



UNIVERSITATEA „POLITEHNICA” din BUCUREȘTI
ȘCOALA DOCTORALĂ INGINERIA SISTEMELOR BIOTEHNICE

REZUMATUL TEZEI DE DOCTORAT

Optimizare convexă pentru demonstratoare reutilizabile cu aplicații spațiale

Autoare: Ing. Ana-Maria NECULĂESCU

Conducător de doctorat: Prof. dr. ing. Ion STROE

COMISIA DE DOCTORAT

Președinte	Prof dr. ing. Andrei CRAIFALEANU	de la	Universitatea Politehnica din București
Conducător de doctorat	Prof. dr. ing. CS I Ion STROE	de la	Universitatea Politehnica din București
Referent	CS I dr. fiz. Marius Ioan PISO	de la	Agencia Spațială Română
Referent	CS I dr. ing. Cătălin NAE	de la	Institutul Național de Cercetare Dezvoltare Aerospațială „Elie Carafoli” - INCAS
Referent	Prof. dr. ing. Adrian-Mihail STOICA	de la	Universitatea Politehnica din București

București

2023

Cuprins

1	Introducere	4
1.1	Conținutul tezei.....	4
1.2	Context.....	5
1.3	Obiectivele tezei.....	6
1.4	Stadiul actual al cercetărilor în domeniu.....	6
1.4.1	Modelarea matematică a problemei de optimizare convexă a traiectoriei	6
1.4.2	Algoritmi de calcul în timp real pentru probleme de optimizare convexă.....	7
2	Optimizare convexă pentru vehicule cu decolare și aterizare verticală cu aplicații spațiale.....	7
3	Minimizarea cantității de combustibil necesară manevrei de aterizare	8
3.1	Formularea matematică a problemei.....	8
3.2	Discretizarea problemei	11
4	Arhitectura de testare a aplicației în timp real (SEP) pentru optimizarea traiectoriei de aterizare	13
5	Prezentarea infrastructurii de cercetare, testare și validare a aplicației SEP	15
6	Implementarea problemei matematice în Matlab/CVX	16
7	Implementarea problemei matematice în generatorul de cod CVXGEN.....	17
7.1	Descrierea și utilizarea pachetului CVXGEN.....	17
7.2	Descrierea codului generat.....	17
7.3	Utilizarea codului generat de CVXGEN.....	17
7.4	Funcții auxiliare necesare.....	18
7.5	Generarea S-funcției ce conține solver-ul generat și testarea offline a acesteia	18
7.6	Misiune nominală.....	19
7.7	Analiză Monte Carlo	22
8	Simulări utilizând modelul dinamic cu șase grade de libertate (6DOF)	23
8.1	Modelare matematică a demonstratorului DTV	23
8.2	Integrarea S-funcției în simulatorul de zbor al vehiculului DTV.....	26
8.3	Rezultate numerice offline	27
9	Testarea aplicației SEP utilizând Real Time Target Machine.....	29
9.1	Arhitectura de testare a aplicației în timp real dezvoltate	29
9.2	Generarea S-funcțiilor necesare pentru integrarea aplicației în timp real în software-ul de zbor.....	29
9.3	Rezultatele simulării misiunii nominale în timp real (HIL).....	30
10	Testarea și validarea aplicației în condiții de zbor captiv	32

Optimizare convexă pentru demonstratoare reutilizabile cu aplicații spațiale

11	Concluzii	36
11.1	Contribuțiile tezei.....	36
11.2	Rezultate obținute	37
11.3	Perspectivă de continuare și dezvoltare ulterioară	38
11.4	Lista publicațiilor	39
	Bibliografie (selectivă).....	41

1 Introducere

Lucrarea de doctorat propune o soluție pentru rezolvarea problemei matematice privind optimizarea în timp real a traiectoriei de aterizare utilizând tehnici convexe pentru un demonstrator reutilizabil în domeniul spațial. Subiectul a vizat dezvoltarea și testarea unei aplicații în timp real (Software Experimental Payload – SEP) iar testarea și validarea pentru această aplicație se face utilizând demonstratorul cu decolare și aterizare verticală, DTV, dezvoltat la Institutul Național de Cercetare-Dezvoltare Aerospațială „Elie Carafoli - INCAS.

Nevoia temei de cercetare vine în contextul interesului crescut la nivel mondial pentru aterizarea, recuperarea și reutilizarea primelor trepte ale lansatoarelor spațiale în vederea diminuării costurilor misiunilor spațiale și accesarea facilă a spațiului cosmic. Această temă de cercetare este de interes și în contextul eforturilor depuse în continuare în vederea colonizării Lunii și a planetei Marte pentru care această tehnologie are aplicabilitate ridicată.

Rezultatele obținute în cadrul acestei teze de doctorat demonstrează fezabilitatea utilizării aplicației dezvoltate ce ar rula în timp real la bordul vehiculelor spațiale pentru procesul de optimizare a traiectoriei.

Cuvinte cheie: *optimizarea traiectoriei la bord, demonstrator cu aplicații spațiale, reutilizare, tehnici convexe, generare automată cod C, testare în timp real, validare în zbor.*

1.1 Conținutul tezei

Primul capitol cuprinde o introducere în domeniul optimizării online a traiectoriei de aterizare a demonstratoarelor reutilizabile spațiale, prezentând contextul actual pe plan național, european și internațional, obiectivul principal al tezei de doctorat, dar și stadiul actual al cercetărilor în domeniul ales. Capitolul al doilea prezintă câteva noțiuni introductive în domeniul optimizării convexe pentru vehicule cu decolare și aterizare verticală cu aplicații spațiale, iar cel de-al treilea capitol descrie modelul matematic pentru minimizarea cantității de combustibil necesare aterizării pentru demonstratoare spațiale reutilizabile. În capitolul trei este totodată prezentată formularea matematică a problemei, dar și modalitatea de discretizare a acesteia pentru rezolvarea ei numerică.

Capitolul al patrulea prezintă arhitectura și metodologia de testare a aplicației pentru optimizarea online a traiectoriei vehiculelor spațiale, aplicație ce a fost dezvoltată și prezentată în cadrul acestei teze. Capitolul cinci cuprinde detalii tehnice despre demonstratorul ce va fi utilizat ca infrastructură de cercetare, testare și validare în zbor a metodologiei dezvoltate în cadrul acestei lucrări și a aplicației SEP.

Capitolul șase prezintă detalii despre implementarea modelului matematic detaliat în capitolul al treilea și utilizarea modului integrat CVX în mediul de simulare Matlab. Pentru testarea aplicației SEP dezvoltate a fost selectat un caz nominal de studiu. Cel de-al șaptelea capitol cuprinde amănunte despre modalitatea de implementare a modelului matematic în generatorul de solver convex, CVXGEN. Sunt oferite detalii despre CVXGEN, despre modul de utilizare a codului C generat, precum și funcțiile adiționale dezvoltate pentru utilizarea acestuia. Rezultate numerice pentru același caz nominal sunt prezentate, împreună cu o analiză de tip Monte Carlo, unde numeroase simulări sunt efectuate în prezența incertitudinilor.

În cel de-al optulea capitol sunt prezentate detalii legate de modelul dinamic 6DOF pentru DTV, modalitatea de integrare a S-funcției în simulatorul cu șase grade de libertate al demonstratorului DTV, împreună cu rezultate numerice pentru misiunea nominală selectată. Capitolul nouă cuprinde arhitectura de testare în timp real a aplicației SEP dezvoltate, rezultatele numerice pentru misiunea nominală analizată fiind prezentate în detaliu.

Capitolul al zecelea cuprinde rezultatele simulărilor numerice necesare pentru pregătirea unei campanii experimentale de zbor, dar și rezultatele înregistrate pentru misiunile experimentale în zbor ancorat, ME#1 și ME#2. Prima misiune experimentală a avut scopul de a valida întregul proces dezvoltat, iar ME#2 reprezintă o scurtă manevră laterală, pentru care au fost efectuate patru zboruri experimentale.

Concluziile tezei de doctorat sunt prezentate în capitolul unsprezece, fiind subliniate contribuțiile tezei, rezultatele obținute, dar și perspectivele de continuare și dezvoltare ulterioară. Totodată este prezentată lista publicațiilor realizate pe parcursul anilor studiilor doctorale, dar și o notișe bibliografice utilizate în realizarea tezei de doctorat.

1.2 Context

În cadrul acestui subcapitol sunt prezentate elementele de noutate în domeniul vehiculelor spațiale reutilizabile la nivel mondial, schițându-se următoarele aspecte:

- Până în decembrie 2015, un vehicul lansator era utilizat pentru o singură lansare. Treapta principală și boosterele care formau lansatorul spațial erau aruncate în ocean, iar treapta de inserție orbitală rămânea pe o orbită în jurul Pământului sau era direcționată pe o traiectorie de reintrare, unde ardea în atmosferă aproape în totalitate;
- Acest scenariu a fost schimbat în totalitate în decembrie 2015 prin succesul companiei private, Space Exploration Technologies Corporation (SpaceX), reprezentat de aterizarea verticală a primei trepte a lansatorului spațial *Falcon 9* [1]. Conceptul de reutilizare pentru sistemele spațiale de lansare este demonstrat și de o altă companie privată din Statele Unite ale Americii, Blue Origin, prin realizarea cu succes a unui zbor suborbital cu lansatorul *New Shepard* [2], dar și de compania privată, Deep Blue Aerospace, cu sediul în China [3];
- Europa a făcut și ea primii pași în dezvoltarea sistemelor spațiale de lansare reutilizabile, prin demonstratoarele *Callisto* și *Themis* [4];
- La nivel regional, în România se află în faza de dezvoltare un demonstrator reutilizabil cu decolare și aterizare verticală (VTVL) propulsat de un motor rachetă denumit *Ascent Descent Autonomous Maneuverable Platform* (ADAMP), proiect european condus de Institutul Național de Cercetare-Dezvoltare Aerospațială „Elie Carafoli” – INCAS [5];
- Reutilizarea sistemelor de lansare reduce masiv costurile unei lansări, ceea ce conduce în final la accesarea mai facilă a spațiului cosmic. NASA estimează că o lansare comercială către Stația Spațială Internațională (ISS) costă în prezent de 4 ori mai puțin decât în urmă cu 20 de ani [6] prin prisma reușitelor tehnologice.

Succesul recuperării și reutilizării primei trepte a lansatorului spațial Falcon 9 se datorează mai multor tehnologii, dar cea mai relevantă în contextul acestei teze de doctorat este optimizarea traiectoriei online la bordul lansatorului și controlul autonom în realizarea aterizării verticale.

1.3 Obiectivele tezei

În această teză de doctorat este expus modelul matematic pentru rezolvarea problemei de aterizare verticală privind minimizarea consumului de combustibil pentru un demonstrator reutilizabil cu aplicații spațiale și detalierea implementării unui algoritm, ce rulează în timp real pe calculatorul de bord al acestuia. Obiectivul principal al tezei este dezvoltarea unei aplicații și propunerea unei metodologii care optimizează traiectoria de aterizare în timp real pentru un demonstrator reutilizabil cu aplicații spațiale, utilizând un software cu licență comercială, CVXGEN, care generează cod C pentru soluționarea problemelor convexe. Aplicația în timp real, denumită SEP, este implementată, testată și validată în zbor utilizând ca platformă de testare un demonstrator cu decolare și aterizare verticală bazat pe un motor turbo-jet, de clasă 1kN.

1.4 Stadiul actual al cercetărilor în domeniu

Lucrarea de doctorat abordează într-o manieră inovativă soluționarea problemei de aterizare pentru vehicule cu decolare și aterizare verticală în retropropulsie utilizând tehnici convexe de control optimal în timp real, pe calculatorul de bord al unui demonstrator reutilizabil. Problematika abordată se poate descompune în două direcții principale de cercetare:

- Problema de control optimal a traiectoriei unui vehicul reutilizabil cu aplicații spațiale, utilizând tehnici convexe;
- Programe de calcul existente pentru soluționarea acestei probleme în timp real.

1.4.1 Modelarea matematică a problemei de optimizare convexă a traiectoriei

Aterizarea unui lansator spațial reprezintă una din cele mai importante provocări tehnologice pentru dezvoltarea algoritmilor de traiectorie și control. Aceste provocări se datorează condițiilor imprevizibile de zbor, dispersiilor semnificative și incertitudinile asociate altor parametri, la care se adaugă constrângerile legate de controlul vehiculului și dimensionării structurale ale acestuia. Considerând aceste constrângeri și adăugând cerințe adiționale riguroase necesare obținerii unei soluții cu acuratețe ridicată pentru manevra de aterizare, problema de aterizare la punct fix pentru vehicule spațiale devine extrem de dificil de rezolvat.

Generarea unei traiectorii pentru vehicule spațiale poate fi exprimată ca o problemă de control optimal, ceea ce conduce la nevoia soluționării unei probleme care are următoarea formă, [7]:

$$\min_{u, p, t_f} J(x, u, p, t_f) \quad (1.1)$$

având următoarele constrângeri:

$$\dot{x}(t) = f(x, u, p, t) \quad (1.2)$$

$$(x(t), u(t), p, t_f) \in C(t), \forall t \in [0, t_f] \quad (1.3)$$

$$(x(t), p) \in X_0, (x(t_f), p) \in X_f \quad (1.4)$$

Funcția obiectiv (1.1) cuprinde ținta misiunii, dinamica sistemului este modelată prin ecuații diferențiale care sunt impuse drept constrângeri (1.2), stările sistemului și controlul sunt impuse

prin intermediul (1.3), iar condițiile la limita sunt impuse prin (1.4).

În acest moment, se cunosc diferite metode de discretizare pentru o problemă de control optimal. O dată ce problema matematică este discretizată, se pot utiliza metode numerice de optimizare pentru obținerea unei soluții, iar aici intervine dificultatea în rezolvarea problemelor reale, cu aplicații în domeniul spațial și nu numai. În funcție de formularea matematică a problemei și a tehnicilor de discretizare selectate, problema de optimizare poate deveni o problemă neliniară (și non-convexă) de control optimal, ceea ce poartă denumirea în literatura de specialitate de *nonlinear optimization problem (NLP)* [8], [9], [10]. Marele dezavantaj al problemei NLP este faptul că necesită resurse foarte ridicate de calcul și că nu există o garanție a obținerii soluției optimale. Din aceste motive, metodele NLP nu sunt potrivite pentru asigurarea deterministă a existenței unei traiectorii optimale pentru vehicule spațiale autonome.

Pentru a atinge acest scop, se disting două metode principale:

- Convexificare fără pierderi (*Lossless Convexification - LCvx*) [11], [12], [13];
- Programare convexă secvențială (*Sequential Convex Programming - SCP*) [14], [15], [16], [17].

1.4.2 Algoritmi de calcul în timp real pentru probleme de optimizare convexă

Cerințele legate de puterea de calcul reprezintă cea de-a doua considerație pentru generarea traiectoriei la bordul vehiculelor. Din punct de vedere istoric, acesta a fost cel mai important factor în dezvoltarea algoritmilor cu aplicații practice. În domeniul optimizării în timp real pe un calculator aflat la bordul unui vehicul (*real-time embedded optimization*) [18], [19], [20], [21], trebuie calculate extrem de rapid (la nivel de milisecundă sau microsecundă) diferite instanțe ale unei probleme, iar în cele mai multe cazuri rezultatul trebuie obținut înainte de un termen strict de execuție. Această cerință tehnică este în contradicție cu algoritmi generici, care au nevoie de un timp mai mare de execuție iar soluția problemei este disponibilă atunci când o anumită precizie a fost obținută [22].

CVXGEN este un generator de cod pentru familii de probleme de optimizare convexă care pot fi reduse la programe pătratice (QP) și programarea conică de ordin doi (*Second Order Cone Programming – SOCP*). CVXGEN pornește de la o descriere a familiei de probleme de optimizare convexă și generează automat cod în limbajul de programare C. Pentru probleme de dimensiuni mici și medii (cu până la câteva sute de variabile de optimizare), CVXGEN generează programe de calcul care rezolvă aceste probleme în câteva microsecunde sau milisecunde, [23].

În cadrul acestei lucrări de doctorat, a fost utilizat generatorul de cod C, CVXGEN, pentru care autoarea a primit o licență academică, mulțumită domnului Jacob Mattingley.

2 Optimizare convexă pentru vehicule cu decolare și aterizare verticală cu aplicații spațiale

În acest capitol este realizată o prezentare comparativă a celor mai cunoscute metode pentru generarea traiectoriei online la bordul vehiculelor reutilizabile, prin utilizarea metodelor de optimizare convexă. Tehnicile de optimizare convexă urmăresc minimizarea unei *funcții obiectiv convexe* în timp ce se respectă un *set de constrângeri convexe*. Sunt prezentate câteva elemente teoretice pentru seturi convexe și funcții convexe utilizate în cadrul acestei teze de doctorat.

3 Minimizarea cantității de combustibil necesară manevrei de aterizare

3.1 Formularea matematică a problemei

Sistemul de referință ales este unul inertial și legat de suprafața Pământului. Originea sistemului este în centrul locului de lansare, direcția pozitivă a axei x este în sus, direcția pozitivă a axei y este către Est, iar axa z are sensul pozitiv către Nord, așa acum este ilustrat în Figura 3.1. Triedrul este denumit în continuare drept *triedrul local de start inertial (LP- "launch pad frame")*.

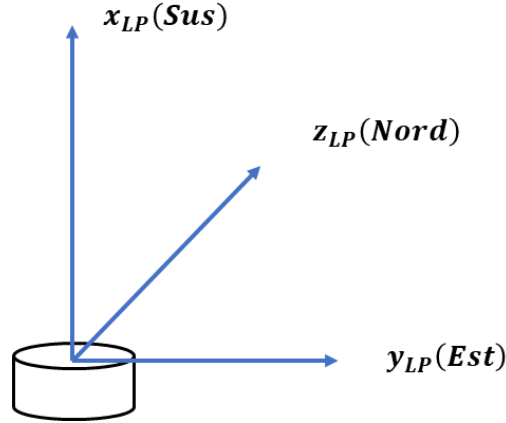


Figura 3.1 Sistem de referință utilizat

Pentru generarea traiectoriei online la bordul unui vehicul reutilizabil astfel încât să fie minimizată cantitatea de combustibil necesară în cazul unei aterizări precise la punct fix, problema se poate formula, conform [24] și [25] în felul următor:

$$\max_{t_f, T_c(\cdot)} m(t_f) = \min_{t_f, T_c(\cdot)} \int_0^{t_f} \|T_c(t)\| dt \quad (3.1)$$

supusă la:

$$\ddot{r}(t) = g + \frac{T_c(t)}{m(t)} \quad (3.2)$$

$$\dot{m}(t) = -\alpha \|T_c(t)\| \quad (3.3)$$

$$0 < \rho_1 \leq \|T_c(t)\| \leq \rho_2 \quad (3.4)$$

$$\|r_d(t)\| \leq \beta r_v(t) \quad (3.5)$$

Condițiile inițiale ale problemei (3.1) au forma:

$$m(0) = m_{\text{initială}} \quad (3.6)$$

$$m(t_f) \geq 0 \quad (3.7)$$

$$r(0) = r_0 \quad (3.8)$$

$$\dot{r}(0) = \dot{r}_0 \quad (3.9)$$

iar condițiile finale au forma:

$$r(t_f) = 0 \quad (3.10)$$

$$\dot{r}(t_f) = 0 \quad (3.11)$$

Pentru simplificarea notațiilor au fost introduși parametrii α, ρ_1, ρ_2 , aceștia fiind definiți în continuare:

$$\alpha = \frac{1}{I_{sp} g_e \cos \Phi} \quad (3.12)$$

$$\rho_1 = n T_1 \cos \Phi \quad (3.13)$$

$$\rho_2 = n T_2 \cos \Phi \quad (3.14)$$

În cadrul formulării problemei au fost folosite următoarele notații:

- Φ reprezintă unghiul de deflexie al propulsoarelor de tip „thrusters”;
- n reprezintă numărul de propulsoare de tip „thrusters”;
- I_{sp} reprezintă impulsul specific;
- $m_{initială}$ reprezintă masa inițială a vehiculului (împreună cu cantitatea inițială de combustibil);
- g reprezintă constanta gravitațională a Pământului;
- T_1 reprezintă limita inferioară a tracțiunii oferite de fiecare propulsor;
- T_2 reprezintă limita superioară a tracțiunii oferite de fiecare propulsor;
- $r_0 \in \mathbb{R}^3$ reprezintă poziția inițială a vehiculului (relativă la punctul de aterizare);
- $\dot{r}_0 \in \mathbb{R}^3$ reprezintă viteza inițială a vehiculului (relativă la punctul de aterizare);
- $\beta = \tan \gamma$, cu $\gamma \in \left(0, \frac{\pi}{2}\right)$ reprezintă unghiul pantei traiectoriei;
- $r_v \in \mathbb{R}$ reprezintă altitudinea vehiculului;
- $r_d \in \mathbb{R}^2$ reprezintă vectorul de poziție laterală a vehiculului („downrange position vector”) astfel încât $r = [r_v, r_d^T]^T$.

De remarcat faptul că în formularea (3.1), inegalitatea (3.4) definește o constrângere de tip non-convexă pe canalul de comandă. Din acest motiv este necesară o relaxare a formulării (3.1), transformând problema acum într-una de optimizare convexă [25]:

$$\min_{t_f, T_c(\cdot), \Gamma(\cdot)} \int_0^{t_f} \Gamma(t) dt \quad (3.15)$$

supusă la:

$$\ddot{r}(t) = g + \frac{T_c(t)}{m(t)} \quad (3.16)$$

$$\dot{m}(t) = -\alpha\Gamma(t) \quad (3.17)$$

$$\|T_c(t)\| \leq \Gamma(t) \quad (3.18)$$

$$0 < \rho_1 \leq \Gamma(t) \leq \rho_2 \quad (3.19)$$

$$\|r_d(t)\| \leq \beta r_v(t) \quad (3.20)$$

Condițiile inițiale și finale ale problemei (3.15) rămân neschimbate, aceste fiind prezentate în relațiile (3.6) - (3.9), respectiv în relațiile (3.10) - (3.11).

Dacă o soluție a problemei (3.15) este de forma $\langle t_f^*, T_c^*(\cdot), \Gamma^*(\cdot) \rangle$, atunci se poate considera că $\langle t_f^*, T_c^*(\cdot) \rangle$ este o soluție și a problemei (3.1), iar $\|T_c^*(t)\| = \rho_1$ sau $\|T_c^*(t)\| = \rho_2$ pentru $t \in [0, t_f^*]$. Astfel, formularea (3.15) convexifică constrângerea problematică (3.4) a formulării (3.1) prin utilizarea a două constrângeri echivalente (3.18) și (3.19).

Folosind cu succes schimbări de variabile într-un mod avantajos ($u = \frac{T_c}{m}$, $\sigma = \frac{\Gamma}{m}$ și $z = \ln m$) și utilizând dezvoltări în serii Taylor pentru mărginirea termenului σ (se păstrează primii 3 termeni pentru limita inferioară și primii 2 termeni pentru limita superioară) se poate rescrie din nou problema minimizării cantității de combustibil necesară în cazul unei aterizări precise, fiind obținută următoarea formulare [25]:

$$\min_{t_f, u(\cdot), \sigma(\cdot)} \int_0^{t_f} \sigma(t) dt \quad (3.21)$$

supusă la:

$$\ddot{r}(t) = g + u(t) \quad (3.22)$$

$$\dot{z}(t) = -\alpha\sigma(t) \quad (3.23)$$

$$\|u(t)\| \leq \sigma(t) \quad (3.24)$$

$$\mu_1(t) \left[1 - (z(t) - z_0(t)) + \frac{(z(t) - z_0(t))^2}{2} \right] \leq \sigma(t) \leq \mu_2(t) [1 - (z(t) - z_0(t))] \quad (3.25)$$

$$z_0(t) \leq z(t) \leq \ln(m_{\text{initială}} - \alpha\rho_1 t) \quad (3.26)$$

$$\|r_d(t)\| \leq \beta r_v(t) \quad (3.27)$$

unde:

$$\mu_1(t) = \rho_1 e^{-z_0(t)} \quad (3.28)$$

$$\mu_2(t) = \rho_2 e^{-z_0(t)} \quad (3.29)$$

$$z_0(t) = \ln(m_{\text{initială}} - \alpha\rho_2 t) \quad (3.30)$$

Condițiile inițiale ale problemei (3.21) au forma:

$$z(0) = \ln m_{\text{initială}} \quad (3.31)$$

$$r(0) = r_0 \quad (3.32)$$

$$\dot{r}(0) = \dot{r}_0 \quad (3.33)$$

iar condițiile finale au forma:

$$r(t_f) = 0 \quad (3.34)$$

$$\dot{r}(t_f) = 0 \quad (3.35)$$

Pentru un timp dat de zbor t_f , folosind formularea (3.21), toate constrângerile de egalitate și inegalitate definesc regiuni fezabile convexe în spațiul de stare și control, având totodată și o funcție cost (obiectiv) de tip convexă. Prin urmare, problema (3.21) poate fi convertită într-o problemă de optimizare convexă dimensional-finită, cea mai eficientă metodă de rezolvare fiind aceea de tip SOCP (*Second Order Cone Programming*), după o discretizare corespunzătoare a acesteia.

3.2 Discretizarea problemei

În acest capitol este prezentat modul de discretizare a problemei curente de rezolvat (3.21):

$$\min_{u_0, \dots, u_N, \sigma_0, \dots, \sigma_N} -z_n \quad (3.36)$$

supusă la:

$$r_{k+1} = r_k + \frac{\Delta t}{2}(\dot{r}_k + \dot{r}_{k+1}) + \frac{\Delta t}{12}(u_k - u_{k+1}) \quad (3.37)$$

$$\dot{r}_{k+1} = \dot{r}_k + \frac{\Delta t}{2}(u_k + u_{k+1}) + g\Delta t \quad (3.38)$$

$$z_{k+1} = z_k - \frac{\alpha\Delta t}{2}(\sigma_k + \sigma_{k+1}) \quad (3.39)$$

$$\|u_k\| \leq \sigma_k \quad (3.40)$$

$$\mu_{1,k} \left[1 - (z_k - z_{0,k}) + \frac{(z_k - z_{0,k})^2}{2} \right] \leq \sigma_k \leq \mu_{2,k} [1 - (z_k - z_{0,k})] \quad (3.41)$$

$$z_{0,k} \leq z_k \leq \ln(m_{\text{initială}} - \alpha\rho_1 t) \quad (3.42)$$

$$\|r_{d_k}\| \leq \beta r_{v_k} \quad (3.43)$$

cu $k = 0, 1 \dots N$, unde:

$$\mu_{1,k} = \rho_1 e^{-z_{0,k}} \quad (3.44)$$

$$\mu_{2,k} = \rho_2 e^{-z_{0,k}} \quad (3.45)$$

$$z_{0,k} = \ln(m_{\text{initială}} - \alpha \rho_2 k \Delta t) \quad (3.46)$$

Condițiile inițiale ale problemei (3.36) au forma:

$$z_0 = \ln m_{\text{initială}} \quad (3.47)$$

$$r(0) = r_0 \quad (3.48)$$

iar condițiile finale au forma:

$$t_f = N \Delta t \quad (3.49)$$

$$r_N = 0 \quad (3.50)$$

$$\dot{r}_N = 0 \quad (3.51)$$

4 Arhitectura de testare a aplicației în timp real (SEP) pentru optimizarea traiectoriei de aterizare

În cadrul lucrării de doctorat a fost dezvoltată o aplicație în timp real pentru generarea la bord a unei traiectorii optimale, împreună cu o modalitate integrată de testare a acesteia (Figura 4.1) utilizând un demonstrator denumit DTV [26], care aparține Institutului Național de Cercetare-Dezvoltare Aerospațială „Elie Carafoli” - INCAS București.

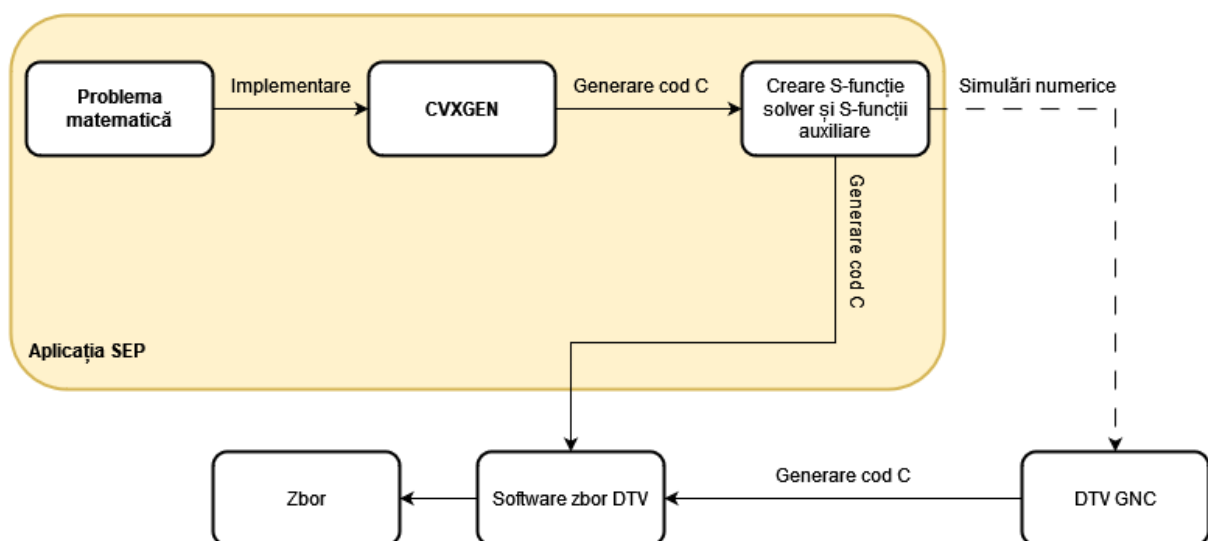


Figura 4.1 Aplicația SEP și integrarea acesteia

Punctul de pornire este modelul matematic al problemei de optimizare convexă pentru minimizarea consumului de combustibil utilizat pentru faza de aterizare. O dată scrisă problema matematică și discretizată, urmează faza de implementare în generatorul de solver convex pentru această familie de probleme matematice. După ce problema a fost implementată în CVXGEN, următorul pas este generarea de cod în limbajul de programare C, rezultând astfel un număr de fișiere ce au fost utilizate mai departe pentru generarea S-funcției, împreună cu funcțiile adiționale dezvoltate. Aceasta S-funcție, ce conține solverul pentru generarea traiectoriei online (la bord în timpul zborului), este mai departe integrată în Simulink, în sub-sistemul de Traietorie, Navigație și Control (Guidance, Navigation and Control - GNC) al demonstratorului ce permite testarea într-un mediu de simulare a aplicației SEP utilizând un model dinamic cu șase grade de libertate (6DOF) pentru DTV. DTV este o platformă de testare și validare pentru produse software și echipamente hardware, iar INCAS a dezvoltat această funcționalitate, de a putea integra un software experimental prin intermediul unui bloc Simulink, punând la dispoziția partenerilor interfețele necesare de care trebuie să țină cont pentru ca integrarea să se facă în mod corect.

După integrarea S-funcției în blocul SEP din GNC-ul demonstratorului și testarea utilizând simulatorul cu șase grade de libertate, se trece la următoarea etapă în procesul de testare a aplicației SEP prin generarea automată de cod C, utilizând funcționalitatea Matlab/Simulink. Codul C se generează pentru un calculator generic pentru testare în timp real (*“Generic Real-Time Target”*), care este integrat mai departe în software-ul de zbor al DTV, aceasta fiind apelată la fiecare 100 milisecunde și rulând ca o aplicație de sine-stătătoare pe calculatorul de bord, spre deosebire de codul generat din sistemul GNC, care este apelat la fiecare 40 milisecunde.



Figura 4.2 Arhitectura de testare a optimizatorului de traiectorie online

Pentru ca aplicația de optimizare online a traiectoriei să poată fi testată în zbor, iar DTV să rămână în siguranță pe perioada desfășurării testelor în zbor, următoarea procedură de testare (detalii în Figura 4.2) este adoptată:

- DTV decolează urmărind o traiectorie de referință, predefinită;
- După ce faza de urcare se încheie, în urma unor condiții predefinite, demonstratorul trece în faza de zbor la punct fix deasupra locului de decolare;
- După 15 secunde de la începutul fazei de zbor la punct fix, programul de optimizare online de traiectorie începe să ruleze;
- În momentul în care este comandată trecerea din faza de zbor pre-programată la traiectorie experimentală generată online, începe execuția programului de optimizare pe baza informațiilor de la sistemul de navigație prin achiziția de date de la sistemul GNC (poziție și viteză) și generează o traiectorie optimală la bord în fereastra de timp alocată, 100 milisecunde (un tabel ce conține trei poziții și trei viteze de referință pentru fiecare punct de discretizare);
- Odată cu generarea online a acestei traiectorii optimale, programul de calcul trimite informații legate de statusul procesului de optimizare, informație determinată prin intermediul S-funcției;
- Dacă procesul de optimizare online se termină cu succes, traiectoria generată la bord se trimite vehicului ca noua referință prin intermediul S-funcțiilor către aplicația GNC care rulează pe calculatorul de bord;
- După ce demonstratorul primește noua referință, algoritmul de control calculează comenzile necesare pentru ca acesta să urmărească traiectoria generată online, astfel fiind asigurată în permanență integritatea vehicului.

5 Prezentarea infrastructurii de cercetare, testare și validare a aplicației SEP

Pentru creșterea nivelului de maturitate a tehnologiei propuse în cadrul acestei teze de doctorat, mai exact a aplicației SEP dezvoltată, este necesară testarea și validarea acesteia în timp real, utilizând o infrastructură de testare în zbor existentă. Institutul Național de Cercetare-Dezvoltare Aerospațială “Elie Carafoli” – INCAS București a dezvoltat această infrastructură unică într-un proiect european finanțat de către Agenția Spațială Europeană (ESA) și cu susținerea Agenției Spațiale Române (ROSA), în cadrul proiectului Demonstrator for Technologies Validation (DTV).

Așa cum este evident încă din titlul proiectului, demonstratorul DTV (Figura 5.1) oferă oportunitatea validării în zbor a tehnologiilor aflate în faza experimentală, atât software, cât și hardware. DTV este un demonstrator reutilizabil propulsat de un motor turbo-jet de clasă 1kN cu decolare și aterizare verticală, complet autonom.



Figura 5.1 Vehicul DTV – configurație. Credit: INCAS/ESA

Pentru realizarea campaniilor experimentale, conducerea INCAS și-a dat acordul în vederea utilizării infrastructurii de cercetare (atât infrastructură la sol – de la sediul INCAS din Măneciu, cât și infrastructură de cercetare în zbor – demonstratorul DTV). De asemenea, conducerea INCAS a fost de acord cu utilizarea în această teză a Simulatorului de zbor (model protejat), ce permite folosirea acestei unelte fără a avea acces la modelele matematice implementate (model dinamic în șase grade de libertate, arhitectură de control și modele de senzori). INCAS a pregătit un model protejat al Simulatorului de zbor, model ce este distribuit partenerilor INCAS pentru interfațarea programelor software dezvoltate de aceștia în proiecte de cercetare.

6 Implementarea problemei matematice în Matlab/CVX

Problema matematică discretizată, prezentată pe larg în capitolul 3, a fost implementată în Matlab, modulul de optimizare convexă CVX. CVX transformă Matlab într-un limbaj de modelare, permițând funcției obiectiv și constrângerilor să fie specificate utilizând sintaxe standard disponibile în Matlab, iar implementarea a fost studiată utilizând un caz nominal de studiu.

Implementarea modelului matematic pentru minimizarea cantității de combustibil în modulul CVX din Matlab și soluționarea acestuia utilizând solverul SeDumi a fost necesară în special pentru verificarea existenței unei soluții numerice, iar în urma realizării simulărilor numerice, timpul de calcul necesar este 2,16 secunde, ceea ce înseamnă implementarea **nu este potrivită** pentru aplicații în timp real. Astfel a apărut necesitatea implementării modelului matematic într-un generator de cod C, CVXGEN, pentru transformarea problemei matematice într-un solver care să soluționeze clasa de probleme studiată într-un timp de calcul de ordinul milisecundelor.

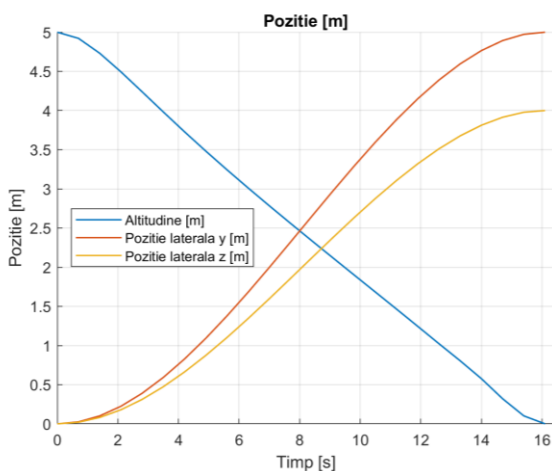


Figura 6.1 Pozitie DTV vs. timp, caz nominal de studiu

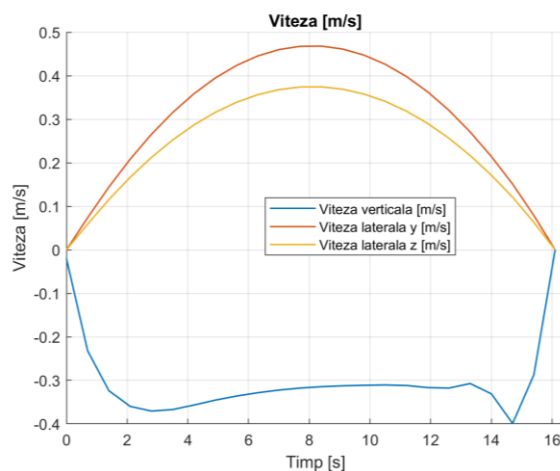


Figura 6.2 Viteză DTV vs. timp, caz nominal de studiu

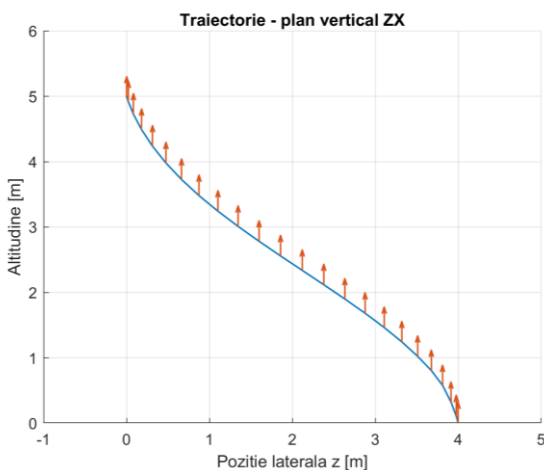


Figura 6.3 Traiectorie ZX vs. timp, caz nominal de studiu

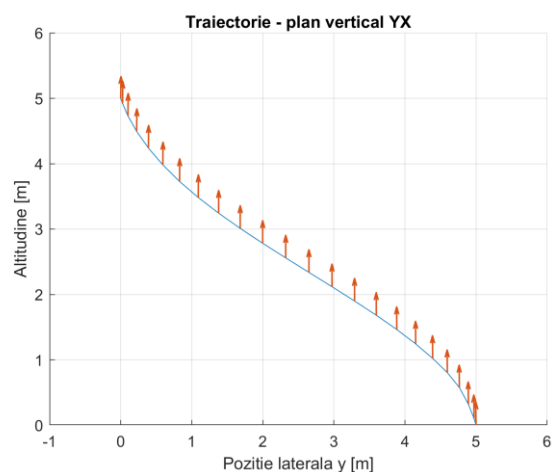


Figura 6.4 Traiectorie YX vs. timp, caz nominal de studiu

7 Implementarea problemei matematice în generatorul de cod CVXGEN

7.1 Descrierea și utilizarea pachetului CVXGEN

Problema matematică descrisă în capitolul 3, aceea a minimizării cantității consumului de combustibil necesară aterizării, a fost implementată în CVXGEN [27]. CVXGEN generează cod în limbajul de programare C pentru probleme de optimizare convexă, utilizând o interfață online și nu necesită instalarea niciunui program adițional. Acest generator de cod transformă o problemă matematică convexă într-un solver extrem de rapid.

Pentru utilizarea CVXGEN, este necesară parcurgerea următorilor pași:

- Pas 1. Modelarea matematică a problemei și transformarea ei într-o problemă convexă;
- Pas 2. Generarea automată a unui solver dedicat clasei de probleme modelate la pasul anterior (pasul 1);
- Pas 3. Obținerea unei soluții numerice.

7.2 Descrierea codului generat

Problema matematică descrisă în capitolul 3, a fost implementată în CVXGEN, unde se setează dimensiunile problemei, se impun dimensiunile parametrilor, a variabilelor de optimizare, funcția obiectiv și se impun constrângerile problemei. În CVXGEN nu se impun valori numerice pentru parametri sau variabile de optimizare (ci doar dimensiunile acestora), acesta fiind motivul pentru care codul generat se poate utiliza în obținerea unei soluții optimale la bord, citind datele de intrare în timp real și având ca rezultat o traiectorie validă în baza acelor valori citite.

După implementarea problemei în CVXGEN, se generează cod C, CVXGEN generează cinci fișiere primare în limbajul de programare C. Algoritmii de rezolvare a problemei se află în fișierul „*solver.c*” care conține funcția principală „*solve*” și rutinele de bază. Factorizarea matricei KKT și soluția sunt rezolvate de funcțiile din „*ldl.c*”, iar „*matrix_support.c*” conține codul C pentru operațiile matriciale și vectoriale. Structurile de date și prototipurile funcțiilor sunt setate în „*solver.h*”, iar „*testsolver.c*” conține un exemplu pentru testarea codului generat. În „*util.c*” se află funcții adiționale pentru testare, iar „*Makefile*” este necesară pentru compilarea automată.

7.3 Utilizarea codului generat de CVXGEN

Deși CVXGEN oferă opțiunea testării individuale a codului generat, în aplicații practice de inginerie, acesta este de obicei integrat în coduri mai complexe. În cazul de față, este necesară integrarea în aplicația de zbor a demonstratorului DTV („flight software”) a codului C generat pentru a asigura posibilitatea testării aplicației SEP în timp real. În cadrul acestei teze, aplicația SEP a fost integrată în software-ul de zbor ca o aplicație de sine-stătătoare și este apelată la fiecare 100 milisecunde. Pe calculatorul de bord este apelată, la fiecare 40 milisecunde, aplicația GNC.

7.4 Funcții auxiliare necesare

În acest subcapitol sunt prezentate funcțiile auxiliare necesare a fi dezvoltate pentru integrarea fișierelor generate utilizând CVXGEN.

7.5 Generarea S-funcției ce conține solver-ul generat și testarea offline a acesteia

S-funcțiile reprezintă funcții ale unui sistem și oferă mecanisme foarte importante pentru extinderea capacităților Simulink. Implementarea unui algoritm scris în limbajul de programare C se poate încapsula într-o S-funcție și apoi utilizată în mediul de dezvoltare Matlab/Simulink. S-funcțiile sunt extrem de utilizate în industrie pentru că se poate integra cod C extern pentru realizarea unor simulări offline și reprezintă diagrame din care se poate genera cod C utilizând un generator automat de cod C (de exemplu, Simulink Coder). Pentru generarea S-funcției care să conțină codul C generat de CVXGEN și funcțiile auxiliare implementate, a fost utilizată funcționalitatea Matlab „Legacy Tool”.

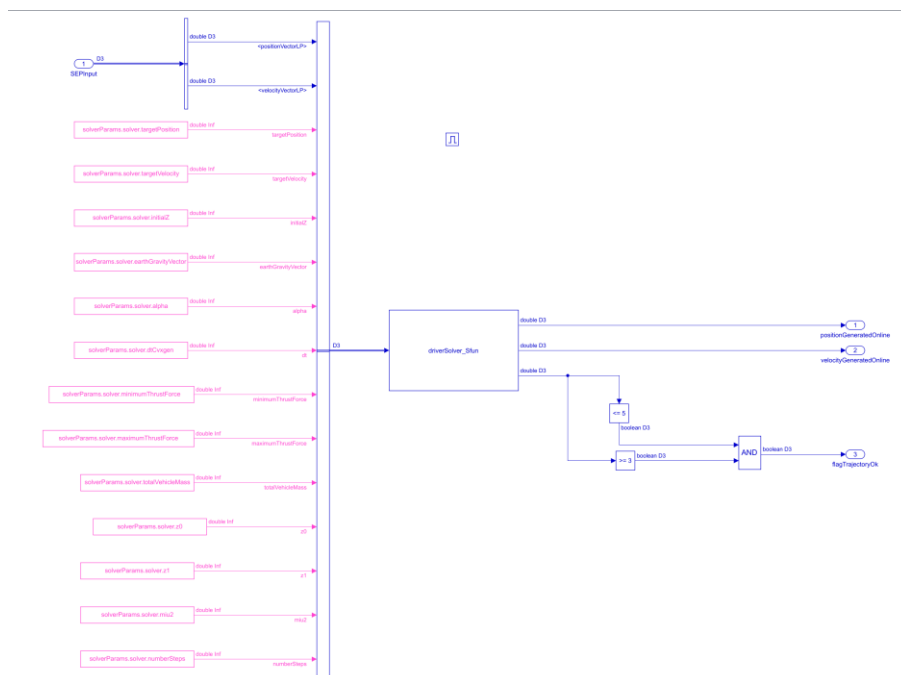


Figura 7.1 Testarea S-funcției generate în Simulink

Datele de ieșire ale S-funcției, ce conține solverul care va rula pe calculatorul de bord sunt următoarele: *Poziția generată online*: o matrice de 3 linii și 24 de coloane; numărul de linii reprezintă poziția pe fiecare dintre cele 3 axe, iar numărul de coloane este numărul de pași preselecțat; *Viteza generată online*: o matrice de 3 linii și 24 de coloane; numărul de linii reprezintă viteza pe fiecare dintre cele 3 axe, iar numărul de coloane este numărul de pași preselecțat; *Valoarea minimă a funcției obiectiv*.

Pentru vizualizarea traiectoriei și realizarea unor grafice în funcție de timp, este necesară o interpolare a valorilor optime calculate de solverul convex. Astfel că procesarea datelor brute de ieșire a fost realizată tot în Simulink.

7.6 Misiune nominală

Pentru testarea codului generat, mai precis a aplicației în timp real dezvoltate, care conține solverul de optimizare online a traiectoriei demonstratorului DTV, a fost selectată aceeași misiune nominală ca în cazul prezentat în capitolul 6, iar lista completă de parametri este prezentată în Tabelul 7.1 și în Tabelul 7.2:

Tabelul 7.1 Valori numerice parametri ficși – caz nominal

Parametru	Valoare numerică
m_i [Kg]	64.5
g_0 $\left[\frac{m}{s^2}\right]$	9.08665
α $\left[\frac{kg}{s}\right]$	0.00043
Δt [s]	0.7
T_1 [N]	600
T_2 [N]	750
z_0 [–]	4.1058
z_1 [–]	4.0967
μ_2	10.47
N	24

Tabelul 7.2 Condiții inițiale și finale – caz nominal

Parametru	Condiții inițiale traiectorie	Condiții finale traiectorie
r_x [m]	5	0
r_y [m]	0	5
r_z [m]	0	4
V_x [m]	0	0
V_y [m]	0	0
V_z [m]	0	0

Utilizând aceste date de intrare și realizând o simulare numerică în Matlab / Simulink, obținem rezultatele numerice prezentate în figurile următoare, testându-se astfel funcționalitatea

S-funcției ce încapsulează solverului convex pentru optimizarea traiectoriei de aterizare pentru DTV.

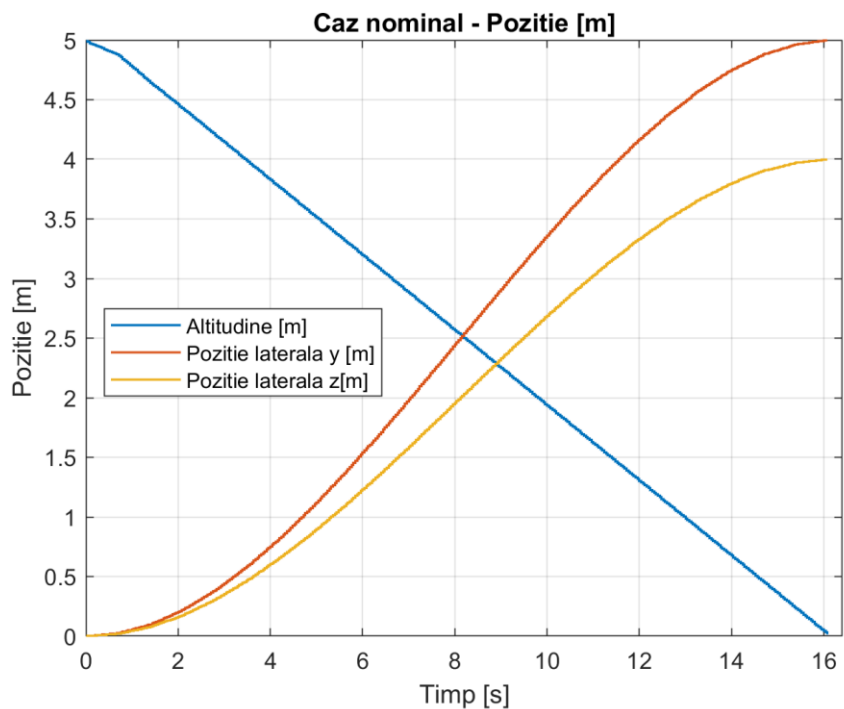


Figura 7.2 Poziție DTV vs. timp, caz nominal

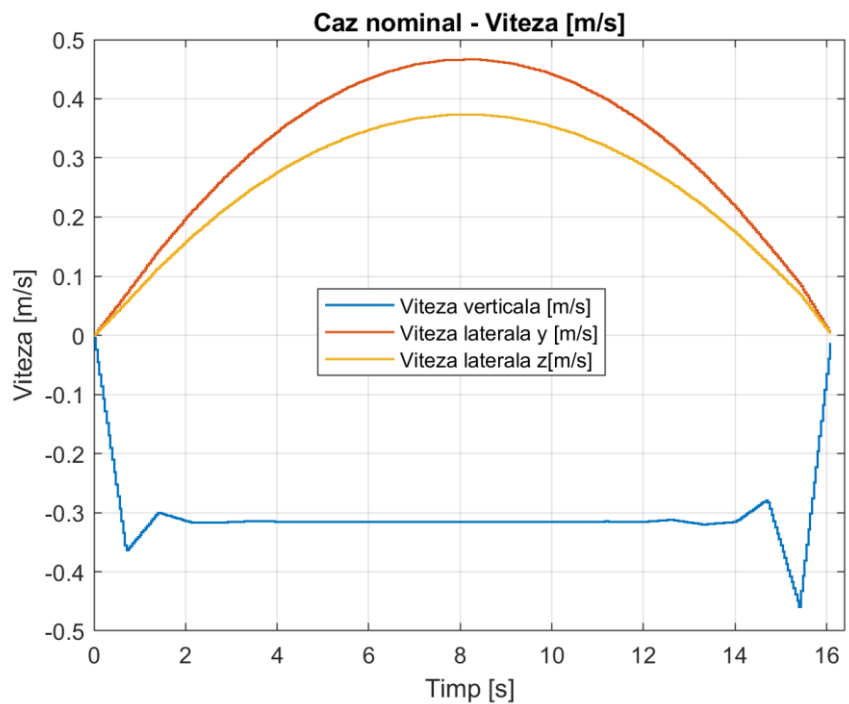


Figura 7.3 Viteza DTV vs. timp, caz nominal

Figura 7.2 conține poziția DTV în funcție de timp, pe cele trei axe. Se poate observa respectarea

condițiilor inițiale și a celor finale pentru traiectorie. Misiunea experimentală are drept altitudine inițială 5 m, iar pozițiile laterale sunt 0 pe ambele axe. Condiția finală pentru poziții este și ea respectată pentru fiecare dintre cele trei axe.

Figura 7.3 prezintă viteza demonstratorului în funcție de timp, pe cele trei axe. De asemenea, se poate observa că sunt respectate condițiile inițiale și finale pentru viteze. O altă observație este legată de vitezele maxime ale vehiculului, care nu depășesc 0.5 m/s, ceea ce conduce la concluzia că traiectoria nominală selectată este una fezabilă pentru capacitățile reale ale DTV. În Figura 7.4 se poate observa traiectoria optimală 3D pentru cazul nominal.

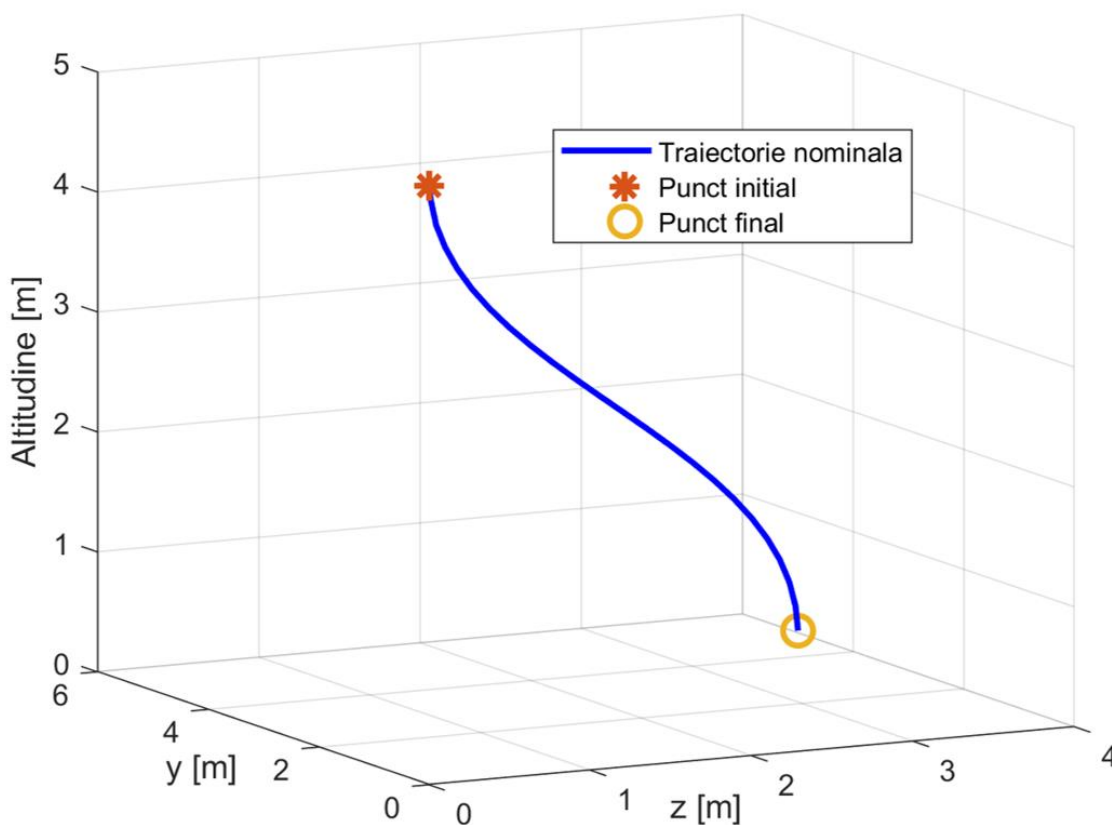


Figura 7.4 Traiectorie 3D, caz nominal

Aceste rezultate numerice pot fi, de asemenea, comparate cu rezultatele prezentate în capitolul 6, obținute prin implementarea problemei matematice în Matlab, utilizând modulul CVX. Deși problema matematică este aceeași și datele de intrare sunt identice pentru această misiune nominală, modalitatea de testare este diferită:

- În cazul simulărilor numerice utilizând CVX, problema matematică este definită în Matlab și soluționată cu solver-ul SeDumi (care este scris în cod C) și utilizat în mediul Matlab prin intermediul fișierelor .mex.
- În cazul simulărilor numerice utilizând CVXGEN și funcțiile adiționale, codul de calcul este scris în limbaj de programare C, ceea ce îl face să fie potrivit pentru rulare în timp real pe calculatorul de bord al unui demonstrator.

7.7 Analiză Monte Carlo

Pentru a testa funcționalitatea în prezența incertitudinilor a aplicației în timp real dezvoltate și implementate, a fost realizată o analiză de tip Monte Carlo în prezența incertitudinilor pentru poziții și viteze inițiale, iar rezultatele demonstrează că solver-ul convex găsește întodeauna o soluție optimală și fezabilă pentru demonstratorul DTV în condițiile considerate.

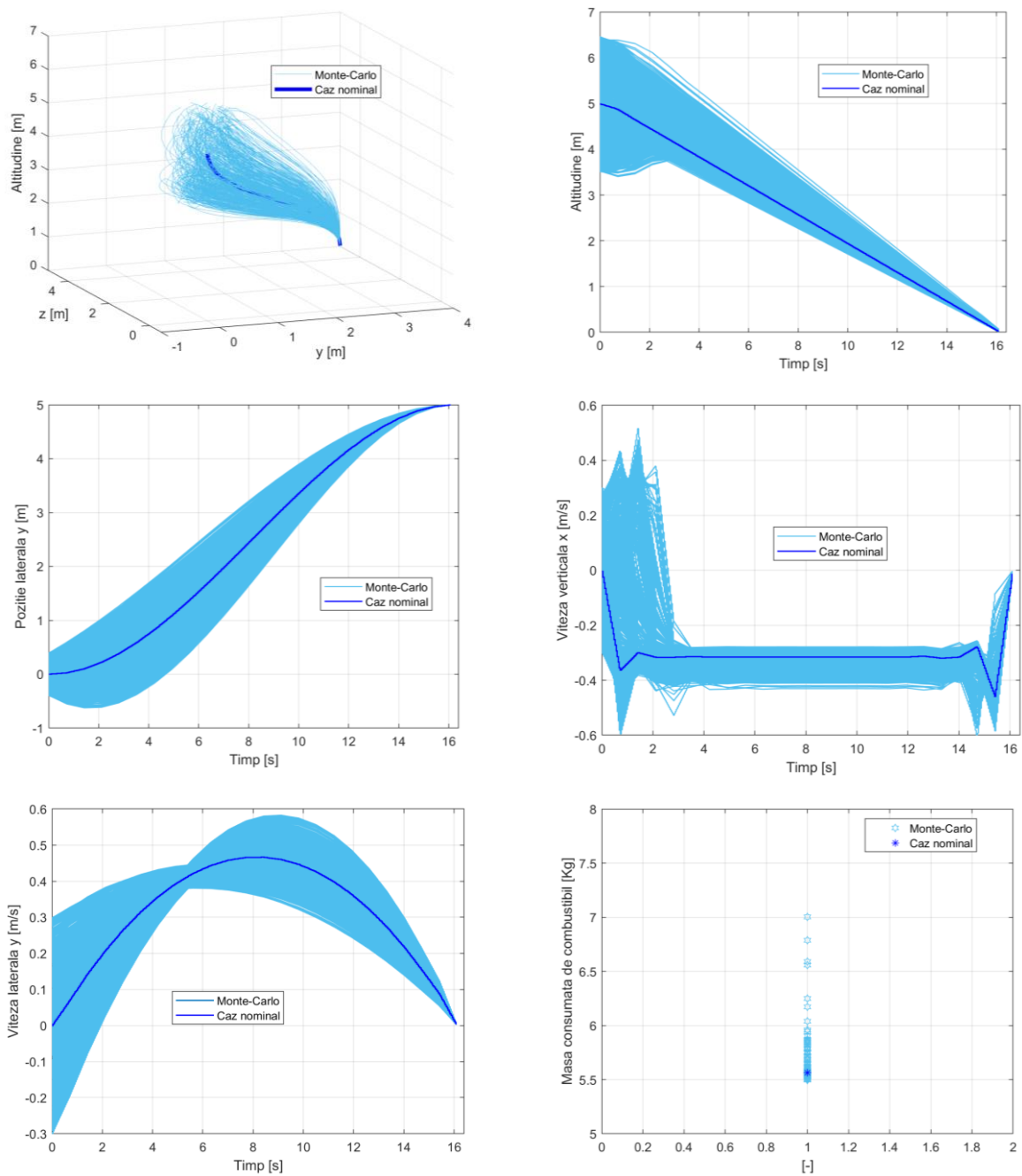


Figura 7.5 Analiză Monte-Carlo – rezultate

8 Simulări utilizând modelul dinamic cu șase grade de libertate (6DOF)

8.1 Modelare matematică a demonstratorului DTV

Formularea mecanicii newtoniene este utilizată pentru a scrie ecuațiile matematice ale modelului dinamic 6DOF specific vehiculului DTV (corp cu masă variabilă) și se vor aplica teoremele generale ale impulsului (pentru a descrie mișcarea de translație a centrului de masă) și ale momentului cinetic (pentru a descrie mișcarea în jurul centrului de masă). În continuare este prezentat modelul dinamic pentru vehiculul DTV plecând de la modelul general detaliat în [28].

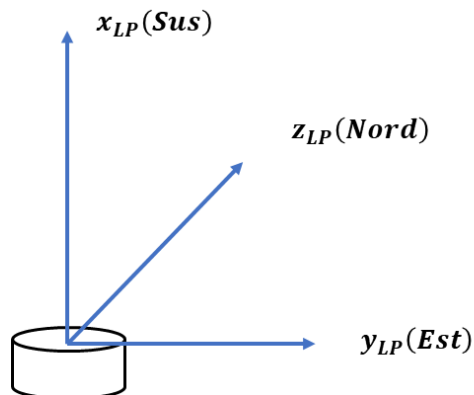


Figura 8.1 Triedrul inerțial LP

Pentru descrierea translației centrului de masă este utilizat triedrul local de start inerțial (LP - “launch pad frame”, prezentat în Figura 8.1). Pentru descrierea mișcării de rotație în jurul centrului de masă, este utilizat sistemul de referință legat de corp (B - “body frame”) și este prezentat în Figura 8.2 având originea sistemului în centrul de masă al vehiculului.

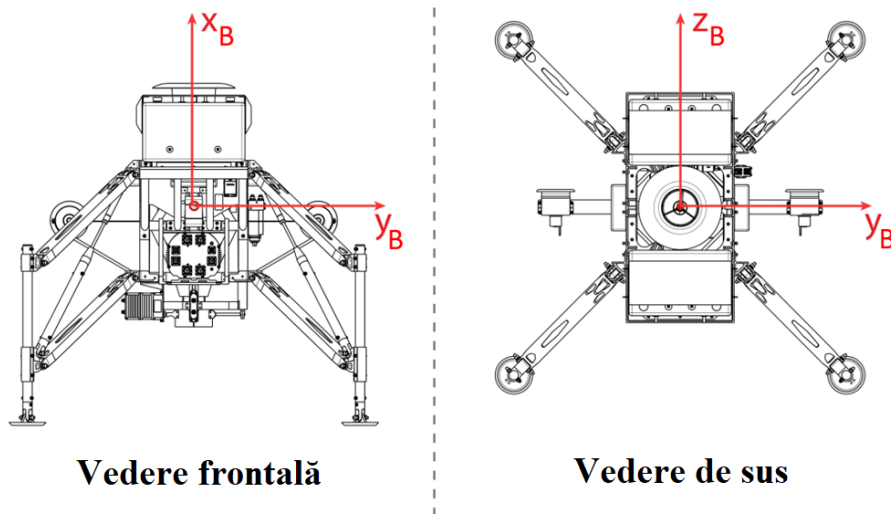


Figura 8.2 Triedrul legat de corp – DTV. Credit: INCAS

Ecuțiile de translație (cinematice și dinamice), în formă vectorială, sunt următoarele:

$$\begin{aligned}\dot{\mathbf{r}} &= \mathbf{v} \\ \mathbf{m}\dot{\mathbf{v}} &= \mathbf{N}\end{aligned}\quad (8.1)$$

unde:

- \mathbf{r} reprezintă vectorul de poziție al vehiculului (al centrului de masă) în coordonate carteziane;
- \mathbf{v} reprezintă viteza vehiculului (al centrului de masă) în coordonate carteziane;
- \mathbf{m} reprezintă masa vehiculului (instantanee);
- \mathbf{N} reprezintă forța aplicată, care este alcătuită din forța de tracțiune generată de motor (\mathbf{F}_T), forța generată de sistemul de vane (\mathbf{F}_V), forța generată de sistemul de ventilatoare (\mathbf{F}_{DF}), forța datorată gravitației (\mathbf{F}_g) și o forță perturbatoare (\mathbf{F}_d):

$$\mathbf{N} = \mathbf{F}_T + \mathbf{F}_V + \mathbf{F}_{DF} + \mathbf{F}_g + \mathbf{F}_d \quad (8.2)$$

Forța de tracțiune generată de motor (\mathbf{F}_T) se calculează folosind:

$$\mathbf{F}_T = \mathbf{A}_{LP_B} \begin{pmatrix} T \\ 0 \\ 0 \end{pmatrix} \quad (8.3)$$

unde T este tracțiunea motorului, iar \mathbf{A}_{LP_B} este matricea de rotație din sistemul legat de corp (B) la sistemul local de start (LP) și are forma:

$$\mathbf{A}_{LP_B} = \begin{bmatrix} \cos\psi \cos\theta & \cos\psi \sin\theta \sin\phi - \sin\psi \cos\phi & \sin\psi \sin\phi + \cos\psi \sin\theta \cos\phi \\ \sin\psi \cos\theta & \cos\psi \cos\phi + \sin\psi \sin\theta \sin\phi & \sin\psi \sin\theta \cos\phi - \cos\psi \sin\phi \\ -\sin\theta & \cos\theta \sin\phi & \cos\theta \cos\phi \end{bmatrix} \quad (8.4)$$

Forța generată de sistemul de vane (\mathbf{F}_V) se calculează folosind:

$$\mathbf{F}_V = \mathbf{A}_{LP_B} \begin{pmatrix} 0 \\ F_{Vy} \\ F_{Vz} \end{pmatrix} \quad (8.5)$$

unde F_{Vy} și F_{Vz} sunt cele două vane ale vehiculului DTV. De menționat este faptul că F_{Vy} și F_{Vz} pot lua valori atât pozitive cât și negative, în funcție de deflexiile acestora. Convenția de semne este prezentată în Figura 8.3.

Forța generată de sistemul de ventilatoare de tip “*duct-fan*” (\mathbf{F}_{DF}) se calculează folosind:

$$\mathbf{F}_{DF} = \mathbf{A}_{LP_B} \begin{pmatrix} 0 \\ 0 \\ F_{DF_stang} + F_{DF_drept} \end{pmatrix} \quad (8.6)$$

De menționat faptul că forțele F_{DF_stang} și F_{DF_drept} sunt întotdeauna pozitive, conform convenției din Figura 8.4.

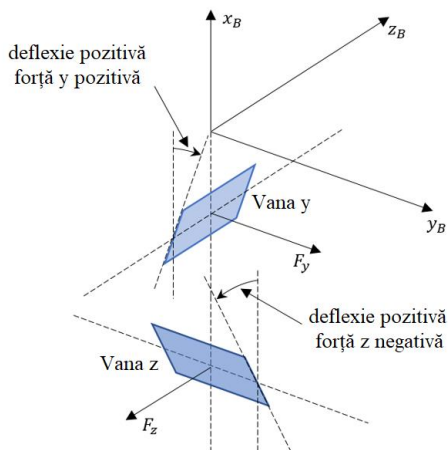


Figura 8.3 Convenția de semne, deflexii pozitive ale vanelor

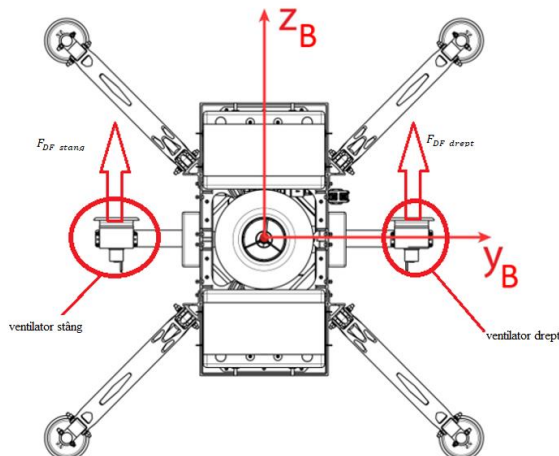


Figura 8.4 Convenția de semne, sistemul de ventilatoare (control al ruliului). Credit: INCAS

Forța datorată gravitației (F_g) se calculează folosind:

$$\mathbf{F}_g = \begin{Bmatrix} -mg \\ 0 \\ 0 \end{Bmatrix} \quad (8.7)$$

unde g este accelerația gravitațională standard.

Din considerente conservative, se ia în calcul și o forță perturbatoare (F_d).

Ecuțiile de rotație în jurul centrului de masă (cinematice și dinamice) în formă vectorială, sunt următoarele:

$$\begin{aligned} \dot{\phi} &= p + q \sin \phi \tan \theta + r \cos \phi \tan \theta \\ \dot{\theta} &= q \cos \phi - r \sin \phi \\ \dot{\psi} &= q \frac{\sin \phi}{\cos \theta} + r \frac{\cos \phi}{\cos \theta} \end{aligned} \quad (8.8)$$

$$\mathbf{I}\dot{\boldsymbol{\omega}} + \boldsymbol{\omega} \times (\mathbf{I}\boldsymbol{\omega}) = \mathbf{M}_T + \mathbf{M}_V + \mathbf{M}_{DF} + \mathbf{M}_r + \mathbf{M}_d$$

unde:

- ϕ , θ și ψ reprezintă unghiurile de ruli, tangaj și girație (Euler ZYX);
- p , q , și r reprezintă componentele vectorului viteză unghiulară $\boldsymbol{\omega}$ în triedrul legat de corp (B);
- \mathbf{I} reprezintă matricea momentelor de inerție în triedrul legat de corp (B);
- \mathbf{M}_T reprezintă momentul produs de forța de tracțiune a motorului \mathbf{F}_T , descrisă în (8.3);
- \mathbf{M}_V reprezintă momentul produs de forța generată de sistemul de vane \mathbf{F}_V , descrisă în (8.5);

Optimizare convexă pentru demonstratoare reutilizabile cu aplicații spațiale

- M_{DF} reprezintă momentul produs de sistemul de ventilatoare F_{DF} , descrisă în (8.6);
- M_r reprezintă momentul produs rotorul motorului principal (I_r este matricea momentelor de inerție ale rotorului relativ la centrul de masă al vehiculului, iar ω_r este viteza unghiulară a rotorului), descris de:

$$M_r = -I_r \dot{\omega}_r - I_r (\omega \times \omega_r) - \omega_r \times (I_r \omega_r) - \omega_r \times (I_r \omega) - \omega \times (I_r \omega_r) \quad (8.9)$$

- M_d reprezintă momentul perturbator (introdus din considerente conservative).

8.2 Integrarea S-funcției în simulatorul de zbor al vehiculului DTV

Simulatorul de zbor dezvoltat de INCAS pentru DTV poate accepta programe de calcul experimentale. Pentru a testa și valida în zbor o aplicație experimentală, este necesară integrarea acesteia în simulatorul de zbor pentru testarea în buclă închisă, utilizând modelele cu șase grade de libertate. Blocul „SEP” (Software Experimental Payload) din Figura 8.5 conține programul de optimizare convexă de traiectorie online dezvoltat în cadrul acestei teze de doctorat, dar și funcțiile suplimentare dezvoltate.

Așa cum este evidențiat în Figura 8.5, blocul SEP are culoarea albastră, ceea ce reprezintă, în acest caz, că execuția lui realizându-se la fiecare 100 milisecunde. Detalii suplimentare pot fi observate în Figura 8.6 și Figura 8.7.

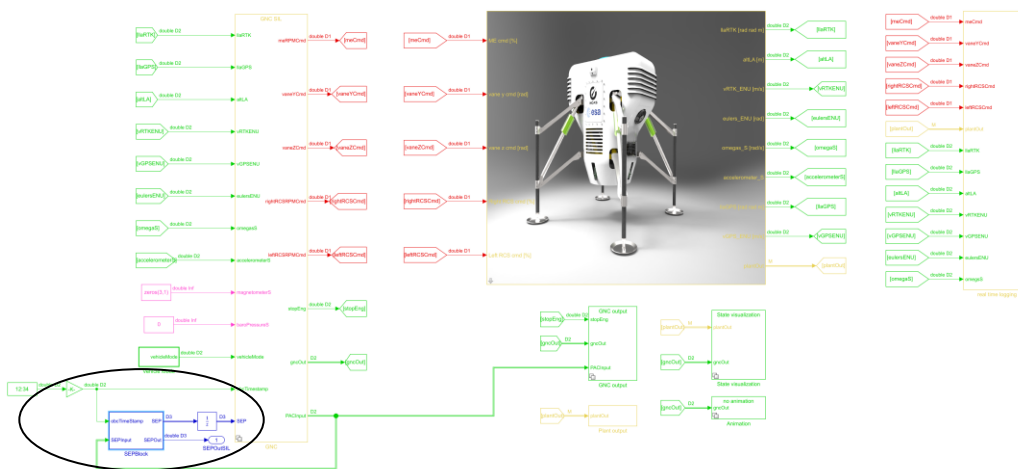


Figure 8.5 Integrarea solverului online în Simulatorul de zbor al DTV

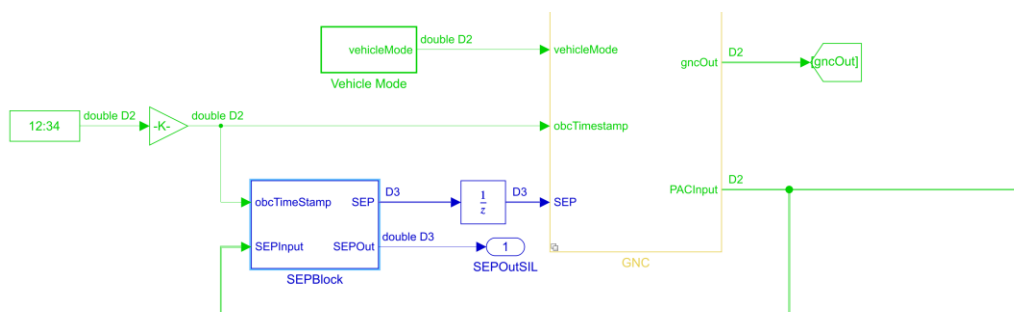


Figure 8.6 Integrarea SEP în Simulatorul de zbor al DTV (detaliu)

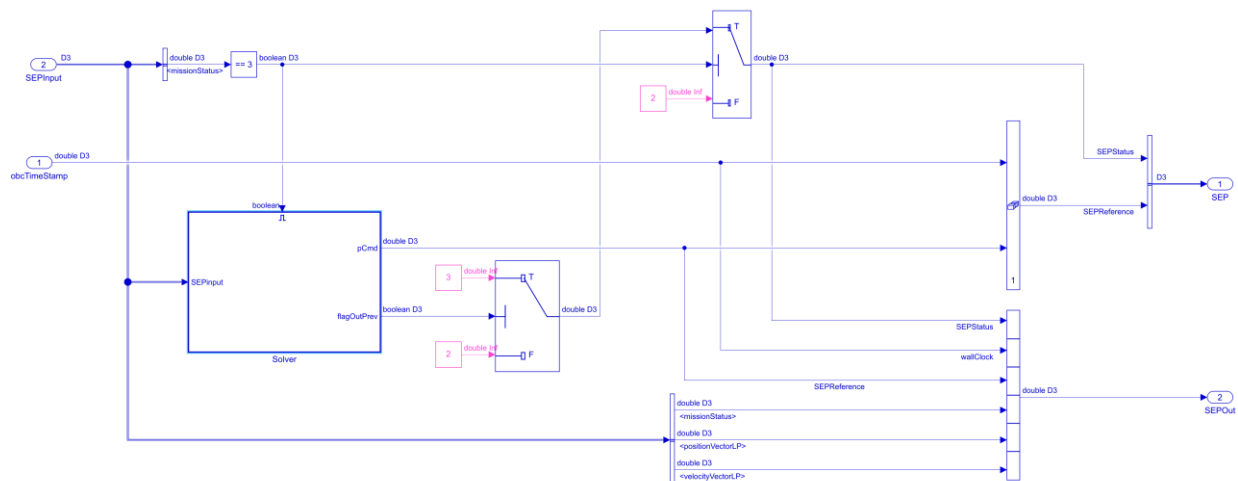


Figura 8.7 Conținutul blocului SEP - Simulink

Datele de ieșire din blocul SEP sunt următoarele:

- **SEPStatus**: reprezintă semnalul care stabilește dacă traiectoria trimisă calculatorului de bord este traiectoria de referință sau traiectoria experimentală. Simulatorul de zbor al DTV conține un program care decide, în baza unor parametri, dacă traiectoria calculată online este validă înainte să fie trimisă vehicului;
- **SEPReference**: conține traiectoria calculată online (Pozițiile și vitezele pe cele 3 axe).

8.3 Rezultate numerice offline

În acest subcapitol sunt prezentate rezultatele obținute utilizând simulatorul de zbor (model protejat) pentru DTV, ceea ce reprezintă testarea în buclă închisă.

DTV pornește cu o traiectorie de referință predefinită: urcare până la o altitudine de 5 metri, trece în faza de zbor la punct fix, iar după 15 secunde aplicația în timp real dezvoltată în cadrul acestei teze de doctorat începe să calculeze la bord o traiectorie. În momentul în care se începe calculul traiectoriei la bord, aplicația cere date de la aplicația GNC despre poziția și vitezele DTV înregistrate în timp real în zbor și utilizează aceste date ca punct de pornire. Poziția țintă în care trebuie să ajungă DTV este 0.65 m deasupra solul ului (poziția centrului de masa), 5 metri în direcția y și 4 metri în direcția z. Parametrii predefiniți pentru aplicația în timp real sunt cei prezentați în subcapitolul 7.6.

Misiunea completă pentru DTV este prezentată în Figura 8.8, unde este evidențiată partea de misiune în care traiectoria de referință este generată la bord.

Figura 8.9 prezintă traiectoria generată online, la bordul DTV. Se poate observa altitudinea de la care începe calculul online (5 metri), dar și pozițiile laterale finale, prezentând-se astfel respectarea constrângerilor impuse aplicației dezvoltate.

Vitezele pe cele 3 axe sunt ilustrate în Figura 8.10 și se poate observa că acestea au valori sub 0.5 m/s, respectând-se astfel constrângerile legate de viteza maximă de zbor. Atitudinea vehicului este prezentată în Figura 8.11, iar vitezele unghiulare de ruliu, tangaj și girație se observă în Figura 8.12.

Optimizare convexă pentru demonstratoare reutilizabile cu aplicații spațiale

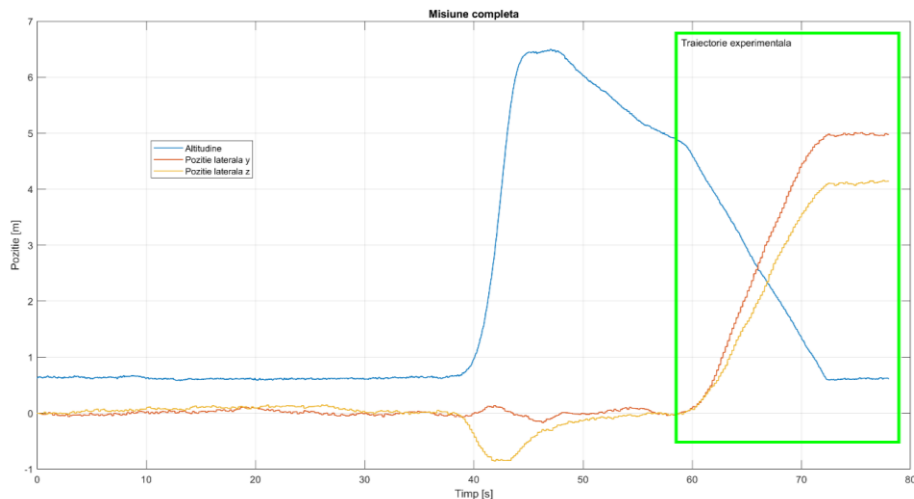


Figura 8.8 Misiune completa – simulare 6DOF

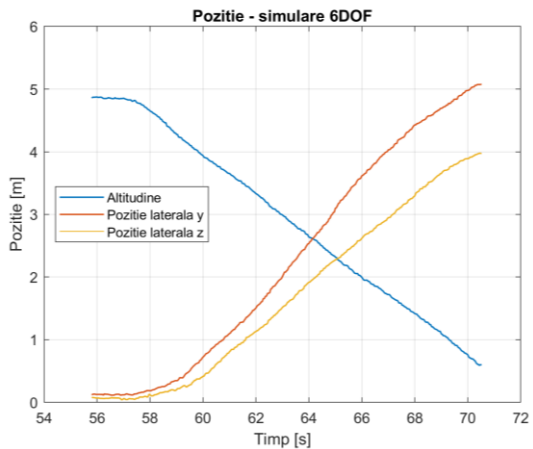


Figura 8.9 Poziție – simulare 6DOF

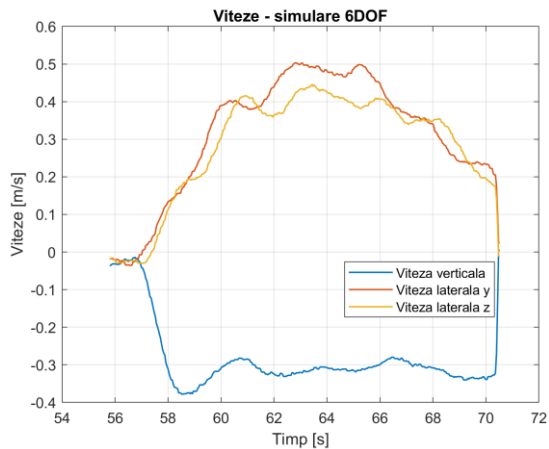


Figura 8.10 Viteze - simulare 6DOF

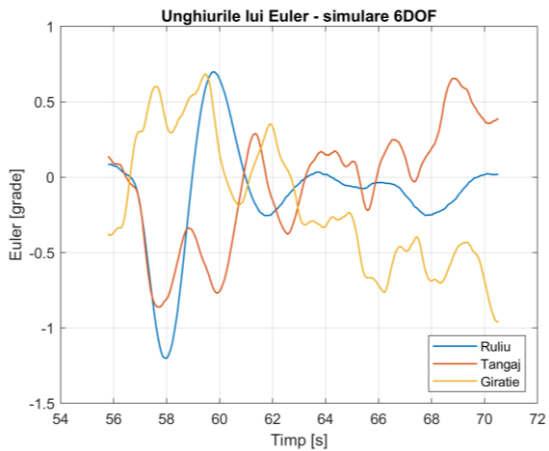


Figura 8.11 Atitudine – simulare 6DOF

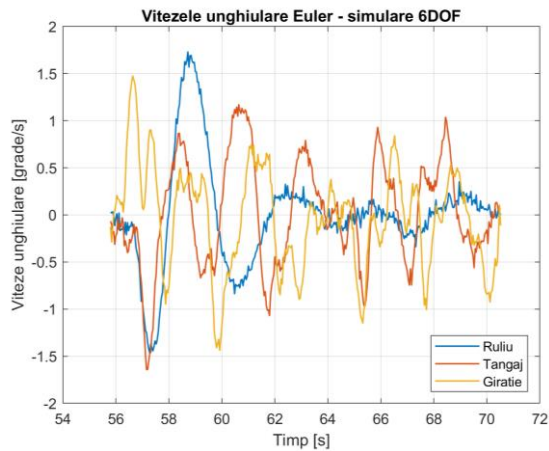


Figura 8.12 Viteze unghiulare – simulare 6DOF

9 Testarea aplicației SEP utilizând Real Time Target Machine

9.1 Arhitectura de testare a aplicației în timp real dezvoltate

Pentru testarea aplicației în timp real, ce conține solverul convex, este necesară integrarea acesteia în software-ul de zbor al DTV. Sistemul de operare care rulează pe calculatorul de bord cu care este echipat DTV este Linux Real Time. Software-ul de zbor este nivelul principal și cel care „decide” ordinea de execuție a proceselor. În cadrul software-ului de zbor există integrat aplicația GNC, iar schimbul de semnale între GNC și aplicația SEP dezvoltată este asigurat de acest software de zbor prin intermediul unor semnale predefinite.

Figura 9.1 ilustrează echipamentele necesare pentru testarea în timp real a aplicației SEP dezvoltate. Utilizând programele realizate în mediul de dezvoltare Matlab/Simulink, se generează automat cod pentru o aplicație de testare în timp real. Sistemele de testare în timp real, SpeedGoat, sunt concepute special pentru „Simulink Real-Time” [29], acesta fiind motivul pentru care a fost selectată această modalitate de testare. Pe SpeedGoat rulează modelul dinamic în șase grade de libertate al DTV și trimite semnale simulând misiunea nominală în timp real. Aceste semnale sunt mai departe trimise către calculatorul de bord pe care rulează sistemul GNC și aplicația de zbor a vehiculului, ca două procese separate. Toate aceste echipamente hardware se află într-o rețea locală LAN putând a fi schimbate astfel semnale și fișiere între ele.

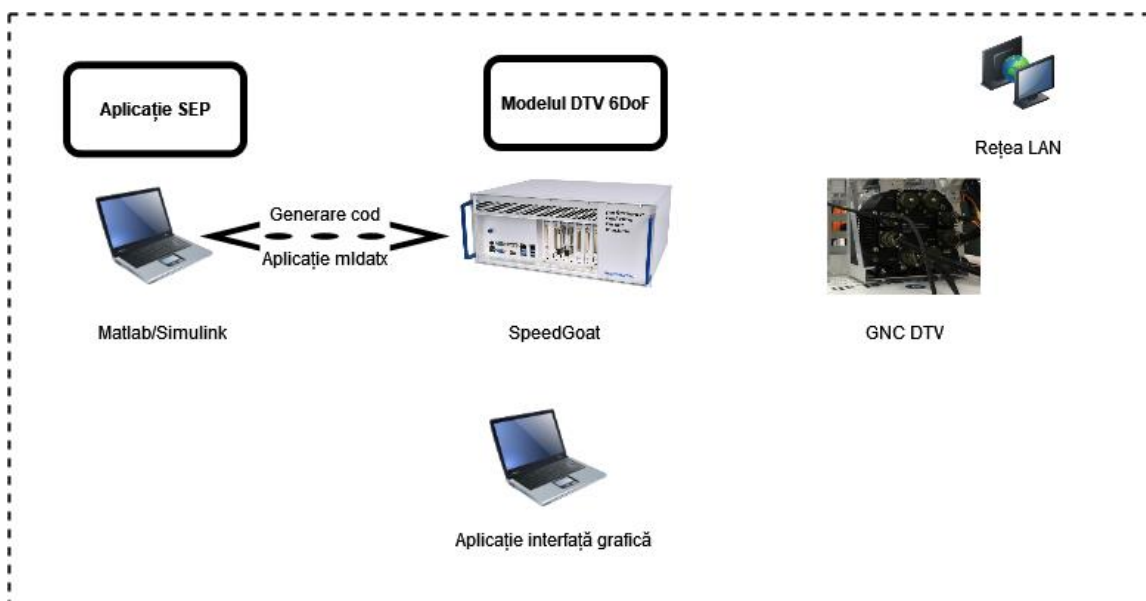


Figura 9.1 Modalitatea de testare în timp real

9.2 Generarea S-funcțiilor necesare pentru integrarea aplicației în timp real în software-ul de zbor

Blocul SEP descris în detaliu în cadrul subcapitolului 8.1, este testat în timp real utilizând procedura descrisă în subcapitolul 9.1, în care se înlocuiesc semnalele simulate cu S-funcții care fac legătura între aplicația SEP și software-ul de zbor. Aceste semnale au rolul de a asigura schimbul de date între software-ul de zbor, aplicația SEP și aplicația GNC.

9.3 Rezultatele simulării misiunii nominale în timp real (HIL)

Rezultatele rulării în timp real utilizând aplicația dezvoltată sunt prezentate în Figura 9.2 și Figura 9.3. Acestea sunt trimise mai departe prin intermediul S-funcțiilor către GNC-ul demonstratorului și devin noile referințe de poziție și viteze pentru DTV.

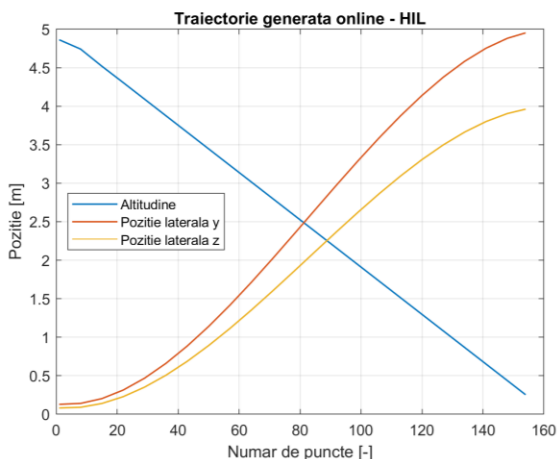


Figura 9.2 Traiectorie optimă online – Simulări HIL

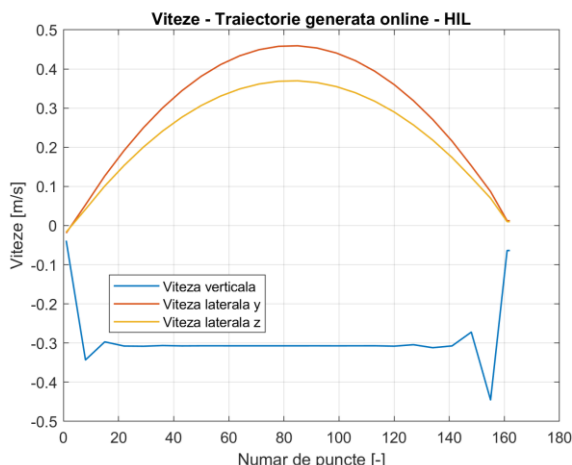


Figura 9.3 Viteze calculate online – Simulări HIL

Traectoria optimă este prezentată în Figura 9.2, iar vitezele de zbor calculate la bord sunt prezentate grafic în Figura 9.3. Se poate observa că toate constrângerile finale impuse ca parametri aplicației sunt respectate.

Pozițiile și vitezele simulate (în șase grade de libertate) în timp real sunt prezentate în Figura 9.4 și Figura 9.5 și se poate observa că sunt similare cu cele calculate de aplicație pe calculatorul de zbor, ceea ce înseamnă că DTV poate urma traiectoria optimizată online cu o eroare mică inclusiv în timp real. Unghiurile Euler și vitezele unghiulare ca urmare a simulărilor în șase grade de libertate pot fi vizualizate în Figura 9.6 și Figura 9.7. Deoarece vitezele comandate sunt mici, unghiurile de atitudine necesare sunt de asemenea mici.

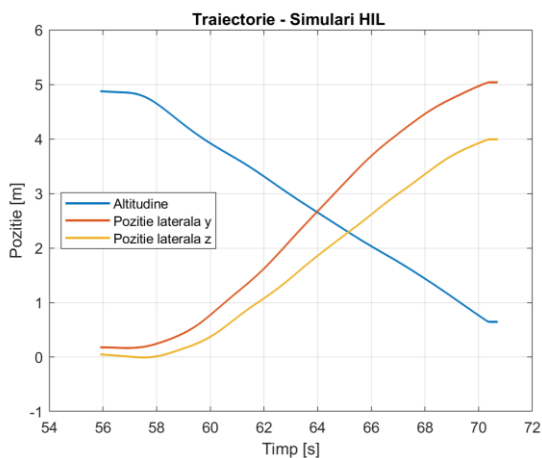


Figura 9.4 Traiectorie – simulări HIL

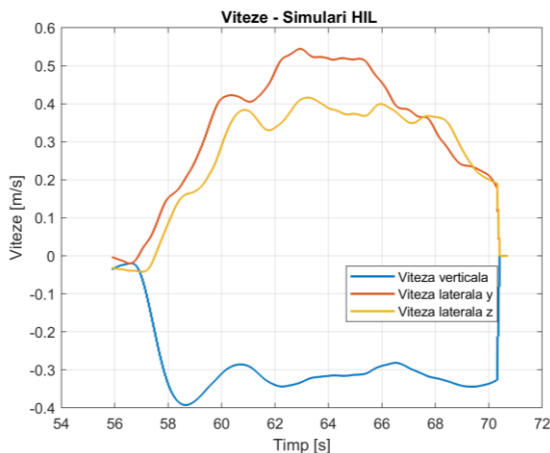


Figura 9.5 Viteze DTV - simulări HIL

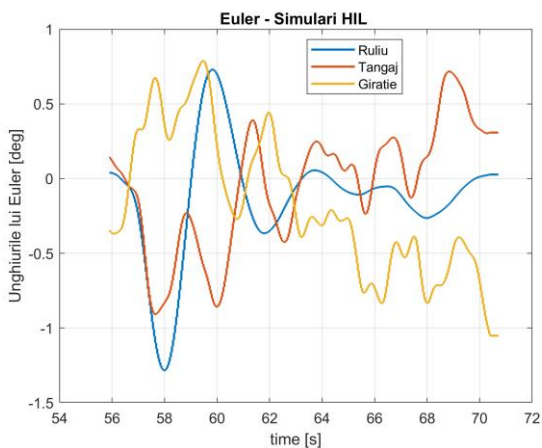


Figura 9.6 Atitudine DTV - simulări HIL

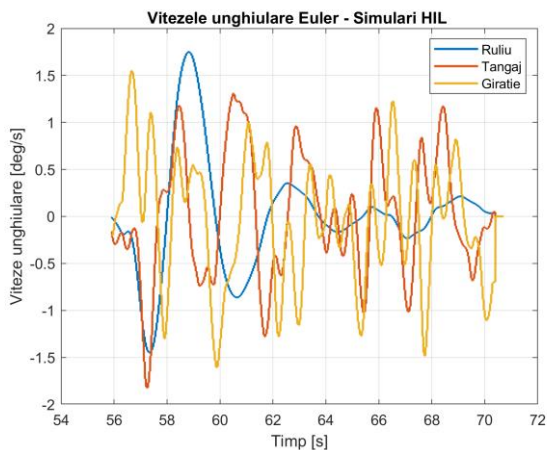


Figura 9.7 Viteze unghiulare Euler - simulări HIL

Rezultatele testării aplicației dezvoltate în condiții de timp real confirmă posibilitatea utilizării metodologiei și a algoritmilor dezvoltați în cadrul acestei lucrări de doctorat la bordul demonstratoarelor reutilizabile cu aplicații spațiale. Timpul de execuție a aplicației SEP dezvoltate pe calculatorul de bord al demonstratorului DTV în timp real a aplicației în timp real dezvoltate este de aproximativ 10 milisecunde, ceea ce confirmă fezabilitatea utilizării acesteia în timpul unei misiuni reale.

10 Testarea și validarea aplicației în condiții de zbor captiv

Pentru validarea în zbor a aplicației dezvoltate, a fost realizată o campanie experimentală de testare la sediul INCAS din Măneciu, Prahova. Cinci zboruri experimentale au fost realizate cu succes și rezultatele sunt prezentate în detaliu în acest capitol. Înaintea fiecărui zbor experimental, au fost realizate simulări numerice utilizând simulatorul de zbor cu șase grade de libertate pentru asigurarea fezabilității acestora. În Figura 10.1 este prezentată o comparație între simulare offline și rezultate experimentale utilizând aplicația SEP dezvoltată.

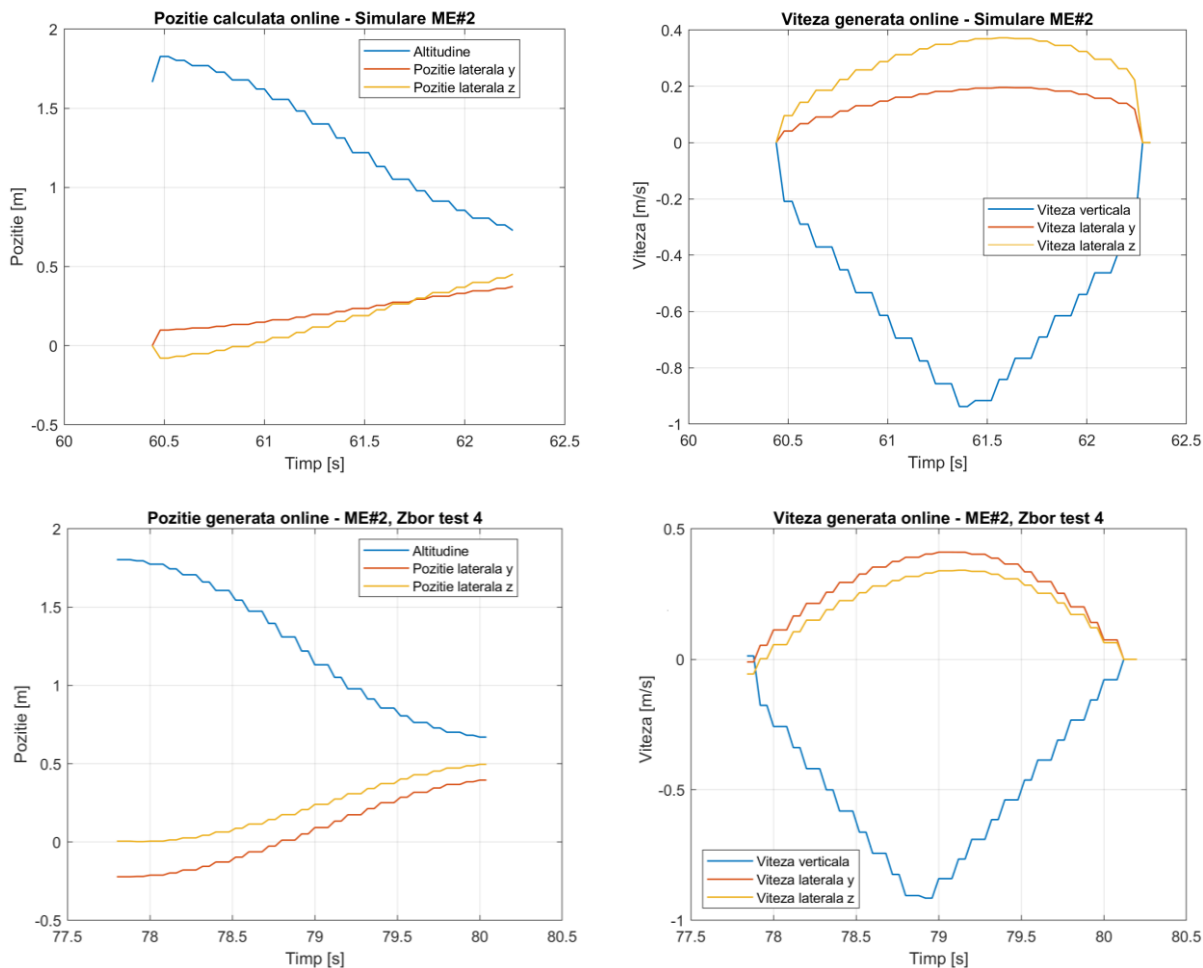


Figura 10.1 Poziție și viteze online (simulare și zbor experimental)

Rezultatele înregistrate în timpul zborului de test nr. 4 sunt prezentate în Figura 10.2. Traectoria calculată la bord respectă constrângerile impuse, iar această traectorie devine noua referință pentru demonstratorul DTV. Sunt prezentate de asemenea comparativ comenzile și stările reale ale DTV și se poate observa că demonstratorul urmărește traectoria calculată la bord cu o precizie foarte bună ținând cont de spațiul fizic de zbor limitat. Comenzile calculate pentru sistemul de vane, forța de tracțiune și unghiurile de atitudine sunt prezentate, de asemenea.

Optimizare convexă pentru demonstratoare reutilizabile cu aplicații spațiale

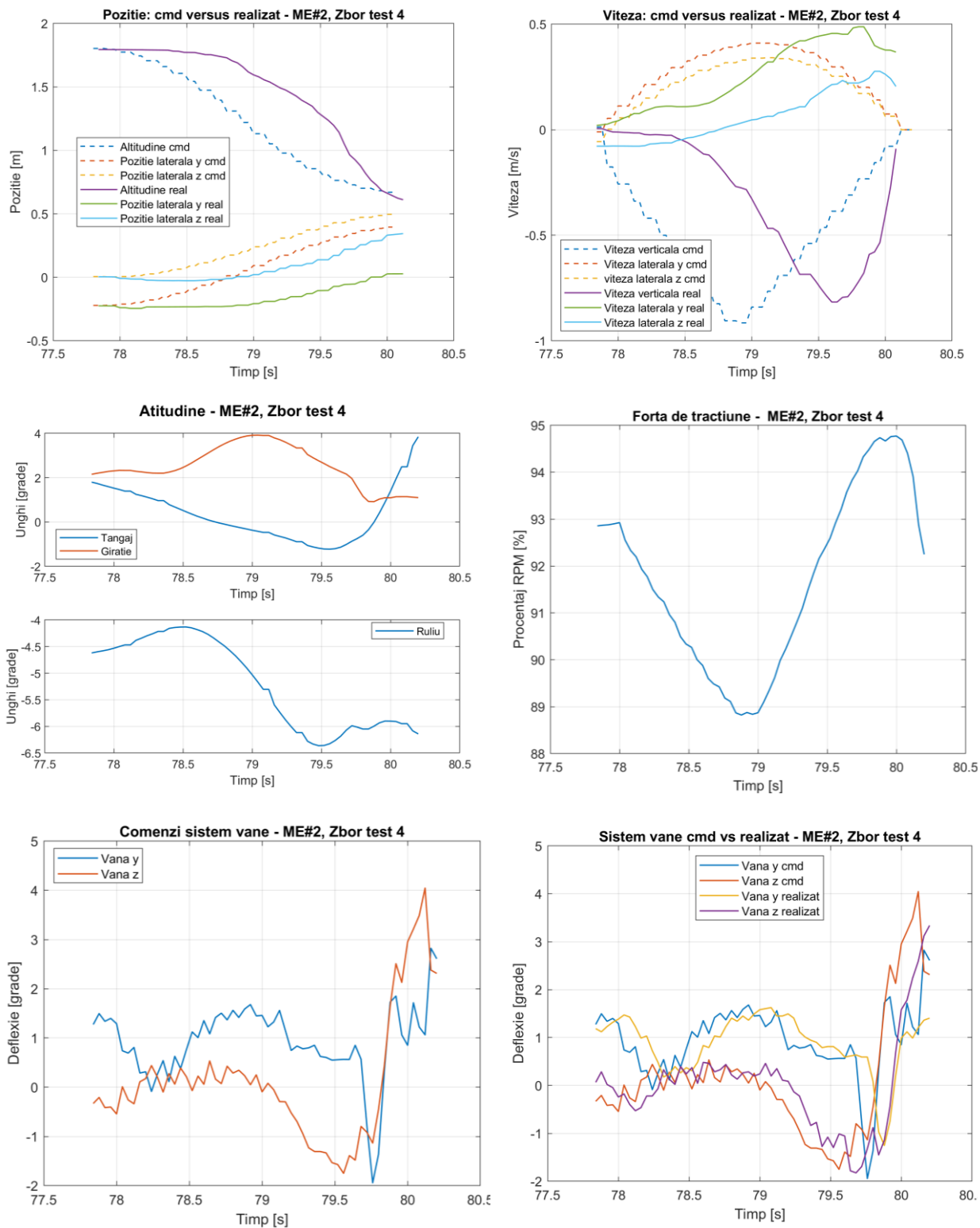


Figura 10.2 Rezultate experimentale, ME#2, zbor test nr. 4

Optimizare convexă pentru demonstratoare reutilizabile cu aplicații spațiale

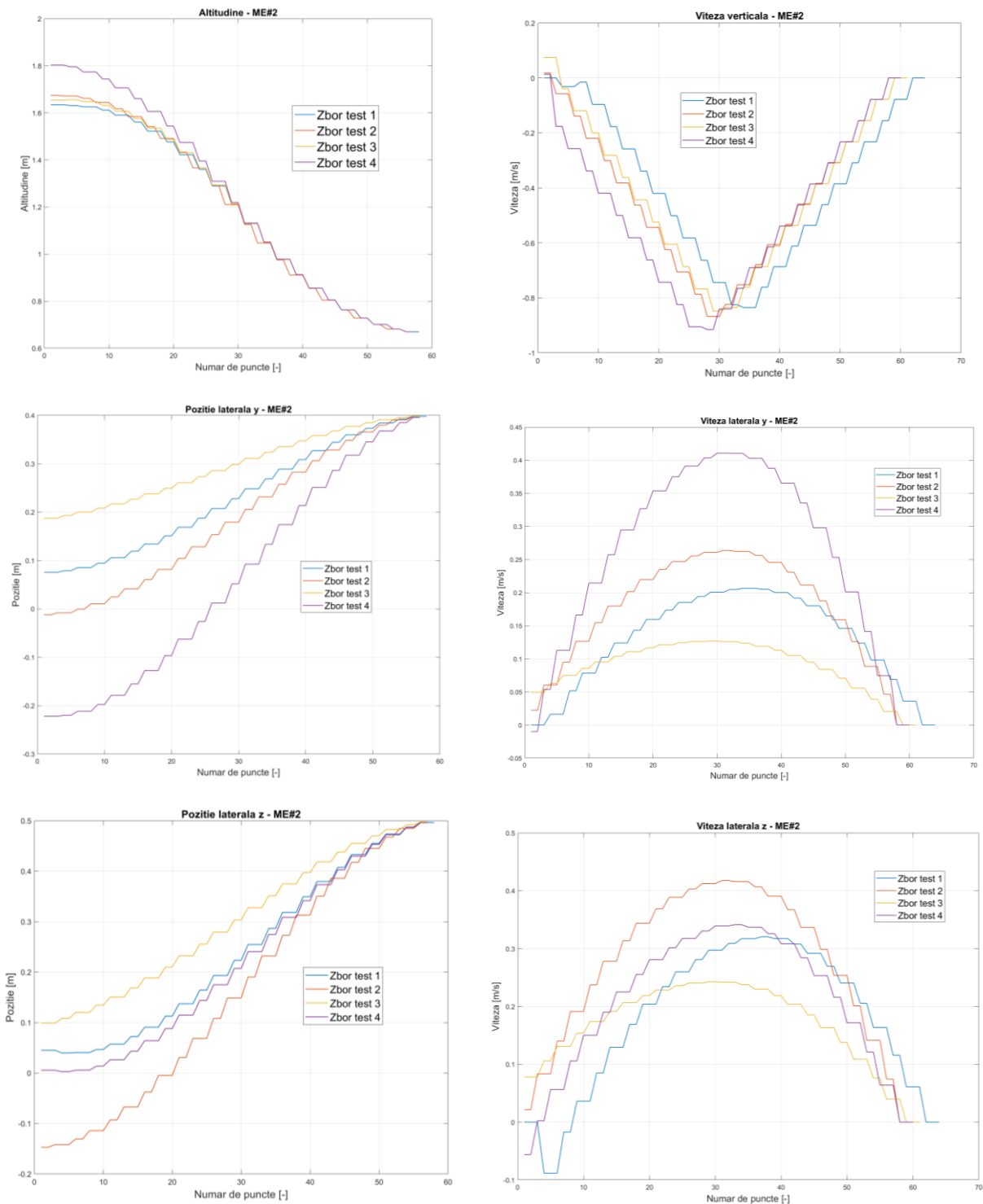


Figura 10.3 Rezultate comparative zboruri experimentale ME#2

Rezultatele experimentale obținute în zbor demonstrează faptul că aplicația propusă și dezvoltată în cadrul acestei lucrări de doctorat se poate utiliza în timp real și poate optimiza traiectoria de aterizare pe calculatorul de bord al vehiculelor spațiale. Toate zborurile de test au fost realizate cu succes, algoritmul dezvoltat a calculat în fiecare caz o traiectorie optimală pe

calculatorul de la bordul demonstratorului, indiferent de dispersia stărilor inițiale ale demonstratorului.

Pentru toate zborurile realizate în campania experimentală, timpul de execuție pe calculatorul de bord al algoritmului pentru generarea unei traiectorii optimale nu a depășit 70 milisecunde. Timpul total este reprezentat de suma următoarelor evenimente care se întâmplă în mod succesiv:

- Algoritmul experimental apelează, prin intermediul S-funcțiilor descrise în subcapitolul 9.2, funcția GNC care rulează ca un proces separat, și cere datele legate de starea reală a demonstratorului (poziție și viteză);
- Generarea unei traiectorii optimale la bord utilizând ca date de intrare starea reală a demonstratorului;
- Efectuarea unei verificări a traiectoriei calculate la bord pentru asigurarea fezabilității soluției optimale obținute;
- Trimiterea traiectoriei optimale ca nouă referință pentru demonstrator.
- Scrierea într-un fișier .mat a tuturor parametrilor de interes la fiecare execuție a aplicației.

În timpul campaniei de testare HIL, timpul de execuție pentru aplicația dezvoltată este de 10 milisecunde, iar în timpul campaniei experimentale, timpul maxim de execuție a fost 70 milisecunde. Această diferență se datorează alocării dinamice a memoriei pentru înregistrarea datelor experimentale. Acest aspect se poate îmbunătăți în viitor, iar timpul total de execuție va fi redus semnificativ.

11 Concluzii

11.1 Contribuțiile tezei

Pentru atingerea obiectivului principal, au fost aduse următoarele contribuții:

- a) Completarea modelului matematic al problemei de optimizare convexă pentru aterizarea verticală a unui demonstrator prin introducerea unor constrângeri suplimentare pentru traiectorie;
- b) Propunerea unei metodologii inovative de testare integrată utilizând diferite medii de programare, Matlab/Simulink, CVXGEN, limbaj de programare C (capitolul 4);
- c) Dezvoltarea unui algoritm de calcul pentru rezolvarea problemei de optimizare convexă pentru aterizarea unui demonstrator reutilizabil utilizând mediul de programare Matlab, împreună cu modulul dedicat programării convexe, CVX, capitolul 6;
- d) Dezvoltarea unui algoritm de calcul pentru soluționarea problemei de optimizare convexă pentru aterizarea unui demonstrator reutilizabil și implementarea acestuia în generatorul de solver convex, CVXGEN, în vederea eficientizării/optimizării timpului de execuție al acestuia pentru a fi asigurată utilizarea pentru aplicații în timp real. Modalitatea de implementare și utilizare a codului generat este expusă în subcapitolele 7.1, 7.2 și 7.3 ceea ce reprezintă o premieră pe plan național și este, în același timp, un punct de pornire în dezvoltarea ulterioară a tehnologiei pentru optimizarea traiectoriei în timp real;
- e) Dezvoltarea unor funcții adiționale scrise în limbajul de programare C pentru utilizarea și integrarea codului generat de CVXGEN în Matlab/Simulink, aducându-se astfel contribuții importante în utilizarea CVXGEN, având în vedere că sunt foarte puține surse științifice disponibile;
- f) Dezvoltarea unui algoritm de simulare Monte Carlo pentru testarea și validarea funcționalității metodei de optimizare implementate, fiind aduse contribuții la demonstrarea fiabilității aplicației dezvoltate prin simulări offline;
- g) Dezvoltarea unui proces pentru generarea și compilarea automată a unei S-funcții care conține solverul generat de CVXGEN și funcțiile adiționale dezvoltate pentru integrarea acestui cod în Matlab/Simulink, prezentate în subcapitolul 8.1;
- h) Dezvoltarea unei aplicații, denumită SEP, pentru testarea întregii metodologii de optimizare convexă în timp real pentru problema studiată utilizând SpeedGoat (Real-Time Target Machine) ;
- i) Verificarea și validarea aplicației SEP în condiții de zbor captiv utilizând o infrastructură de cercetare și validare, contribuind astfel la creșterea maturității tehnologiei de optimizare online pentru traiectoria vehiculelor spațiale;
- j) Demonstrarea repetabilității și fiabilității utilizării algoritmului SEP dezvoltat pentru optimizarea traiectoriei de aterizare pentru demonstratoare reutilizabile, cu decolare și aterizare verticală prin realizarea unor zboruri experimentale în condiții similare.

11.2 Rezultate obținute

Activitățile de cercetare din cadrul lucrării de doctorat s-au concentrat pe dezvoltarea unei metodologii integrate și a unei aplicații în timp real, ce conține un solver pentru obținerea unei soluții optimale a problemei convexe pentru optimizarea traiectoriei de aterizare a unui demonstrator reutilizabil în retropropulsie cu aplicații spațiale. Pentru atingerea acestui obiectiv, a fost derivat modelul matematic și soluționat numeric cu ajutorul modului Matlab, CVX/SeDumi. Modelul matematic a fost implementat și în generatorul de cod C, CVXGEN, încapsulat într-o S-funcție și integrat într-un modul de simulare în șase grade de libertate pentru demonstratorul reutilizabil, DTV. Au fost realizate simulări numerice pentru o misiune nominală pentru testarea aplicației în timp real dezvoltate, împreună cu simulări Monte Carlo în prezența incertitudinilor. Aplicația dezvoltată a fost testată și validată în timp real utilizând o mașină de testare în timp real, SpeedGoat înainte de efectuarea testelor în zbor pentru a diminua riscurile. A fost realizată o campanie experimentală pentru testarea și validarea în zbor a aplicației denumite SEP, dezvoltate în cadrul acestei lucrări de doctorat.

În primul capitol a fost prezentat contextul actual al cercetărilor în domeniul optimizării convexe, dar și a algoritmilor de calcul pentru soluționarea problemei în timp real utilizând calculatoarele de bord ce echipează vehiculele spațiale. Capitolul al doilea prezintă câteva noțiuni introductive legate de tehnica optimizării convexe, iar capitolul al treilea prezintă deducerea matematică a problemei minimizării cantității de combustibil necesară aterizării verticale a demonstratoarelor spațiale și discretizarea acesteia.

Capitolul patru oferă o imagine de ansamblu asupra metodologiei propuse și urmărite în cadrul acestei lucrări de doctorat, iar în capitolul cinci este prezentat succint platforma reutilizabilă DTV ce este utilizată pentru testarea și validarea în zbor a aplicației în timp real dezvoltate.

În capitolul șase sunt prezentate rezultatele numerice pentru o misiune considerată nominală pentru validarea modelului matematic derivat pentru problema de interes. Rezultatele demonstrează că problema este una convexă, iar modulul CVX selectat este potrivit pentru rezolvarea acesteia offline. Capitolul șapte cuprinde detalii despre implementarea numerică în generatorul comercial de cod C, CVXGEN. Au fost realizate simulări Monte Carlo în prezența incertitudinilor condițiilor inițiale și rezultatele obținute confirmă robustețea aplicației în timp real ce conține solverul, dar și aplicabilitatea ei în probleme de optimizare convexă în timp real.

Aplicația în timp real dezvoltată este integrată în simulatorul de zbor al DTV, iar rezultatele prezentate în capitolul opt confirmă faptul că traiectoria optimizată de solver este una fezabilă din punct de vedere operațional ținând cont de constrângerilor demonstratorului DTV.

Rezultatele testării și validării în timp real sunt prezentate în capitolul al nouălea, iar rezultatele obținute în urma testării utilizând sistemul SpeedGoat confirmă faptul că această aplicație poate fi utilizată într-un zbor real pentru optimizarea traiectoriei demonstratorului. Timpul de calcul necesar obținerii unei traiectorii optimale în timpul zborului este de ordinul a 10 milisecunde, ceea ce o face potrivită pentru generarea traiectoriei optimale în timpul zborului.

Aplicația SEP dezvoltată a fost testată și validată în urma efectuării unor campanii experimentale de zbor, iar rezultatele sunt prezentate în capitolul 10. Două misiuni experimentale au fost considerate, iar în total cinci zboruri au fost efectuate la sediul INCAS din Măneciu. Rezultatele campaniei experimentale demonstrează fezabilitatea utilizării metodologiei propuse în

cadrul prezentei lucrări de doctorat. Aplicația SEP dezvoltată a fost utilizată cu succes în toate zborurile experimentale și a furnizat de fiecare dată, indiferent de condițiile inițiale ale demonstratorului, o traiectorie optimă într-un timp foarte scurt (de ordinul zecilor de milisecunde), ceea ce demonstrează potențialul utilizării acestei metodologii în misiuni spațiale.

11.3 Perspective de continuare și dezvoltare ulterioară

Această teză de doctorat s-a concentrat pe dezvoltarea unor algoritmi și a unei metodologii complete integrate în aplicația SEP pentru soluționarea în timp real a problemei de optimizare convexă pentru vehicule reutilizabile spațiale.

Problema matematică studiată este problema de optimizare convexă cu timp fix de zbor. Această ipoteză este valabilă în condițiile studiate în aceasta teză, însă pentru cazul în care se dorește studierea unui vehicul care reintră în atmosfera terestră, atunci este necesară și optimizarea la bord a timpului de zbor. O îmbunătățire ulterioară este studierea și implementarea problemei de aterizare cu timp liber de zbor și optimizarea acesteia la bordul vehiculului. O altă îmbunătățire ce poate fi adusă modelului matematic este adăugarea constrângerilor legate de atitudinea vehiculului, ceea ce conduce în final la o problemă matematică cu șase grade de libertate.

În cadrul acestei teze, pentru generarea automată de cod în limbajul de programare C, a fost utilizat un generator de solver comercial, CVXGEN. Având în vedere stadiul actual al cercetărilor în domeniu și direcțiile pe care le urmărește industria spațială, este de interes dezvoltarea unui generator automat de cod C dedicat care să soluționeze problema aterizării la punct fix, pentru că CVXGEN sau alternativele comerciale tratează probleme generale de optimizare convexă, ceea ce nu reprezintă întotdeauna o variantă optimă de implementare. O altă perspectivă viitoare de interes este dezvoltarea unui proces complet care să ofere utilizatorilor un mediu dedicat pentru integrarea și utilizarea ulterioară a acestui cod generat, împreună cu o interfață grafică flexibilă.

O altă direcție de cercetare și dezvoltare ulterioară este utilizarea aplicației SEP propuse în cadrul acestei lucrări într-un program complex pentru aselenizare și amartizare. Pentru aterizarea cu precizie ridicată este necesară utilizarea unei aplicații tip SEP concomitent cu programe software pentru procesarea de imagini la bord. Analizând în timpul reintrării imagini în timp real, algoritmul poate selecta un loc de aselenizare/amartizare în baza constrângerilor impuse și transmiterea coordonatelor locului țintă ales unei aplicații de tip SEP. Punctul țintă selectat de algoritmul de procesare a imaginilor poate fi trimis către aplicația SEP devenind astfel parte integrată în procesul de optimizare a traiectoriei de aterizare în timp real. Având în vedere cercetările actuale în domeniul spațial, este de așteptat ca scenariul descris mai sus să devină practică recurentă în viitoarele misiuni spațiale, iar eforturile de cercetare descrise în prezenta lucrare de doctorat reprezintă un punct solid de pornire, demonstrându-se fezabilitatea implementării și utilizării aplicației SEP în timp real.

11.4 Lista publicațiilor

În perioada elaborării lucrării de doctorat, au fost realizate și publicate în colaborare cu alți cercetători 12 articole științifice, ce pot fi accesate în reviste naționale și internaționale și un capitol de carte, publicat de Springer.

2017
T. V. Chelaru, A. I. Onel, T. P. Afilipoae, A. M. Neculăescu , “Mathematical Model for Microlauncher performances evaluation,” UPB Scientific Bulletin, Series D: Mechanical Engineering, vol. 79, nr. 4, pp. 49-66, 2017.
B. Oving, A. Kleef, B. Haemmerli, A. Boiron, Markus Kuhn, Ilja Müller, Ivaylo Petkov, A.M Neculaescu , T.P. Afilipoae and Marina Petrozzi, "Small Innovative Launcher for Europe: achievement of the H2020 project SMILE", 7th European Conference for Aeronautics and Space Sciences (Eucass); DOI: 10.13009/EUCASS20170600, 2017
M.V. Pricop, M.G. Cojocaru, C.I. Stoica, M.L. Niculescu, A. M. Neculăescu , A.G. Persinaru, M.Boscoianu, "Glide back booster wind tunnel model testing", AIP Conference Proceedings 1863, DOI: 10.1063/1.4992593, 2017.
2018
A. M. Neculăescu , T. P. Afilipoae, A. I. Onel, M. V. Pricop, I. Stroe, “Trajectory Optimization for Small Launchers Using a Genetic Algorithm Approach,” AIP Conference Proceedings, vol. 2046, 2018.
A. I. Onel, T. P. Afilipoae, A. M. Neculăescu , M. V. Pricop, “MDO approach for a two-stage microlauncher,” <i>INCAS Bulletin</i> , vol. 10, nr. 3, pp. 127-138, 2018.
T. P. Afilipoae, A. M. Neculăescu , A. I. Onel, M. V. Pricop, A. Marin, A. G. Perșinaru, A. M. Cișmilianu, I. C. Oncescu, A. Toader, A. Sirbi, S. Bennani, T. V. Chelaru, “Launch Vehicle - MDO in the development of a Microlauncher,” <i>Transportation Research Procedia</i> , vol. 29, pp. 1-11, 2018.
A. I. Onel, A. Stăvărescu, M. G. Cojocaru, M. V. Pricop, M. L. Niculescu, A. M. Neculăescu , T. P. Afilipoae, “Computation of the Hypersonic Heat Flux with Application to Small Launchers,” AIP Conference Proceedings, vol. 2046, 2018.
A. I. Onel, T. P. Afilipoae, A. M. Neculăescu , M. V. Pricop, “Drag coefficient modelling in the context of small launcher optimisation,” <i>INCAS Bulletin</i> , vol. 10, nr. 4, pp. 103-116, 2018.

2019
A. I. Onel, O. I. Popescu, A. M. Neculăescu , T. P. Afilipoae, T. V. Chelaru, “Liquid rocket engine performance assessment in the context of small launcher optimisation”, <i>INCAS Bulletin</i> , vol. 11, nr. 3, pp. 135-145, 2019.
A. M. Neculăescu , A. Marin, A. Toader, A. G. Persinaru, A. M. Cismilianu, M. Tudose, Camelia-Elena Munteanu, I. Popescu, H. Strauch, S. Dussy, “System Identification and Testing for a VTVL vehicle”, EUCASS 2019, DOI: 10.13009/EUCASS-925, 2019.
2021 (capitol de carte)
A. M. Cismilianu, I. Chirita, A. G. Persinaru, A. Marin, C. E. Munteanu, A. M. Neculăescu , C. Dragoman, „Re-entry vehicle structural optimization for mass minimization”, publicat în cartea „Innovations in Mechanical Engineering”, DOI: 10.1007/978-3-030-79165-0_9, 2021.
2022
T. P. Afilipoae, A. M. Neculăescu , P. Simplicio, S. Bennani, H. Strauch, „Control strategies comparison and performance evaluation for a reusable VTVL platform based on a rocket engine”, EUCAS 2022, DOI: 10.13009/EUCASS2022-4480
2023
A.M. Neculăescu , A.I. Onel, C. B. Briceag, A. Toader, “Online convex optimization for a reusable vertical take-off and vertical landing demonstrator”, în curs de publicare.

Bibliografie (selectivă)

- [1] Musk, E., „Space X, Falcon 9 launcher,” <https://www.spacex.com/vehicles/falcon-9/>.
- [2] J. Foust, „Blue Origin flies six on first New Shepard suborbital flight of 2022,” <https://spacenews.com/new-shepard-completes-fifth-crewed-suborbital-flight/>, 2022.
- [3] A. Jones, „Deep Blue Aerospace completes kilometer-level rocket launch and landing test,” <https://spacenews.com/deep-blue-aerospace-completes-kilometer-level-rocket-launch-and-landing-test/>, 2022.
- [4] ESA Communication team, „ESA plans demonstration of a reusable rocket stage,” https://www.esa.int/Enabling_Support/Space_Engineering_Technology/ESA_plans_demonstration_of_a_reusable_rocket_stage, 2021.
- [5] ESA GSTP Annual Report, „GSTP Annual Report 2020,” <https://indd.adobe.com/view/fb37fecf-5ab6-4cd1-868d-a0679e7432c7>, 2020.
- [6] H. W. Jones, „The Recent Large Reduction in Space Launch Cost,” vol. 48th International Conference on Environmental Systems, 2018.
- [7] D. Malyuta, T. P. Reynolds et al, „Convex optimization for Trajectory Generation,” <https://arxiv.org/abs/2106.09125>, 2021.
- [8] Mordecai Avriel, *Nonlinear Programming: Analysis and Methods*, Dover Publications, 2003.
- [9] Mokhtar S. Bazaraa, Hanif D. Sherali, C.M. Shetty, *Nonlinear programming, Theory and algorithms*, Third Edition, John Wiley & Sons, 2006.
- [10] Andrzej Ruszczyński, *Nonlinear Optimization*, Princeton University Press, 2006.
- [11] Behçet Açıkmeşe, Lars Blackmore, „Lossless convexification of a class of optimal control problems with non-convex control constraints,” *Automatica*, vol. 47, nr. 2, pp. 341-347, 2011.
- [12] Lars Blackmore, Behçet Açıkmeşe, John M. Carson III, „Lossless convexification of control constraints for a class of nonlinear optimal control problems,” *Systems & Control Letters*, vol. 61, nr. 8, pp. 863-870, 2012.
- [13] Behçet Açıkmeşe, John M. Carson III and Lars Blackmore, „Lossless Convexification of Nonconvex Control Bound and Pointing Constraints of the Soft Landing Optimal Control Problem,” *IEEE Transactions on Control Systems Technology*, vol. 21, nr. 6, 2013.
- [14] Quoc Tran Dinh, „Sequential Convex Programming and Decomposition Approaches for Nonlinear Optimization,” Arenberg Doctoral School of Science, Engineering & Technology, 2012.

- [15] John Duch, „Sequential Convex Programming,” Stanford University, 2018.
- [16] Taylor P. Reynolds, Mehran Mesbahi, „The Crawling Phenomenon in Sequential Convex Programming,” *American Control Conference*, p. 2020.
- [17] M. Szmuk, T. P. Reynolds, B. Acikmese, „Successive convexification for real-time 6-DoF powered descent guidance with state-triggered constraints,” *AIAA Journal of Guidance, Control, and Dynamics*, 2018.
- [18] Meikang Qiu, Jiayin Li, Real-Time Embedded Systems: Optimization, Synthesis, and Networking, 1st Edition, CRC Press, 2011.
- [19] Xue Liu, Xi Chen, Fanxin Kong, Utilization Control and Optimization of Real-Time Embedded Systems, now Publishers Inc., 2015.
- [20] Fateh Boutekkouk, Bout Fat, „Real-Time Embedded Systems Scheduling Optimization: A Review on Bio-Inspired Approaches,” *International Journal of Applied Evolutionary Computation*, vol. 12, nr. 1, 2021.
- [21] X. Liu, „Fuel-optimal rocket landing with aerodynamic controls,” *Journal of Guidance, Control and Dynamics*, vol. 42, pp. 65-77, 2019.
- [22] Yuanqi Mao, Behcet Acikmese, Pierre-Loïc Garoche, Alexandre Chapoutot, „Successive Convexification for Optimal Control with Signal Temporal Logic Specifications,” în *25th ACM International Conference on Hybrid Systems: Computation and Control (HSCC '22)*, Milan, Italy, 2022.
- [23] Mattingley, J. E., „CODE GENERATION FOR EMBEDDED CONVEX OPTIMIZATION,” https://stanford.edu/~boyd/papers/pdf/code_gen_impl.pdf, 2011.
- [24] B. Açıkmeşe, L. Blackmore, D. P. Scharf, A. Wolf, „Enhancements on the Convex Programming Based Powered,” în *AIAA/AAS Astrodynamics Specialist Conference, Honolulu, Hawaii*, 2008.
- [25] B. Açıkmeşe, S. R. Ploen, „Convex Programming Approach to Powered Descent Guidance,” *Journal of Guidance, Control, and Dynamics*, vol. 30, nr. 5, pp. 1353-1366, 2007.
- [26] ESA, INCAS, „Demonstrator masters flight sequences for reusable rocket stages,” https://www.esa.int/Enabling_Support/Space_Transportation/Future_space_transportation/Demonstrator_masters_flight_sequences_for_reusable_rocket_stages, 2020.
- [27] J. Mattingley, S. Boyd, „CVXGEN: Code Generation for Convex Optimization,” <https://cvxgen.com/docs/index.html>, 2022.
- [28] I. E. Mooij, „The motion of a vehicle in a planetary atmosphere,” TU Delft, Delft, 1994.
- [29] Speedgoat Products and Services, „<https://www.speedgoat.com/products-services>”.