2: Performanse krstarenja (eng. cruise performance) [12] 24
Tablica 3: Performanse penjanja (eng. cruise performance) [12] 24
Tablica 4: Vjetar (eng. wind) [13] 25
Tablica 5: Rezultati [11] 27

Popis slika

Slika 1: Primjer penjanja povećavanjem snage 5
Slika 2: Primjer penjanja s konstantnom snagom 6
Slika 3: Sile koje djeluju na zrakoplov u horizontalnom letu [4] 7
Slika 4: Dijagram ovisnosti potrebne snage s brzinom zrakoplova [5] 8
Slika 5: Dijagram toka 13
Slika 7: Vertikalni profil leta [11] 23

Prilog 1

```
 import numpy as np
import math
from matplotlib import pyplot as plt

def interpoliraj(fa, a, fb, b, x):
    if x == a:
        return fa
    if x == b:
        return fb
    else:
        return fa + ((fb-fa)*(x-a)/(b-a))


# cruise performance tablica
# u unutarnjim poljima, prvo mjesto je %BHP, drugo je KTAS, trece je GPH

ft2000 = {
    "2500" : [47,94,5.6],
    "2400" : [67, 111, 7.5],
    "2100" : [47,94,5.6]
}

ft4000 = {
    "2500" : [46,93,5.5],
    "2100" : [46,93,5.5]
}

ft6000 = {
    "2500" : [44,92,5.4],
    "2100" : [44,92,5.4],
}

ft8000 = {
    "2500" : [47,97,5.5],
    "2200" : [47,97,5.5]
```

```
}

ft10000 = {
    "2500" : [45,96,5.4],
    "2200" : [45,96,5.4]
}

cruisePerformance = {
    "2000"  : ft2000,
    "4000"  : ft4000,
    "6000"  : ft6000,
    "8000"  : ft8000,
    "10000" : ft10000
}


# time, fuel and distance to climb
# varijable su redom: visina (ft), temp(C), climb speed (kias), roc (fpm)
# time (min), fuel used (gallons), distance (nm)

timeFuelDistance = {
    "0"    : [15, 73, 770, 0, 0,0],
    "1000" : [13, 73, 725, 1, 0.3, 2],
    "2000" : [11, 72, 675, 3, 0.6, 3],
    "3000" : [9, 72, 630, 4, 0.9, 5],
    "4000" : [7, 71, 580, 6, 1.2, 8],
    "5000" : [5, 71, 535, 8, 1.6, 10],
    "6000" : [3, 70, 485, 10, 1.9, 12],
    "7000" : [1, 69, 440, 12, 2.3, 15],
    "8000" : [-1, 69, 390, 15, 2.7, 19],
    "9000" : [-3, 68, 345, 17, 3.2, 22],
    "10000": [-5, 68, 295, 21, 3.7, 27],
    "12000": [-9, 67, 200, 29, 4.9, 38]
}
# prvo biramo visinu, onda biramo tocku
# 1: LDZA -> S1, 2: S1->Sisak, 3: Sisak->VBA, 4: VBA->A1, 5:A1->2DOS
```

```
# onda biramo temperaturu i vjetar, vjetar ima smjer i jacinu

flyingConditions = {
    "2000" :  [[19,[15,7.5]],[19,[10,5]], [20,[350,5]], [20,[355,5]], [20,
[20,10]]],
    "5000" :  [[12,[20,10]],[12,[50,10]],[12,[45,10]], [13,[40,5]],  [13,[
45,5]]],
    "10000" : [[1,[60,10]],[1,[60,10]], [0,[40,10]],  [0,[50,13]],  [0,[45
,15]]]
}


def kruzanje(D, trueTrack, wind, windVelocity, visina):
    # visina u ft
    visinaIspod = math.floor(visina/2000)*2000
    visinaIznad = math.ceil(visina/2000)*2000
    #TAS
    TASispod = cruisePerformance[str(visinaIspod)]["2500"][1]
    TASiznad = cruisePerformance[str(visinaIznad)]["2500"][1]
    TAS = interpoliraj(TASispod, visinaIspod, TASiznad, visinaIznad, visin
a)
    #GPH
    GPHispod = cruisePerformance[str(visinaIspod)]["2500"][2]
    GPHiznad = cruisePerformance[str(visinaIznad)]["2500"][2]
    GPH = interpoliraj(GPHispod, visinaIspod, GPHiznad, visinaIznad, visin
a)
    HDG = trueTrack - 4

    # wind correction angle
    if(abs(trueTrack-wind)<=90):
        alpha = abs(trueTrack-wind)
    else:
        alpha = abs(trueTrack-(wind+180))

    CWC = windVelocity * math.sin(math.radians(alpha))
    TWC = windVelocity * math.cos(math.radians(alpha))
```

```
    GS = TAS + TWC

    DA = CWC*60/TAS
    WCA = 0-DA
    tU = 60*D/GS
    FuelRequired = GPH * tU / 60

    return [round(FuelRequired,2), round(tU,2)]


def penjanje(visinaAerodroma, visinaPenjanja, trueTrack, wind, windVelocit
y):
    # visinaAerodroma = 325
    visinaIspod = math.floor(visinaAerodroma/1000)*1000
    visinaIznad = math.ceil(visinaAerodroma/1000)*1000


    #visinaPenjanja = 10000
    penjanjeIspod = math.floor(visinaPenjanja/1000)*1000
    penjanjeIznad = math.ceil(visinaPenjanja/1000)*1000

    # racunamo potrosnju goriva
    f1 = timeFuelDistance[str(visinaIspod)][4]
    f2 = timeFuelDistance[str(visinaIznad)][4]
    fA = interpoliraj(f1, visinaIspod, f2, visinaIznad, visinaAerodroma)

    f1 = timeFuelDistance[str(penjanjeIspod)][4]
    f2 = timeFuelDistance[str(penjanjeIznad)][4]
    fP = interpoliraj(f1, penjanjeIspod, f2, penjanjeIznad, visinaPenjanja
)
    fU = fP - fA

    # racunamo vrijeme
    t1 = timeFuelDistance[str(visinaIspod)][3]
    t2 = timeFuelDistance[str(visinaIznad)][3]
```

```
    tA = interpoliraj(t1, visinaIspod, t2, visinaIznad, visinaAerodroma)

    t1 = timeFuelDistance[str(penjanjeIspod)][3]
    t2 = timeFuelDistance[str(penjanjeIznad)][3]
    tP = interpoliraj(t1, penjanjeIspod, t2, penjanjeIznad, visinaPenjanja
)

    tU = tP - tA

    # racunamo udaljenost
    D1 = timeFuelDistance[str(visinaIspod)][5]
    D2 = timeFuelDistance[str(visinaIznad)][5]
    DA = interpoliraj(D1, visinaIspod, D2, visinaIznad, visinaAerodroma)

    D1 = timeFuelDistance[str(penjanjeIspod)][5]
    D2 = timeFuelDistance[str(penjanjeIznad)][5]
    DP = interpoliraj(D1, penjanjeIspod, D2, penjanjeIznad, visinaPenjanja
)

    DU = DP - DA



    # iz karte smo ocitali vjetar
    # trueTrack, wind u stupnjevima
    # windVelocity u cvorovima


    # wind correction angle
    if(abs(trueTrack-wind)<=90):
        alpha = abs(trueTrack-wind)
    else:
        alpha = abs(trueTrack-(wind+180))

    CWC = windVelocity * math.sin(math.radians(alpha))
    HWC = (-1) * windVelocity * math.cos(math.radians(alpha))
```

```
    # racunamo potrosnju goriva
    tas1 = timeFuelDistance[str(visinaIspod)][1]
    tas2 = timeFuelDistance[str(visinaIznad)][1]
    TAS = interpoliraj(tas1, visinaIspod, tas2, visinaIznad, visinaPenjanj
a)
    WCA = 60 * CWC / TAS


    # true heading
    if wind - trueTrack == 0 or wind-trueTrack == 180:
        trueHDG = trueTrack
    elif wind - trueTrack > 0:
        trueHDG = trueTrack + WCA
    elif wind - trueTrack < 0:
        trueHDG = trueTrack - WCA


    # magnetski heading
    magHDG = trueHDG - 4


    # racunamo ground speed ovisno o tome od gdje nam dolazi vjetar

    if abs(wind-trueTrack) <= 90:
        GS = TAS + HWC
    else:
        GS = TAS - HWC

    # udaljenost u penjanju
    if trueTrack - wind == 0:
        Dv = HWC*tU/60
        #print('Dv je ', Dv)
        D = DU + Dv
    elif trueTrack - wind == 180:
        Dv = HWC*tU/60
        #print('Dv je ', Dv)
```

```
            D = DU + Dv
        elif abs(wind-trueTrack) <= 90:
            Dv = HWC*tU/60
            #print('Dv je ', Dv)
            D = DU + Dv
        elif abs(wind-trueTrack) > 90:
            Dv = HWC*tU/60
            #print('Dv je ', Dv)
            D = DU + Dv

        # potrosnja goriva tj. fuel flow, specificno za penjanje
        FF = fU * 60 / tU

        return [round(fU,2), round(tU,2), round(D,2)]


podatci = []
for visinaPenjanja in [item for item in np.linspace(2000,10000,81)]:
    p = penjanje(325, visinaPenjanja, 192, 10, 5)


    # fuel<
    fU = p[0]
    # ukupno vrijeme
    tt = p[1]
    # udaljenost do TOC
    toc = p[2]

    if(p[2]<15.8):
        print('prije smo dosli')
        trenutanDistance = p[2]

        # wind

        if(visinaPenjanja < 5000):
            # interpoliramo smjer vjetra
```

```
            windispod = flyingConditions['2000'][0][1][0]
            windiznad = flyingConditions['5000'][0][1][0]
            wind = interpoliraj(windispod, 2000, windiznad, 5000, visinaPe
njanja)


            # interpoliramo brzinu vjetra (slučaj između 2 i 5000)
            windispod = flyingConditions['2000'][0][1][1]
            windiznad = flyingConditions['5000'][0][1][1]
            windVelocity = interpoliraj(windispod, 2000, windiznad, 5000,
visinaPenjanja)
        elif(visinaPenjanja == 5000):
            # interpoliramo smjer vjetra (ovo je za 5000)
            wind = flyingConditions['5000'][0][1][0]
            windVelocity = flyingConditions['5000'][0][1][1]
        else:
            # interpoliramo smjer vjetra (ovo je između 5000 i 10000)
            windispod = flyingConditions['5000'][0][1][0]
            windiznad = flyingConditions['10000'][0][1][0]
            wind = interpoliraj(windispod, 5000, windiznad, 10000, visinaP
enjanja)

            # interpoliramo brzinu vjetra
            windispod = flyingConditions['5000'][0][1][1]
            windiznad = flyingConditions['10000'][0][1][1]
            windVelocity = interpoliraj(windispod, 5000, windiznad, 10000,
 visinaPenjanja)

        var = kruzanje(15.8-
trenutanDistance,192,wind,windVelocity, visinaPenjanja)
        fU += var[0]
        tt += var[1]

        # dosli smo do S1, sad letimo kroz sve ostale tocke
        for i in range(1,5):
            if(visinaPenjanja < 5000):
```

```
                # interpoliramo smjer vjetra
                windispod = flyingConditions['2000'][i][1][0]
                windiznad = flyingConditions['5000'][i][1][0]
                wind = interpoliraj(windispod, 2000, windiznad, 5000, visi
naPenjanja)

                # interpoliramo brzinu vjetra
                windispod = flyingConditions['2000'][i][1][1]
                windiznad = flyingConditions['5000'][i][1][1]
                windVelocity = interpoliraj(windispod, 2000, windiznad, 50
00, visinaPenjanja)

            elif(visinaPenjanja == 5000):
                # interpoliramo smjer vjetra
                wind = flyingConditions['5000'][i][1][0]
                windVelocity = flyingConditions['5000'][i][1][1]

            else:
                # interpoliramo smjer vjetra
                windispod = flyingConditions['5000'][i][1][0]
                windiznad = flyingConditions['10000'][i][1][0]
                wind = interpoliraj(windispod, 5000, windiznad, 10000, vis
inaPenjanja)

                # interpoliramo brzinu vjetra
                windispod = flyingConditions['5000'][i][1][1]
                windiznad = flyingConditions['10000'][i][1][1]
                windVelocity = interpoliraj(windispod, 5000, windiznad, 10
000, visinaPenjanja)


            var = kruzanje(distances[i],tablicaSmjerVjetar[i][0],wind,wind
Velocity, visinaPenjanja)
            fU += var[0]
            tt += var[1]
```

```
    else:
        print('poslije smo dosli')

        p1 = penjanje(325, 7800, 192, 10, 5)
        p2 = penjanje(7800, visinaPenjanja, 95, 10, 5)

        fU  = p1[0] + p2[0]
        tt  = p1[1] + p2[1]
        toc = p1[2] + p2[2]


        if(visinaPenjanja < 5000):
            # interpoliramo smjer vjetra
            windispod = flyingConditions['2000'][1][1][0]
            windiznad = flyingConditions['5000'][1][1][0]
            wind = interpoliraj(windispod, 2000, windiznad, 5000, visinaPe
njanja)

            # interpoliramo brzinu vjetra
            windispod = flyingConditions['2000'][1][1][1]
            windiznad = flyingConditions['5000'][1][1][1]
            windVelocity = interpoliraj(windispod, 2000, windiznad, 5000,
visinaPenjanja)
        elif(visinaPenjanja == 5000):
            # interpoliramo smjer vjetra
            wind = flyingConditions['5000'][1][1][0]
            windVelocity = flyingConditions['5000'][1][1][1]
        else:
            # interpoliramo smjer vjetra
            windispod = flyingConditions['5000'][1][1][0]
            windiznad = flyingConditions['10000'][1][1][0]
            wind = interpoliraj(windispod, 5000, windiznad, 10000, visinaP
enjanja)

            # interpoliramo brzinu vjetra
```

```
            windispod = flyingConditions['5000'][1][1][1]
            windiznad = flyingConditions['10000'][1][1][1]
            windVelocity = interpoliraj(windispod, 5000, windiznad, 10000,
 visinaPenjanja)


        var = kruzanje(16.3-p2[2],95,wind,windVelocity, visinaPenjanja)
        fU += var[0]
        tt += var[1]


        for i in range(2,5):
            if(visinaPenjanja < 5000):
                # interpoliramo smjer vjetra
                windispod = flyingConditions['2000'][i][1][0]
                windiznad = flyingConditions['5000'][i][1][0]
                wind = interpoliraj(windispod, 2000, windiznad, 5000, visi
naPenjanja)

                # interpoliramo brzinu vjetra
                windispod = flyingConditions['2000'][i][1][1]
                windiznad = flyingConditions['5000'][i][1][1]
                windVelocity = interpoliraj(windispod, 2000, windiznad, 50
00, visinaPenjanja)
            elif(visinaPenjanja == 5000):
                # interpoliramo smjer vjetra
                wind = flyingConditions['5000'][i][1][0]
                windVelocity = flyingConditions['5000'][i][1][1]
            else:
                # interpoliramo smjer vjetra
                windispod = flyingConditions['5000'][i][1][0]
                windiznad = flyingConditions['10000'][i][1][0]
                wind = interpoliraj(windispod, 5000, windiznad, 10000, vis
inaPenjanja)

                # interpoliramo brzinu vjetra
```

```
                windispod = flyingConditions['5000'][i][1][1]
                windiznad = flyingConditions['10000'][i][1][1]
                windVelocity = interpoliraj(windispod, 5000, windiznad, 10
000, visinaPenjanja)


            var = kruzanje(distances[i],tablicaSmjerVjetar[i][0],wind,wind
Velocity, visinaPenjanja)
            fU += var[0]
            tt += var[1]



    print('Za visinu',visinaPenjanja,'put je trajao', round(tt,2), ', potr
osili smo', round(fU,2), 'goriva i TOC je na', round(toc,2),'.')
    podatci += [[round(visinaPenjanja,2), round(tt,2), round(fU,2), round(
toc,2)]]


SMALL_SIZE = 12
MEDIUM_SIZE = 14
BIGGER_SIZE = 16

plt.rc('font', size=SMALL_SIZE)          # controls default text sizes
plt.rc('axes', titlesize=SMALL_SIZE)     # fontsize of the axes title
plt.rc('axes', labelsize=MEDIUM_SIZE)    # fontsize of the x and y labels
plt.rc('xtick', labelsize=SMALL_SIZE)    # fontsize of the tick labels
plt.rc('ytick', labelsize=SMALL_SIZE)    # fontsize of the tick labels
plt.rc('legend', fontsize=SMALL_SIZE)    # legend fontsize
plt.rc('figure', titlesize=BIGGER_SIZE)  # fontsize of the figure title

fig = plt.figure(figsize=(12,8))
legenda = []
for item in podatci:
    x = [0,item[3],141]
    y = [0,item[0],item[0]]
    plt.plot(x,y)
```

```
    plt.xticks([0,15.8,32.1,68.8,118.3,141],['LDZA','S1','SISAK','VBA','A1
','LDOS'])
    legenda += ['t_u = '+str(item[1])+', f_u= '+str(item[2])]

plt.legend(legenda,loc='lower right')
fig.savefig('basic.png',dpi=300)
```

Sveučilište u Zagrebu
Fakultet prometnih znanosti
10000 Zagreb
Vukelićeva 4

IZJAVA O AKADEMSKOJ ČESTITOSTI I SUGLASNOST

Izjavljujem i svojim potpisom potvrđujem kako je ovaj završni rad isključivo rezultat mog vlastitog rada koji se temelji na mojim istraživanjima i oslanja se na objavljenu literaturu što pokazuju korištene bilješke i bibliografija.

Izjavljujem kako nijedan dio rada nije napisan na nedozvoljen način, niti je prepisan iz necitiranog rada, te nijedan dio rada ne krši bilo čija autorska prava.

Izjavljujem također, kako nijedan dio rada nije iskorišten za bilo koji drugi rad u bilo kojoj drugoj visokoškolskoj, znanstvenoj ili obrazovnoj ustanovi.

Svojim potpisom potvrđujem i dajem suglasnost za javnu objavu završnog rada pod naslovom **Određivanje optimalne visine leta jednomotornog klipnog zrakoplova na odabranoj ruti.** na internetskim stranicama i repozitoriju Fakulteta prometnih znanosti, Digitalnom akademskom repozitoriju (DAR) pri Nacionalnoj i sveučilišnoj knjižnici u Zagrebu.

U Zagrebu, 5.9.2021

Student/ica:

(potpis)

Scanned with MOBILE SCANNER