



Domeniul Ingineria Sistemelor

TEZĂ DE DOCTORAT

- REZUMAT -

**Dezvoltarea și Testarea Unui Sistem Multi-agent de Căutare și
Localizare Folosind Drone**

Student-doctorand:
Daniel-Dumitru TIMIȘ

Conducător științific:
Prof. Dr. Ing. Eva H. DULF

Comisia de evaluare a tezei de doctorat:

Președinte: Prof. Dr. Ing. **Vlad MUREȘAN** - Universitatea Tehnică din Cluj-Napoca;
Conducător științific: Prof. Dr. Ing. **Eva H. DULF** - Universitatea Tehnică din Cluj-Napoca;

Referenți:

- Prof. Dr. Ing. **Radu-E. PRECUP** - Universitatea Politehnică din Timișoara;
- Prof. Dr. Ing. **Monica LEBA** - Universitatea din Petroșani;
- Prof. Dr. Ing. **Liviu-C. MICLEA** - Universitatea Tehnică din Cluj-Napoca.

- Cluj-Napoca -
2024

CUPRINS

1.	INTRODUCERE	3
2.	STADIUL ACTUAL AL CUNOAȘTERII.....	4
2.1.	<i>Vehicule zburătoare fără pilot – UAV.....</i>	<i>5</i>
2.2.	<i>Aplicații cu un singur UAV.....</i>	<i>7</i>
2.3.	<i>Roiuri inteligente de UAV-uri</i>	<i>7</i>
2.4.	<i>Concluzia studiului literaturii</i>	<i>8</i>
3.	Ipoteza de lucru/obiective	9
4.	Metodologie generală.....	10
5.	Controlul clasic al dronei.....	10
5.1.	<i>Primul studiu comparativ.....</i>	<i>11</i>
5.2.	<i>Al doilea studiu comparativ.....</i>	<i>12</i>
5.3.	<i>Concluzii.....</i>	<i>14</i>
6.	Controlul de la distanță al dronei	14
6.1.	<i>Evaluare control dronă.....</i>	<i>15</i>
6.2.	<i>Conceperea aplicației de control.....</i>	<i>16</i>
6.3.	<i>Implementarea serverului și a aplicației de operare.....</i>	<i>17</i>
6.4.	<i>Concluzii.....</i>	<i>17</i>
7.	Procesare de imagine.....	18
8.	Sistem distribuit de căutare și localizare	19
8.1.	<i>Algoritmul PSO – fundament teoretic</i>	<i>20</i>
8.2.	<i>Validare PSO pe sistemul real.....</i>	<i>21</i>
8.3.	<i>Concluzii.....</i>	<i>22</i>
9.	Discuții	23
10.	Concluzii	24
	REFERINȚE	26
	LISTĂ DE PUBLICAȚII	27

1. INTRODUCERE

Se preconizează că roboții vor deveni indispensabili în modul rapid în care evoluează societatea actuală. În ultimii ani, creșterea capacităților și performanțelor hardware, dezvoltarea de software complexe și necesitatea acestor entități în aplicațiile de zi cu zi au determinat o evoluție rapidă a roboților autonomi. Cu toate că ne referim la roboți software sau la roboți fizici, aceștia sunt unul dintre principalii factori care accelerează evoluția societății umane.

Stadiul actual al tehnologiei a permis cercetătorilor să proiecteze și să efectueze experimente pe diferite tipuri de vehicule fără pilot. Unul dintre acestea sunt vehiculele aeriene fără pilot (UAV - Unmanned Aerial Vehicles în engleză) sau cunoscute în mod obișnuit ca „drone”. Având o mare flexibilitate, o desfășurare ușoară, o manevrabilitate excelentă și cu numeroase soluții comerciale existente, acestea sunt utilizate într-o gamă largă de aplicații. Printre aceste aplicații putem aminti: gestionarea dezastrelor, căutarea și salvarea, construcțiile și industria civilă, monitorizarea traficului, agricultura și multe altele. În plus, potențialul de extindere a utilizării dronelor în alte domenii este posibil datorită flexibilității acestora și oportunităților de piață.

Motivația acestui studiu a constat în proiectarea și dezvoltarea unui sistem de drone capabile să colaboreze, în vederea identificării și localizării persoanelor aflate în situații critice. Anual în toată lumea numeroase persoane sunt date dispărută în urma unor calamități naturale sau dezastre provocate de om. Un astfel de sistem, capabil să fie desfășurat oriunde și în orice condiții, pentru a gestiona dezastrul, poate salva numeroase vieți și poate reduce riscul de vătămare al salvatorilor. Practic, dronelor li se asignează o misiune de căutare într-o anumită zonă bine determinată geografic, iar acestea folosind senzorii cu care sunt echipate, sunt capabile să localizeze persoane sau grupuri de persoane. Acestea pot fi folosite și pentru a transporta o serie de necesități pentru primul ajutor, iar pe baza informațiilor despre mediu culese, pot ajuta la conceperea unui plan de salvare care să minimizeze orice risc la care ar putea fi supuse echipele de salvare.

Această teză își propune în primă fază să studieze și să identifice cum poate fi construit un astfel de sistem, care sunt soluțiile, problemele și direcțiile actuale de dezvoltare al acestui domeniu. Pe baza informațiilor culese, se propune dezvoltarea și testarea într-un mediu controlat a unui astfel de sistem de căutare și localizare.

Prima parte a acestei teze (capitolele 1 și 2) realizează un studiu detaliat al literaturii de specialitate, în care mai mult de 100 de lucrări științifice actuale, în domeniul dronelor sunt evaluate. Sunt prezentate soluțiile și metodele actuale folosite pentru integrarea UAV-urilor în diverse aplicații cum ar fi: căutare și salvare, monitorizarea construcțiilor, agricultură, monitorizarea traficului, transport și livrare. Au fost evaluate metodele și rezultatele pe care literatura le propune ca soluționare a aceste integrări și tendințele viitoare în această direcție. Pe lângă studiul aplicațiilor, au fost evaluate și sistemele multi-agent de drone (capitolul 3), efort care s-a concentrat pe a înțelege cum un astfel de sistem este conceput și care sunt cele mai importante componente a acestora. Plecând de la rezultatele analizei de mai sus au fost formulate principiile pe care ar trebui să se bazeze dezvoltarea unei soluții care să satisfacă scopul acestei teze.

A doua parte a lucrării este împărțită în 4 studii de caz aplicate pe tematica dronelor, în vederea evaluării performanțelor acestora și a identificării celei mai bune modalități de integrare a acestora într-o aplicație reală.

Primul studiu (capitolul 7) s-a concentrat pe structuri de reglare a stabilității zborului unui quadcopter. Au fost evaluate o serie de tehnici moderne, cum ar fi regulatoarele PID fracționale și structurile adaptive de control. **Al doilea studiu (capitolul 8)** s-a concentrat pe

dezvoltarea și testarea infrastructurii sistemului distribuit de control, la care o serie de drone trebuie să se conecteze pentru a colabora. Au fost definite și concepute aplicațiile software necesare îndeplinirii acestui obiectiv și a fost evaluat modul în care o dronă poate să fie controlată cu ajutorul acestei structuri.

Al treilea studiu (capitolul 9) prezintă modul în care folosind procesarea de imagine, se pot identifica anumite obiecte de anumite forme și culori. Cu ajutorul acestei identificări dronele pot zbura într-o zonă de căutare predefinită, cu scopul de a localiza ținta. **Ultimul studiu (capitolul 10)** se concentrează pe conceperea unui sistem distribuit de drone, care cu ajutorul algoritmului de optim Particle Swarm Optimization (PSO) va calcula poziția viitoare a tuturor dronelor, cu scopul de a identifica obiectul țintă definit anterior. Studiul analizează rezultatele simulate și experimentale ale acestui sistem.

Capitolele 11 și 12 prezintă o serie de discuții generale a rezultatelor obținute, concluziile teze și contribuția personală adusă acestui domeniu.

2. STADIUL ACTUAL AL CUNOAȘTERII

De-a lungul istoriei, omenirea s-a confruntat cu o mulțime de dezastre naturale sau provocate de om. În astfel de circumstanțe, este nevoie de un răspuns rapid și bine coordonat pentru a salva cât mai multe vieți după ce un astfel de eveniment lovește o zonă populată. În acest caz, conceptul de căutare și salvare sau Search and Rescue (SAR) în limba engleză s-a născut și a început să evolueze. SAR este un concept în care o echipă sau echipe organizate de căutători sau salvatori sunt trimise într-o zonă în care a avut loc un eveniment periculos și în care unul sau mai mulți oameni au fost puși în pericol din această cauză. Aceste echipe încearcă să identifice și să localizeze victimele și își depun eforturile, experiența și cunoștințele pentru a le aduce înapoi într-un loc sigur.

Chiar la începutul istoriei, operațiunile de căutare și salvare erau dedicate exclusiv expedițiilor maritime. Conceptul este întâlnit prima dată în istorie, în timpul antichității în Grecia sau Roma antică, unde comunitățile situate pe coastă erau adesea implicate în acțiuni de salvare a marinarilor naufragați în apropierea țărmului. Primul exemplu documentat de acțiune SAR a avut loc în 1656, când o navă comercială olandeză „Vergulde Draeck” a naufragiat lângă coasta australiană. Trei echipe de salvare separate s-au aventurat să salveze marinarii supraviețuitori, însă fără succes [1]. Prima organizație din istorie cu scopul principal de a salva vieți a fost înființată de Sir William Hillary în 1824, sub numele de „Royal National Lifeboat Institution” (RNLI). Sir Hillary, care locuia pe Insula Man, a salvat multe vieți din naufragii, iar efortul său a pus bazele primei organizații dedicate salvării marinarilor naufragați [2]. Începând cu secolul al XX-lea și mai precis în timpul celui de-al Doilea Război Mondial, odată cu dezvoltarea aviației și creșterea frecvenței accidentelor aeriene în zone îndepărtate, a crescut nevoia înființării unor grupuri specializate care să întreprindă acțiuni de salvare. Pe această bază, lucrurile au evoluat foarte rapid, iar până la sfârșitul secolului al XX-lea existau deja în lume echipe dedicate de profesioniști al căror principal domeniu de activitate era căutarea și salvarea. S-au dezvoltat domenii emergente, cum ar fi salvarea în caz de inundații, incendii, salvare montană sau alte acțiuni de salvare în urma dezastrelor naturale sau provocate de om. De exemplu, în România, organizația de salvare montană (Salvamont) a fost înființată în anii 1960 printr-o serie de decrete guvernamentale și inițiative voluntare. Aceasta reprezentând la acel moment un răspuns la numărul tot mai mare de persoane care se angajează în activități în aer liber, cum ar fi drumeții, schi și alpinism, în special în Munții Carpați [3]. Pentru a sprijini efortul de dezvoltare al acestui sector și pentru a-l standardiza la nivel global, în cadrul Organizației Națiunilor Unite, în 1991 a fost înființat grupul consultativ internațional pentru căutare și salvare sau „International Search and Rescue Advisory Group” (INSARAG) în limba engleză. Scopul acestei organizații este de a

standardiza operațiunile SAR transfrontaliere, asigurând un răspuns coordonat la dezastrele pe scară largă.

Căutarea și salvarea au evoluat de la eforturi comunităților de voluntari, la un domeniu sofisticat, bazat pe tehnologie, esențială pentru gestionarea dezastrelor. Operațiunile SAR moderne sunt caracterizate de tehnologii avansate, precum GPS, drone și inteligență artificială, și sunt sprijinite de organizații structurate și cadre de cooperare internațională. Pe măsură ce frecvența și complexitatea dezastrelor continuă să crească, rolul SAR va continua să fie îmbunătățit pentru a spori capacitatea de a salva vieți.

Componenta integrării dronelor în acest domeniu, este una intens studiată, care se bucură de soluții reale și aplicate nevoilor SAR. În acest sens, această lucrare a avut ca prim obiectiv, realizarea unui studiu detaliat al stadiului actual al literaturii, pentru a înțelege cum sunt dronele folosite și integrate, nu doar în domeniul SAR, dar și în alte industrii. Studiul literaturii s-a bazat pe 3 direcții: clasificare UAV-urilor, aplicații ale dronelor în diferite industrii de interes și pe un concept mai actual cel al sistemelor multi-agent de UAV-uri sau „Swarm of drones” în limba engleză.

2.1. Vehicule zburătoare fără pilot – UAV

Pentru a oferi o imagine de ansamblu completă a subiectului UAV-urilor (termen provenit din engleză „Unmanned Aerial Vehicle”) și înainte de a intra în detalii cu privire la aplicații și funcționalitate, trebuie făcută o clasificare a obiectelor zburătoare. Termenul UAV este prea generic, poate lua diferite forme și poate avea diferite soluții de implementare. Acest lucru este posibil mai ales datorită interesului ridicat din ultimii ani, atât din partea sectorului privat, cât și a celui științific.

Pot fi definite mai multe criterii de clasificare, cum ar fi: domeniul de aplicare, forma, dimensiunea, raza de acțiune, aerodinamica, manevrabilitatea și structura. Cel mai comun UAV sau dronă este quadcopterul sau UAV-urile multirotoare în general, datorită manevrabilității lor și a ușurinței de a fi integrate în diferite aplicații [4].

2.1.1. Clase de UAV-uri

Principalul criteriu prin care dronele pot fi clasate se referă la altitudinea la care acestea pot zbura. În acest sens sunt definite 3 clase de UAV-uri [5] [6]:

- Joasă altitudine (Low Altitude UAVs - LAU)
- Înaltă altitudine (High Altitude UAVs - HAU)
- Sateliți

Prima clasă se referă la UAV-urile care au o rază de acțiune limitată și funcționează la altitudine mică. Cel mai comun tip este cel al UAV-urilor multirotoare. Această clasă ar trebui să fie ușor și rapid de operat, pe distanțe scurte și medii. A doua clasă operează la altitudini mai mari și are nevoie de o rută planificată în prealabil pentru a funcționa. Acestea au o rază de acțiune mai mare, și adesea iau forma unor avioane fără echipaj. A treia clasă operează în spațiu, fiind de fapt sateliții comerciali sau militari operaționali. Unul dintre scopurile lor fiind supravegherea și acoperirea comunicațiilor pentru LAU și HAU.

În momentul de față nu există un sistem care să integreze cele trei clase într-o aplicație reală, dar se fac studii și cercetări în această direcție.

2.1.2. Aerodinamica

Merită menționată clasificarea UAV-urilor în funcție de forma sau aerodinamica lor. Acestea sunt disponibile în multe forme și variante. Principalele criterii de evaluare sunt clasificarea acestora în funcție de capacitatea lor de aterizare sau de decolare. Astfel, pot fi definite 2 tipuri de vehicule zburătoare: decolare și aterizare orizontală (HTOL) sau decolare și aterizare verticală (VTOL) [7].

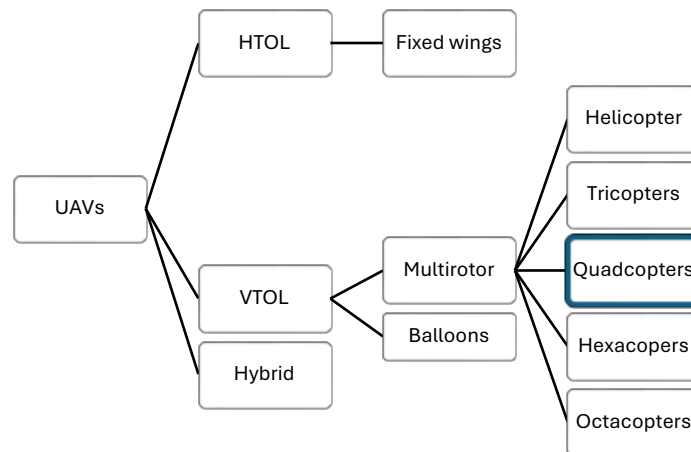


Figura 1: Clasificarea dronelor în funcție de aerodinamică

În Figura 1 este prezentată structura de clasificare a UAV-urilor pe baza aerodinamicii lor. Categoria HTOL este compusă în principal din aeronave cu aripă fixă sau dintr-un avion clasic. Pe aceste tipuri de aeronave pot fi instalate sisteme de control la distanță pentru a fi transformate în UAV-uri.

A doua categorie prezentată (VTOL) este compusă în principal din sisteme multirotor, capabile de decolare verticală. Cel mai răspândit UAV sau drone multirotor din zilele noastre este quadcopterul, datorită robusteții și capacității sale de a fi ușor de operat. La această secțiune pot fi adăugate și baloanele. Atât baloanele meteorologice, cât și cele de supraveghere sunt autonome și capabile să decoleze și să aterizeze pe verticală. Acest tip de UAV-uri pot fi utilizate ca stații terminale sau terminale de comunicații în anumite aplicații, deoarece pot rămâne în aer atâta timp cât este necesar și pot transporta echipamente mai grele, în comparație cu multirotoarele, cu un consum redus de energie. A treia categorie este cea hibridă. Acest tip de aeronavă este capabil să decoleze vertical sau orizontal, în funcție de necesitățile aplicației.

2.1.3. Dimensiunea și distanța de operare

Dimensiunea și raza de acțiune sunt două caracteristici importante pe baza cărora pot fi clasificate UAV-urile. Ambele vor defini tipul de aplicație și complexitatea acesteia. Unele aplicații vor necesita drone mai mici cu rază de acțiune limitată, în timp ce alte aplicații vor necesita drone mai grele cu rază de acțiune mare. Autorii din [7] și [8] clasifică UAV-urile în funcție de greutatea și dimensiunea lor. Clasificarea prezentată în aceste 2 articole este aplicată pentru întregul spectru al UAV-urilor.

Tabel 1 clasifică quadcopterele destinate sectorului civil, pe baza greutății și a razei de acțiune. Acest tabel poate fi utilizat ca referință la construirea unei aplicații dedicate unui anumit sector. De asemenea, legislația trebuie confruntată și respectată pentru a dezvolta orice tip de sisteme UAV aplicate pentru o nevoie reală.

În Europa și în SUA, categoria Nano este singura din tabelul de mai jos pentru care nu este necesară certificarea, înregistrarea sau instruirea specială pentru a opera un astfel de aparat, conform Agenției Uniunii Europene pentru Siguranța Aviației (EASA) [9] și Federal Aviation Administration (FAA) [10].

Categorie	Masă	Rază de acțiune	Tip
Nano	< 250 g	5 km	Aripă fixă, multirotor
Micro	< 2 kg	25 km	Aripă fixă, multirotor
Ușoare cu rază mică	< 20 kg	40 km	Aripă fixă, multirotor
Ușoare cu rază medie	< 50 kg	100 km	Aripă fixă, multirotor
Mici sau UAV-uri medii	< 150 kg	150 km	Aripă fixă

Tabel 1: Clasificarea UAV-urilor în funcție de masă și raza de acțiune

2.2. Aplicații cu un singur UAV

Această secțiune realizează o analiză detaliată a literaturii pentru a identifica în ce industrii și cum sunt integrate UAV-urile. Au fost analizate pentru început numeroase lucrări care evaluează aplicațiile în care pot fi integrate dronele. Pornind de la această evaluare, au fost identificate cele mai importante domenii care merită o analiză detaliată pentru a înțelege cum pot fi UAV-urile integrate și care sunt rezultatele acestei integrări.

În ultimii 15 ani, cercetătorii s-au concentrat pe dezvoltarea unei drone stabile, capabilă de zbor autonom. În [11] autorii au făcut o trecere în revistă a celor mai relevante cercetări în domeniul UAV-urilor. Lucrarea prezintă faptul că 31% din articolele examinate au avut ca subiect principal dezvoltarea hardware pentru quadcoptere sau drone și 28% controlul și modelarea. Hardware-ul și strategia de control reprezintă prima necesitate în construirea unei aplicații care integrează UAV-uri. Aceste două subiecte reprezintă pilonii principali dezvoltați de cercetători pentru a obține o dronă robustă capabilă să opereze cu succes în diverse misiuni de zbor. Această primă necesitate este evidențiată de 59% din efortul principal dedicat acestui domeniu. Restul de 41% din lucrări au avut în vedere scopuri mai înalte pentru UAV-uri, cum ar fi 17% - planificarea traseului, 12% - cartografiere și inspecții, 7% - teleoperare și 5% evitarea coliziunilor.

Scopul acestei secțiuni este de a extinde cercetarea și de a se concentra doar pe cele mai relevante și importante aplicații. După cum se poate observa, **căutarea și salvarea (sau gestionarea dezastrelor), construcțiile și inspecția infrastructurii, agricultura de precizie, transportul și livrarea de bunuri, monitorizarea traficului în timp real și supravegherea** sunt aplicațiile în care interesul cercetătorilor și crearea unei soluții reale sunt mai mari. Prin urmare, această secțiune se va concentra pe subiectele prezentate mai sus, prin efectuarea unui studiu amplu și detaliat asupra articolelor în care principalul obiectiv de cercetare este integrarea UAV-urilor în unul din domeniile enumerate mai sus.

Acest capitol prezintă importanța dronelor în sectoare-cheie ale economiei și societății actuale. Pe baza acestei analize, se poate concluziona că aceste sisteme sunt extrem de importante, dar soluțiile sunt încă în curs de dezvoltare, ceea ce reprezintă o oportunitate reală pentru continuarea cercetării. Această secțiune s-a bazat pe aplicații care necesită utilizarea unei singure drone, însă domeniile menționate mai sus pot fi îmbunătățite prin integrarea unui sistem multi-agent de drone (sau a unui roi de drone), capabil să interacționeze și să lucreze împreună pentru a rezolva sarcini și a îmbunătăți astfel sectoare-cheie ale economiei.

2.3. Roiuri inteligente de UAV-uri

Pentru această secțiune s-a efectuat o analiză extinsă a literaturii de specialitate pe tema roiurilor de drone sau „Swarms of Drones” în engleză. Cercetarea urmărește să înțeleagă modul în care mai multe UAV-uri pot fi integrate în mod eficient într-o structură colaborativă (sau roi) pentru a crește performanța și capacitățile de utilizare a unui singur agent în diferite aplicații. În introducere a fost menționat faptul că acest studiu a reprezentat o cercetare preliminară, pentru această teză de doctorat, în urma căreia se dorește implementarea unui sistem colaborativ cu mai multe UAV-uri pentru o aplicație de căutare și salvare, capabilă să identifice persoane în pericol.

Primul pas a fost analiza celor mai relevante și actualizate lucrări de cercetare pe tema UAV-urilor multi-agent sau a roiurilor de UAV-uri. În urma acestei lecturi, prezentul studiu își propune să identifice modul în care sunt structurate astfel de sisteme, care sunt cele mai importante componente și care sunt cele mai relevante aspecte care trebuie examinate în continuare.

Pe baza analizei efectuate ca parte al acestui capitol, se poate observa un model în ceea ce privește aspectele sau componentele cheie ale unui roi de UAV-uri. Toate articolele au menționat importanța comunicării, care joacă un rol fundamental și constituie un strat de bază pentru un sistem robust și eficient. În al doilea rând, controlul formației și importanța ca roiurile să mențină forma, să evite coliziunile și obstacolele din mediu și să planifice traiectoria de zbor a întregii structuri. În cele din urmă, algoritmul de control al roiului, care este centrul roiului, este responsabil de asigurarea faptului că întreaga structură acționează ca o entitate unificată capabilă să atingă obiectivele comune.

Aceasta secțiune trece în revistă stadiul actual al tehnologiei pe baza articolelor care discută una dintre următoarele idei cheie: **comunicarea** în cadrul roiului de UAV-uri, **colaborarea** dronelor (gestionarea formației, gestionarea sarcinilor și planificarea traiectoriei) și algoritmi de **control al roiurilor**.

2.4. Concluzia studiului literaturii

În viitoarea eră a inteligenței artificiale și a aplicațiilor de operare la distanță, utilizarea vehiculelor aeriene fără pilot este aproape inevitabilă. Piața UAV se află într-o continuă expansiune și se preconizează că va crește considerabil în următorul deceniu din cauza nevoii tot mai mari de lucru de la distanță cu ajutorul dronelor și a cercetării și dezvoltării continue care se investește în acest sector. Prin efectuarea unei analize extinse a primelor cinci sectoare sau industrii în care aceste sisteme pot fi integrate sau aplicate, această lucrare a identificat starea actuală și soluțiile actuale ale aplicațiilor UAV. În toate cele cinci industrii care au fost analizate: Căutare și salvare, Construcții, Agricultură, Transporturi și Monitorizarea traficului, utilizarea dronelor contribuie la îmbunătățirea acestora.

Cea mai vitală componentă sau aspect care influențează calitatea unei astfel de aplicații este capacitatea acesteia de a identifica mediul înconjurător, utilizând procesarea imaginilor sau alți senzori, cum ar fi: LiDAR, camere spectrale sau termice. Pentru procesarea imaginilor există un set mare de algoritmi precum: TensorFlow, YoLo sau Meshlab, combinate cu rețele neurale de convoluție (CNN), s-au dovedit a fi instrumente fiabile pentru identificarea mediului înconjurător sau a altor forme-cheie (oameni, clădiri, infrastructură sau obstacole). Găsirea unei modalități de a integra informațiile prelucrate de imagini cu datele colectate de la ceilalți senzori sensibili menționați mai sus este esențială pentru dezvoltarea unui UAV autonom capabil să navigheze într-un mediu real. Cercetările actuale se concentrează în continuare pe realizarea acestei etape importante și au fost realizate progrese substanțiale în această direcție.

În plus, sistemele UAV multi-agent, în care un roi de drone poate fi desfășurat pentru a atinge un obiectiv comun, este în continuă creștere. Aceste soluții pot fi integrate în oricare dintre industriile menționate anterior și pot îmbunătăți capacitățile unei aplicații cu o singură dronă. Pentru a realiza un astfel de sistem, este important să se asigure trei componente vitale: Comunicarea, controlul formației de zbor și controlul roiului. Comunicarea între componentele roiului este vitală și ar trebui să fie cât mai stabilă posibil pentru a asigura colaborarea generală a agenților. Pe lângă aceasta, comunicarea cu baza este, de asemenea, importantă, dar nu vitală, deoarece acest sistem ar trebui să funcționeze autonom. Tehnologia 5G actuală este un instrument fiabil și accesibil care contribuie la îmbunătățirea acestei funcționalități. Componenta de control al formațiunii reprezintă algoritmul de coordonare utilizat pentru a se asigura că toți agenții păstrează forma dorită a roiului, evitând coliziunea între fiecare agent cu obstacolele din jur. Controlul roiului este responsabil pentru a se asigura că obiectivul comun este atins, prin implementarea unor algoritmi avansați, cum ar fi: Particle Swarm Optimization, Algoritmi genetici, Ant Colony Optimization, Model Predictive Control și multe altele.

În concluzie, acest studiu evidențiază stadiul actual al aplicațiilor UAV în general, prezentând tehnologiile actuale, algoritmii, rezultatele și tendințele viitoare ale acestui subiect de cercetare. Soluțiile concrete au arătat că, din punct de vedere tehnic, aceste sisteme sunt aproape capabile de a fi implementate în aplicații reale. Orizontul de timp pentru ca acest lucru să se întâmple depinde, de asemenea, de modul și momentul în care va fi reglementată legislația în acest domeniu.

3. Ipoteza de lucru/obiective

Scopul acestei cercetări este conceperea unui sistem de căutare și salvare bazat pe un roi de 3 sau mai multe drone, capabile să localizeze persoane pierdute sau/și aflate în situație critică. Din studiul bibliografic detaliat realizat în capitolele de mai sus, se poate constata că din ce în ce mai multe industrii găsesc aplicabilitate UAV-urilor, iar numeroase soluții au fost deja identificate în această direcție. În același timp interesul pentru conceperea unui sistem multi-agent, format din mai multe drone sau roi inteligent de UAV-uri este în creștere.

Componenta principală a acestei lucrări o reprezintă sistemul multi-agent sau roiul de drone. În studiul bibliografic s-a identificat faptul că un astfel de sistem trebuie să conțină 3 elemente fundamentare: comunicarea, controlul formației și controlul întregului roi. Plecând de la aceste 3 componente fundamentale studiul propune următoarea soluționare:

1. **Comunicarea:** printr-o rețea locală Wi-Fi cu o acoperire limitată ariei folosită pentru validarea rezultatelor. Această rețea izolată va asigura eliminarea întârzierilor.
2. **Controlul formației:** nu se impune o formație propriu zisă (sferă sau triunghi) ci mai degrabă se dorește proiectarea unei structuri la nivel de agent, care să împiedice coliziunea cu un alt agent, prin impunerea unui „câmp” de forță, care să nu permită ca 2 drone să se afle în aceeași poziție la același moment de timp. Pentru aceasta, dronele vor ajusta mișcarea lor următoare în funcție de poziția actuală a celorlalți agenți.
3. **Controlul roiului:** este propus algoritmul PSO pentru a poziționa fiecare agent din roi. Pe baza poziției actuale a fiecărei drone și a locului țintă, algoritmul va calcula și va transmite către fiecare dronă poziția următoare astfel încât întreg sistemul să-și atingă obiectivul fixat. În plus, componenta de control preia toate informațiile (poziție, imagine video, rezultat funcție de cost) și le stochează pentru analiză. Această componentă reprezintă și interfața dintre sistem și operator. Acesta din urmă poate oricând să solicite controlul manual de la distanță al oricărei drone din roi.

Pornind de la scopul acestei cercetări și de la specificațiile sistemului multi-agent sunt identificate următoarele componente și direcții de dezvoltare:

1. **Control UAV:** realizarea unui sistem de control pentru un singur UAV, care va fi replicat pentru toți agenții sistemului distribuit. Mai multe metode de control vor fi examinate în vederea identificării celei mai robuste metode de control. Pe baza rezultatelor experimentale se va realiza sau alege hardware-ul și software-ul potrivit.
2. **Server de control al roiului:** implementarea unui server cu care să se conecteze fiecare agent din sistem, folosind o rețea de internet locală. Serverul trebuie să fie capabil să preia informațiile (video și de poziționare) de la fiecare agent și să le stocheze. Prin intermediul acestei componente, operatorul uman poate interacționa cu tot sistemul sau cu un singur element al acestuia, prin control manual.
3. **Identificare video:** dezvoltarea unei metode de identificare video a unei anumite ținte predefinite. Conceperea și validarea programului de procesare de imagine pentru identificare șablonului definit (o figură geometrică de o anumită culoare).
4. **Implementare algoritm de căutare:** algoritm de căutare bazat pe algoritmul Particle Swarm Optimization (PSO), care să poziționeze fiecare agent în spațiul de căutare. Funcția de cost pe care algoritmul trebuie să o minimizeze fiind cea mai mică distanță a

unei drone fața de forma șablon, identificată prin intermediul algoritmului de procesare de imagine.

Pentru fiecare componentă de mai sus va fi realizat un studiu independent, care implementează o soluție pentru problema definită și o validează prin rezultate experimentale.

4. Metodologie generală

Fiecare componentă va fi testată în prima instanță individual, iar în momentul în care rezultatele satisfac, întreg sistemul de control va fi validat pe un scenariu real.

1. **Control UAV:** este validat folosind un stand experimental pentru testele statice. Dacă testele statice sunt satisfăcătoare, soluția va fi validată și în zbor real, într-un mediu controlat.
2. **Server de control:** prima validare este reprezentată de comunicare dintre server și o dronă. Dacă soluția garantează conectare automată la pornirea aplicației și o conexiune stabilă (fără nici-o pierdere de comunicare) atunci același test se va repeta pentru 3 agenți. Sistemul este considerat apt de zbor în momentul în care toți cei 3 agenți prezintă o conexiune stabilă cu serverul. Serverul e valid doar dacă dronele pot fi controlate manual de către operatorul uman, iar imaginea video de la fiecare agent poate fi redată.
3. **Identificare video:** soluție validată inițial în laborator, prin teste statice. Dacă identificare prezintă eroare de sub 5% și timp de procesare mai mic de 0,1 secunde, soluție este considerată stabilă și poate fi integrată în sistem.
4. **Algoritm de căutare:** soluția va fi propusă spre validare în momentul în care toate componentele de mai sus îndeplinesc condițiile impuse. În primă fază algoritmul va fi validat manual, prin poziționarea statică a dronelor la poziția calculată de algoritm. Dacă acest test trece, atunci algoritmul va fi validat cu echipamentele aflate în zbor, dar cu viteza de calcul și de poziționare redusă pentru a putea fi identificate și corectate de factorul uman posibilele erori.

5. Controlul clasic al dronei

Au fost alese structurile PID clasic și PID fracționar pentru această evaluare, deoarece ele au prezentat cele mai bune rezultate simulate. În plus aceasta au fost cuplate cu un algoritm adaptiv, care a își propune să crească robustețea sistemului și să asigure o rejectare cât mai bună a perturbațiilor și variațiilor fizice ale sistemului.

Dinamica sistemului este prezentată în Figura 2. De aici se poate înțelege că mișcarea quadcopterului este direct influențată de controlul perechilor de motoare. Spre exemplu, pentru o deplasare în față, viteza motoarelor M11 și M12 trebuie scăzută cu aceeași valoare, iar viteza motoarelor M21 și M22 trebuie crescută cu aceeași valoare. Astfel drone se va roti în jurul axei Y, ceea ce va determina o deplasare a sistemului în direcția respectivă. Același principiu este aplicabil și celorlalte motoare. Cu cât structura de reglare este mai performantă, cu atât viteza fiecărui motor este ajustată independent de conexiunea dintre ele, pentru a reduce efectul cuplării dintre intrări.

Folosind standul din Figura 3 și programul de control descris mai sus, vor fi realizate 2 studii comparative a unor metode de control avansate în vederea identificării celei mai bune soluții pentru controlul zborului prototipului de quadcopter. Pe baza rezultatelor testelor statice de pe stand se vor determina parametrii inițiali de control pentru zborul dronei.

Metodele analizate:

- Comparație între un PID clasic cu mecanism de anti-saturație și un regulator fracțional.

- Comparație între un regulator PID fracțional și o metoda de reglare adaptivă pentru ajustarea parametrilor regulatorului fracțional.

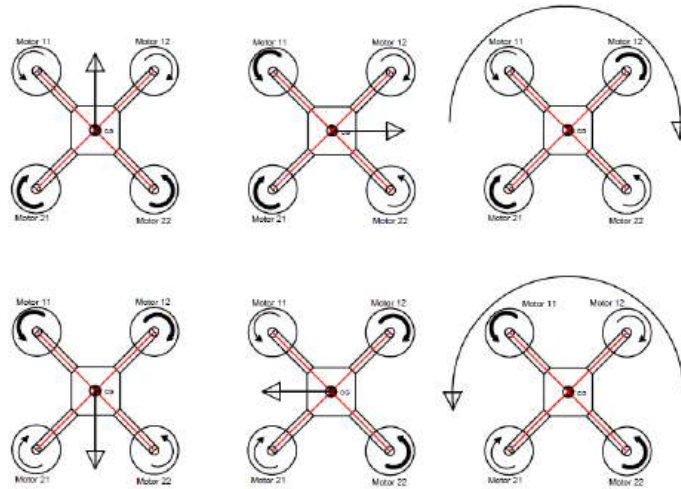


Figura 2: Dinamica de mișcare a unui quadcopter



Figura 3: Standul și prototipul experimental

5.1. Primul studiu comparativ

Au fost evaluate pozițiile unghiulare în jurul axelor X (roll) și Y (pitch), pentru a determina stabilitatea quadcopterului pe standul experimental. Atât pentru regulatorul PID clasic cât și pentru cel fracțional, s-a folosit metoda PSO.

S-a impus ca mărimi de performanță $\omega_{gc} = 20 \text{ rad/sec}$ și $\gamma_m = 100 \text{ deg}$. Rulând algoritmul de determinare al parametrilor regulatorului s-au ajuns la următoarele valori:

Tip regulator	K_p	K_i	K_d	λ	μ
PID	4,65	2,2	0,15	1	1
FO – PID	4,98	1,8	0,5	0,89	0,6

Tabel 2: Parametrii reglatoarelor studiu 1

Pentru regulatorul PID rezultatul a fost foarte apropiat de 0, iar pentru FO-PID, rezultatul returnat prin minimizarea funcției de cost a fost 0,295, care reprezintă valoarea celei de-a 3-a ecuații evaluate, care nu a reușit să fie satisfăcută pe deplin, dar rezultatele sunt satisfăcătoare.

Reglatoarele obținute au fost implementate pe micro-controler-ul prezentat mai sus, prin aplicarea funcțiilor de transfer discretizate. Regulatorul fracțional a fost discretizat folosit metoda de aproximare „Crone” [12], [13]. Ambele metode au rulate cu o frecvență de 250 Hz.

Figura 4 prezintă răspunsul în timp al modelului fizic, atunci când un impuls mecanic scurt este aplicat pe axa Y (pitch). PID-ul clasic răspunde mai rapid, datorită componentei sale derivative. În mai puțin de 1 secundă, variația vitezei unghiulare este contracarată. Singurul dezavantaj este că și celelalte axe sunt perturbate. În cazul regulatorului fracțional, timpul de reglare este scurt, de aproximativ 1 secundă, iar principalul avantaj este că interacțiunile sunt eliminate.

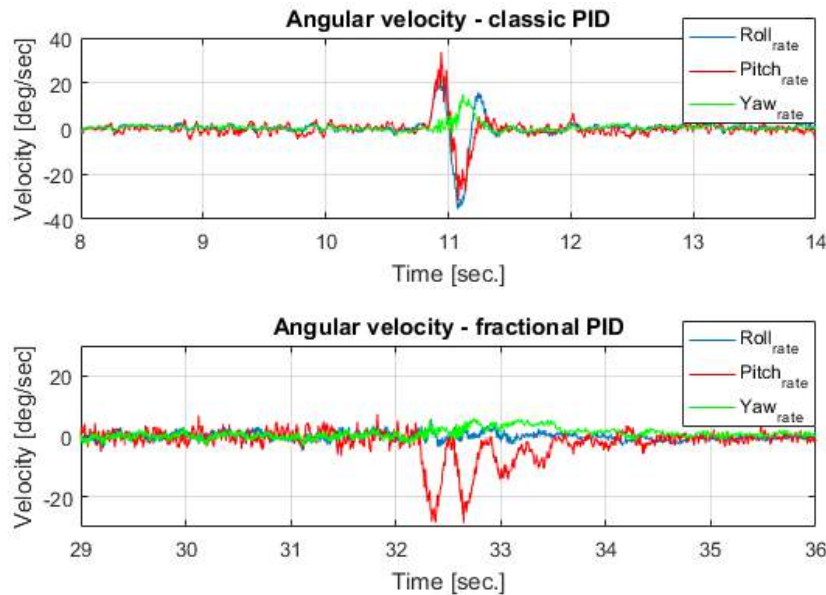


Figura 4: Răspuns la rejectarea perturbațiilor

5.2. Al doilea studiu comparativ

Plecând de la rezultatele obținute în studiul de mai sus, s-a ajuns la concluzia că un simplu regulator, fie și el fracțional, nu poate răspunde adecvat condițiilor instabile de zbor. În aceste situații, evoluțiile modelului datorate turației nominale a motoarelor pentru a ajunge la o anumită altitudine și cel mai important, datorită condițiilor de mediu imprevizibil se impune analiza unei structuri adaptive. O astfel de structură are rolul de a identifica continuu modelul actual al sistemului și de a ajusta parametrii de acord în funcție de aceste modificări. Astfel variațiile de mediu și perturbațiile pot fi rejectate, conducând la un sistem stabil.

Acest studiu a realizat o comparație între un PID clasic și un FO-PID, în jurul cărora a fost construită o structură adaptivă. Pentru această etapă a fost realizat un studiu comparativ între o structură non-adaptivă și una adaptivă.

Modelul a fost aplicat pentru axa X și Y, acesta este prezentat în schema din Figura 5.

Blocul „System FO Controller” implementează modelul regulatorului, fie că vorbim de PID sau de FO-PID. Acesta primește valorile de acord de la blocul „PSO Controller Designer”, care implementează algoritmul PSO pentru determinarea acestor parametri. Acest bloc, primește de la „System Parameters Estimator” parametri fizici ai procesului, prin identificarea continuă a acestora pe baza intrării și a răspunsului sistemului controlat. „Quadcopter Axis” reprezintă o axă de reglare a quadcopterului.

La fel ca pentru prima analiză, au fost impuse constrângerile pentru structura de reglare. Astfel, performanțele impuse trebuie să respecte: frecvența de tăiere $\omega_{gc} = 20$ rad/sec, marginea de fază $\varphi_m = 100^\circ$ și se impune să fie robust, adică variația fazei (sau derivata acesteia) să fie nulă. Specificațiile de proiectare au fost alese pentru a crește stabilitatea generală a sistemului (marginea de fază mare) și pentru a asigura un timp de stabilizare rapid (marginea de câștig mare).

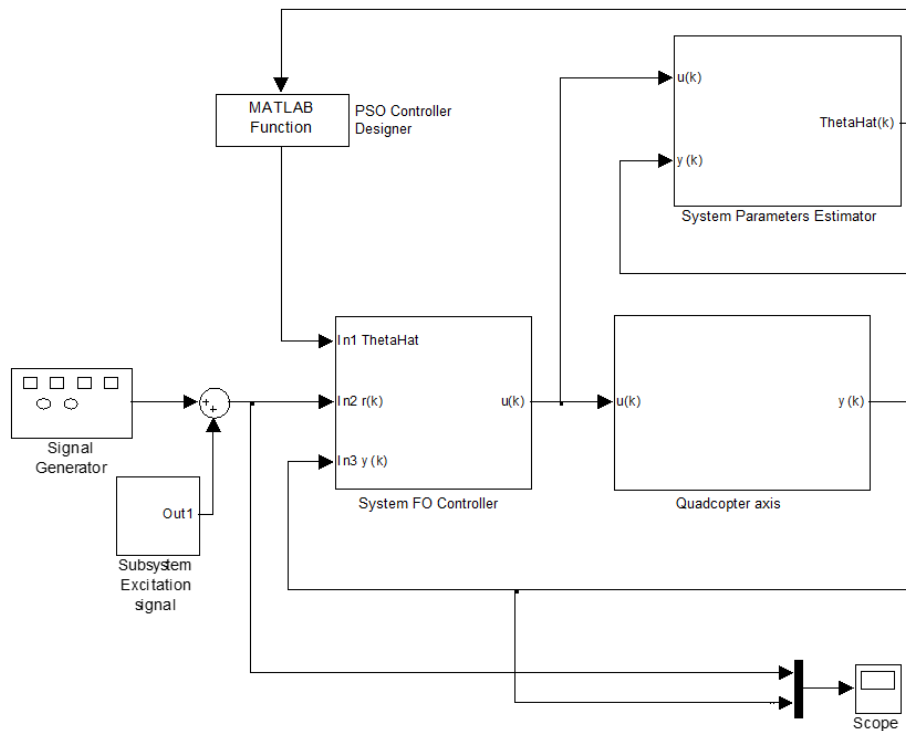


Figura 5: Structura strategiei de control adaptivă propusă

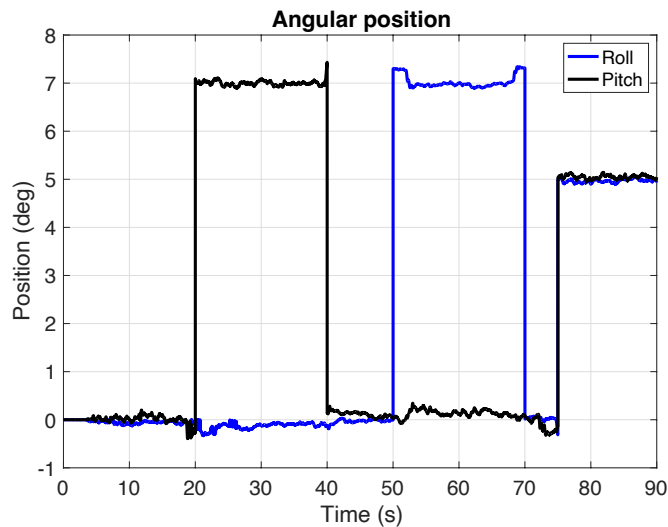


Figura 6: Răspunsul în buclă închisă a structurii adaptive cu FO-PID

Rezultatele experimentale obținute aplicate modelului fizic, cu reglatoarele PID adaptive de ordin întreg și de ordin fracțional, sunt indicate în Figura 6. Aceste rezultate urmăresc să demonstreze modul în care sistemul răspunde la o anumită referință aplicat pe axa X(pitch), Y(roll) și modul în care acestea se afectează reciproc. Punctele de referință în ambele cazuri au fost de 7 grade. Rezultatele au arătat clar că ambele strategii de control sunt capabile să asigure urmărirea referinței, deși reglatoarele de ordin întreg prezintă unele oscilații mari din cauza interacțiunilor axelor sistemului.

Tip regulator	K_p	K_i	K_d	λ	μ
PID	4,65	2,2	0,15	1	1
FO - PID	5,0501	0,2758	0,0097	0,6747	0,5085

Tabel 3: Parametrii inițiali de acord pentru reglatoarele PID și FO-PID adaptive

5.3. Concluzii

Pe standul experimental toate cele 4 metode au satisfăcut condițiile impuse, iar algoritmul PSO, a demonstrat că poate foarte simplu și repede să găsească un set de valori bune pentru specificațiile impuse.

Algoritmul adaptiv aplicat pe stand s-a dovedit a fi cel mai performant, iar regulatorul FO-PID, a furnizat un răspuns mult mai bun din punct de vedere al efortului de comandă, decât regulatorul clasic.

Atât pe stand cât mai ales pentru experimentele de zbor real, componenta adaptiva a controlului și-a dovedit importanța. Mai ales că această nevoie s-a observat prin introducerea unei funcții de ajustare manuală a regulatorului, care subliniază această nevoie de control adaptiv. Aici putem concluziona că pentru un astfel de proces se impune implementarea unei structuri de control adaptivă, pentru a asigura performanțele minime de zbor și pentru a identifica variațiile de proces, cum ar fi cele meteorologice.

Pentru a putea integra acest prototip în cadrul unei aplicații extinse, cum ar fi cea de Search and Rescue sau cea de monitorizare aeriană, este necesar un control al altitudinii și al poziției geografice, pentru a se menține un punct fix de zbor în spațiul 3D. Așadar, echipamente suplimentare trebuie introduse pentru asta, cum ar fi barometru pentru altitudine și GPS pentru poziționarea geografică. Pe lângă aceste componente hardware, noi bucle de control, care să asigure poziționarea corectă sunt necesare a fi introduse în algoritmul de control.

În concluzie, pentru acest studiu, folosirea unui Arduino UNO pentru implementarea unui control simplu de zbor al unui quadcopter poate fi realizată cu ușurință, doar că acest model de control necesită un efort din partea operatorului mult mai mare și chiar învățarea modului în care modelul poate să fie controlat. Pentru a extinde scopul aplicației și pentru a îmbunătăți performanțele de control este necesară implementarea algoritmului de control pe un alt hardware, care să fie capabil să implementeze algoritmi adaptiv de control (cel puțin elementari) și care să fie capabilă să asigure un control de poziționare.

6. Controlul de la distanță al dronei

Obiectivul final al acestei teze este acela da a construi o aplicație multi-agent, în care fiecare agent este capabil să identifice, folosind o cameră video, un anumit obiect țintă. Pentru a realiza acest lucru, atunci când alegem hardware-ul, trebuie să ținem cont că acesta este capabil să realizeze operații de procesare de imagine, poate să controleze zborul dronei și poate să comunice cu serverul de monitorizare.

Odată ce există soluția de control pentru dronă se impune crearea aplicației de monitorizare propriu zise, care reprezintă un sistem distribuit, alcătuit din dronă, un server și o interfață grafică pentru operare. Aplicația de operare și serverul pot rula pe aceeași componentă, dar ca aplicații de sine stătătoare.

Pornind de la rezultatele obținute în capitolul anterior și de la cele scrise mai sus, acest studiu are definite următoarele obiective:

1. Identificare unei soluții hardware și software.
2. Identificarea unei soluții hardware care să fie capabilă să capteze imagini și să le proceseze. De asemenea această componentă trebuie să fie capabilă să se conecteze la un server conectat la internet, printr-o rețea locală.
3. Evaluarea performanțelor de zbor a soluției de control aleasă.
4. Crearea unui server și a unui protocol de comunicare, capabil să schimbe date bidirecțional cu drona.
5. Crearea unei aplicații de monitorizare simple, care să interfațeze operatorul uman cu drona. Aplicația trebuie să fie capabilă să preia și să afișeze datele de zbor (GPS,

altitudine și poziție unghiulară) și să trimită comenzi simple pentru zbor (deplasare în spațiu folosind butoane pentru cele 6 direcții de deplasare).

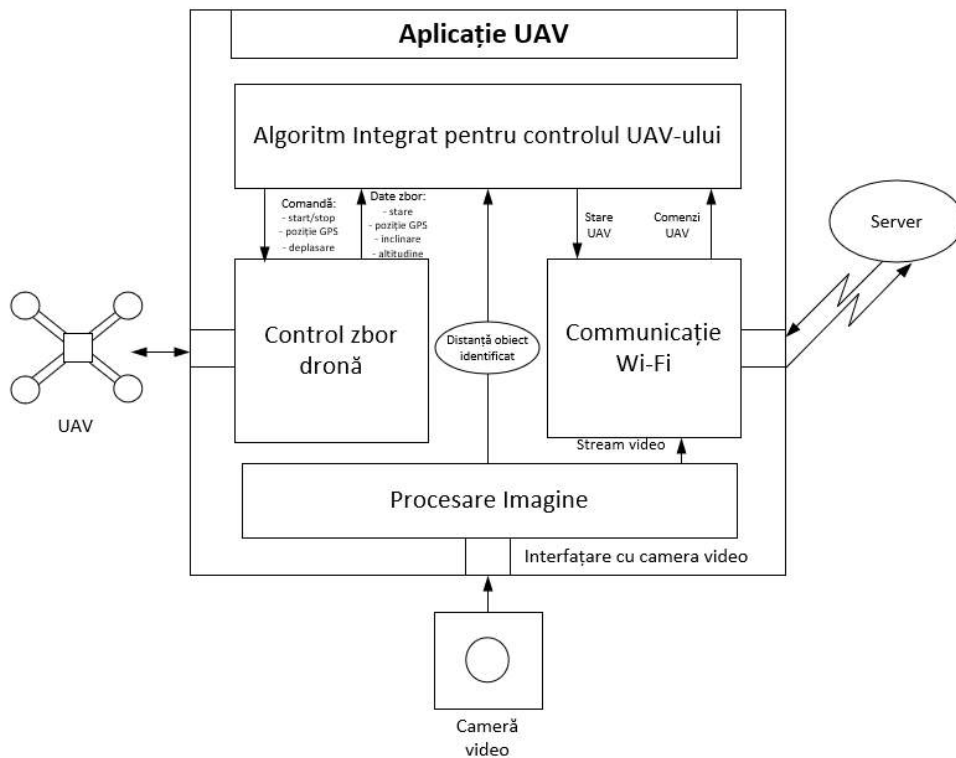


Figura 7: Diagrama aplicației integrate de control al UAV-ului

Figura 7 prezintă diagrama aplicației de control al UAV-ului. Nu este obligatoriu, ca aceasta să fie formată dintr-o singură componentă hardware, ea poate fi formată din mai multe componente dedicate, dar doar una dintre ele va realiza controlul integral și va comunica cu serverul. În vârful structurii propuse, se află programul principal de control al droniei. Acesta este conectat la server prin intermediul unui protocol de comunicare TCP-IP și va schimba informații cu acesta. Serverul poate trimite de exemplu următoarea poziție a droniei, și poate recepționa care este poziția curentă. Pe baza informațiilor primite de la server, algoritmul de control le va procesa și le va trimite mai departe către componenta de control al zborului. În același timp aceasta poate solicita datele actuale legate de poziționare GPS, de exemplu, și le poate trimite mai departe către server.

A fost ales un Raspberry Pi 4B, care dispune de un port dedicat interfațării cu o cameră video, folosită pentru procesarea de imagine. Pentru controlul stabilității droniei, a fost ales controlerul PixHawk 4 împreună cu firmware-ul PX4, care oferă performanțe superioare în controlul zborului. De asemenea, a fost ales un modul GPS auxiliar de tipul M9N, care are o eroare pe poziționare de 1 metru. Controlerul de zbor va fi conectat la Raspberry PI, folosind un port serial UART.

6.1. Evaluare control dronă

Primul pas pentru a seta drona să zboare este necesară instalarea programului de control pe plăcută. Există mai multe variante disponibile pentru PixHawk 4, dar cel mai performant este PX4, versiunea 1.14. Folosind aplicația QGroundControl, a fost instalat acest program și s-a configurat mai departe quadcopterul. Este necesară calibrarea senzorilor, definirea numărului de motoare, definirea și măsurarea bateriei și instalarea radiocomenzii, pentru controlul manual al droniei.

Performanțele hardware și software al acestui controler de zbor au fost validate prin control manual, folosind o radiocomandă. Acestea au fost evaluate direct în zbor, nefiind necesară testarea pe standul static. Prin acest test s-a urmărit modul în care modelul menține poziția fixă în aer cu oscilații minime, care să nu determine o mișcare de deplasare. În al doilea rând s-a urmărit cum răspunde modelul la o serie de comenzi de poziționare unghiulară pe axa X și Y.

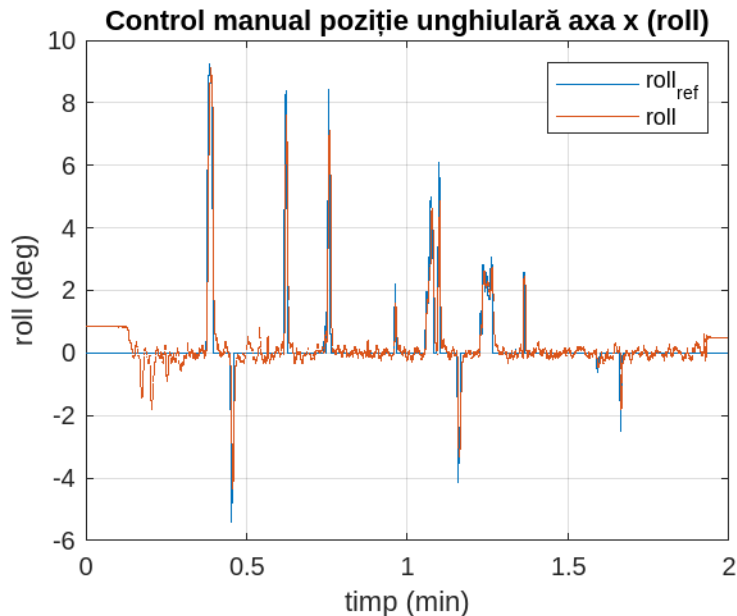


Figura 8: Testare control manual rezultate poziție unghiulară axa X (roll)

După cum se poate observa în Figura 8, poziția în jurul axei X este destul de stabilă. Pe prima perioadă de testare, se observă o înclinare maximă până la aproximativ -2° . Amplitudinile sunt mici, chiar neglijabile, ceea ce demonstrează că modelul poate să mențină o postură fixă. În concluzie, acest tip de controler de zbor este unul care îndeplinește criteriile de performanță impusă, și poate fi integrat cu ușurință în orice aplicație care implică quadcoptere.

6.2. Conceperea aplicației de control

Controlerul de zbor PX4, poate fi interfațat cu ușurință cu Raspberry Pi 4, prin intermediul portului serial. Pentru comunicarea acestora folosesc un protocol de comunicare numit MAVLink (Micro Air Vehicle Link). Acesta este open source și foarte răspândit în integrarea UAV-urilor cu computere companion.

Cel mai important avantaj al acestei librării pe lângă implementarea asincronă, este faptul că gestionează singură comunicarea cu controlerul de zbor. Astfel programul companion, de control poate seta placa în modul OFFBOARD, și poate fixa referințe de zbor pentru dronă. Aici există trei tipuri de referință care pot fi date:

1. **Control manual:** prin intermediul căruia se pot seta direct valorile de poziție unghiulară pe fiecare axă și de turație a motoarelor. Practic aici există 4 comenzi care pot fi aplicate. Acest tip de control este similar cu cel nativ prin radiocomandă.
2. **Control poziție NED:** NED este un acronim pentru Nord, Est și Jos (down în engleză). Folosind această funcționalitate, se poate transmite o poziție viitoare în aceste coordonate. Spre exemplu dacă se dorește o deplasare în față cu 2 metrii (nord), la dreapta cu 4 metrii (est) și în sus cu -10 metrii (- pentru deplasare în sens opus, sus) atunci se transmit aceste valori către dronă, iar aceasta va zbura spre noua poziție care se află la direcțiile specificate, față de punctul actual în care se află drona.

3. **Control coordonate GPS:** prin transmiterea directă a latitudinii, a longitudinii și a altitudinii, folosind senzorul GPS, drona va zbura spre coordonatele indicate.

Aceste moduri de control sunt ideale pentru a defini mai multe timpuri de control. Spre exemplu, dacă se dorește ca o dronă să fie poziționată de către operator la o anumită poziție, atunci acesta poate opta pentru unul dintre primele 3 moduri. Pentru o aplicație de poziționare, care va face acest lucru automat pe baza unui algoritm, cea mai simplă variantă este de a folosi ultimul mod, în care poziția este transmisă în mod direct.

6.3. Implementarea serverului și a aplicației de operare

Serverul și aplicația de operare sunt 2 entități total diferite, care pot rula pe platforme sau sisteme diferite. Aplicație de operare se conectează la server folosit protocolul TCP/IP. Aceasta poate fi să ruleze pe același dispozitiv, caz în care IP-ul serverului este IP-ul calculatorului, fie se poate conecta de pe un alt dispozitiv din rețea prin accesarea IP-ului serverului. Spre exemplu poate rula pe o tabletă sau pe un telefon mobil.

Pentru această aplicație în mod particular a fost ales Java ca limbaj de programare al serverului și al aplicației de monitorizare. În Java se poate folosi framework-ul Spring, care facilitează crearea de aplicații client – server, prin încorporarea propriu zisă a serverului de comunicare și prin folosirea unor funcții predefinite pentru conectarea la funcțiile serverului.

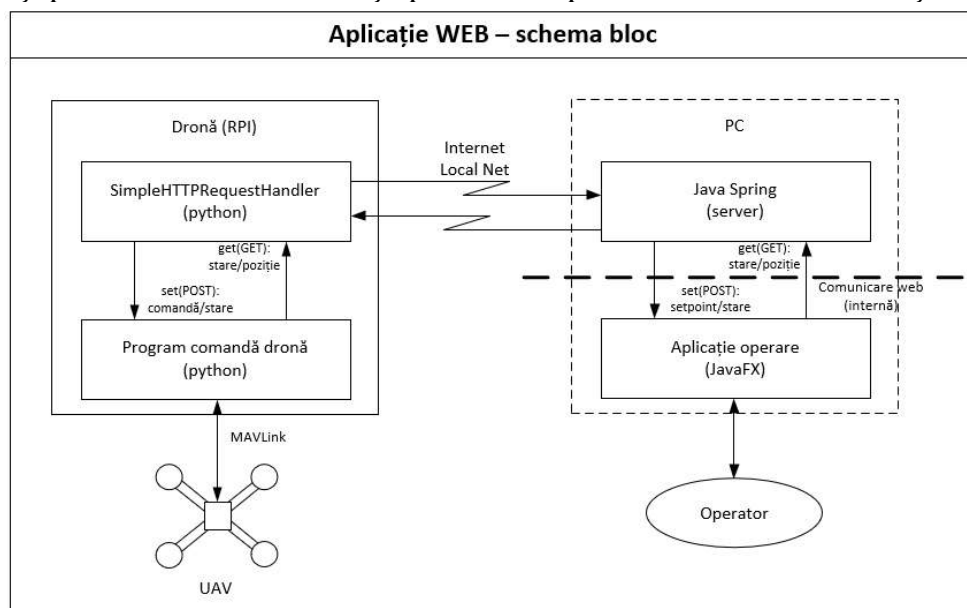


Figura 9: Structură integrare aplicații web

Figura 9 prezintă schema bloc a aplicației web. Algoritmul de comandă a dronei a fost prezentat în secțiunea anterioară. Funcțiile interne ale acestuia sunt conectate la un obiect care gestionează comunicarea între el și Java, SimpleHTTPRequestHandler în cazul de față. Acest obiect HTTP implementat la nivel de Raspberry Pi, gestionează comunicarea între Java Spring și programul de comandă al dronei. Practic acesta reprezintă o interfață web între cele 2 programe.

6.4. Concluzii

În concluzie prin studiul de față s-a reușit crearea infrastructurii de comunicare pentru aplicația finală de căutare și salvare. Această infrastructură a permis evaluarea performanțelor de zbor a controlerului ales pentru dronele din sistem.

Performanțele sunt mulțumitoare. Drona este stabilă în poziția statică de zbor, iar la referințele de poziționare unghiulară răspunde foarte bine. Pentru evaluarea controlului

poziției liniare rezultatele sunt la fel de mulțumitoare. Drona a reușit să urmărească referința impusă. Chiar dacă eroarea la poziție este prezentă, acest aspect este normal, ținând cont că nu a fost folosit modulul GPS pentru a corecta poziționarea. Estimarea poziției s-a bazat doar pe senzorii interni de măsurare inerțială, prin integrarea accelerațiilor liniare citite. Estimări care în mod normal sunt predispuse la erori de integrare.

Studiul acesta și cel anterior reprezintă componenta critică a întregului sistem, componentă care a fost implementată și validată cu succes. Folosind acest model, studiul poate fi extins către următoarele 2 componente, algoritmul de procesare de imagine și cel de căutare.

7. Procesare de imagine

Se va utiliza plăcuța Raspberry Pi aleasă în cadrul studiului anterior, împreună cu o cameră video dedicată acesteia, Raspberry Pi Camera 2. Cele două componente hardware vor fi montate pe structura dronei, pentru a captura imaginea din timpul zborului.

Se va construi aplicația de procesare de imagine folosind librăria OpenCV din Python, iar datele vor fi validate static. Algoritmul are ca obiectiv identificarea unui pătrat roșu și determinarea unghiului și al distanței pe care figura identificată îl face față de centrul imaginii.

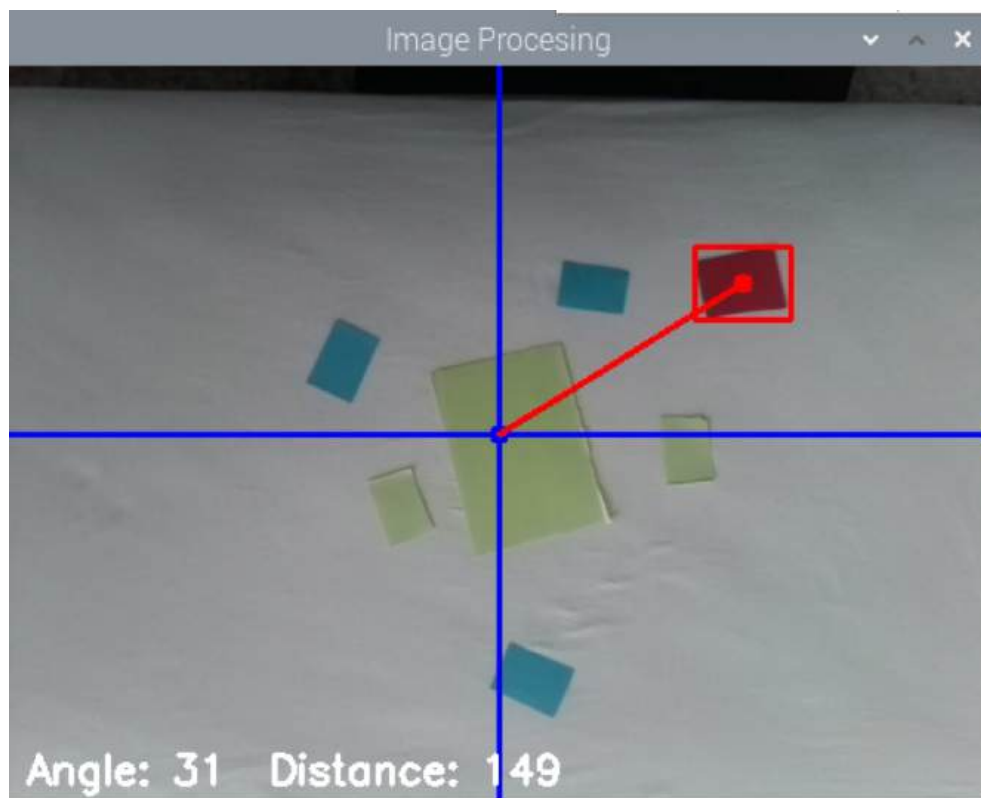


Figura 10: Identificare figură de referință, rezultat final

Figura 10 prezintă rezultatul final al algoritmului de procesare de imagine. Se poate observa cum pătratul roșu este încadrat, iar distanța și unghiul pe care acesta-l face relativ la centrul imaginii este calculat și afișat pe imagine.

Pe baza rezultatelor obținute și prezentate mai sus, putem integra programul de procesare de imagine cu programul de control al dronei. Acesta va rula separat, ca un serviciu de sine stătător și va transmite serverului Java Spring, distanța și unghiul pe care le-a

determinat. Aceste două informații vor fi folosite pentru poziționarea quadcopterului. Spre exemplu, dacă s-a identificat forma unui obiect în colțul din stânga jos al imaginii (cadrantul 4), asta înseamnă că drona se depărtează de obiect, dar înseamnă că tocmai a identificat o posibilă țintă. Ținând cont că direcția de zbor a dronei este opusă locației obiectului, controlul roiului v-a actualiza orientarea quadcopterului în direcția opusă unghiului identificat relativ la poziția imaginii. Astfel, la următoarea iterație, ținta va apărea în partea de sus a imaginii, între cadrantul 1 și 2, cu un unghi relativ de 90°. Drona își va continua direcția de deplasare spre acel punct, poziția lui fiind actualizată de algoritmul de control al sistemului multi-agent.

În concluzie, studiul de față a reușit să prezinte o soluție simplă și rapidă de procesare de imagine care poate fi folosită pentru identificare unei forme țintă. Nu doar imaginea procesată, dar și informațiile calculate (distanță și unghi) vor putea fi folosite de componenta de control al sistemului distribuit pentru a îndeplini scopului acestei aplicații. Urmează ca acest rezultat să fie integrat și adaptat în cadrul soluției finale.

8. Sistem distribuit de căutare și localizare

În urma analizei literaturii făcute în prima parte a acestei lucrări s-a constatat că pentru un swarm de drone sunt esențiale 3 componente fundamentale: comunicația, controlul formației și controlul propriu zis al dronelor. Pentru comunicație este necesar să se asigure o infrastructură și un protocol stabil, pentru a facilita schimbul de informații între toți agenții sistemului. Controlul formației asigură că întreaga structură păstrează postura sau forma impusă de zbor. Această componentă se asigură ca fiecare dronă se află la poziția corectă și că nu există coliziuni între ele. Controlul propriu zis al roiului ce impune tipul de formație și dirijează sistemul către obiectivul comun sau către noua poziție. Această componentă reprezintă și legătura între roi și factorul uman.

Sunt mai mulți algoritmi care pot fi aplicați pentru a implementa un astfel de sistem. Spre exemplu Particle Swarm Optimization (PSO), care este folosit frecvent în rezolvarea problemelor de cost, ar putea fi aplicat pentru controlul unei astfel de soluții. Există și alți algoritmi de optim care pot fi implementați, dar PSO ar avea o aplicabilitate directă, fiind inspirat tocmai din modul în care stolurile de păsări se deplasează în vederea identificării hranei.

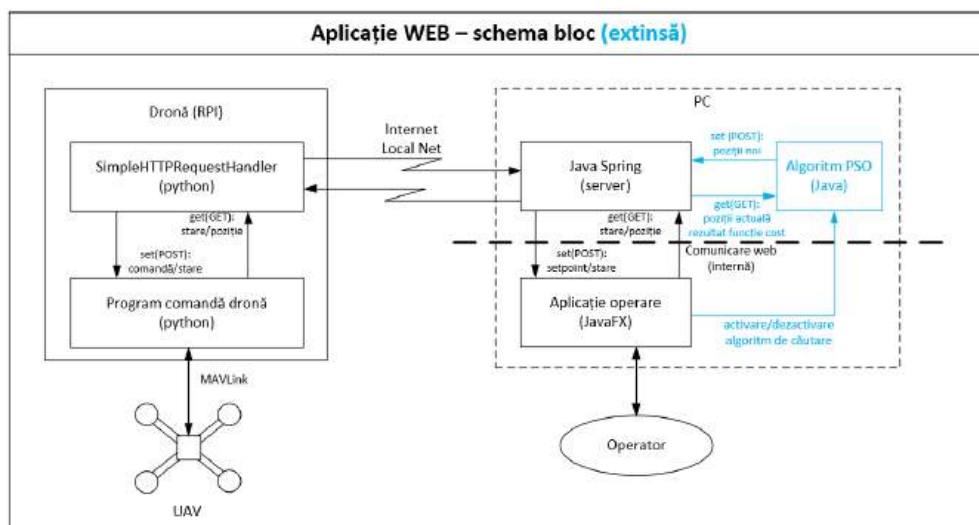


Figura 11: Diagramă bloc aplicație extinsă pentru algoritmul PSO

Figura 11 prezintă cu albastră componenta algoritmului PSO, care este adăugată soluției deja existente. Aceasta poate fi activată sau dezactivată de către operator. Spre exemplu, dacă acesta dorește să repositioneze o dronă, atunci v-a dezactiva algoritmul, va repositiona drona

și va reactiva algoritmul. Prin re poziționare se va actualiza și poziția agentului în programul PSO, pentru a lucra cu noua poziție actualizată.

8.1. Algoritmul PSO – fundament teoretic

Algoritmul Particle Swarm Optimization (PSO) este o tehnică de optimizare inspirată din natură care imită comportamentul social al organismelor, în special comportamentul de grup al păsărilor sau al peștilor. Introdus inițial de Kennedy și Eberhart în 1995, PSO face parte din categoria algoritmilor de optimizare bazată pe populație, în care un grup de soluții candidate (numite particule) traversează spațiul de căutare pentru a găsi soluția optimă. Fiecare particulă din roi reprezintă o soluție potențială la problema de optimizare și își ajustează poziția pe baza a doi factori-cheie: cea mai bună poziție proprie anterioară (prin evaluarea funcției de cost) și cea mai bună poziție găsită în cadrul roiului (poziția care returnează cel mai bun rezultat pentru funcția de cost). Acest aspect cheie face ca PSO să caute eficient în spațiul soluțiilor. Comportamentul PSO este guvernat de un set de ecuații care dictează modul în care particulele își actualizează pozițiile și vitezele [14].

$$v_i(t+1) = w \cdot v_i(t) + c_1 \cdot r_1 \cdot (p_i - x_i(t)) + c_2 \cdot r_2 \cdot (g - x_i(t)) \quad (1)$$

$$x_i(t+1) = x_i(t) + v_i(t+1) \quad (2)$$

$$p_i = \begin{cases} x_i(t), & \text{dacă } f(x_i(t)) < f(p_i) \\ p_i, & \text{în caz contrar} \end{cases} \quad (3)$$

$$g = \begin{cases} x_i(t), & \text{dacă } f(x_i(t)) < f(g) \\ g, & \text{în caz contrar} \end{cases} \quad (4)$$

Ecuațiile (1 - 4) descriu module de execuție al algoritmului PSO, iar variabilele sunt definite astfel:

- $v_i(t)$ – viteza particulei i .
- $x_i(t)$ – poziția particulei i .
- w – componenta inerțială, ajustează influența vitezei.
- c_1 – coeficient cognitiv, influențează atracția particulei către cea mai bună poziție a sa.
- c_2 – coeficient social, influențează atracția particulei către cea mai bună poziție globală.
- r_1 și r_2 – numere aleatorii între 0 și 1
- p_i – cea mai bună poziție a particulei i .
- g – cea mai bună poziție globală.
- $f(x)$ – funcția de cost.

În urma testării algoritmului pe o funcție de cost de antrenare, s-a constatat că rata cu care variază pozițiile particulelor este mult prea mare, iar de cele mai multe ori acestea sfârșeau în limitele spațiului de căutare. Pentru a combate acest efect a fost introdusă o nouă formulă în algoritmul PSO, cu scopul de a modifica componenta inerțială w , astfel permițând o variație mult mai lentă a poziție unghiulare [15].

$$w(i) = m - n \frac{1}{p^{best(i)} + 1} + q \frac{1}{r^{bestf} + 1} \quad (5)$$

unde m , n , p și r sunt valori subunitare care au fost alese experimental, iar $best(i)$ reprezintă cea mai bună soluție personală a particulei i și $bestf$ reprezintă cea mai bună soluție găsită până acuma.

8.2. Validare PSO pe sistemul real

Testul își propune să valideze algoritmul de căutare pe sistemul real, folosind 3 drone. A fost aleasă o zonă de căutare deschisă, a fost setată toată infrastructura sistemului (rețea locală, server și aplicația de operare). Algoritmului PSO i-a fost impusă o arie de căutare, care nu putea să depășească coordonatele GPS [46,777159 23,707359] și [46,777449 23,707512].

Evoluția Dronei 2 este prezentată în Figura 12. Aceasta a fost inițializată în colțul din stânga – sus. Pentru această particulă se observă că poziția ei este actualizată cu o variație destul de mare. Adică următoarea poziție calculată se află în extremitatea opusă poziției actuale. Acest comportament poate să reprezinte și un avantaj, pentru că drona acoperă o arie mai mare de căutare și poate avea șansa să identifice figura țintă. Drona a trecut de mai multe ori pe lângă figură, a identificat-o, dar nu s-a oprit la figură, deoarece poziția impusă de algoritmul se afla în altă parte.

După cum se poate observa, faptul că această drona a identificat la un moment dat figura, a făcut-o să găsească și locația acesteia. Ultima poziție, în momentul în care algoritmul și-a finalizat execuția a fost chiar lângă țintă.



Figura 12: Validare PSO – evoluția poziției Dronei 2

Fluxul de date preluat de la Drona 2 este prezentat în Figura 13. Figura 13 (a) și (b) arată imaginea preluată în momentul ultimei iterații, atunci când algoritmul s-a finalizat și a returnat poziția obiectului țintă.

Figura 13 (c) și (d) arată un exemplu în care în timpul zborului drona trece pe lângă obiectul țintă (10.9.(c)), îl identifică, dar se îndepărtează de acesta (10.9.(d)) înaintând spre următoare poziție. În acest caz, valoarea furnizată algoritmului a fost de aproximativ 182, fiind ultima distanță identificată cu succes. Acest aspect poate fi și un avantaj, deoarece algoritmul PSO determină că se apropie de punctul dorit. Pe de altă parte acest comportament poate să ducă particula curentă (sau celelalte particule) într-un punct de minim local. Aceasta deoarece programul poate să creadă că poziția curentă e aproape de soluție, cu toate că aceasta poate fi în extremitatea opusă.

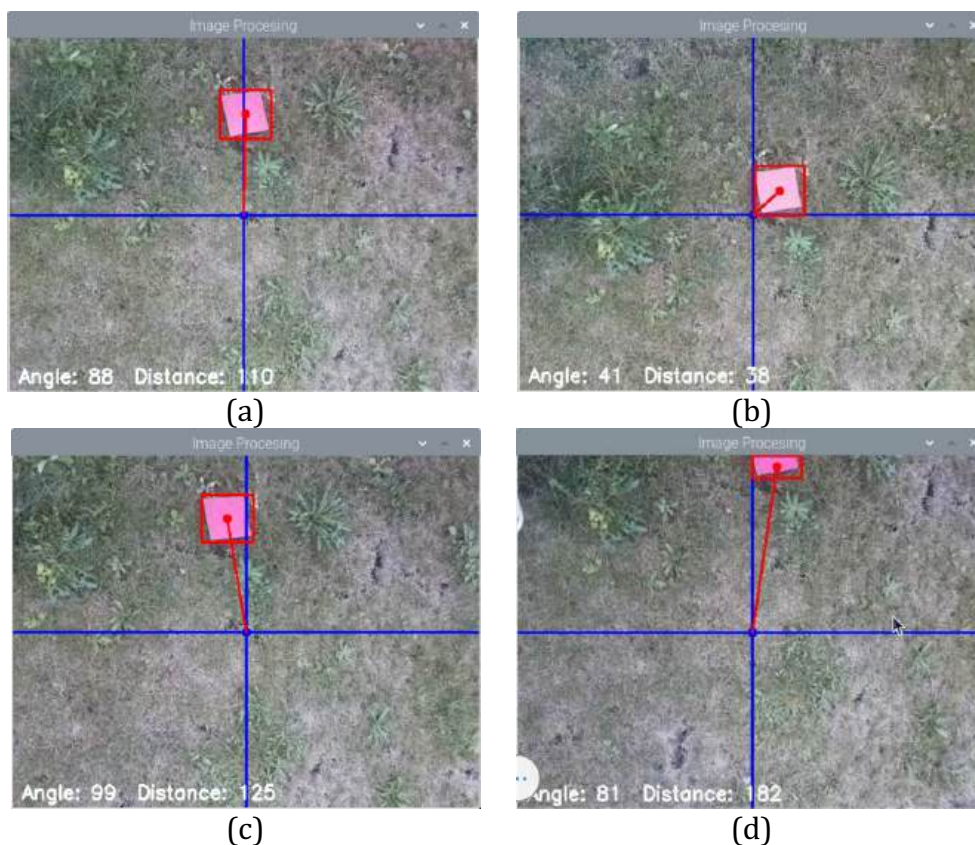


Figura 13: Validare PSO – flux video Drona 2

8.3. Concluzii

Algoritmul PSO se poate considera adecvat pentru o astfel de aplicație. Cu toate că soluția oferită nu a oferit cel mai optim răspuns (2 drone din 3 au rămas blocate într-un punct de minim local), aceasta și-a dovedit aplicabilitatea, satisfăcând obiectivul impus prin localizarea figurii țintă (pătratul roșu).

Observația principală este că algoritmul de față necesită îmbunătățire și adaptare pentru un astfel de sistem, pentru a-i crește eficiența și randamentul. În primul rând execuția secvențială împiedică localizarea în timp real. Pentru a combate acest aspect programul PSO trebuie adaptat ca să funcționeze și între iterații. În momentul în care drona localizează o posibilă țintă poziția ei să fie reactualizată în direcția acelei ținte, fără a mai aștepta ca drona să ajungă în poziția viitoare, calculată inițial. Aceasta se poate face fie prin micșorarea pasului poziție viitoare și creșterea vitezei de execuție a algoritmului independentă de faptul că dronele au atins poziția calculată inițial sau nu. Sau prin introducerea unei bucle interne PSO, dedicată pe fiecare agent, care să-i permită să-și actualizeze poziția în spațiul curent de căutare, dacă a identificat o posibilă țintă. În acel moment celelalte drone care nu au identificat încă nimic, vor continua execuția lor după programul din bucla exterioară, în timp ce cealaltă dronă va funcționa după propria buclă internă. Rezultate pe care le va suprascrie și algoritmului din bucla exterioară.

A doua observație care se face este cea că este necesar să se țină cont și de unghiul care se face în momentul în care figura este identificată. Acesta poate reprezenta ca o direcție, sau un vector al mișcării următoare, iar punctul viitor să fie influențat direct de acea direcție.

Ținând cont că aria de căutare de cele mai multe ori se cunoaște, algoritmului PSO ar trebui forțat să caute soluții către partea interioară zonei de căutare și nu înspre extremități. Optimizând astfel algoritmul cresc șansele de a identifica ținta impusă.

S-a plecat de la ipoteza ca un algoritm PSO poate să fie folosit într-o aplicație de căutare și localizare. Acest studiu demonstrează că sistemul creat pentru această cercetare poate îndeplini scopul cercetării, și anume căutarea și localizarea. Rezultatele culese oferă o înțelegere și formulează o traiectorie de dezvoltare viitoare, care poate să crească eficiența și timpul de răspuns algoritmului.

9. Discuții

Potențialul UAV-urilor și aplicabilitatea acestora sunt de foarte mare interes în momentul actual, iar cercetătorii și mediul de afaceri fac tot posibilul de a furniza o soluție cât mai stabilă și sigură pentru integrarea în orice aplicație. Deja rezultatele obținute sunt extrem de bune și de promițătoare. Agricultură utilizează drone nu numai pentru monitorizarea culturii, dar și pentru acțiuni specifice cum ar fi fertilizarea sau combaterea bolilor. Industria construcțiilor se bucură de dezvoltarea de instrumente prin integrarea de UAV-uri, care să evalueze și să compare modul în care un proiect este executat în raport cu schița acestuia. Pentru gestionarea dezastrelor, dronele de bucură de asemenea de popularitate. Acestea pot fi desfășurate în zone greu accesibile pentru a obține informații despre zona respectivă sau pot oferi ajutor victimelor acelor dezastre. În toate aceste aplicații omul este principalul câștigător, beneficiind de siguranță, timp de răspuns și de implementare mai mic și de optimizarea costurilor cu care o anumită activitate se poate realiza.

Sistemele distribuite de drone au început să fie populare în ultimi 8 – 10 ani, când o serie de cercetări pe algoritmi (inteligentă colectivă) de gestiune a populațiilor extinse de roboți au început să fie dezvoltate. Încă acest sector nu este în totalitate explorat și sunt multe întrebări care caută un răspuns, dar direcția este una promițătoare. Mai mulți agenți implicați pot rezolva o problemă mult mai rapid și eficient decât unul singur.

După cum a reieșit și în urma rezultatelor prezentate în această teză, un sistem de căutare și localizare poate fi dezvoltat cu oarecare ușurință și poate fi aplicat pe un caz real. Tehnologia actuală permite atât metode de control stabil și robuste pentru drone, cât și procesare eficientă de imagine cu costuri relativ mici și performanțe ridicate. De asemenea se constată că un algoritm de optim precum PSO este destul de promițător pentru a fi folosit ca algoritm de inteligență colectivă pentru controlul sistemelor distribuite de drone.

Bineînțeles, pe lângă aceste rezultate și direcții promițătoare și motivaționale de dezvoltare viitoare, există și piedici. Principalul obstacol este capacitatea mică a bateriilor, care limitează timpul de zbor a unei drone. Impediment care poate fi combătut pentru moment printr-un sistem multi-agent de drone, pentru că o dronă poate să substituie activitatea altei drone atunci când cea din urmă rămâne fără energie.

Acoperirea rețelelor de comunicare constituie și ele un impediment. Infrastructura de telecomunicații existentă nu asigură o acoperire globală. În anumite zone trebuie desfășurate echipamente costisitoare de comunicații pentru a face aceste sisteme funcționale. O soluție viitoare este la orizont, prin internetul oferit de sateliți, la care ar putea să se conecteze și dronele. Acest internet de generație nouă ar putea sigura o acoperire globală și pentru un sistem distribuit de drone.

Legislația americană și europeană actuală încă nu aprobă folosirea pe scară largă a UAV-urilor cu scop comercial, decât pentru o anumită serie de cazuri excepționale. Practic soluțiile comerciale, de care să se bucure utilizatorul final sunt momentan restricționate, în special pe motive de siguranță. Odată cu identificare unor soluții robuste, capabile să răspundă la defecțiuni și la criteriile de siguranță publică, legislația va începe să se deschidă și să reglementeze folosirea controlată, dar pe scară largă a UAV-urilor.

Tocmai problemele nerezolvate fac din acest subiect unul de o importanță și cu un potențial enorm. O industrie în acest sens a început să se dezvolte, care începe să angajeze din

ce în ce mai mulți oameni în fiecare an, iar impactul economic pe care acesta-l va avea va fi de asemenea unul important pentru economia globală.

10. Concluzii

În concluzie teza de față a reușit în primă fază un studiu bibliografic extins a celor mai importante lucrări de specialitate în domeniul aplicațiilor cu UAV-uri și al sistemelor distribuite de UAV-uri. Aceste informații au constituit un fundament pentru teză în continuare. Pe baza principiilor și a concluziilor formulate în urma acestei etape s-au clădit următoarele studii care au fost prezentate în această teză.

Primul studiu concluzionează că structurile moderne de reglare pot îmbunătăți stabilitatea în timpul zborului a unui quadcopter. O structură de reglare adaptivă folosind regulator PID fracțional a reprezentat noutatea. S-a constatat că o astfel de structură poate să răspundă specificațiilor impuse și poate să fie eficientă din punct de vedere al efortului de comandă, în comparație cu o structură clasică.

Studiul al doilea și-a propus să construiască o aplicație distribuită cu ajutorul căreia să fie realizat controlul de la distanță a unui quadcopter. Rezultatele obținute au fost mulțumitoare. Aplicația creată a fost stabilă și a putut comanda mișcare unei drone în aer. S-a observat că drona urmărește referința impusă, se întoarce la locația inițială, iar problemele de comunicare au fost inexistente. Această aplicație reprezintă temelia sistemului distribuit care va fi folosit în activități de căutare și localizare.

Procesarea de imagine în vederea identificării unei figuri țintă a fost prezentată în studiul 3. În cazul de față drona trebuia să identifice un pătrat de culoare roșie. Rezultatele au fost mai mult decât mulțumitoare. Prin folosirea librăriei OpenCV, care a fost dezvoltată și îmbunătățită în această direcție s-a reușit identificarea, urmărirea și calcularea unghiului și distanței de la centrul imaginii până la forma țintă. Ulterior această soluție a fost analizată și într-un scenariu real.

În urma testării sistemului distribuit pe un scenariu real s-a concluzionat că o astfel de soluție este implementabilă și oferă soluții viabile. Rezultatele au fost prezentate în cadrul studiului 4, unde algoritmul PSO a fost aplicat pentru a determina pozițiile GPS viitoare în vederea atingerii obiectivului. Cea mai importantă concluzie și observație în urma acestui studiu constă în identificare modului în care performanțele unui astfel de sistem pot fi optimizate, pentru un răspuns mai rapid și mai bun. Lipsa eficienței consta în faptul că atunci când drona zbura de la o locație la alta și trecea pe lângă țintă, aceasta nu se oprea, ci din contră, în majoritatea cazuri aceasta ieșea din raza vizuală a țintei. Ca soluție aici au fost propuse următoarele 3 direcții:

1. Creșterea frecvenței de execuție a programului. Pentru soluția prezentată algoritmul aștepta re poziționarea tuturor dronelor, apoi se trecea la calculul următoarei iterații. Prin creșterea frecvenței, algoritmul PSO poate să ajusteze traiectoria, prin preluarea informațiilor pe care drona le preia în timpul zborului. În plus, în acest caz pasul de integrare folosit pentru calculul poziției următoare trebuie redus. Astfel traiectoria calculată inițial va fi segmentată în puncte intermediare mai mici.
2. Crearea unei bucle interne PSO pentru fiecare dronă. Bucle care v-a interveni doar dacă identifică o posibilă țintă. În acest caz, drona va lucra independent de algoritmul PSO, va prelua control și își va corecta traiectoria în funcție de informațiile video pe care le preia. Dacă într-adevăr obiectul găsit este cu adevărat cel căutat atunci drona se oprește și va comunica rezultatul către algoritmul mare, care v-a opri execuția.
3. Introducerea unei componente care să oblige PSO să poziționeze dronele în interiorul zonei de căutare, astfel toate pozițiile viitoare să fie calculate cât mai

aproape de centru ariei. Această soluție ar funcționa doar pentru cazurile în care se cunoaște zona de căutare.

În concluzie, teza de față a reușit să demonstreze că algoritmul de optim PSO, poate fi aplicat în aplicații de căutare și localizare. Sistemul implementat a reușit să ofere cel puțin o soluție care să confirme această ipoteză.

În plus se propun următoarele direcții viitoare pentru dezvoltarea și îmbunătățirea performanțelor soluției prezentate:

1. Optimizarea algoritmului PSO, prin introducerea unei strategii noi care să permită segmentarea traiectoriilor de zbor cu scopul de a identifica mai rapid ținta.
2. Implementarea unei componente software care să împiedice ciocnirea dintre drone.
3. Validarea noului sistemului pe un sistem real folosind drone aflate în zbor.

REFERINȚE

- [1] "Wikipedia - Search and Rescue." [Online]. Available: https://en.wikipedia.org/wiki/Search_and_rescue
- [2] "RNLI foundation." [Online]. Available: <https://rnli.org/about-us/our-history/timeline/1824-our-foundation>
- [3] "Salvamont Romania - History." [Online]. Available: <https://www.salvamontromania.ro/istoric.php>
- [4] Y. Alghamdi, A. Munir, and H. M. La, "Architecture, Classification, and Applications of Contemporary Unmanned Aerial Vehicles," *IEEE Consum. Electron. Mag.*, vol. 10, no. 6, pp. 9–20, Nov. 2021, doi: 10.1109/MCE.2021.3063945.
- [5] B. Alzahrani, O. S. Oubbati, A. Barnawi, M. Atiquzzaman, and D. Alghazzawi, "UAV assistance paradigm: State-of-the-art in applications and challenges," *J. Netw. Comput. Appl.*, vol. 166, p. 102706, Sep. 2020, doi: 10.1016/j.jnca.2020.102706.
- [6] A. Gupta, T. Afrin, E. Scully, and N. Yodo, "Advances of UAVs toward Future Transportation: The State-of-the-Art, Challenges, and Opportunities," *Future Transp.*, vol. 1, no. 2, pp. 326–350, Sep. 2021, doi: 10.3390/futuretransp1020019.
- [7] G. Singhal, B. Bansod, and L. Mathew, "Unmanned Aerial Vehicle Classification, Applications and Challenges: A Review," Nov. 27, 2018. doi: 10.20944/preprints201811.0601.v1.
- [8] M. Hassanalian and A. Abdelkefi, "Classifications, applications, and design challenges of drones: A review," *Prog. Aerosp. Sci.*, vol. 91, pp. 99–131, May 2017, doi: 10.1016/j.paerosci.2017.04.003.
- [9] "Open Category - Low Risk - Civil Drones | EASA." Accessed: Apr. 24, 2024. [Online]. Available: <https://www.easa.europa.eu/en/domains/civil-drones-rpas/open-category-civil-drones>
- [10] "How to Register Your Drone | Federal Aviation Administration." Accessed: Apr. 24, 2024. [Online]. Available: https://www.faa.gov/uas/getting_started/register_drone
- [11] F. Ahmed, J. C. Mohanta, A. Keshari, and P. S. Yadav, "Recent Advances in Unmanned Aerial Vehicles: A Review," *Arab. J. Sci. Eng.*, vol. 47, no. 7, pp. 7963–7984, Jul. 2022, doi: 10.1007/s13369-022-06738-0.
- [12] P. Lanusse, J. Sabatier, and A. Oustaloup, "Fractional Order PID and First Generation CRONE Control System Design," in *Fractional Order Differentiation and Robust Control Design*, vol. 77, in Intelligent Systems, Control and Automation: Science and Engineering, vol. 77., Dordrecht: Springer Netherlands, 2015, pp. 63–105. doi: 10.1007/978-94-017-9807-5_2.
- [13] P. Lanusse, J. Sabatier, and A. Oustaloup, "Extension of PID to fractional orders controllers: a frequency-domain tutorial presentation," *IFAC Proc. Vol.*, vol. 47, no. 3, pp. 7436–7442, 2014, doi: 10.3182/20140824-6-ZA-1003.01053.
- [14] J. Kennedy, "Particle Swarm Optimization," in *Encyclopedia of Machine Learning*, C. Sammut and G. I. Webb, Eds., Boston, MA: Springer US, 2011, pp. 760–766. doi: 10.1007/978-0-387-30164-8_630.
- [15] E.-H. Dulf, C. I. Muresan, and Daniel D. Timis, "Adaptive fractional order control of a quadrotor," *Dyn. Syst. – Theory Appl.*, pp. 129–138, 2019, doi: <https://doi.org/10.34658/9788366287303>.

LISTĂ DE PUBLICAȚII

1. Timiș, D. D. & Dulf, E. H. (2024). Unmanned Aerial Vehicles: State-of-the-Art on Swarm Control, *Journal of Control Engineering and Applied Informatics*, Vol. 26 No. 4, 69-80, DOI: 10.61416/ceai.v26i4.9270.
2. Timiș, D. D., Muresan, C. I., & Dulf, E. H. (2022). Design and experimental results of an adaptive fractional-order controller for a quadrotor. *Fractal and Fractional*, 6(4), 204. WOS:000786793700001(Q1), <https://doi.org/10.3390/fractalfract6040204>.
3. Timis, D. D., & Dulf, E. H. (2020). Software implementation and test of an advanced robust control applied to a quad-copter. In 2020 IEEE International Conference on Automation, Quality and Testing, Robotics (AQTR) (pp. 1-4). IEEE., WOS:000643732600075 <https://ieeexplore.ieee.org/document/9130018>.
4. Dulf, E. H., Timis, D. D., Szekely, L., & Miclea, L. C. (2019). Adaptive fractional order control applied to a multi-rotor system. In 2019 22nd International Conference on Control Systems and Computer Science (CSCS) (pp. 696-699). IEEE. WOS:000491270300117 <https://ieeexplore.ieee.org/document/8744692>.
5. Dulf, E.H., Muresan, C.I. and Timis, D. D. (2019) Adaptive fractional order control of a quadrotor. In: 2019 International Conference "Dynamical Systems – Theory and Applications" (DSTA) (pp. 129-138). ISBN 978-83-66287-30-3. <https://doi.org/10.34658/9788366287303>.

ARTICOLE ÎN CURS DE PUBLICARE

6. Timiș, D. D. & Dulf, E. H. (2024). Unmanned Aerial Vehicles: State-of-the-Art in Applications and Swarm Control, *IEEE ACCESS*, ISSN 2169-3536.