



Universitatea
Ștefan cel Mare
Suceava

Facultatea de Inginerie
Electrică și Știința
Calculatoarelor

CALCULATOARE ȘI TEHNOLOGIA INFORMAȚIEI

Ing. Dragoș-Alexandru ANDRIOAIA

REZUMAT TEZĂ DE DOCTORAT

Cercetări privind îmbunătățirea sistemelor de
mentenanță predictivă utilizând IoT

Coordonator științific:
Prof.univ.dr.ing Vasile-Gheorghică GĂITAN

Suceava 2024

Această lucrare a fost susținută de proiectul „Proinvent”, contractul nr. 62487/03.06.2022 - POCU/993/6/13 - Cod 153299, finanțat de Programul operațional de capital uman 2014-2020 (POCU), România.

Doresc să adresez mulțumirile mele domnului profesor Vasile Gheorghiță GĂITAN pentru îndrumarea și susținerea de-a lungul întregii perioade de formare prin studii doctorale. Pentru observațiile făcute și sfaturile date asupra tezei de doctorat, mai doresc să mulțumesc domnilor: Prof. univ. dr. ing George CULEA, Prof. univ. dr. ing Cornel TURCU, Conf. univ. dr. ing. Ioan UNGUREAN, Ș. l. dr. ing. Ionel ZAGAN și Ș. l. dr. ing Petru Gabriel PUIU.

Cuprins

1. Motivația cercetării	1
1.1. Obiectivele cercetării	1
1.2. Structura lucrării	2
2. Stadiul actual și perspectivele sistemelor de mentenanță predictivă	3
2.1. Paradigma „Industrializarea 4.0”	3
2.2. Aspecte teoretice privind mentenanța.....	3
2.3. Elementele componente ale sistemelor de mentenanță predictivă	4
2.4. Componenta dispozitivelor IoT ce permit transferul datelor de la senzori în cadrul sistemelor de mentenanță predictivă.....	6
2.4.1. Arhitecturi ale sistemelor IoT ce pot fi utilizate în domeniul mentenanței predictive.....	6
2.4.2. Protocoale de rețea utilizate în IoT	6
2.4.3. Tehnici de achiziție a datelor	7
2.5. Tehnici de preprocesare a datelor utilizate de sistemele de mentenanță predictivă	7
2.6. Algoritmi de învățare automată utilizați de sistemele de mentenanță predictivă	9
2.6.1. Învățarea supravegheată	9
2.6.1.1. Algoritmi utilizați de către sistemele de mentenanță predictivă pentru clasificare și regresie	9
2.6.1.2. Algoritmi clasici de învățare automată supravegheată	10
2.6.1.3. Învățare profundă	11
2.6.1.4. Metodele ansamblului	11
2.6.2. Învățare nesupravegheată	12
2.6.3. Învățare semi-supravegheată	12
2.6.4. Strategii de validare și evaluare	12
2.6.5. Parametrii ce rezultă în urma unui sistem de mentenanță predictivă.....	12
2.7. Concluzii.....	13
2.8. Stabilirea direcțiilor de cercetare	13
3. Cercetări privind realizarea unui sistem de mentenanță predictivă pentru UAV-uri.....	14
3.1. Proiectarea și realizarea standurilor experimentale utilizate pentru extragerea datelor de antrenare ale modelelor de învățare automată	14
3.2. Analiza comparativă a performanțelor algoritmilor de tratare a valorilor anormale, ce pot fi utilizați în cadrul sistemelor de mentenanță predictivă pentru predicția RUL al acumulatorilor de Li-ion și identificarea tipului de defect a motoarelor BLDC ale UAV-urilor propulsate electric	17
3.2.1. Introducere	17
3.2.2. Analiza comparativă a algoritmilor utilizați în detecția valorilor anormale	18
3.2.3. Date experimentale, rezultate și discuții	18
3.2.4. Concluzii	19
3.3. Analiza comparativă a performanțelor algoritmilor de tratare a valorilor lipsă ce pot fi utilizați în cadrul sistemelor de mentenanță predictivă pentru, predicția RUL al acumulatorilor de Li-ion și identificarea tipului de defect a motoarelor BLDC din cadrul UAV-urilor propulsate electric.....	20
3.3.1. Introducere	20

3.3.2.	Analiza comparativă a algoritmilor utilizați pentru completarea valorilor lipsă	20
3.3.3.	Date experimentale, rezultate și discuții	21
3.3.4.	Concluzii	22
3.4.	Predicția RUL al acumulatorilor de Li-ion din cadrul UAV-urilor propulsate electric utilizând tehnicile de învățare automată	22
3.4.1.	Introducere	22
3.4.2.	Metodologia de predicție a RUL utilizând învățarea automată	23
3.4.3.	Setup experimental	24
3.4.4.	Rezultate și discuții	25
3.4.5.	Concluzii	25
3.5.	Identificarea tipului de defect a motoarelor BLDC din cadrul UAV-urilor, utilizând tehnicile de învățare automată	25
3.5.1.	Introducere	25
3.5.2.	Metodologia de identificare a tipului de defect a motoarelor BLDC	26
3.5.3.	Setup experimental	27
3.5.4.	Rezultate și discuții	30
3.5.5.	Concluzii	30
4.	Realizarea unui sistem de mentenanță predictivă pentru UAV-uri folosind Fog Computing	31
4.1.	Introducere	31
4.2.	Conceperea și realizarea practică a unui sistem de mentenanță predictivă pentru UAV-uri	31
4.2.1.	Schemele bloc și standul experimental	33
4.3.	Rezultate și discuții	39
4.4.	Concluzii	40
5.	Concluziile lucrării	40
6.	Contribuții și direcții de cercetare viitoare	42
6.1.	Contribuții teoretice	42
6.2.	Contribuții demonstrate pe baza unei cercetări practice	42
6.3.	Măsuri și direcții viitoare de cercetare	44
6.4.	Diseminarea rezultatelor	44

1. Motivația cercetării

1.1. Obiectivele cercetării

Teza de doctorat „Cercetări privind îmbunătățirea sistemelor de mentenanță predictivă utilizând IoT” își propune să vină cu contribuții la îmbunătățirea sistemelor de mentenanță predictivă, utilizând progresele făcute în ultimii ani în domeniul tehnologiilor apărute recent precum Internet of Things (IoT) dar și a altor tehnologii emergente precum Big Data (BD), Data Mining (DM), Cloud Computing (CC) și Internet of Services (IoS). Din cercetările făcute în cadrul primului raport „Stadiul actual și perspectivele sistemelor de mentenanță predictivă” s-a constatat că sistemele de mentenanță predictivă din cadrul Unmanned Aerial Vehicles (UAV) nu au fost suficient cercetate.

Datorită unui potențial neexploatat, UAV-urile au început să fie utilizate în diverse aplicații cu scopuri diferite precum: transport, agrement, monitorizarea mediului, căutare și salvare, precum și în domeniul militar pentru identificarea țintelor de luptă.

Odată cu creșterea numărului de UAV-uri, au crescut proporțional și numărul accidentelor. Pentru a scădea numărul accidentelor este necesară implementarea unei politici de întreținere.

Mentenanța este un cumul de activități organizatorice ce are ca și scop menținerea echipamentelor dintr-un sistem, în stare de funcționare [1]. Mentenanța poate fi efectuată prin aplicarea unei politici de întreținere reactive sau proactive. Prin aplicarea strategiilor de întreținere reactive, echipamentele sunt reparate după ce s-au defectat [2]. Acest lucru, în cazul UAV-urilor, este de nepermis deoarece nu elimină producerea accidentului. Aplicarea unei strategii de mentenanță preventivă presupune schimbarea componentelor înainte ca acestea să se defecteze, lucru ce poate crește costurile cu mentenanța. Mentenanța predictivă presupune monitorizarea parametrilor ce ne pot da informații despre starea de sănătate a echipamentului pentru a interveni înainte ca acesta să se defecteze. Mentenanța predictivă este standardul dorit de majoritatea fabricilor în cadrul standardului „Industrializarea 4.0”. Prin aplicarea unei strategii de întreținere predictivă echipamentul este reparat înaintea apariției eșecului funcțional deci, costurile cu întreținerea sunt reduse considerabil.

Plecând de la necesitatea implementării sistemelor de mentenanță predictivă în cadrul UAV-urilor propulsate de motoarelor electrice Brushless DC (BLDC), în această teză de doctorat s-a cercetat realizarea sistemelor de mentenanță predictivă pentru UAV-urile propulsate electric, care să monitorizeze componentele cu o fiabilitate scăzută. Printre componentele cu o fiabilitate scăzută avem motoarele electrice BLDC, precum și acumulatorii de Li-ion ce le alimentează. Astfel, pentru implementarea acestei cerințe, au fost stabilite următoarele obiective:

1. Analiza sistemelor de mentenanță predictivă pe baza literaturii de specialitate și stabilirea principalelor elemente componente ale sistemelor de mentenanță predictivă;
2. Conceperea unui model pentru predicția Remaining Useful Life (RUL) al acumulatorilor de Li-ion din componența UAV-urilor propulsate electric;
3. Conceperea unei metode pentru identificare a tipului de defect a motoarelor electrice BLDC din componența UAV-urilor propulsate electric;

4. Selectarea algoritmului din cadrul etapei de preprocesare a datelor ce are performanțele cele mai bune în completare a valorilor lipsă pe setul de date provenit de la un sistem de mentenanță predictivă al unui UAV propulsat electric;
5. Selectarea algoritmului din cadrul etapei de preprocesare a datelor ce are performanțele cele mai bune în identificarea valorilor anormale pe setul de date provenit de la un sistem de mentenanță predictivă al unui UAV propulsat electric;
6. Stabilirea unei arhitecturi pe care ar trebui să o aibă sistemele de mentenanță predictivă pentru UAV-uri propulsate electric;
7. Realizarea practică a unui sistem de mentenanță predictivă pentru UAV-urile propulsate electric care să monitorizeze componentele cu o fiabilitate scăzută precum acumulatorii de Li-ion și motoarele electrice BLDC.

1.2. Structura lucrării

Lucrarea este structurată pe un număr de șase capitole, prezentate după cum urmează:

Primul capitol prezintă obiectivele trasate în cadrul acestei teze de doctorat și structura acestei lucrări.

În al doilea capitol, sunt prezentate principalele aspecte privind strategiile de întreținere, elementele componente ale sistemelor de mentenanță predictivă, arhitecturi IoT utilizate, stiva de rețea adoptată prin convenție în IoT, precum și tehnicile de extragere a informației din date.

În cadrul celui de-al treilea capitol, au fost selectați algoritmi și concepute modelele din componența unui sistem de mentenanță predictivă pentru UAV-uri. Astfel, utilizând datele generate din monitorizarea funcționării unui acumulator de Li-ion și din monitorizarea funcționării unui motor BLDC, aflate în componența unui UAV, a fost: identificat algoritmul ce are performanțele cele mai bune în identificarea valorilor anormale, identificat algoritmul ce are performanțele cele mai bune în completarea valorilor lipsă, concepută o metodă ce poate fi utilizată pentru predicția timpului rămas din viață (Remaining Useful Life, RUL) al acumulatorilor de Li-ion și concepută o metodă ce poate fi utilizată pentru identificarea tipului de defect a motoarelor electrice BLDC. Performanța metodelor concepute și algoritmilor selectați a fost dovedită prin intermediul rezultatelor experimentale obținute.

În cel de-al patrulea capitol, au fost analizate două posibilități de realizare a unui sistem de mentenanță predictivă pentru UAV-uri, ce utilizează o arhitectură IoT bazată pe Fog Computing. Cele două variante de sisteme de mentenanță predictivă diferă prin modul de transmisie a datelor de pe nivelul de fog, pe nivelul de cloud, Long Range (LoRa) și WiFi. Tot în cadrul acestui capitol, este prezentată și realizarea practică a celor două variante de sisteme de mentenanță predictivă pentru UAV-urile acționate electric. Ambele variante de sisteme de mentenanță predictivă au o arhitectură bazată pe Fog Computing ce include: nivelul dispozitivelor, nivelul de fog și nivelul de cloud. Nivelul dispozitivelor este utilizat pentru colectarea datelor de la senzori prin intermediul dispozitivelor ce au rol de hub senzor. Pe nivelul de fog datele sunt preprocesate. Preprocesarea include utilizarea algoritmilor de completare a valorilor lipsă și de identificare a valorilor anormale, selectați în capitolul doi. Datele preprocesate sunt trimise de pe nivelul fog pe nivelul de cloud unde a fost realizată o aplicație în Python 3.6 ce utilizează metoda de identificare a tipului de defect a motoarelor BLDC și metoda de predicție a RUL al acumulatorilor de Li-ion, concepute în capitolul doi. La final prin intermediul unei interfețe web (Flask) sunt afișate defectele motorului BLDC și RUL al acumulatorului de Li-ion din cadrul UAV-ului.

Cel de-al cincilea capitol prezintă concluziile acestei lucrări precum și viitoarele direcții de cercetare.

În capitolul șase, pe baza lucrării au fost scoase în evidență contribuțiile teoretice și practice.

Teza de doctorat conține un număr de 74 de figuri și 17 tabele. Bibliografia conține un număr de 265 de referințe bibliografice. Peste 75% din referințele bibliografice sunt publicații din ultimii cinci ani. Marea majoritate a publicațiilor utilizate au fost selectate din jurnale și conferințele domeniului abordat, conținute de platformele Elsevier și IEEE.

2. Stadiul actual și perspectivele sistemelor de mentenanță predictivă

2.1. Paradigma „Industrializarea 4.0”

Evoluția în domeniul tehnologiei de producție, care era complet diferită în trecut, poartă și denumirea de „*Revoluție Industrială*”. Noile tehnologii de producție au schimbat fundamental condițiile de lucru și stilul de viață al oamenilor.

„Industrializarea 4.0” presupune transformarea fabricilor de astăzi, în fabrici inteligente, prin utilizarea tehnologiilor emergente, cum ar fi: IoT, Cyber Physical Systems (CPS), Internet of Services (IoS), BD&DM și Cloud Computing (CC) [3].

„Industrializarea 4.0” utilizează proceduri de auto configurare, de auto optimizare, de analiză predictivă și de auto diagnoză, fiind în măsură să asiste și să sprijine operatorul uman în activități complexe și dificile atât tehnic cât și fizic.

Termenul de *Internetul lucrurilor (Internet of Things, IoT)* a fost folosit pentru prima dată de Kevin Ashton în anul 1999 și se referea la acea vreme mai mult la rețele de senzori Radio-Frequency Identification (RFID) [4]. Astăzi, *IoT se referă la o societate în care o mulțime de obiecte/dispozitive sunt conectate [5]. Sistemele cyber-fizice (Cyber Physical Systems, CPS), sunt sisteme încorporate complexe. CPS sunt capabile să se descopere, să comunice și să se controleze reciproc [6], [7]. CC-ul este o tehnologie care se bazează pe procesarea distribuită a datelor și pe tehnologiile Internet. Internetul serviciilor (Internet of Services, IoS), a apărut prin convergența conceptelor de Web 2.0 și Service Oriented Architecture (SOA) [8].*

Întreținerea predictivă îmbunătățește considerabil suportul tehnic prin identificarea erorilor pe care nici un om nu le poate identifica, eliminând astfel timpul de nefuncționare neplanificat al mașinii și crescând siguranța persoanelor ce utilizează mașina. *O întreținere deficitară poate reduce productivitatea unei fabrici cu 5 - 20 %.* Deci, putem spune că întreținerea predictivă aduce o serie de *beneficii precum: creșterea productivității, evitarea defectelor, eliminarea perioadelor de nefuncționare neplanificate, utilizarea optimă a resurselor umane și financiare și optimizarea planificării intervențiilor de întreținere.* Înainte să se defecteze, echipamentele industriale au simptome că ceva nu merge [9].

2.2. Aspecte teoretice privind mentenanța

Necesitatea realizării unei producții cât mai mari, la o calitate cât mai bună și cu costuri cât mai mici, a determinat o îmbunătățire a politicilor de mentenanță în cadrul companiilor axate pe producție [1].

În general, strategiile de întreținere pot fi împărțite în [2]: *Strategii de întreținere reactivă* și *strategii de întreținere proactivă*, Fig. 1.

Pentru *Mentenanța reactivă (Run-to-Failure, R2F)*, reparațiile se fac după ce echipamentele s-au defectat. Reparațiile în regim de urgență au un cost mult mai mare decât reparațiile planificate, astfel încât planurile de întreținere care se bazează pe întreținerea reactivă sunt cele mai scumpe [2].

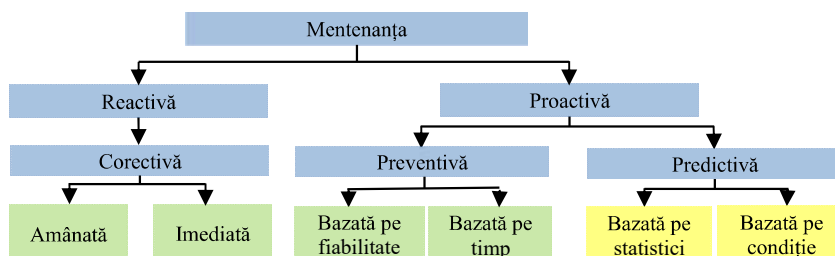


Fig. 1 Clasificarea strategiilor de întreținere [2].

Mentenanța proactivă este o strategie de întreținere preventivă pentru menținerea stării de funcționare a mașinilor sau echipamentelor [10].

Mentenanța preventivă este orice lucru făcut preventiv pentru a evita apariția defecțiunilor și a uzurii inutile. Se poate baza pe timpul calendaristic, pe timpul de funcționare al echipamentului sau pe cicluri de funcționare [11].

Strategia *mentenanței predictivă* are la bază monitorizarea echipamentului/instalației cu ajutorul unor instrumente de măsurare capabile să evalueze starea acestuia direct în timpul funcționării, permanent sau într-o anumită perioadă de timp [10], [12].

Întreținerea predictivă aduce o serie de avantaje precum: crește durata de viață a echipamentelor, elimină opririle accidentale, micșorează timpul de nefuncționare al utilajelor, costuri de mentenanță reduse și siguranță în exploatarea mediului. Întreținerea predictivă poate detecta apariția unui defect prin diagnosticare și poate preciza timpul rămas până la apariția unui defect prin prognoză [10].

Printre companiile de top ce oferă servicii de mentenanță predictivă sunt [13], [14], [15]: IBM, SAP, Microsoft și Siemens.

2.3. Elementele componente ale sistemelor de mentenanță predictivă

Sistemele de mentenanță predictivă au capacitatea de a detecta modificări ale stării fizice a echipamentelor (semne de avarie) pentru a se efectua lucrările de întreținere, cu scopul de a mări durata de viață a echipamentelor și de a scădea probabilitatea apariției unei avarii funcționale.

În general, sistemele de mentenanță predictivă au în componență următoarele module (Fig. 2): (1) modul utilizat pentru achiziția datelor; (2) modul de preprocesare și stocare date; (3) modul de identificare defecte și predicție; (4) modul de analiză a indicatorilor de performanță și optimizare a programului de întreținere; (5) modul de implementare a programului de mentenanță [2], [10], [16], [17].

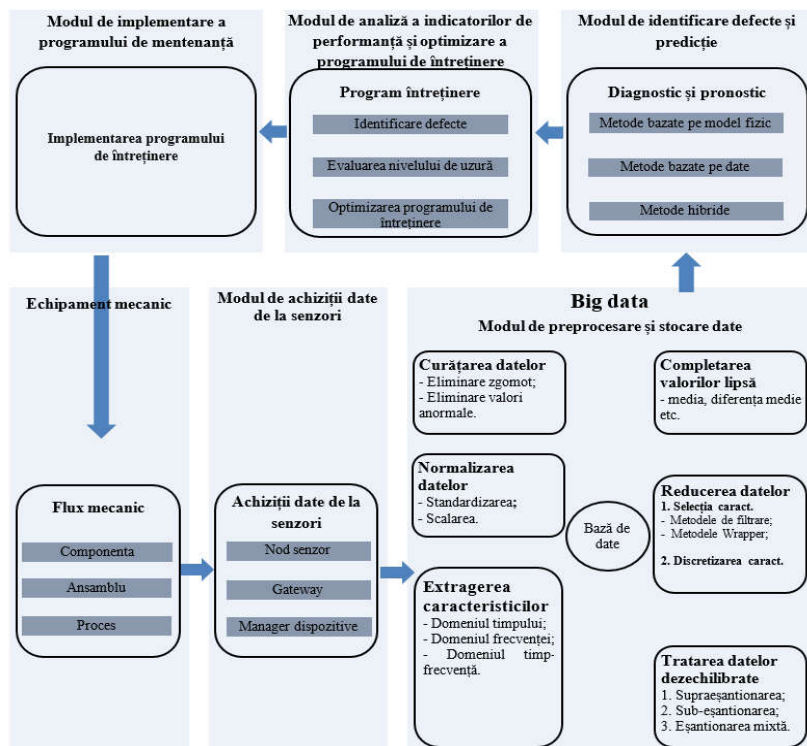


Fig. 2 Schema generală a sistemelor de mentenanță predictivă [2], [10], [16], [17].

Modulul de achiziții date de la senzori permite *colectarea datelor de la senzori prin intermediul dispozitivelor IoT* [16], [17], [18].

După ce datele sunt achiziționate, toate datele colectate vor fi *stocate într-o bază de date cu scopul de a extrage diagnosticul și prognoza*. Datele vor fi prelucrate prin intermediul etapelor prezentate în cadrul *modulului de preprocesare și stocare date* din Fig. 2 [10], [16], [17], [19].

Modulul de identificare defecte și predicție se concentrează în principal pe *detectarea, clasificarea și predicția defectelor* pentru a implementa o strategie de întreținere predictivă. *Diagnosticul* este axat în general pe detectarea și clasificarea defectelor [10]. *Prognosticul* se bazează în general pe predicția timpului rămas din viață al unui echipament, *indicele RUL (Remaining Useful Life)* [20]. Pentru analiza datelor, cu scopul de a diagnostica și a anticipa defecte se pot utiliza [1]: metode bazate pe model fizic, metode bazate pe date, metode bazate pe cunoștințe și metode hibride.

Modulul de analiză a indicatorilor de performanță și optimizare a programului de întreținere, afișează rezultatul obținut în urma modulului de analiză a datelor și ne va furniza un program de întreținere [10], [16], [17].

Modulul de implementare a programului de mentenanță, în cadrul acestui modul, programul de întreținere va fi implementat după ce decidenții vor alege strategia de întreținere [16], [17].

2.4. Componenta dispozitivelor IoT ce permit transferul datelor de la senzori în cadrul sistemelor de mentenanță predictivă

2.4.1. Arhitecturi ale sistemelor IoT ce pot fi utilizate în domeniul mentenanței predictive

În ceea ce privește arhitectura sistemelor IoT, *nu există o arhitectură generală adoptată prin convenție*. În literatura de specialitate au fost propuse mai multe arhitecturi IoT. Cele mai utilizate arhitecturi în cadrul lucrărilor de specialitate sunt [21], [22], [23]:

- Arhitecturile *clasice* cu trei, patru și cinci niveluri, Fig. 3;
- Arhitecturile bazate pe *Edge Computing (EC)*, Fig. 4;
- Arhitecturile bazate pe *Fog Computing (FC)*, Fig. 5.

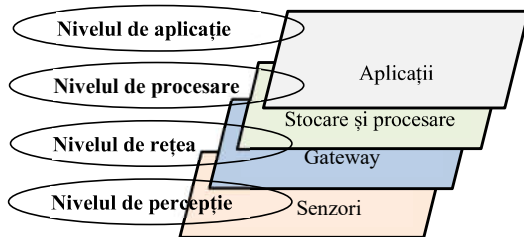


Fig. 3 Arhitectura sistemelor IoT, cu patru niveluri [21], [24].

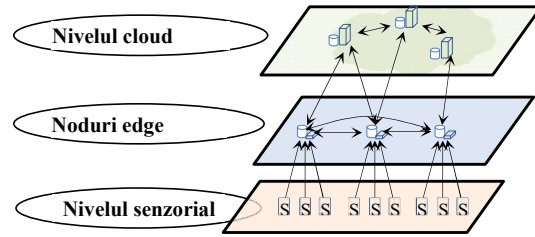


Fig. 4 Arhitectura sistemelor IoT, bazată EC [25], [26].

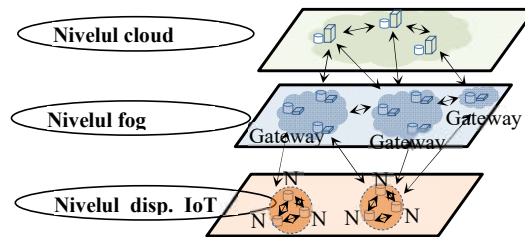


Fig. 5 Arhitectura sistemelor IoT, bazată FC [27], [28].

P. Ferrari prezintă o analiză comparativă în lucrarea [29] a arhitecturii bazate pe EC și a arhitecturii bazate pe cloud, utilizate în cadrul sistemelor de mentenanță predictivă. Pe baza datelor experimentale, se constată că arhitectura bazată pe cloud este cu 350% mai eficientă, dar este consumatoare de bandă.

2.4.2. Protocoale de rețea utilizate în IoT

În ceea ce privește protocoalele de rețea, nu există o stivă IoT adoptată până în prezent. O serie de lucrări încadrează protocoalele de rețea utilizate în IoT pe modelul Transmission Control Protocol (TCP)/IP, Fig. 6. [21], [30], [31].

HTTPS, XMPP, CoAP, MQTT, AMQP
UDP, TCP
IPv4, IPv6, 6 LoPAN
IEEE802.15.4 WiFi, Bluetooth, BLE

Fig. 6 Protocoalele de rețea IoT după modelul TCP/IP [21] [30], [31].

Multe tehnologii de rețea au fost adoptate în IoT. Tehnologiile dezvoltate vizează diferite piețe, automatizarea locuinței, transport inteligent etc.

2.4.3. Tehnici de achiziție a datelor

Sistemele de mentenanță predictivă utilizează în general tehnici de achiziție a datelor, bazate pe extragere (pull) sau pe împingere (push). La un sistem de achiziție a datelor bazat pe tehnici de extragere, utilizatorul definește o interogare și o trimite către sistem care va returna valoarea. La sistemele de achiziție a datelor bazate pe tehnici de împingere, apariția unei situații noi (date noi) va fi comunicată în mod autonom de sistem (Fig. 7) [32], [33], [34].

Sistemele de achiziție a datelor pot fi de mai multe tipuri, astfel avem:

- **Sisteme de achiziție a datelor în timp real;**
- **Sisteme de achiziție a datelor bazate intervale de timp;**
- **Sisteme de achiziție a datelor bazate pe evenimente.**

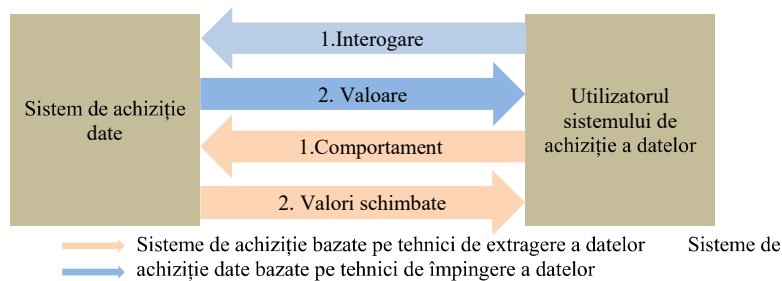


Fig. 7 Tehnici de achiziție a datelor utilizate de sistemele de mentenanță predictivă.

Folosirea sistemelor de achiziție a datelor în timp real în domeniul mentenanței predictive, va transfera o cantitate considerabilă de informații inutile ce va crește costurile de transmisie și procesare a datelor. Folosirea sistemelor de achiziție a datelor bazate intervale de timp și bazate pe evenimente în cadrul sistemelor de mentenanță predictivă va scădea costurile cu transmisia și procesarea datelor, iar în final va rezulta și o economie de energie.

2.5. Tehnici de preprocesare a datelor utilizate de sistemele de mentenanță predictivă

În general, datele brute din lumea reală pot să ascundă anumite tendințe, putând fi inconsistente sau incomplete. Ca și exemplu, *datele ce fac referire la anumite defecte sunt minoritare, fiind dezechilibrate, deoarece informațiile ce fac referire la defecte sunt mult mai puțin frecvente decât cele ce fac referire la situații fără defecte* [19].

Rezultatul extragerii cunoștințelor din date prin intermediul tehnicilor de DM depinde foarte mult de calitatea datelor [19].

Preprocesarea datelor se referă la pregătirea și transformarea setului de date brute cu scopul de a îmbunătăți eficiența modelului de extragere a cunoștințelor din date, prin: *creșterea preciziei, creșterea robusteții, scăderea cerințelor de memorie/putere de calcul și reducerea timpului de calcul* [19].

Metodele de preprocesare a datelor cuprind următoarele etape [19]:

1. *Curățarea datelor;*
2. *Completerea valorilor lipsă;*

3. *Reducerea datelor;*
4. *Tratarea datelor dezechilibrate;*
5. *Extragerea caracteristicilor;*
6. *Normalizarea datelor;*
7. *Etichetarea datelor.*

Curățarea datelor - Pentru a avea performanțe ridicate, metodele de extragere a cunoștințelor din date utilizează date fără valori lipsă, fără valori anormale și fără zgomot. *Seturile de date ce conțin valori lipsă, zgomot și valori anormale pot provoca confuzii în cadrul metodelor extragere a cunoștințelor din date* [19], [35].

Completarea valorilor lipsă - urmărește aproximarea valorilor ce lipsesc din seriile de date achiziționate de la senzori [16].

Reducerea datelor - După etapele anterioare în care semnalul a fost procesat, setul de date în care se găsesc caracteristicile semnalului poate avea dimensiuni considerabile. *Pentru a crește precizia și a reduce timpul de procesare al modelului utilizat pentru extragerea cunoștințelor, este necesară reducerea numărului de caracteristici.* Caracteristicile irelevante și redundante pot provoca deteriorarea performanței unui model de învățare automată [36].

Tratarea datelor dezechilibrate - *Dacă o clasă conține un număr mai mare de probe decât clasele alternative, atunci seturile de date sunt dezechilibrate. Clasa care are mai multe probe este, în general, o clasă majoritară iar clasa care conține probe foarte rare este o clasă minoritară* [37]. Algoritmii modelelor de învățare automată au performanțe slabe atunci când datele sunt dezechilibrate. *Echilibrarea claselor, se poate face în două moduri, prin adăugarea de eșantioane în clasa minoritară (supra-eșantionarea) sau prin eliminarea eșantioanelor din clasa majoritară (sub-eșantionarea)* [19], [37], [38]. *Întrucât fiecare metodă prezintă avantaje și dezavantaje, pentru a beneficia de avantaje este recomandată o utilizare hibridă (eșantionare mixtă)*

Extragerea caracteristicilor - Majoritatea algoritmilor de învățare automată necesită extragerea caracteristicilor din datele provenite de la senzori. *Extragerea caracteristicilor este procesul de dezvoltare a informațiilor ascunse despre semnalul de intrare* [1], [39], [40]. Datele ce au formă de undă pot fi observate în domeniul timpului și domeniul frecvenței. Astfel, *se disting trei tipuri de tehnici de analiză a semnalelor: domeniul timpului, domeniul frecvenței și domeniul timp-frecvență* [19].

Normalizarea datelor - Datele provenite de la senzori au diverse intervale de valori. *Majoritatea algoritmilor ai modelului de extragere a cunoștințelor din date sunt foarte sensibili la o multitudine de intervale și de scale ale variabilelor, ceea ce duce de obicei la scăderea performanței modelului.* Normalizarea datelor se poate face utilizând: *centrarea medie, standardizarea* sau *Scalarea* [19].

Etichetarea datelor - *În întreținerea predictivă, învățarea automată supravegheată este cea mai utilizată tehnică pentru predicția cu ajutorul clasificatoarelor a tipului de defect și pentru predicția indicelui RUL. Învățarea automată supravegheată utilizează date istorice care trebuie etichetate* [41].

2.6. Algoritmi de învățare automată utilizați de sistemele de mentenanță predictivă

Învățarea automată utilizează tehnici statistice care învață să reprezinte o funcție ce reprezintă valorile sau categoriile țintă. În funcție de tipul datelor disponibile, algoritmi de învățare automată se pot clasifica după Fig. 8 [42], [43].

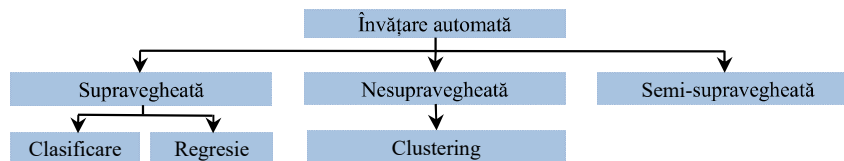


Fig. 8 Clasificarea algoritmilor de învățare automată [44].

În cadrul *învățării automate supravegheate*, antrenarea modelului se face cu date istorice ce au putut fi etichetate [42], [40].

Învățarea automată nesupravegheată nu se bazează pe date istorice etichetate. Acesta descoperă modele în datele de intrare, dar fără să știe ce reprezintă acele tipare [40], [42], [43].

Învățarea automată semi-supravegheată, datele de intrare sunt un amestec de exemple etichetate și neetichetate. Pentru a face posibilele predicții, modelul trebuie să învețe structurile pentru a-și organiza datele [44].

În domeniul *mentenanței predictive*, cea mai utilizată tehnică de învățare automată este *învățarea automate supravegheată*, și este folosită pentru conceperea [43]:

- *Modelelor de detectare a anomaliilor*, detectează un comportament atipic al sistemului sau componentei;
- *Modelelor de clasificare*, prezic eșecul într-un interval de timp dat sau clasifică tipurile de defecte;
- *Modelelor de regresie*, prezic RUL.

2.6.1. Învățarea supravegheată

Algoritmii de învățare supravegheată învață dintr-un set de date istorice, format din date perechi (intrarea, ieșire). Regresia și clasificarea sunt două tipuri de învățare automată supravegheată [19], [42].

Clasificarea Algoritmii de clasificare sunt folosiți pentru predicția etichetelor datelor de intrare ale căror etichete nu sunt cunoscute. Metodele de clasificare pot fi utilizate pentru clasificare multi-clasă și binară. În *sistemele de mentenanță predictivă*, *clasificarea* este utilizată în general pentru *clasificarea defectelor a unui echipament* [42], [45].

Regresia Algoritmii de regresie sunt utilizați pentru predicția unei ieșiri care este o variabilă continuă. Variabilele de intrare pentru modelele de regresie pot fi numerice sau categorice [42], [45]. În *sistemele de mentenanță predictivă*, *regresia* este utilizată pentru *predicția RUL a unui echipament*.

2.6.1.1. Algoritmi utilizați de către sistemele de mentenanță predictivă pentru clasificare și regresie

Principali algoritmi de învățare automată, utilizați în literatura de specialitate pentru detecția anomaliilor, identificarea tipului de defect și predicția RUL al echipamentelor Tab. 1.

În urma analizei rezultatelor obținute, se constată că interesul cercetătorilor față de identificarea tipului de defect este mult mai mare față de predicția RUL al echipamentelor.

Tab. 1 Algoritmi de învățare automată utilizați în literatura de specialitate pentru, predicția indicelui RUL, identificarea tipului de defect, precum și pentru detectarea anomaliilor.

Referință	LRM	LR	BN	SVM	DT	RF	KNN	CNN	LSTM	AE	MLP	Bagging	Boosting	Alți algoritmi	Scopul utilizării algoritmilor		
															Regresie, Predicție	Identificarea tipului de defect	Deteție anomalii
K., S.-Jong și colab. [46]	X														X		
L. Feng și colab. [47]		X													X		
X. Tangbin și colab. [48]												X	X			X	
Y. Jianbo [49]										X						X	
S. Snehsheel și colab. [50]				X												X	
R. Thomas Walter și colab. [51]				X		X	X	X								X	
A. Zenghui și colab. [52]									X							X	
T. Benkedjouh și colab. [53]								X								X	
L. Hedda și colab. [54]						X							X			X	
Y. C. Liang și colab. [55]								X								X	
S. S. Gustavo și colab. [56]				X	X	X					X				X		
J. C., Alberto și colab. [57]						X							X	X	X		
S. Uferah și colab. [58]				X	X	X	X									X	
Y. Rui și colab [59]									X							X	
K. H. Petia și colab. [60]														X			X
P. Tianyang și colab. [61]			X													X	
H. Z. Mohammad și colab. [62]						X										X	
V. Gunasegaran și colab. [63]					X												
S. Mohamed și colab. [64]									X						X		
Y. Yong și colab. [65]									X						X		
X. Ling și colab. [66]								X	X								X
Suma	1	1	1	4	3	6	2	4	5	1	1	1	3	2	6	12	2

Dintre algoritmi folosiți pentru clasificare și regresie *RF* și *Long Short Term Memory (LSTM)* și *sunt cei mai utilizați algoritmi* datorită faptului că aceștia pot memora dependențe pe termen lung.

2.6.1.2. Algoritmi clasici de învățare automată supravegheată

Principalii algoritmi de învățare automată utilizați sunt:

- **Regresia liniară:** Linear Regression Model (LRM), Multiple Linear Regression (MLR) etc [45], [67];
- **Regresia logistică:** Binary Logistic Regression (LR), Ordinal Logistic Regression (OLR) etc [67], [68];

- **Clasificatorul bayesian naiv:** Bayesian Networks (BN), Gaussian Naive Bayes (GaussianNB), Bayesian Belief Network (BBN) etc [31], [42], [69], [70];
- **Mașini cu vectori suport:** Suport Vector Machine (SVM), Support Vector Regression (SVR) etc [42];
- **Arbori de decizie: Decision Tree (DT):** Classification And Regression Tree (CART), C4.5 (succesorul lui ID3) și C5.0. etc [45], [68], [71];
- **Algoritmii pădurilor aleatorii:** Random Forests (RF), Kernel Random Forest (KRF), Regression Tree (RT), Extremely Randomized Trees (ERT), Breiman-RF (BRF) sau metode de boost cum ar fi, Adaptive Boosting (AdaBoost) etc [72], [73], [74], [75];
- **Algoritmii celor mai apropiați k vecini:** K-Nearest Neighbors (KNN), Weighted Nearest Neighbor (WNN), Fuzzy K-Nearest Neighbor (FKNN), Bonferroni-Mean Based Fuzzy K-Nearest Neighbor (BM-FKNN) etc [42], [76], [77].

2.6.1.3. Învățare profundă

Învățarea profundă este un subtip a învățării automate ce utilizează de obicei arhitecturi ale rețelelor neuronale cu niveluri profunde. Rețelele neuronale tradiționale au, de obicei, în componență câteva niveluri ascunse de neuroni, în timp ce metodele învățării profunde au în componența rețelei neuronale un număr mult mai mare de niveluri ascunse după cum e și denumirea termenului „*Deep = adânc*” [40], [42].

Principalii algoritmi de învățare profundă sunt:

- **Rețele neuronale convoluționale, Convolutional Neural Network (CNN):** *LeNet*, *AlexNet* etc. [78].
- **Rețele neuronale recurente:** Recurrent Neural Network, (RNN), Long Short Term Memory (LSTM), Recursive Neural Tensor Network (RNTN), Recurrent Neural Network-Restricted Boltzmann Machine (RNNRBM), Elman’s Simple Recurrent Networks (SRN) și Echo State Networks (ESN) etc. [31], [60], [67], [68], [79], [80].
- **Autoencoderul:** Autoencoders (AE), Stacked Autoencoders (SAE), Denoising Autoencoder (DAE), Discriminant Autoencoder (DisAE), Stacked Denoising Autoencoders (SDAE) etc. [67], [81], [82], [83].
- **Rețele multistrat de perceptroni, Multilayer Perceptron (MLP)** [84].

2.6.1.4. Metodele ansamblului

Utilizează predicția făcută de mai multe modele analitice în loc să folosească doar un model. Astfel, aceste metode se bazează pe ideea că mai multe modele pot acoperi diferite subseturi ale datelor de antrenament și ca atare fiecare completează deficiența celuilalt [85].

Bagging, *datele de antrenament sunt împărțite în mai multe eșantioane cu un înlocuitor numit eșantioane bootstrap.* Un eșantion bootstrap va avea aceeași dimensiune cu dimensiunea eșantionului ordinal și va păstra 75% din valorile originale, iar 25% din vari vor fi înlocuite prin reperarea valorilor din eșantionul original [81], [86].

Boosting, a fost introdus în 1995 de Freund și Schapire și *se bazează pe faptul că combinarea ipotezelor într-o ordine secvențială crește precizia.* Algoritmii de boosting transformă cursanții slabi în cursanți puternici. Acești algoritmi sunt bine proiectați pentru a

rezolva problemele de părtinire. *Printre cei mai utilizați algoritmi de stimulare*, putem preciza: *AdaBoost, Gradient Boosting Decision Trees (GBDT), eXtreme Gradient Boosting (XGBoost)* [81], [86].

2.6.2. Învățare nesupravegheată

În *învățarea nesupravegheată*, algoritmi primesc date neetichetate și extrag tipare, fac corelații din acestea. [42]. Printre cei mai utilizați algoritmi de învățare nesupravegheată: *K-Means, Principal Component Analysis (PCA), Independent Component Analysis (ICA), Hierarchical Clustering and Spectral Clustering (HCSC), Equivalence Class Transformation (ECLAT) și Hierarchical Clustering (HC)* etc [67].

2.6.3. Învățare semi-supravegheată

Există și un model hibrid al ambelor modele numit *învățare semi-supravegheată*, ce combină date neetichetate cu date etichetate. Scopul acestui tip de învățare este să organizeze datele, precum și să facă predicții. Poate fi utilizată în probleme de clasificare și regresie [31], [42].

2.6.4. Strategii de validare și evaluare

Algoritmii utilizați pentru clasificări de date sunt evaluați pe baza matricei de confuzie. Matricea de confuzie este construită aplicând la intrarea unui model antrenat, un set de date pentru care cunoaștem răspunsul corect (utilizăm seturile de date pentru testare). Principalii indicatori utilizați pentru evaluarea performanțelor algoritmilor de clasificare sunt: *True-Positive Rate (TPR)* sau *Recall*, *True-Negative Rate (TNR)*, *Accuracy (ACC)*, *Precision (PR)* [40], [42].

Deoarece algoritmi de regresie prezic o valoare, performanța lor este măsurată prin evaluarea numărului de erori de predicție. Modelele de regresie utilizează pentru evaluarea performanței următorii indicatori: *Mean Absolute Error (MAE)*, *Mean Squared Error (MSE)*, *Root Mean Square Error (RMSE)* și *Scorul R2* [40].

Evaluarea algoritmilor de învățare nesupravegheată este greu de definit, deoarece algoritmi sunt subiectivi, iar evaluarea depinde de tipul aplicațiilor în care sunt utilizați [96].

2.6.5. Parametrii ce rezultă în urma unui sistem de mentenanță predictivă

Modelele de clasificare și regresie ale sistemelor de mentenanță predictivă permit:

- *Predicția timpului rămas din viață (RUL) al unei componente (regresie);*
- *Identificarea tipului de defect (clasificare);*
- *Aflarea probabilității defectării activului în diferite ferestre de timp (clasificare);*
- *Predicția defectelor ce pot apărea într-o fereastră de timp viitoare (fereastră fixă, clasificare).*

Modelele de regresie sunt utilizate pentru predicția duratei de viață utilă rămasă (Remaining Useful Life, RUL), ce este timpul rămas până când activul va eșua. Utilizarea modelelor care se bazează pe învățare automată pentru a estima RUL al unui echipament necesită disponibilitatea datelor istorice ce conțin informații despre procesul de degradare.

Datele disponibile trebuie să fie etichetate iar observațiile trebuie să fie făcute în diferite momente de viață a activului. *Indicele RUL poate fi prezis în funcție de modul în care se comportă la momentul t* [12], [19], [43].

Metodele de clasificare pot fi utilizate pentru clasificare multi-clasă și binară.

Modelele de clasificare binare sunt utilizate pentru predicția defectelor ce pot apărea într-o fereastră de timp viitoare (fereastra este fixă), cea ce duce la predicții mai puțin precise.

În unele abordări *modelele de clasificare multi-clasă* sunt utilizate pentru *predicția probabilității defectării activului în diferite ferestre de timp* corespunzătoare claselor. De exemplu, C_1 poate exprima probabilitatea defectării activului săptămâna aceasta, C_2 poate exprima probabilitatea defectării activului săptămâna viitoare etc [12], [19], [43].

În alte abordări *modelele de clasificare multi-clasă* sunt utilizate pentru *identificarea anumitor tipuri de defecte*. De exemplu C_0 nu corespunde niciunui defect, C_1 poate fi utilizat pentru tipul I de defect etc. Defectele pot fi identificate după ce ele s-au produs sau înainte să se producă [12], [19].

2.7. Concluzii

Spre deosebire de celelalte tipuri de întrețineri, *întreținerea predictivă a devenit standardul dorit al industriei pe majoritatea piețelor. Activitățile de service au loc numai atunci când sunt necesare.*

În cadrul fabricilor inteligente, *tehnologia IoT este un factor esențial pentru întreținerea predictivă.*

Arhitecturile IOT bazate pe EC și FC încep să fie din ce în ce mai utilizate datorită beneficiilor pe care le aduc. Prin utilizarea acestor arhitecturi se asigură o procesare parțială a datelor pe nivelurile edge și fog, reducând astfel cantitatea datelor transferate pe nivelul de cloud. Gradul de confidențialitate a datelor crește datorită faptului că pe nivelul cloud este stocat doar rezultatele procesării datelor de pe nivelurile edge și fog.

Spre deosebire de metodele bazate pe modele fizice, *diagnosticul și prognosticul bazat pe date nu necesită o înțelegere profundă a fizicii* care stă la baza proceselor și nu necesită o cunoaștere completă a comportamentului sistemului. *Pentru predicția și diagnosticul defectelor, majoritatea autorilor de lucrări de specialitate utilizează algoritmi de învățare automată.*

În cadrul algoritmilor de învățare automată, *regresia este utilizată pentru predicția RUL al unui echipament, iar clasificarea poate fi utilizată pentru identificarea tipului de defect, dar și pentru predicția probabilității defectării activului* în diferite ferestre de timp.

2.8. Stabilirea direcțiilor de cercetare

Pornind de la principalul obiectiv stabilit prin titlul acestei teze de doctorat „*Cercetări privind îmbunătățirea sistemelor de mentenanță predictivă utilizând IoT*”, din cercetările făcute în cadrul acestui capitol, s-a constatat că *lucrările de specialitate despre sistemele de mentenanță predictivă ale UAV-urilor propulsate electric sunt limitate.* Astfel, *cercetările următoare s-au concentrat pe îmbunătățirea sistemelor de mentenanță predictivă ale UAV-urilor propulsate electric utilizând IoT.* Pentru realizarea obiectivului propus, au fost tratate următoarele mini obiective:

- *Identificarea metodei ce are performanțele cele mai bune în găsirea valorilor anormale ale setului de date provenit de la senzorii unui sistem de mentenanță predictivă pentru UAV-urile propulsate electric;*
- *Identificarea metodei ce are performanțele cele mai bune în completarea valorilor lipsă ale setului de date provenit de la senzorii unui sistem de mentenanță predictivă pentru UAV-urile propulsate electric;*
- *Realizarea unei metode pentru predicția RUL al acumulatorilor de Li-ion din componența UAV-urilor propulsate electric;*
- *Realizarea unei metode pentru identificarea tipului de defect a motoarelor BLDC din componența UAV-urilor propulsate electric;*
- Conceperea unei arhitecturi pe care ar trebui să o aibă sistemele de mentenanță predictivă pentru UAV-urile propulsate electric ce utilizează IoT;
- *Realizarea practică a unui sistem de mentenanță predictivă care să monitorizeze componentele cu o fiabilitate scăzută precum acumulatorii de Li-ion și motoarele electrice BLDC, pentru a se preveni apariția unui eșec funcțional. Pentru a scădea lungimea de bandă utilă necesară transmiterii datelor în cloud și pentru a crește siguranța și confidențialitatea datelor de la senzori, se dorește să se implementeze o arhitectură bazată pe FC.*

3. Cercetări privind realizarea unui sistem de mentenanță predictivă pentru UAV-uri

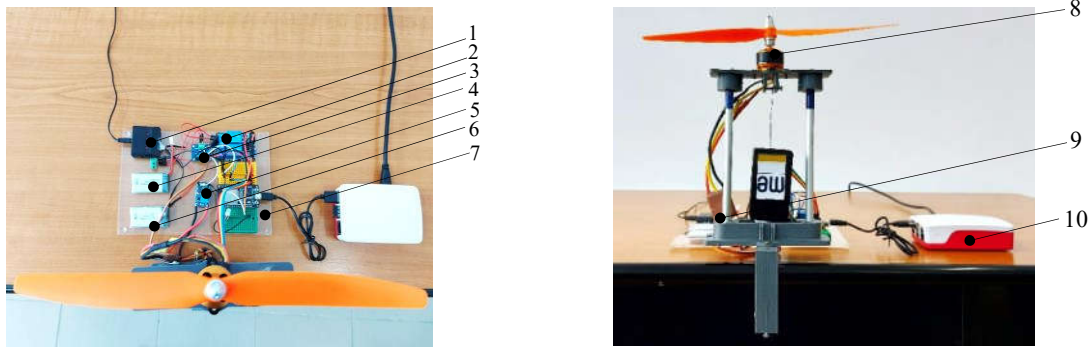
3.1. Proiectarea și realizarea standurilor experimentale utilizate pentru extragerea datelor de antrenare ale modelelor de învățare automată

Standurile experimentale permit studiul degradării acumulatorilor de Li-ion odată cu creșterea numărului de cicluri de încărcare/descărcare precum și analiza funcționării motoarelor BLDC sub influența defectelor. *Prin proiectarea și realizarea celor două standuri experimentale, se urmărește crearea unor baze de date care să permită:*

- Analiza comparativă a performanțelor algoritmilor de tratare a valorilor anormale ce pot fi utilizați în cadrul sistemelor de mentenanță predictivă pentru predicția RUL al acumulatorilor de Li-ion și identificarea tipului de defect a motoarelor BLDC din cadrul UAV-urilor propulsate electric;
- Analiza comparativă a performanțelor algoritmilor de tratare a valorilor lipsă ce pot fi utilizați în cadrul sistemelor de mentenanță predictivă pentru predicția RUL al acumulatorilor de Li-ion și identificarea tipului de defect a motoarelor BLDC din cadrul UAV-urilor propulsate electric;
- Conceperea, utilizând tehnicile de învățare automată a unei metode de predicție a RUL al acumulatorilor de Li-ion din cadrul UAV-urilor propulsate electric;
- Conceperea, utilizând tehnicile de învățare automată a unei metode de identificare cu ajutorul clasificatoarelor a tipului de defect la motoarele BLDC din cadrul UAV-urilor propulsate electric.

Standul experimental utilizat pentru a extrage caracteristicile de degradare a acumulatorilor de Li-ion realizează încărcarea și descărcarea automată a acumulatorilor de Li-

ion. Standul experimental este prezentat în Fig. 9, iar schema electrică după care a fost realizat este prezentată în Fig. 10.



a. vedere de sus: 1 – încărcător acumulatori; 2 – releu; 3 – modul INA219; 4 – acumulator Li-ion 2; 5 – ridicător tensiune MT3608; 6 – acumulator Li-ion 1; 7 – modul NodeMCU-32S.

b. vedere din fata: 8 - motor BLDC A2208; 9 - ESC; 10 - Raspberry Pi 4.

Fig. 9 Standul experimental:

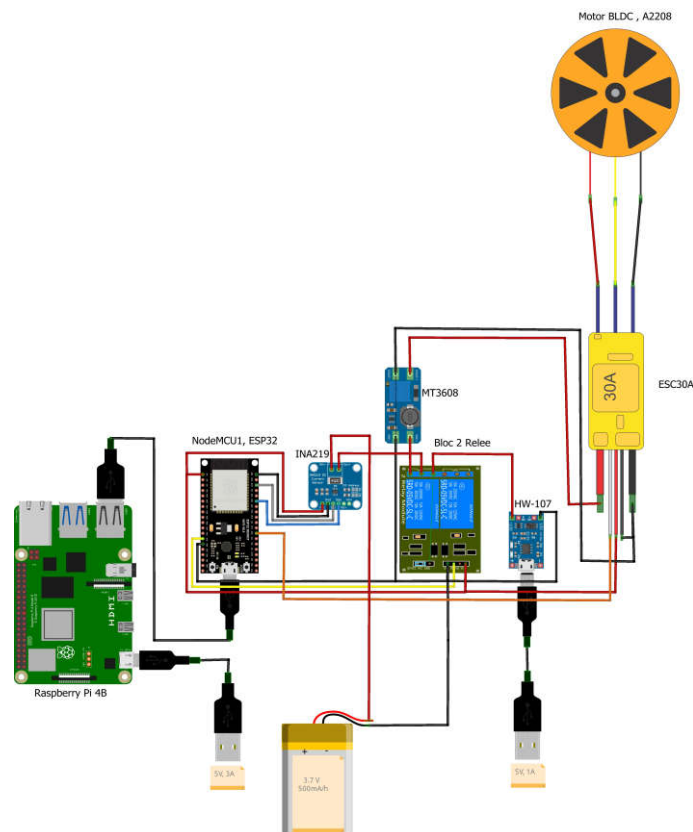


Fig. 10 Schema electrică a standului experimental utilizat pentru extragerea caracteristicilor de degradare a acumulatorilor de Li-ion, *Fritzing*.

În cadrul standului experimental, acumulatorul de Li-ion este încărcat și descărcat repetat. *Pe fiecare ciclu de descărcare a acumulatorilor de Li-ion (Fig. 9, - 4, - 6; Fig. 10). este monitorizată tensiunea și curentul de descărcare apoi este extrasă în timp real capacitatea descărcată.* Datele experimentale au fost generate utilizând acumulatori Li-ion de 3.7 [V] cu o capacitate de 500 [mAh].

În cadrul acestui studiu a fost monitorizată tensiunea și curentul pentru doi acumulatori Li-ion, pe parcursul a 1200 de cicluri de descărcare. Pe parcursul fiecărui ciclu, prin integrare,

a fost calculată în timp real capacitatea descărcată. Pentru acest studiu, s-a considerat că un ciclu constă în încărcare acumulatorilor la 3.7 [V] și descărcare lor până la 3.1 [V].

Standul experimental utilizat pentru monitorizarea motoarelor BLDC permite monitorizarea temperaturii, vibrației, curentului și tensiunii prin intermediul senzorilor, cu scopul de a identifica defectele acestuia. Standul experimental utilizat pentru extragerea caracteristicilor defectelor motoarelor BLDC din cadrul UAV-urilor este prezentat în Fig. 11, iar schema electrică după care a fost realizat este prezentată în Fig. 12.

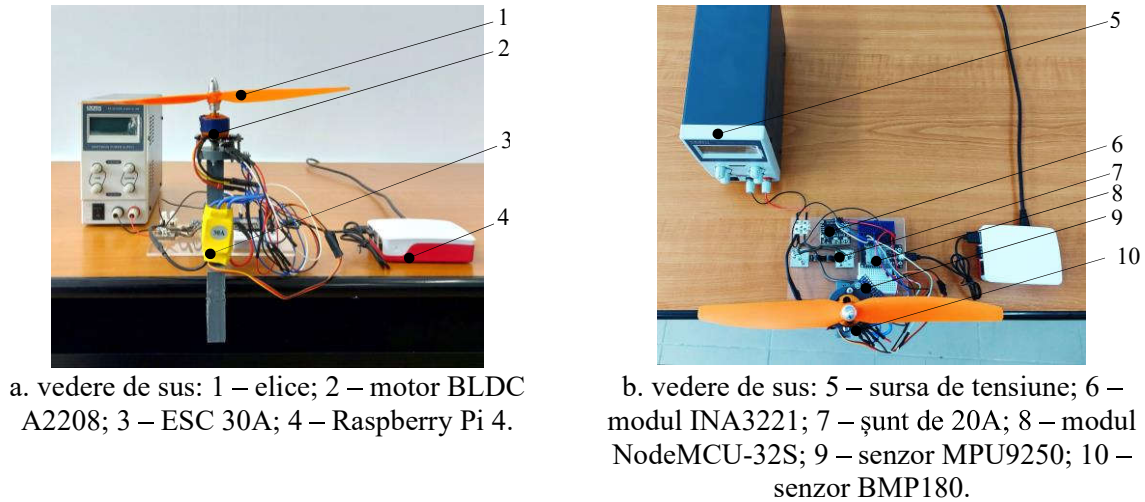


Fig. 11 Standul experimental:

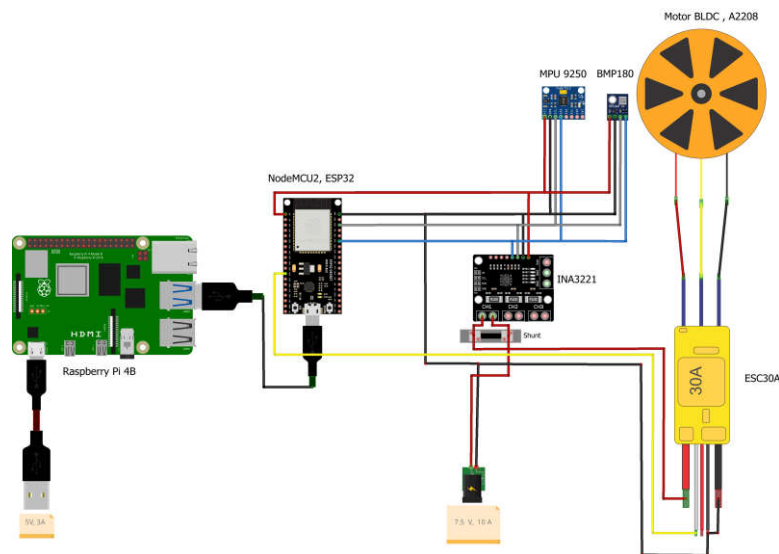


Fig. 12 Schema electrică a standului experimental utilizat pentru extragerea caracteristicilor defectelor motoarelor BLDC din cadrul UAV-urilor, *Fritzing*.

În timpul funcționării motorului electric BLDC, sub influența fiecărei clase analizate (mai multe tipuri de defecte), prin intermediul senzorului MPU9250 (Fig. 11, -9; Fig. 12) a fost monitorizată accelerația pe cele trei axe, prin intermediul modului INA3221 (Fig. 11,-6; Fig. 12) a fost monitorizată tensiunea și curentul ce se închide prin motor, iar prin intermediul senzorului BMP180 (Fig. 11, -10; Fig. 12) a fost monitorizată temperatura.

Pentru acest studiu, au fost create defecțiuni ale elicei, de excentricitate și ale ESC-ului, frecvent întâlnite la motoarele BLDC din componența UAV-urilor. Setul de date a fost împărțit

în patru clase: C_1 - sănătos, C_2 - elice ciobită, C_3 - arbore excentric și C_4 – ESC defect (tranzistor MOSFET FDD8896 defect), (Fig. 13).

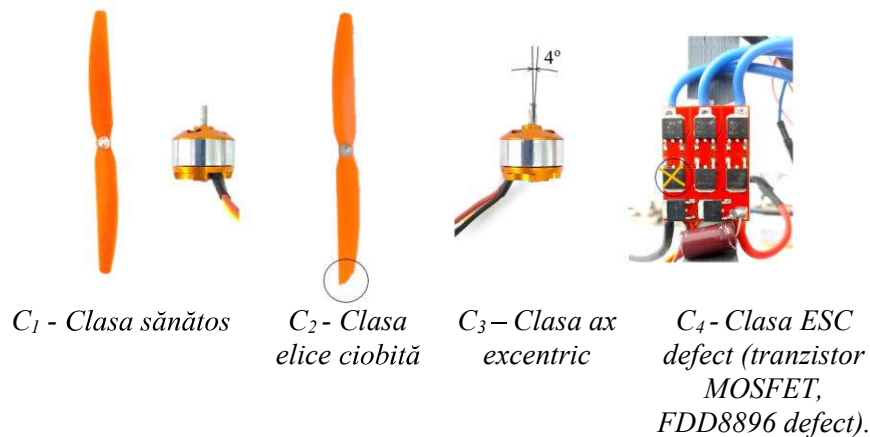


Fig. 13 Clasele analizate:

Fiecare clasă implementată a fost monitorizată cu ajutorul standului experimental. Astfel, au fost generate date experimentale care conțin informații despre fiecare clasă analizată. Pe parcursul monitorizării, motorul a fost utilizat la turații diferite.

3.2. Analiza comparativă a performanțelor algoritmilor de tratare a valorilor anormale, ce pot fi utilizați în cadrul sistemelor de mentenanță predictivă pentru predicția RUL al acumulatorilor de Li-ion și identificarea tipului de defect a motoarelor BLDC ale UAV-urilor propulsate electric

3.2.1. Introducere

Datele colectate de la sistemele de mentenanță predictivă ale UAV-urilor pot să conțină valori anormale.

O valoarea anormală este o observație care se află anormal de departe de restul valorilor ale setului de date analizat. Detectarea valorilor anormale este o problema importantă în analiza unui set de date deoarece rezultatul analizei datelor ar putea fi influențat de valorile anormale din setul de date. *Valorile anormale afectează: variația standard generală a datelor, media generală a datelor, distribuția datelor precum și statisticile setului de date.* Valorile anormale sunt problematice deoarece afectează rezultatul unei analize [87], [88]. Valorile anormale din cadrul seturilor de date ale sistemelor de mentenanță predictivă pot afecta modelul utilizat pentru predicția RUL al unui echipament.

Procesul de detectare al valorilor anormale dintr-un set de date face parte din faza de preprocesare a analizei datelor. Algoritmii de detectare a valorilor anormale se pot baza: pe statistici, pe distanță, pe tehnici de grupare (clustering) și pe densitate [87], [89]. Învățarea automată poate fi utilizată și ea în detectarea valorilor anormale.

G. Xunqiang și colab. realizează o analiză comparativă privind performanța a trei metode de detecție a valorilor anormale. Metodele analizate sunt: Z-score, Boxplot și Median Absolute Deviation (MAD). Analiza a fost realizată pe două seturi de date, cu un conținut informațional diferit. Rezultatele experimentelor au arătat că numărul de valori anormale detectate cu metoda MAD depășește semnificativ pe cele ale metodelor Z-score și Boxplot [90].

3.2.2. Analiza comparativă a algoritmilor utilizați în detecția valorilor anormale

În cadrul acestui studiu, au fost utilizate metodele *IForest*, *AE* și *KNN*, pentru a estima valorile anormale din seturile de date ale unui sistem de mentenanță predictivă din componența UAV-urilor. Seturile de date conțin valori provenite de la standul experimental utilizat pentru extragerea caracteristicilor de degradare a acumulatorilor de Li-ion (senzorii de curent și tensiune) și standul experimental utilizat pentru extragerea caracteristicilor defectelor motoarelor BLDC din cadrul UAV-urilor (senzorii, temperatură, accelerație, curent și tensiune).

IForest este un algoritm care utilizează învățarea automată nesupravegheată ce poate fi utilizat pentru detectarea valorilor anormale. Instanțele normale se vor situa la capătul mai adânc al copacului în timp ce anomaliile sunt situate la baza copacului datorită caracterului lor singular.

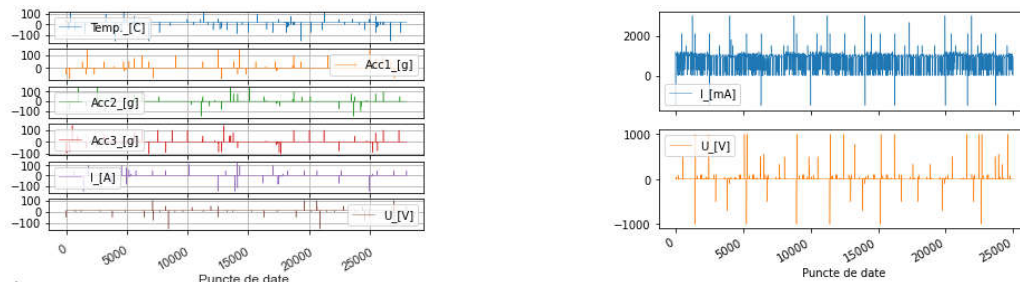
AE este un algoritm de învățare profundă ce are în componență o rețea neuronală care este instruită să reproducă intrarea. Varianta nesupravegheată a lui *AE* din lucrarea [91] a fost utilizată pentru detectarea valorilor anormale. În cadrul procesului de antrenament a variantei nesupravegheate, *AE* învață minimizând eroarea de reconstrucție.

KNN este un algoritm de învățare automată supravegheată frecvent utilizat în probleme de clasificare dar și de regresie [89], [92]. Varianta utilizată de *KNN* ce utilizează învățarea nesupravegheată este întâlnită în lucrarea [93].

Procesul de identificare a valorilor anormale este privit ca fiind o problemă de clasificare binară. În acord cu matricea de confuzie se poate calcula: *ACC*, *PR* și *Recall* [94], [95].

3.2.3. Date experimentale, rezultate și discuții

Pentru acest studiu, au fost utilizate datele provenite de la, standul experimental din Fig. 9 și Fig. 10 și standul experimental prezentat în Fig. 11 și Fig. 12.



a. *Temp* [°C], *Acc1* [g], *Acc2* [g], *Acc3* [g], *I* [A] b. *I* [A], *U* [V] ai setului doi de date, *Python* 3.6.
și *U* [V] ai primului set de date *Python* 3.6;

Fig. 14 Variația parametrilor:

În Fig. 14.a sunt prezentate graficele de variație a parametrilor *Temp* [°C], *Acc1* [g], *Acc2* [g], *Acc3* [g], *I* [A] și *U* [V] din setul de date utilizat pentru monitorizarea motorului BLDC iar Fig. 14.b sunt prezentate graficele de variație a parametrilor *U* [V] și *I* [A] din setul de date utilizat, generat prin monitorizarea descărcării acumulatorilor de Li-ion, ce conține valori anormale. Din Fig. 14 .a și Fig. 14 .b, se pot observa reprezentate valorile care sunt prea depărtate de tendința valorilor normale.

Performanța de a clasifica valorile anormale ale celor trei metode pe cele două seturi de date, a fost analizată în limbajul *Python* 3.6. prin intermediul indicatorilor: *ACC*, *PR*, *Recall* și *AUC*. Rezultatele indicatorilor de performanță sunt prezentate în Tab. 2 și Tab. 3.

Prin analiza rezultatelor obținute pentru ACC, PR, Recall și AUC, prezentate în cadrul Tab. 2 se constată că valorile anormale sunt identificate de metoda KNN cu o acuratețe mai bună, iar metoda AE oferă rezultatele cele mai nesatisfăcătoare, dar apropiate de metoda IForest.

Tab. 2 Performanțele algoritmilor de identificare a valorilor anormale asupra datelor rezultate din monitorizarea motoarelor electrice BLDC, Python 3.6.

Indicator de performanță		IForest	AE	KNN
ACC	Temp.	0.9030	0.9091	0.9975
	Acc1, Acc2, Acc3	0.8891	0.8946	0.9043
	I, U	0.9012	0.9097	0.9872
	Medie ACC	0.8978	0.9045	0.9630
PR	Temp.	0.0476	0.0506	0.6666
	Acc1, Acc2, Acc3	0.0854	0.0851	0.0977
	I, U	0.0355	0.0387	0.2222
	Medie PR	0.0561	0.0581	0.3288
Recall	Temp.	1.0	1.0	1.0
	Acc1, Acc2, Acc3	0.9444	0.8888	0.9444
	I, U	1.0	1.0	1.0
	Medie Recall	0.9814	0.9629	0.9814
AUC	Temp.	0.9513	0.9543	0.9987
	Acc1, Acc2, Acc3	0.916	0.8917	0.9241
	I, U	0.9504	0.9547	0.9936
	Medie AUC	0.9394	0.9336	0.9721

Tab. 3 Performanțele algoritmilor de identificare a valorilor anormale asupra datelor utilizate pentru monitorizarea acumulatorilor de Li-ion, Python 3.6.

Indicator de performanță		IForest	AE	KNN
ACC	I, U	0.9169	0.9109	0.9066
PR	I, U	0.2359	0.1976	0.1828
Recall	I, U	0.9767	0.7906	0.7441
AUC	I, U	0.9460	0.8524	0.8276

Prin analiza rezultatelor obținute pentru ACC, PR, Recall și AUC, prezentate în cadrul Tab. 3, se constată că metoda IForest identifică valorile anormale cu o acuratețe mai bună iar metoda KNN oferă rezultatele cele mai nesatisfăcătoare, dar apropiate de metoda AE.

3.2.4. Concluzii

Rezultatele experimentale arată că algoritmul KNN este mai eficient pe primul set de date iar algoritmul IForest este mai eficient pe cel deal doilea set de date. Cu toate acestea, pe alte seturi de date cu un alt conținut informațional performanța lor poate fi diferită. Performanța fiecărei metode variază în funcție de tipul datelor aplicate și de scopul detectării valorilor anormale.

3.3. Analiza comparativă a performanțelor algoritmilor de tratare a valorilor lipsă ce pot fi utilizați în cadrul sistemelor de mentenanță predictivă pentru, predicția RUL al acumulatorilor de Li-ion și identificarea tipului de defect a motoarelor BLDC din cadrul UAV-urilor propulsate electric.

3.3.1. Introducere

Datele colectate de la senzorii sistemelor de mentenanță predictivă ale UAV-urilor pot să conțină valori lipsă. Seturile de date incomplete sunt o problemă frecvent întâlnită în domeniul IoT.

Predicția RUL a unei componente poate fi afectată de seturile de date ce prezintă valori lipsă. Din acest motiv este necesar să găsim cea mai bună metodă pentru a estima valorile lipsă astfel încât datele analizate să reflecte mediul real [96].

Un set de date ce conține valori lipsă poate cauza distorsiuni din cauza diferențelor dintre datele observate și cele neobservate. Valorile lipsă din seturile de date se pot datora și valorilor anormale, majoritatea algoritmilor înlocuiesc valorile anormale cu valori lipsă. Seturile de date incomplete pot duce la rezultate care sunt diferite de cele care ar fi fost obținute dintr-un set de date complet [97].

F. Yingpeng și colab. prezintă o analiză comparativă a metodelor precum, RF, SVR, ANN, Mean și Multiple Imputation (MI), ce pot fi utilizate pentru înlocuirea valorilor lipsă. Analiza a fost efectuată pe o baza de date gratuită denumită UNSODA, în care sunt analizate proprietățile solului. Studiul a scos în evidență că metodele RF și MI au avut performanțele cele mai bune [98].

3.3.2. Analiza comparativă a algoritmilor utilizați pentru completarea valorilor lipsă

În cadrul acestui studiu, au fost utilizate patru metode pentru a estima valorile lipsă din seturile de date ale unui sistem de mentenanță predictivă din componența UAV-urilor, iar performanțele acestor metode au fost comparate. Cele patru metode sunt: metoda interpolării, valoarea medie, metoda KNN și RF. Seturile de date conțin datele provenite de la, standul experimental utilizat pentru extragerea caracteristicilor de degradare a acumulatorilor de Li-ion (senzorii: curent și tensiune) și standul experimental utilizat pentru extragerea caracteristicilor defectelor motoarelor BLDC din cadrul UAV-urilor (senzorii: temperatură, accelerație, tensiune și curent).

Interpolarea este o metodă frecvent utilizată pentru completarea valorilor lipsă, deoarece generează noi puncte de date în intervalul seriei de date cu valori lipsă.

Completarea valorilor lipsă utilizând valoarea medie, înlocuiește valorile lipsă cu o media tuturor datelor din seria de date.

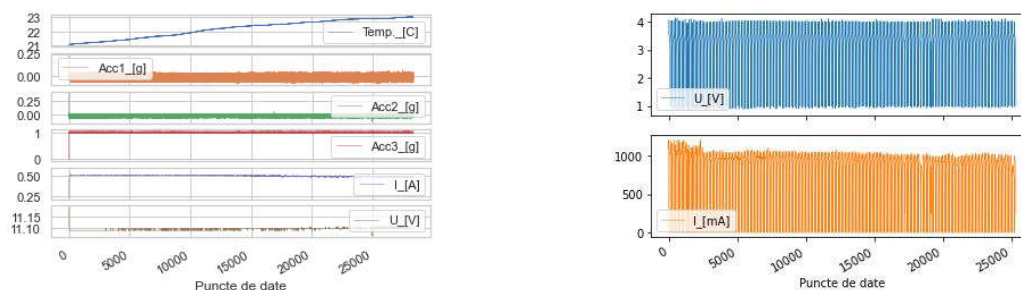
KNN este un algoritm de învățare automată supravegheată utilizat în probleme de clasificare dar și regresie. În ceea ce privește găsirea valorilor lipsă este utilizată varianta sa nesupravegheată [76], [99].

RF se bazează pe ansambluri ce funcționează pe principiul că un grup de modele de învățare automată slabe vor forma un model puternic [72].

Algoritmii de completarea a valorilor lipsă pot fi evaluați utilizând indicatorii folosiți pentru evaluarea algoritmilor de regresie din cadrul modelelor de învățare automată. Astfel, cei mai frecvent utilizați indicatori sunt: MSE, RMSE și MAE [97], [100].

3.3.3. Date experimentale, rezultate și discuții

În cadrul acestei secțiuni au fost utilizate datele ce provin de la două standuri experimentale: primul stand experimental este prezentat în Fig. 9 și Fig. 10 iar al doilea stand experimental este prezentat în Fig. 11 și Fig. 12. Performanța algoritmilor utilizați pentru generarea valorilor lipsă a fost analizată în limbajul Python 3.6.



a. $Temp$ [°C], $Acc1$ [g], $Acc2$ [g], $Acc3$ [g], I [A] și U [V] al primului set de date, Python 3.6; b. I [A] și U [V] al celui de-al doilea set de date, Python 3.6.

Fig. 15 Variația parametrilor:

Graficele de variație a parametrilor $Temp$ [°C], $Acc1$ [g], $Acc2$ [g], $Acc3$ [g], I [A] și U [V] pe setul de date obținut din *monitorizarea funcționării motoarelor BLDC* sunt prezentate în Fig. 15.a iar graficele de variație a parametrilor I [A] și U [V] pe setul de date obținut din *monitorizarea acumulatorilor de Li-ion* sunt prezentate în Fig. 15.b

Deoarece, ar fi fost dificil de evaluat performanța metodelor de înlocuire a valorilor lipsă atunci când valorile reale sunt necunoscute, în seturile de date utilizate au fost introduse pe poziții alese aliaitor valori lipsă, cu scopul de a evalua capacitatea strategiilor de înlocuire a valorilor lipsă.

În Tab. 4 sunt prezentate performanțele algoritmilor de înlocuire a valorilor lipsă asupra datelor generate prin monitorizarea motorului electric BLDC iar în Tab. 5 sunt prezentate performanțele algoritmilor de înlocuire a valorilor lipsă asupra datelor generate prin monitorizarea acumulatorilor de Li-ion pe parcursul a 200 de cicluri de funcționare.

Din analiza rezultatelor obținute pentru MAE, MSE și RMSE, prezentate în cadrul Tab. 4 se constată că atunci când valorile lipsă sunt completate utilizând metoda interpolării liniare, valorile obținute sunt cele mai apropiate de valorile reale, iar completarea valorilor lipsă cu valoarea medie oferă rezultatele cele mai depărtate față de valorile reale.

Tab. 4 Performanțele algoritmilor de înlocuire a valorilor lipsă asupra datelor generate prin monitorizarea motoarelor electrice BLDC, Python 3.6.

Indicator de performanță	Interpolare l.	Media	KNN	RF	
MAE	$Temp.$ [°C]	0.0040	0.0988	0.0834	0.0780
	$Acc1$ [g]	0.0206	0.0246	0.0128	0.0090
	$Acc2$ [g]	0.0152	0.0134	0.0119	0.0106
	$Acc3$ [g]	0.0145	0.0195	0.0079	0.0082
	I [A]	0.0006	0.0015	0.0006	0.0004
	U [V]	0.0006	0.0061	0.0054	0.0011
	Medie MAE	0.0092	0.0273	0.0203	0.01792
MSE	$Temp.$ [°C]	0.0000	0.0135	0.0122	0.0094
	$Acc1$ [g]	0.0007	0.0009	0.0002	0.0001
	$Acc2$ [g]	0.0003	0.0002	0.0002	0.0001
	$Acc3$ [g]	0.0002	0.0004	0.0001	0.0001

	I [A]	0.0000	5.2582	0.0000	0.0000
	U [V]	0.0000	0.0002	0.0003	0.0000
	Medie MSE	0.0002	0.0025	0.0022	0.0016
RMSE	$Temp.$ [°C]	0.0055	0.1164	0.1100	0.0972
	$Acc1$ [g]	0.0271	0.0304	0.0172	0.0128
	$Acc2$ [g]	0.0196	0.0165	0.0155	0.0136
	$Acc3$ [g]	0.0156	0.0209	0.0123	0.0119
	I [A]	0.0015	0.0022	0.0000	0.0013
	U [V]	0.0015	0.0171	0.0174	0.0023
	Medie RMSE	0.0118	0.0339	0.0291	0.0232

Tab. 5 Performanțele algoritmilor de înlocuire a valorilor lipsă asupra datelor provenite din monitorizarea proceselor de descărcare a acumulatorilor de Li-ion, Python 3.6.

Indicator de performanță		Interpolare	Media	KNN	RF
MAE	I [A]	1.6145	74.9881	25.3267	24.9732
	U [V]	0.0059	0.1751	0.1209	0.0879
	Medie MAE	0.8102	37.5816	12.7156	12.5276
MSE	I [A]	13.4588	11486.1273	1906.2743	1816.2647
	U [V]	0.0006	0.0447	0.0245	0.0151
	Medie MSE	6.7297	5743.0860	953.14721	908.1385
RMSE	I [A]	3.6686	107.1733	43.6609	42.6176
	U [V]	0.0253	0.2115	0.1568	0.1230
	Medie RMSE	1.8469	53.6924	21.9013	21.3645

Prin analiza rezultatelor obținute pentru **MAE**, **MSE** și **RMSE**, prezentate în cadrul Tab. 5, se constată că atunci când valorile lipsă sunt completate utilizând **metoda de interpolării**, valorile obținute sunt cele mai apropiate de valorile reale iar **metoda de completare a valorilor lipsă cu media** acestora oferă rezultatele cele mai depărtate față de valorile reale.

3.3.4. Concluzii

Prin analiza valorilor indicatorilor de performanță, cea mai bună metodă de completare a valorilor lipsă s-a dovedit a fi **metoda de interpolare liniară**, iar completarea valorilor lipsă cu **media** datelor oferă rezultatele cele mai depărtate de valorile dorite.

Metoda ce are rezultatele cele mai bune pe aceste date s-ar putea să nu producă același rezultat pe un alt set de date cu un conținut informațional diferit.

Pentru a alege metoda optimă de completare a valorilor lipsă, trebuie să se studieze performanța metodelor candidate pe un set de date cu același conținut informațional.

3.4. Predicția RUL al acumulatorilor de Li-ion din cadrul UAV-urilor propulsate electric utilizând tehnicile de învățare automată

3.4.1. Introducere

Datorită avantajelor pe care le posedă precum, putere mare/densitate energetică, număr mare al ciclurilor de încărcare-descărcare, rată scăzută de autodescărcare, interval larg al temperaturilor de funcționare .etc, acumulatorii de Li-ion sunt utilizați într-o gamă largă de aplicații [101], [102].

Predicția RUL și predicția capacității acumulatorilor cu Li-ion, utilizați în cadrul UAV-urilor este un mijloc important pentru a preveni o serie de probleme precum pierderea

autonomiei ce poate duce la apariția unor accidente sau defecțiuni. Având informații precum capacitatea și valoarea RUL al acumulatorilor de Li-ion, persoanele responsabile pot monitoriza performanța celulelor și pot programa în avans operațiile de întreținere sau de înlocuire.

Degradarea acumulatorilor de Li-ion este influențată de chimia bateriei precum și de condițiile de funcționare [103].

Metodele de predicție a capacității acumulatorilor pot fi împărțite în: **metode bazate pe model, metode bazate pe date și metode hibride**. Metodele bazate pe date nu necesită un model matematic pentru a descrie dinamica procesului de îmbătrânire a acumulatorilor Li-ion, ele consideră bateria ca o cutie neagră [104]. Metodele bazate pe date utilizează învățarea automată supravegheată. Metodele hibride folosesc pe ambele metode [102], [105].

A. Majid și colab. au propus o metodă în [103] ce utilizează un model ce ia în calcul și procesele chimice din bateriile Li-ion pentru a estima RUL. Ca și concluzie, autorii precizează că dacă se menține bateria la un nivel de încărcare de 70% se poate obține o creștere a duratei de viață a bateriei de aproximativ 55% cu costul unei reduceri de 23% a energiei electrice stocate.

3.4.2. Metodologia de predicție a RUL utilizând învățarea automată

Metodele de învățare automată pot fi utilizate pentru predicția ciclurilor din viață rămase ale unui acumulator de Li-ion. Acestea pot învăța tipare din date pentru a face predicții.

În cadrul acestui studiu, a fost utilizată o metodă bazată pe date ce utilizează învățarea automată pentru predicția ciclurilor din viață rămase ale unei acumulator de Li-ion din componența UAV-urilor.

În Fig. 16, sunt prezentați pașii necesari ce trebuie utilizați atunci când utilizăm învățarea automată pentru predicția ciclurilor rămase din viața. Metoda utilizează date istorice pentru a face predicții. Predicția se va face în următorii pași:

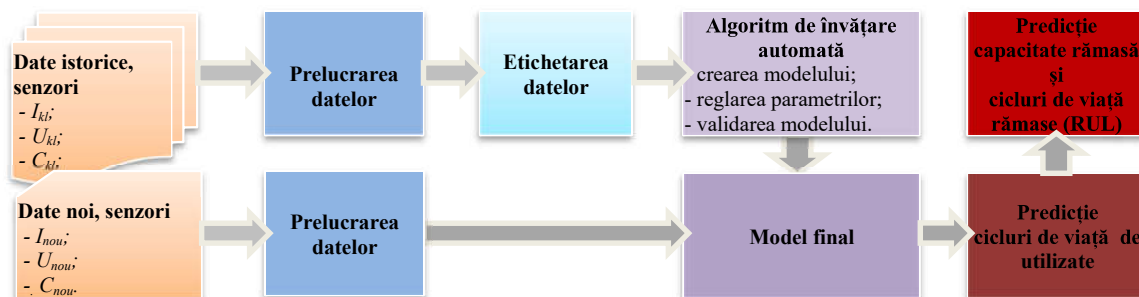


Fig. 16 Diagrama procesului utilizat pentru predicția RUL în timp real.

În cadrul primului pas se vor colecta datele istorice de la senzorii utilizați pentru monitorizarea acumulatorului. Datele utilizate în acest studiu au fost obținute prin studierea procesului de degradare a doi acumulatori Li-ion. În pasul doi, datele colectate vor fi preprocesate. În pasul trei, utilizând datele istorice, va fi antrenat modelul. În pasul patru se va face predicția numărului de cicluri rămase și a capacității rămase.

În cadrul acestui studiu, au fost studiate performanțele mai multor modele de învățare automată realizate cu ajutorul, SVMR, MLR și RF, în a estima RUL.

SVM sunt algoritmi de învățare automată supravegheată ce pot fi utilizați atât în probleme de clasificare cât și în probleme de regresie. În domeniul regresiei, metoda poartă numele de Mașini de regresie cu vectori suport (Support Vector Machine Regression, SVMR) [106].

MLR sunt tehnici de învățare automată parametrică intens utilizate în statistică ce utilizează învățarea automată supravegheată [107]. Deși algoritmul este simplu, în cele mai multe cazuri, produce rezultate satisfăcătoare și foarte rapide datorită formei sale simple.

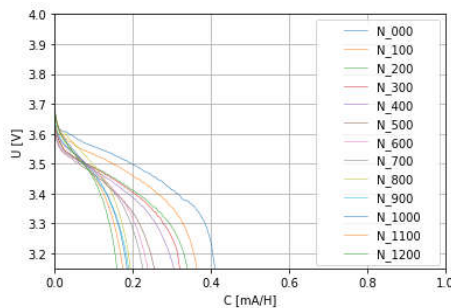
RF este un algoritm de învățare automată supravegheată, bazat pe ansambluri [74], [108]. Un model de regresie ce utilizează RF are în componență mai mulți DT care nu sunt legați unul de celălalt.

Performanțele algoritmilor de învățare automată utilizați în probleme de regresie sunt cel mai frecvent evaluați prin intermediul indicatorilor: MSE, RMSE, MAE și R².

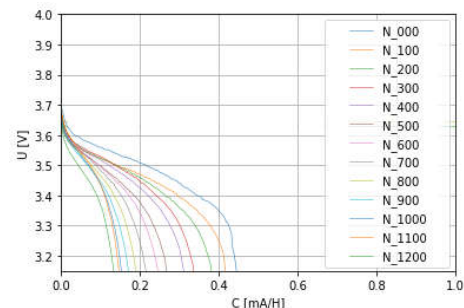
3.4.3. Setup experimental

În cadrul acestui studiu, au fost utilizate datele provenite de la standul experimental utilizat pentru a extrage caracteristicile de degradare a acumulatorilor de Li-ion, prezentat în cadrul Fig. 9 și Fig. 10.

Datele experimentale au fost generate utilizând acumulatori Li-ion de 3.7 [V] cu o capacitate de 500 [mAh]. Această secțiune prezintă datele obținute din monitorizarea bateriilor Li-ion, care au fost utilizate pentru antrenarea modelelor de învățare automată. Datele utilizate pentru antrenarea modelelor au fost reprezentate în Fig. 17- Fig. 18.

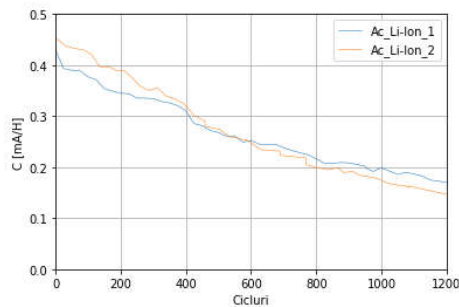


a. acumulatorului Li-ion 1, Python 3.6;

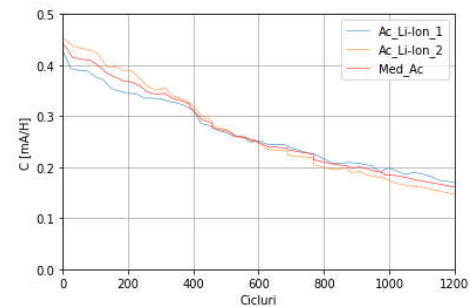


b. acumulatorului Li-ion 2, Python 3.6.

Fig. 17 Variația tensiunii în raport cu capacitatea descărcată pentru fiecare ciclu de descărcare analizat a:



a. acumulatorul Li-ion 1 și 2, Python 3.6



b. acumulatorul Li-ion 1 și 2 precum și media valorilor lor, Python 3.6.

Fig. 18 Variația capacității descărcată în raport cu ciclurile de descărcare pentru:

Din Fig. 18 se constată că, odată cu creșterea numărului de cicluri de încărcare descărcare, capacitatea scade. Capacitatea scade pe parcursul ciclurilor de funcționare datorită procesului de îmbătrânire. După aproximativ 800 de cicluri de încărcare descărcare capacitatea acumulatorilor de Li-ion scade aproximativ la jumătate.

3.4.4. Rezultate și discuții

Metoda propusă a fost implementată utilizând *Python 3.6*. *Au fost comparate performanțele a trei algoritmi de învățarea automată, SVMR, MLR și RF ce au fost utilizați pentru predicția numărului de cicluri de încărcare descărcare al acumulatorilor cu scopul de a extrage numărul de cicluri rămase.*

Tab. 6 Performanța modelelor utilizate pentru predicția ciclurilor de viață a acumulatorilor de Li-ion, *Python 3.6.*

Indicator de performanță	SVMR	MLR	RF
<i>MAE</i>	1.02	67.21	1.53
<i>MSE</i>	51.02	87.25	63.77
<i>RMSE</i>	7.14	7614.10	7.98
R^2	0.99	0.93	0.99

Prin analiza rezultatelor obținute pentru MAE, MSE, RMSE și R^2 , prezentate în Tab. 6 se constată că atunci când pentru predicția ciclurilor de viață scurse ale acumulatorilor de Li-ion sunt utilizați algoritmi SVMR și RF, se obțin performanțele cele mai bune în predicția indicelui RUL, valorile obținute sunt cele mai apropiate de valorile reale. Din Tab. 6, valorile pentru Scorul R^2 ne spun că variabilele predicatoare explică 99% (SVMR, RF) și 93% (MLR) din variația variabilei răspuns. Acest studiu arată că metoda are predicții mai realiste sau similare cu alte metode existente, cum ar fi [109], [110] și [111] ce estimează RUL al acumulatorilor ce sunt utilizați în diverse domenii.

3.4.5. Concluzii

În urma acestui studiu, se constată că modelul prezintă predicții similare sau mai realiste decât cele ale altor modele ce estimează RUL al acumulatorilor ce sunt utilizați în diverse domenii. Metoda poate fi implementată în cadrul unui sistem de mentenanță predictivă din componența UAV-urilor, pentru predicția indicelui RUL al acumulatorilor de Li-ion. Capacitatea de generalizare a modelului va crește dacă modelul va fi antrenat cu alte date de antrenament care vor conține caracteristicile de descărcare a unui număr mai mare de acumulatori.

3.5. Identificarea tipului de defect a motoarelor BLDC din cadrul UAV-urilor, utilizând tehnicile de învățare automată

3.5.1. Introducere

UAV-urile sunt din ce în ce mai utilizate în diverse domenii având scopuri precum: supraveghere, agrement, transport, monitorizarea mediului, căutare și salvare, cartografiere, cercetarea zonelor afectate de dezastre, precum și pentru identificarea țintelor și luptă. Accidentele legate de UAV-uri au crescut proporțional cu numărul acestora [112], [113], [114].

Defecțiunile frecvent observate la motoarele UAV-uri includ, defecțiunile la elice, defecțiunile de excentricitate precum și alte defecțiuni electrice. Funcționarea defectuoasă a UAV-urilor care apare în timpul zborului trebuie identificată și diagnosticată din timp. Unele defecțiuni pot duce la căderea necontrolată a UAV-ului, astfel încât poate fi o amenințare pentru oameni sau obiecte [114], [115].

Din motivele enumerate mai sus, este necesar să se conceapă metode care să poată identifica defectele motoarelor BLDC din componența UAV-urilor propulsate electric. Detectarea stărilor anormale ale motoarelor UAV-urilor poate îmbunătăți siguranța de

exploatare și poate duce la o durată de viață mai lungă. În cadrul acestui studiu este propusă o metodă ce poate identifica 3 tipuri de defecte ale motoarelor BLDC de la UAV-uri.

O. Yaman Yol și F. Altinors propun o metodă ce permite identificarea defectelor motoarelor BLDC din componența UAV-urilor, ce se bazează pe analiza semnalelor sonore. Semnalele sonore au fost colectate de la motoarele BLDC a mai multor tipuri UAV-uri (helicopter, duocopter, tricopter și quadcopter). Precizia modelelor depindea de tipul de UAV analizat și este între 90.53% (Quadcopter) și 100% (pentru elicopter și duocopter) [115].

3.5.2. Metodologia de identificare a tipului de defect a motoarelor BLDC

În cadrul acestui studiu, pentru identificarea tipului de defect a motoarelor BLDC de la UAV-uri, a fost utilizată o metoda bazată pe date, ce utilizează date istorice provenite din monitorizarea funcționării sub influența a diferitor clase de defecte. Datele istorice au fost obținute prin măsurarea cu ajutorul senzorilor, a vibrației, temperaturii, curentului și tensiunii, atunci când motorul BLDC funcționează sub influența unor clase precum Sănătos, Elice ciobită, Ax excentric și ESC defect, care sunt cele mai întâlnite defecte la UAV-uri.

Diagrama procesului utilizat pentru identificarea în timp real a tipului de defect a motoarelor BLDC este prezentată în Fig. 19.

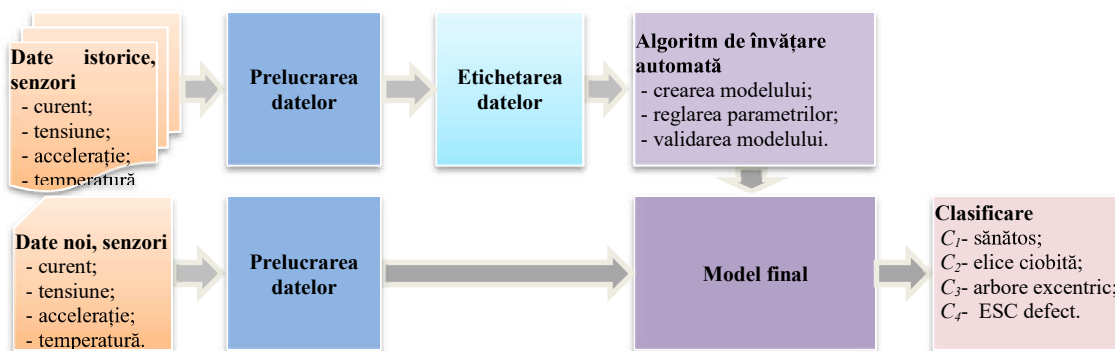


Fig. 19 Diagrama procesului utilizat pentru identificarea tipului de defect a motoarelor BLDC.

Modelul a fost antrenat utilizând date istorice. Acestea conțin informații despre defectele analizate. O dată ce modelul a fost creat, acesta poate fi utilizat pentru a clasifica defecte.

Problema de clasificare a defectelor conține clasele: $C_1 - S$, clasă sănătos; $C_2 - D_{EL}$, clasa elice ciobită; $C_3 - D_{A_E}$, clasa arbore excentric; $C_4 - D_T$, clasa ESC defect (tranzistor MOSFET FDD8896 defect). Dacă o problemă de clasificare are mai mult de două rezultate, atunci sunt utilizați clasificatorii cu mai multe clase.

Performanțele algoritmilor precum, KNN, SVM și BN au fost comparați, cu scopul de a identifica algoritmul ce are performanța cea mai bună în a clasifica defectele motoarelor electrice BLDC.

KNN este un algoritm de învățare automată supravegheată neparametric, frecvent utilizat în probleme de clasificare dar și regresie. O instanță de probă este clasificată prin votul majoritar al vecinilor săi k , astfel instanței de probă i se va atribui clasa ce mai frecvent întâlnește în cazul vecinilor săi k [77].

SVM sunt tehnici de învățare automată des utilizate în probleme de clasificare dar și de regresie, fiind mai utilizate în probleme de clasificare. SVM-urile pot fi folosite în clasificarea datelor liniare, neliniare și multi-clasă [116], [117].

BN este un clasificator probabilistic bazat pe teorema lui Bayes care presupune că fiecare caracteristică depinde doar de clasă. În ciuda simplității sale, clasificatorul bayesian naiv produce rezultate foarte bune și este intens utilizat deoarece depășește adesea metodele de clasificare mai sofisticate. Este rapid din punct de vedere computațional, simplu de implementat și funcționează bine pe seturi de date cu o dimensionalitate ridicată.

Măsurarea performanței unui clasificator multi-clasă este asemănătoare cu măsurarea performanței unui clasificator binar. *Principalii indicatori utilizați prin care se poate măsura performanțele clasificatorilor sunt: ACC , $PR_{clasa\ i}$, $Recall_{clasa\ i}$ și $F1\ Score_{clasa\ i}$. Precizia, Sensibilitatea și Scorul F1 se calculează separat pentru fiecare clasă. Pentru a exprima performanța tuturor claselor cu o singură valoare, se utilizează: $PR_{macro\ avg}$, $Recall_{macro\ avg}$ și $F1\ Score_{macro\ avg}$.*

3.5.3. Setup experimental

Această secțiune conține datele obținute din monitorizarea funcționării motorului BLDC, care au fost utilizate pentru antrenarea modelelor de învățare automată.

În cadrul acestui studiu au fost utilizate datele provenite de la standul experimental utilizat pentru monitorizarea parametrilor motoarelor BLDC, prezentat în cadrul Fig. 11 și Fig. 12. Standul experimental permite monitorizarea, temperaturii, vibrației, curentului și tensiunii motorului BLDC prin intermediul senzorilor, sub influența defectelor analizate. Setul de date utilizat conține informații privind variația accelerației, temperaturii, curentului, tensiunii și turației, atunci când motorul BLDC funcționează sub influența unor defecte precum: sănătos, elice ciobită, ax excentric și ESC defect (tranzistor MOSFET FDD8896 defect).

O reprezentare a variației T [°C], I [mA], $Mag. Acc.$ [g]. pentru clasele analizate, la turațiile de, 1000 [RPM], 2000 [RPM] și 4000 [RPM] este prezentată în cadrul Fig. 20 – Fig. 28, în care au fost utilizate notațiile: C_1-S , clasă sănătos; C_2-D_{EL} , clasa elice ciobită; $C_3-D_{A_E}$, clasa arbore excentric; C_4-D_T , clasa ESC defect (tranzistor MOSFET, FDD8896 defect).

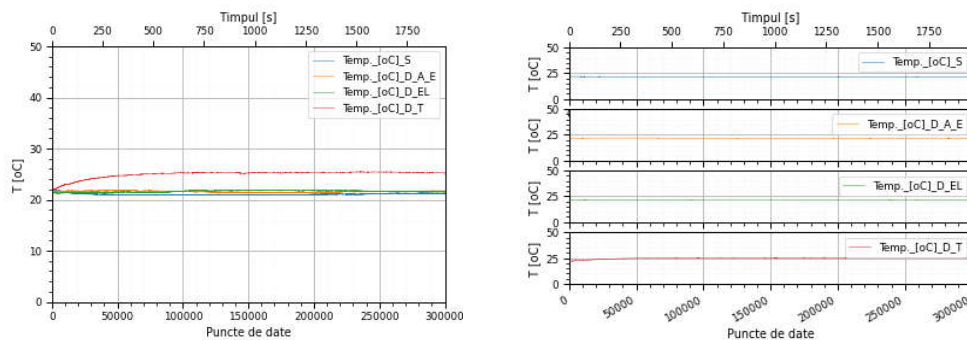


Fig. 20 Variația T [°C] pentru clasele analizate pe parcursul celor 300000 de puncte de date achiziționate, la viteza de rotație 1000 [RPM], Python 3.6.

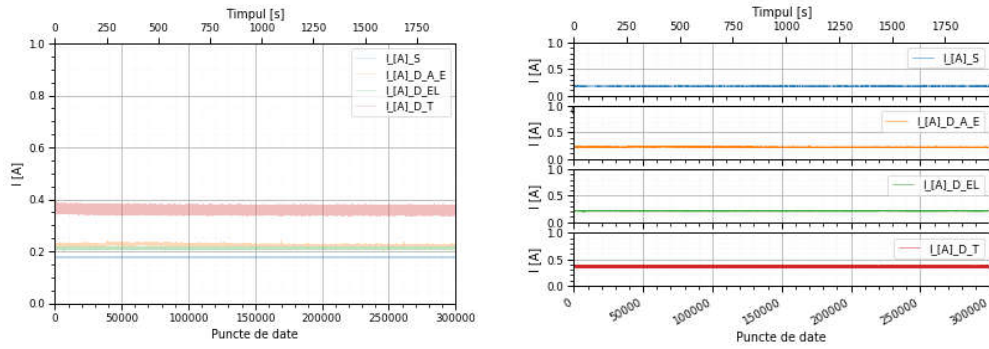


Fig. 21 Variația I [A] pentru clasele analizate pe parcursul celor 300000 de puncte de date achiziționate, la viteza de rotație 1000 [RPM], Python 3.6.

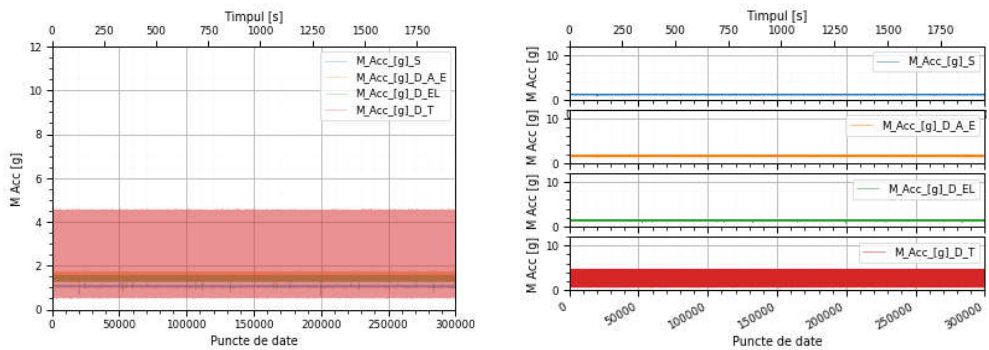


Fig. 22 Variația $Mag. Acc.$ [g] pentru clasele analizate pe parcursul celor 300000 de puncte de date achiziționate, la viteza de rotație 1000 [RPM], Python 3.6.

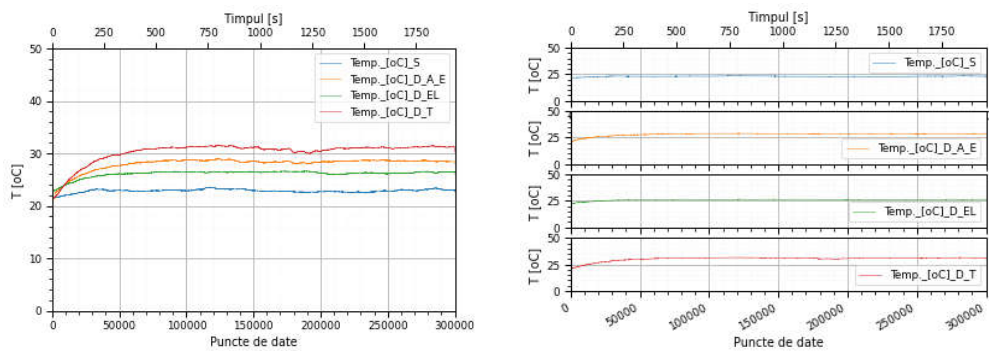


Fig. 23 Variația T [°C] pentru clasele analizate pe parcursul celor 300000 de puncte de date achiziționate, la viteza de rotație 2000 [RPM], Python 3.6.

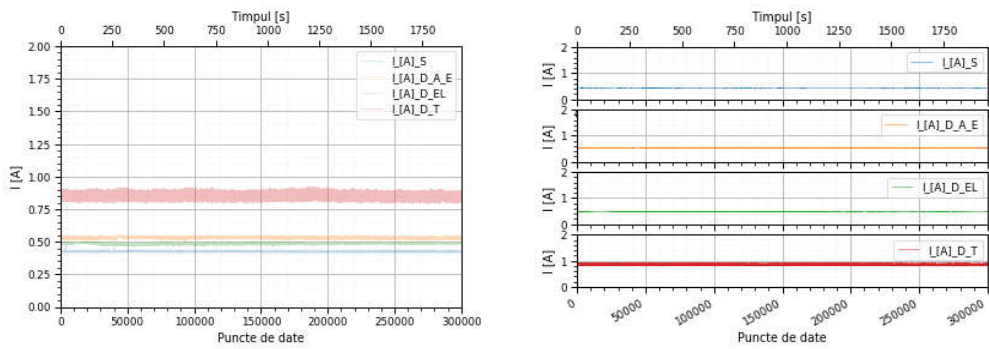


Fig. 24 Variația I [A] pentru clasele analizate pe parcursul celor 300000 de puncte de date achiziționate, la viteza de rotație 2000 [RPM], Python 3.6.

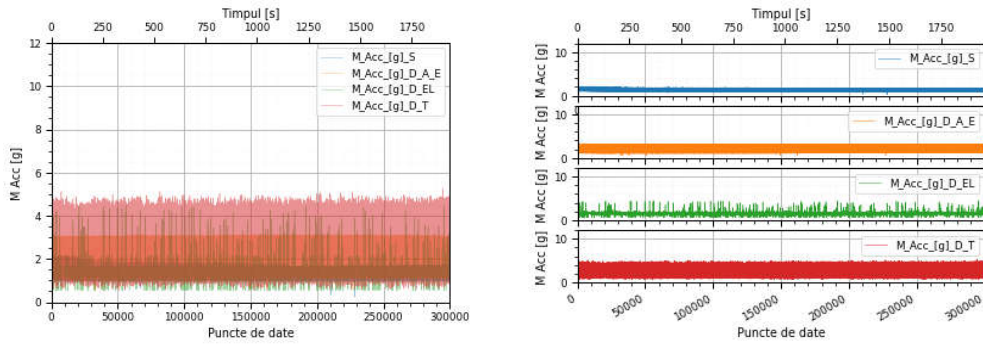


Fig. 25 Variația *Mag. Acc.* [g] pentru clasele analizate pe parcursul celor 300000 de puncte de date achiziționate, la viteza de rotație 2000 [RPM], *Python 3.6*.

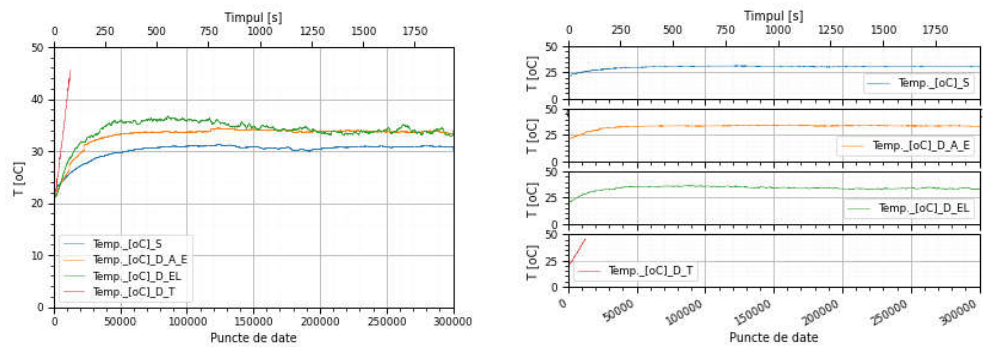


Fig. 26 Variația *T* [°C] pentru clasele analizate pe parcursul celor 300000 de puncte de date achiziționate, la viteza de rotație 4000 [RPM], *Python 3.6*.

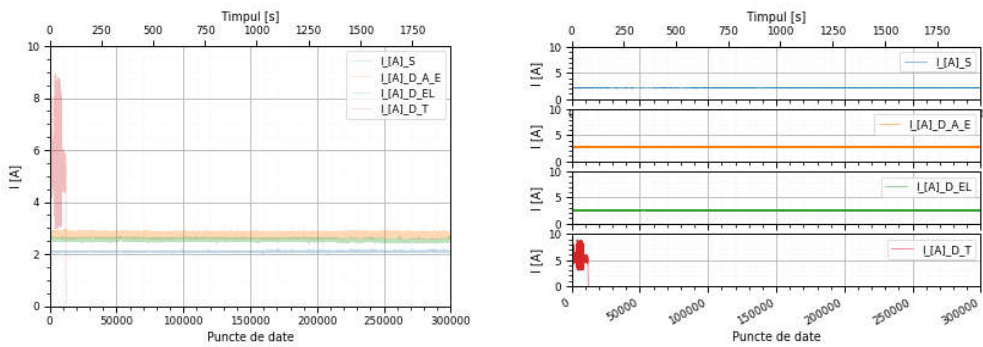


Fig. 27 Variația *I* [A] pentru clasele analizate pe parcursul celor 300000 de puncte de date achiziționate, la viteza de rotație 4000 [RPM], *Python 3.6*.

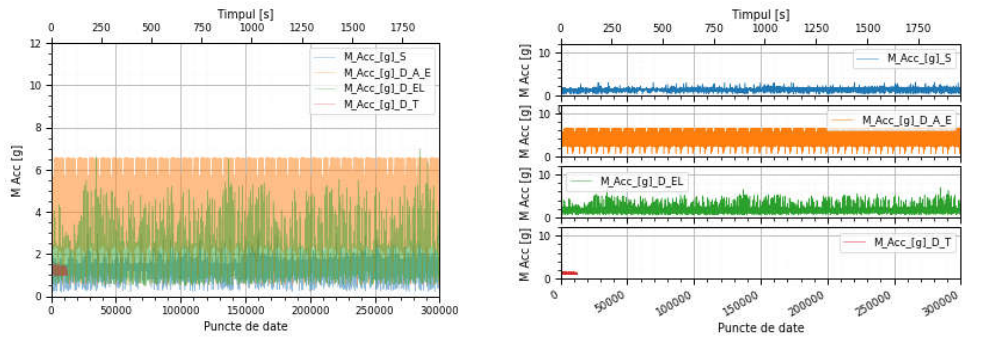


Fig. 28 Variația *Mag. Acc.* [g] pentru clasele analizate pe parcursul celor 300000 de puncte de date achiziționate, la viteza de rotație 4000 [RPM], *Python 3.6*.

Pentru clasa ESC defect, la viteza de 4000 [RPM], Fig. 26-Fig. 28, ESC-ul a întrerupt funcționarea motorului după aproximativ 80 [s], intrând în modul de protecție datorită amplitudinii și variației ridicate a curentului de alimentare a motorului.

Din graficele de variație a parametrilor T [°C], I [mA], $Mag. Acc.$ [g], în raport cu punctele achiziționate, la vitezele axului motor de, 1000 [RPM], 2000 [RPM] și 4000 [RPM] (Fig. 20- Fig. 28), se poate constata:

- T [°C], I [mA], $Mag. Acc.$ [g], *au amplitudinile cele mai mari pentru clasa ESC defect, spre deosebire de celelalte clase. Motorul are pentru această clasă un consum de energie foarte ridicat și funcționează zgomotos.* Pentru a se obține turațiile comandate, motorul consumă mult mai mult, aproximativ dublu față de clasa sănătos;

- T [°C], I [mA], $Mag. Acc.$ [g], *au amplitudinile cele mai mici pentru clasa sănătos.* Motorul vibrează foarte puțin iar consumul de energie este minim;

- valorile parametrilor T [°C], I [mA], $Mag. Acc.$ [g], *variază în funcție de viteza de rotație a arborelui rotor. Aceștia cresc o dată cu creșterea turației;*

- *fiecare clasă analizată are o variație diferită a parametrilor T [°C], I [mA], $Mag. Acc.$ [g], creându-se astfel câte o amprentă unică pentru fiecare clasă;*

- *funcționarea cu defect produce creșterea zgomotului și a consumului de energie.*

3.5.4. Rezultate și discuții

Metoda propusă a fost implementată utilizând *Python 3.6* cu pachetele *Tensorflow* și *Pandas*. Pentru generarea rezultatelor experimentale, au fost utilizate trei motoare BLDC, trei elice și două ESC-uri. Setul de date de antrenament a modelelor conține informații despre fiecare clasă.

Tab. 7 Performanțele algoritmilor utilizați pentru identificarea tipului de defect a motoarelor electrice BLDC, *Python 3.6*.

Indicator de performanță	SVM	KNN	BN
ACC	0.96	0.95	0.91
$PR_{macro\ avg}$	0.96	0.94	0.90
$Recall_{macro\ avg}$	0.97	0.95	0.91
$F1_{macro\ avg}$	0.96	0.95	0.91

Din Tab. 7 SVM are performanțele cele mai bune și este mult mai rapid în comparație cu algoritmi KNN și BN. Cel mai lent algoritm este BN. Performanțele clasificatorului KNN sunt similare clasificatorului SVM.

3.5.5. Concluzii

În această lucrare, autorii prezintă o nouă abordare privind identificarea tipului de defect a motoarelor BLDC de la UAV-uri, ce utilizează învățarea automată. Pentru a maximaliza performanța metodei de a clasifica defectele au fost comparate rezultatele clasificatorilor SVM, KNN și BN. Performanțele clasificatorului SVM sunt mai ridicate în comparației cu ceilalți clasificatori.

Din datele experimentale rezultate se constată că atunci când motorul BLDC funcționează sub influența clasei ESC defect (tranzistor MOSFET FDD8896 defect), motorul funcționează zgomotos, consumul de energie este ridicat iar temperatura este ridicată spre deosebire de

funcționarea sub influența celorlalte clase. În final funcționarea îndelungată sub această clasă duce la pierderea izolației înfășurării de pe stator, ceea ce duce la apariția eșecului funcțional.

În urma acestui studiu se constată că modelul propus prezintă predicții mai realiste decât cele ale altor modele, având o capacitate de generalizare mai mare. Acesta poate fi implementat în cadrul unui sistem de mentenanță predictivă din componența unui UAV, pentru a identifica defectele motoarelor BLDC.

4. Realizarea unui sistem de mentenanță predictivă pentru UAV-uri folosind Fog Computing

4.1. Introducere

În ultimii ani UAV-urile au început să fie din ce în ce mai utilizate datorită potențialului lor neexploatat [118], [119].

O dată cu creșterea numărului de UAV-uri a crescut exponențial și numărul de accidente. Defectarea UAV-urilor poate duce la, prăbușirea UAV-urilor, pierderea UAV-urilor, rănirea oamenilor și distrugerea obiectelor [120], [121]. *Datorită creșterii capacității electrice stocate cu ajutorul acumulatorilor, numărul de UAV-uri propulsate de motoare electrice este în creștere. Majoritatea UAV-urilor propulsate electric utilizează motoare BLDC [112], [115].*

Pentru a crește siguranța de exploatare a UAV-urilor este necesar să se implementeze un sistem de mentenanță predictivă care să monitorizeze componentele cu o fiabilitate mai scăzută.

Întrucât prin utilizarea unei arhitecturi bazate pe cloud, pentru transmisia datelor s-ar consuma o lățime de bandă considerabilă iar siguranța datelor din cloud ar fi scăzută, la realizarea sistemului de mentenanță predictivă din cadrul UAV-urilor a fost utilizată o arhitectură bazată pe FC.

Sistemul de mentenanță predictivă propus, are în componență trei niveluri, nivelul dispozitivelor, nivelul fog și nivelul cloud. Primele două niveluri, dispozitivelor și fog, fac parte din componența UAV-ului iar nivelul de cloud se află în cadrul centrului de mentenanță.

K. Patrick și colab., în cadrul standardului „Industrializarea 4.0” propun arhitectura unui sistem de mentenanță predictivă pentru un parc de autobuze. Arhitectura este împărțită pe trei niveluri: 1. Nivelul de percepție, implementat pe fiecare autobuz, conține senzori, suport de stocare limitată, dispozitive de comunicație; 2. Nivelul middleware se caracterizează prin spațiu de stocare mai mare, învățare automată, oferă interfața cu nodurile nivelului de percepție; 3. Nivelul aplicației, afișează rezultatele, management utilizatori, administrare platformă. Datele sunt preluate de la interfața CAN a autobuzelor. Comunicația la nivel de aplicație se realizează prin intermediul protocolului MQTT [122].

4.2. Conceperea și realizarea practică a unui sistem de mentenanță predictivă pentru UAV-uri

*Sistemul de mentenanță predictivă propus va avea trei niveluri, **nivelul dispozitivelor, nivelul fog și nivelul de cloud.** Nivelul dispozitivelor se ocupă cu achiziția și transmiterea datelor de la senzori, nivelul fog cu preprocesarea datelor iar nivelul cloud cu procesarea, stocarea și afișarea datelor. Pe nivelul dispozitivelor avem două dispozitive, un dispozitiv monitorizează parametrii motorului BLDC iar al doilea dispozitiv monitorizează parametri*

acumulatorului de Li-ion. Datele sunt transmise de pe nivelul dispozitivelor pe nivelul fog unde sunt preprocesate prin intermediul unui SBC iar la final sunt transmise pe nivelul cloud. Pe nivelul de cloud se face analiza datelor și afișarea rezultatelor.

Diagrama UML de secvență a aplicației poate fi vizualizată în Fig. 29. Diagrama de secvență arată cum funcționează procesele între ele și în ce ordine. Prin apăsarea butonului *Conectare* aflat în cadrul interfeței *Conectare dispozitive IoT*, aplicația de pe nivelul cloud se conectează la Broker-ul de MQTT, Fig. 29.

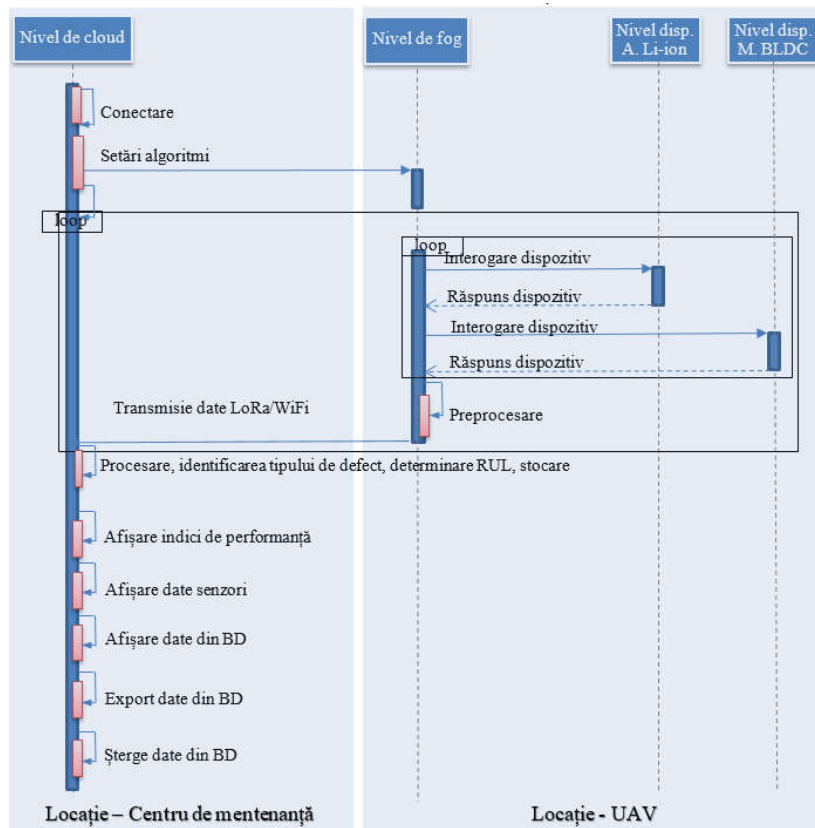


Fig. 29 Diagrama UML de secvență a aplicației.

Cu scopul de a determina soluția optimă, au fost studiate două posibilități de transmisie a datelor de pe nivelul de fog pe nivelul de cloud. În cadrul primei variante, datele sunt transmise de pe nivelul fog pe nivelul cloud prin intermediul protocolului LoRA. Cea de-a doua variantă utilizează pentru a transmite datele de pe nivelul fog pe nivelul de cloud protocolul WiFi. Deci, au fost analizate două variante de sisteme de mentenanță predictivă pentru UAV-urile propulsate electric.

Primul dispozitiv *Nivel disp. A. Li-ion* (Fig. 29) de pe nivelul dispozitivelor, monitorizează parametrii acumulatorului de Li-ion iar cel de-al doilea dispozitiv *Nivel disp. M. BLDC* (Fig. 29) de pe nivelul dispozitivelor monitorizează parametrii motorului BLDC. În mod continuu SBC-ul de pe nivelul fog interoghează fiecare dispozitiv pentru a obține datele de la senzori, Fig. 29. Pe nivelul de cloud datele sunt stocate apoi este extras diagnosticul și pronosticul. Aplicația realizată pe nivelul cloud conține mai multe interfețe. Afișarea rezultatelor se face cu ajutorul interfeței *Afișare indici de performanță*. Prin intermediul interfeței *Afișare date senzori*, se pot vizualiza în timp real datele provenite de la senzori. Interfețele *Vizualizare date din BD*, *Ștergere date din BD*, *Export date din BD în csv* permit afișarea, exportul și ștergerea datelor din baza de date.

Utilizatorul poate seta diverse combinații de algoritmi, prin intermediul interfeței *Setari date*, Fig. 29.

4.2.1. Schemele bloc și standul experimental

Pentru a identifica soluția optimă a sistemului de mentenanță predictivă, au fost concepute două scheme bloc ce diferă prin modul de transmisie a datelor de pe nivelul fog pe nivelul de cloud, prima utilizează pentru transmisia datelor protocolul LoRa iar a doua utilizează pentru transmisia datelor protocolul WiFi.

În cadrul primei scheme bloc Fig. 30, transmisia datelor de pe nivelul de fog pe nivelul de cloud se face prin intermediul **LoRa**, existând astfel o legătură continuă între nivelul de fog și nivelul de cloud. Componentele UAV-ului sunt monitorizate în mod continuu, permițând astfel operatorului să vadă starea activelor în timp real.

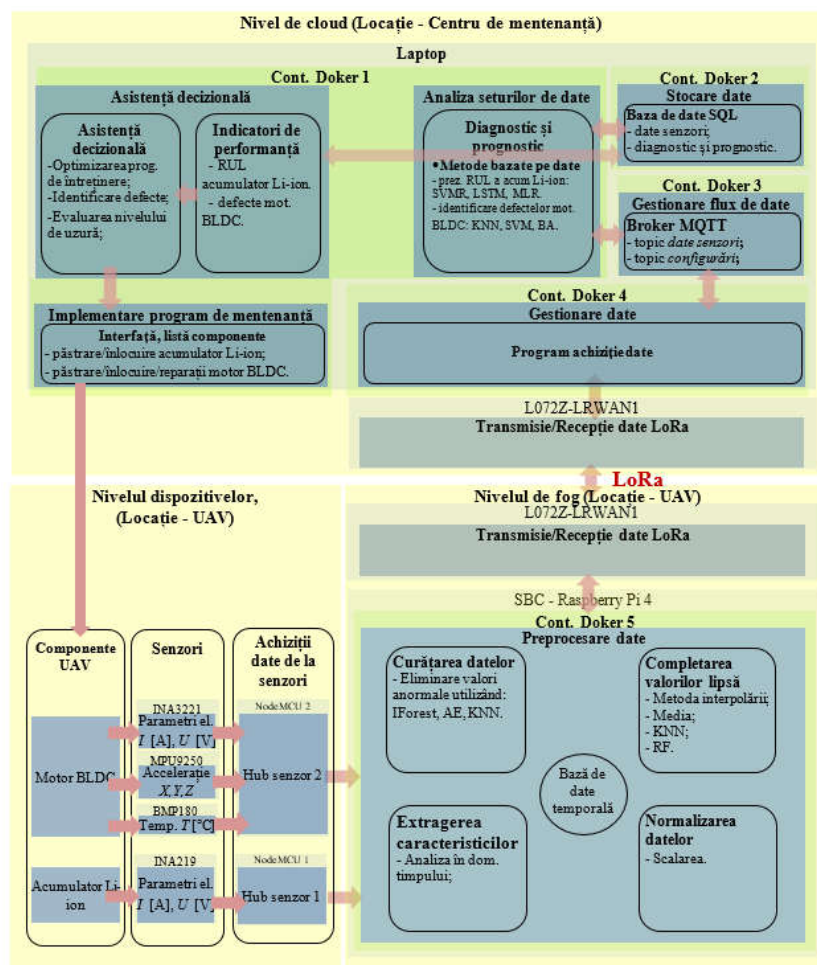


Fig. 30 Schema bloc a standului experimental, LoRa.

Fluxul informațional pentru schema bloc din Fig. 30 este: datele provenite de pe nivelul dispozitivelor sunt transmise pe nivelul fog (Raspberry Pi 4) unde sunt preprocesate, apoi sunt transmise prin intermediul modulelor LoRa (L072Z-LRWAN1) pe nivelul cloud (simulat cu un Laptop) la *Programului de achiziție date* ce transmite datele prin intermediul protocolului MQTT, modulului *Analiza seturilor de date* pentru a extrage diagnosticul și prognosticul. Apoi, diagnosticul, prognosticul, precum și datele de la senzori sunt stocate în BD. La final mai avem

modulele *Asistență decizională* și *Implementare program de mentenanță* care iau decizii pe baza rezultatului dat de modulul *Analiza seturilor de date*.

Standul experimental în care s-a implementat schema bloc din Fig. 30 este prezentat în Fig. 31 și Fig. 32. Datele provenite din monitorizarea acumulatorului Li-ion (Fig. 31, - 7) prin intermediul modulului senzorial INA2019 (Fig. 31, - 5) ce monitorizează I [A], U [V] sunt transmise de către placa de dezvoltare NodeMCU 1 (Fig. 31, -6), modulului SBC Raspberry Pi 4 (Fig. 31, - 1). Totodată, datele provenite din monitorizarea motorului electric BLDC (Fig. 32, - 13) precum I [A], U [V], utilizând modul senzorial INA3221 (Fig. 31, - 9), T [°C], utilizând modul senzorial BMP180 (Fig. 32, - 14), accelerația pe axele X , Y , Z , utilizând modulul senzorial MPU9250 (Fig. 32, - 15) sunt transmise de către placa de dezvoltare NodeMCU 2 (Fig. 31, - 10) modulului SBC Raspberry Pi 4 (Fig. 31, -1). Modulul SBC Raspberry Pi 4 (Fig. 31, -1) preprocesează datele și transmite rezultatul preprocesării prin intermediul modulului LoRa (Fig. 31, -2) la laptopul (Fig. 32, -16) care le recepționează prin intermediul modulului LoRa (Fig. 31, -8).

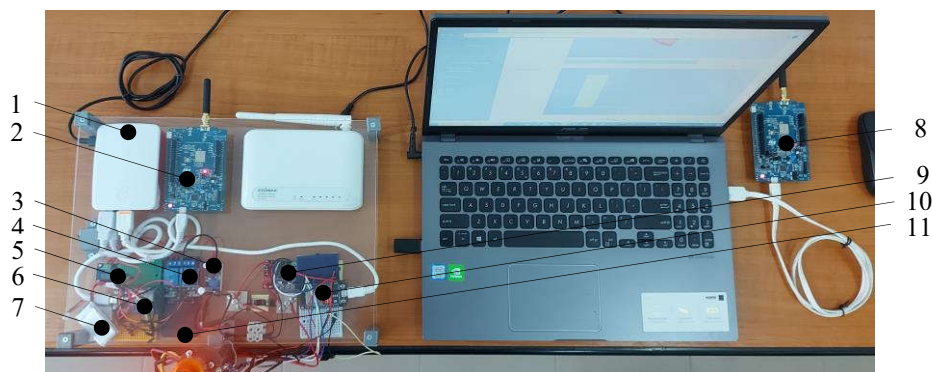


Fig. 31 Standul experimental, LoRa, vedere de sus: 1 - Raspberry Pi 4; 2 – modul LoRa 1, L072Z-LRWAN1; 3 - modulul ridicător de tensiune MT3607; 4 – modul comutare 2 rele; 5 – circuit senzorial INA2019; 6 – placă de dezvoltare NodeMCU 1; 7- acumulator Li-ion; 8 – modul LoRa 2, L072Z-LRWAN1; 9 – circuit senzorial INA3221; 10 – placă de dezvoltare NodeMCU 2; 11 – modul de încărcare acumulator Li-ion HW-107.

În cadrul acestei variante a fost implementată o conexiune LoRa peer-to-peer. Laptopul simulează nivelul cloud.

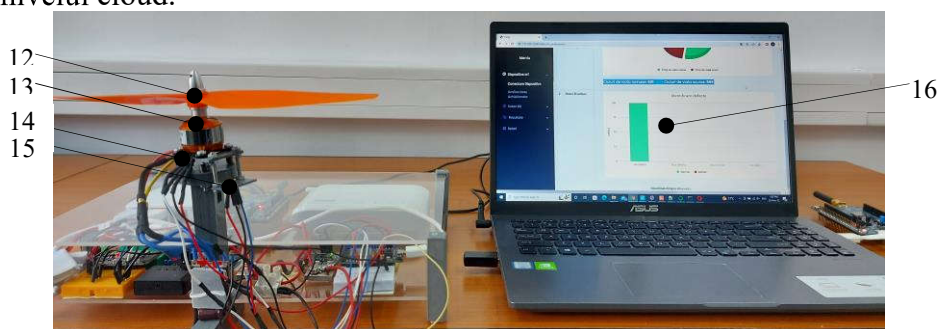


Fig. 32 Standul experimental, LoRa, vedere din fata: 12 – elice; 13 – motor BLDC, A2208; 14 – circuit senzorial temperatură BMP180; 15 – circuit senzorial vibrații MPU9250; 16 – Laptop.

În cadrul celei de-a doua scheme bloc Fig. 33, transmisia datelor între nivelul de fog și nivelul de cloud se face prin intermediul conexiunii WiFi.

Fluxul informațional conform Fig. 33, datele provenite de pe nivelul dispozitivelor sunt transmise pe nivelul fog la Raspberry Pi 4 unde sunt preprocesate, apoi sunt transmise utilizând protocolul MQTT prin intermediul modulului WiFi spre Router Wireless (Edimax BR6208ac),

apoi spre nivelul cloud (Laptop conectat la WiFi Router Wireless) la modulului *Analiza seturilor de date* pentru a extrage diagnosticul și prognosticul. Toate datele sunt stocate în BD. În ultimul pas, modulele *Asistență decizională* și *Implementare program de mentenanță* iau decizii pe baza rezultatului dat de modulul *Analiza seturilor de date*.

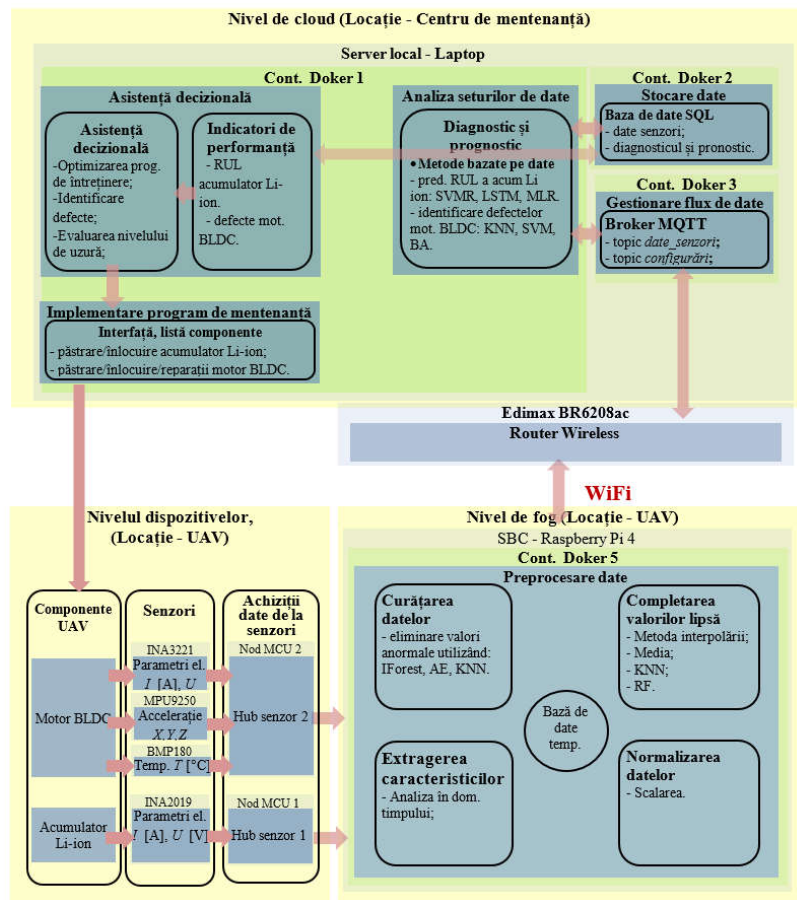


Fig. 33 Schema bloc a sistemului de mentenanță predictivă propus pentru o flotă de UAV-uri, WiFi.

În Fig. 34 și Fig. 35 este prezentat standul experimental descris cu ajutorul schemei bloc din Fig. 33. Datele provenite din monitorizarea acumulatorului de Li-ion (Fig. 34, - 6), prin intermediul modulului senzorial ce monitorizează I [A], U [V] (Fig. 34, - 4) sunt transmise de placa de dezvoltare NodeMCU 1 (Fig. 34, - 5) către SBC-ului Raspberry Pi 4 (Fig. 34, - 1).

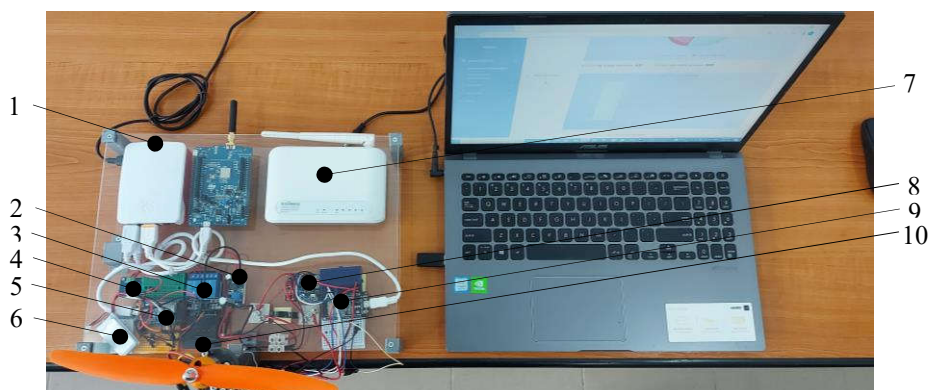


Fig. 34 Standul experimental, WiFi, vedere de sus: 1 - Raspberry Pi 4; 2 - modulul ridicător de tensiune MT3607; 3 – modul comutare 2 rele; 4 – circuit senzorial INA2019; 5 – placă de dezvoltare NodeMCU 1; 6- acumulator Li-ion; 7 – router WiFi, Edimax BR6208ac; 8 – circuit senzorial INA3221; 9 – placă de dezvoltare NodeMCU 2; 10 – modul de încărcare acumulator Li-ion HW-107.

Datele provenite din monitorizarea motorului electric BLDC (Fig. 35, - 12) precum T [°C], utilizând modulul senzorial BMP180 (Fig. 35, - 13), I [A], U [V], utilizând modulul senzorial INA2019 (Fig. 34, - 4) și accelerația pe X , Y , Z , utilizând modulul senzorial MPU9250 (Fig. 34, - 14) sunt transmise de către placa de dezvoltare NodeMCU 2 (Fig. 34, - 9) SBC-ului Raspberry Pi 4 (Fig. 34, -1).

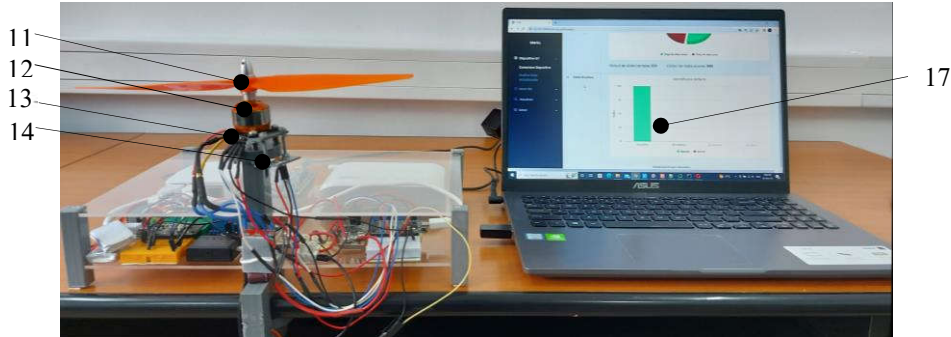


Fig. 35 Standul experimental, WiFi, vedere din fata: 11 – elice; 12 – motor BLDC, A2208; 13 – circuit senzorial temperatură BMP180; 14 – circuit senzorial vibrații MPU9250.

La final, SBC-ul Raspberry Pi 4 (Fig. 34, -1) are rolul de a preprocesa datele și de a transmite rezultatul preprocesării WiFi, prin intermediul router-ului Edimax BR6208ac (Fig. 34, -7) la laptop (Fig. 35, -17) ce simulează nivelul de cloud. Laptop-ul face procesarea datelor și se afișează rezultatul.

Pentru realizarea celor două variante de sisteme de mentenanță predictivă, a fost utilizată o tehnologie de virtualizare bazată pe micro servicii. Aplicația conține mai multe containere Docker, astfel Fig. 30, Fig. 33: Cont. Docker 1 conține programele pentru analiza seturilor de date și programul de asistență decizională; Cont. Docker 2 conține BD; Cont. Docker 3 conține broker-ul MQTT; Cont. Docker 4 conține în cadrul primei variante Fig. 30, programul pentru achiziția datelor; Cont. Docker 5 conține programul pentru preprocesarea datelor.

SBC-ul Raspberry Pi 4 (Fig. 31 - 1, Fig. 34 -1), interoghează prin intermediul protocolului serial, placa dezvoltare NodeMCU 1 pentru a extrage I [A], U [V] și C [mAh] descărcată apoi interoghează placa dezvoltare NodeMCU 2 pentru a extrage I [A], U [V], $AccX$ [g], $AccY$ [g], $AccZ$ [g] și T [°C] de funcționare a motorului electric BLDC. Datele sunt introduse într-o stivă FIFO în grupuri de câte 50 de linii, existând în stiva tot timpul trei grupuri. Grupul din mijloc este preprocesat. La final ultimul grup iese din stiva pentru a face loc altuia.

Pe nivelul fog, datele sunt preprocesate reducând astfel cantitatea datelor transmise pe nivelul cloud. Astfel, pe nivelul fog datele sunt curățate, completate, transformate și normalizate.

În cadrul schemei bloc din Fig. 30, datele sunt transmise prin intermediul protocolului LoRa. Placa SBC Raspberry Pi 4 va transmite prin intermediul protocolului serial datele, plăcii de dezvoltare L072Z-LRWAN1 pentru a le transmite prin intermediul protocolului LoRa pe nivelul cloud.

Schema bloc din Fig. 33 presupune transmiterea datelor de pe nivelul fog pe nivelul de cloud prin intermediul conexiunii WiFi. Astfel placa SBC Raspberry Pi 4 transmite datele în cloud-ul platformei cu ajutorul modulului WiFi aflat în componența sa.

Pe nivelul cloud, aplicația recepționează datele transmise prin intermediul protocolului LoRa/WiFi, le interpretează, apoi le stochează în baza de date iar în final afișează defectele motorului BLDC și RUL al acumulatorului de Li-ion. Aplicația conține mai multe interfețe

web ce permit: conectarea aplicației la broker-ul MQTT, afișarea/exportul/ștergere datelor din BD, realizarea setărilor și afișarea rezultatelor. Aceste interfețe sunt: *Conectare dispozitive IoT, Vizualizare date de la senzori, Vizualizare date din BD, Ștergere date din BD, Export date din BD în csv, Afișare indici de performanță și Setări date.*

Comunicația la nivel de aplicație se face prin intermediul protocolului MQTT (Fig. 30, Fig. 33). Datele transmise pe nivelul cloud sunt stocate în baza de date *sql*, găzduită prin intermediul aplicației *XAMPP* [123].

La schema bloc din Fig. 33 editorul conținut de *Programul pentru preprocesare* aflat pe nivelul fog, trimite datele provenite de la senzori aplicației din containerul Docker 1 pentru a le interpreta și afișa.

Utilizând blocul de *Diagnostic și prognostic* (Fig. 30, Fig. 33) este extras RUL al acumulatorului de Li-ion și este identificat tipul de defect a motorului BLDC. Pentru extragerea RUL și identificarea tipului de defect au fost utilizate metode bazate pe date. Deciziile de întreținere sunt afișate prin intermediul blocului de *Asistență decizională* (Fig. 30, Fig. 33).

La realizarea modelelor de învățare automată, a fost utilizată învățarea supravegheată. Pentru a clasifica defectele motoarelor electrice BLDC, se pot selecta mai mulți algoritmi, KNN, SVM, NB, iar pentru a estima RUL al acumulatorilor de Li-ion, se pot selecta algoritmi, RF, SVMR, MLR. Din Tab. 6 se constată că atunci când pentru predicția ciclurilor de viață scurse ale acumulatorilor de Li-ion sunt utilizați algoritmi SVMR și RF, se obțin performanțele cele mai bune în predicția RUL al acumulatorilor de Li-ion. Din Tab. 7, se constată că algoritmul SVM are performanțele cele mai bune în identificarea tipului de defect a motoarelor electrice BLDC.

Datele de antrenament utilizate pentru antrenarea modelului ce permite identificarea tipului de defect a motoarelor electrice BLDC provin de la sistemul de mentenanță predictivă prezentat mai sus și au fost obținute prin monitorizarea funcționării unui motor BLDC, sub influența claselor precum, *C₁- Sănătos, C₂- Elice ciobită, C₃- Arbore excentric și C₄- Esc defect (tranzistor MOSFET, FDD8896 defect)*. Datele de antrenament utilizate pentru antrenarea modelului ce permite predicția RUL al acumulatorilor au fost obținute de la sistemul de mentenanță predictivă realizat, prin monitorizarea ciclurilor de încărcare descărcare a doi acumulatori de Li-ion. Datele au fost împărțite în date de instruire (70%) și de testare (30%). Modelul a fost antrenat folosind datele de antrenament și performanța modelului a fost verificată pe datele de testare. Rezultatele au fost analizate și evaluate pe baza principalilor indicatori de performanță utilizați de algoritmi de învățare automată. La final, modelele au fost implementate în cadrul celor două variante de sisteme de mentenanță predictivă.

Pictogramele cu interfețele web, conținute de către aplicația de pe nivelul cloud, sunt prezentate în continuare.

Interfața Conectare dispozitive IoT. Interfața afișează statusul fiecărui dispozitiv din aplicație, Neconectat /Conectat, Fig. 36/Fig. 37.



Fig. 36 Interfața *Conectare dispozitive IoT*, status – Neconectat, *Flask, Python 3.6*.

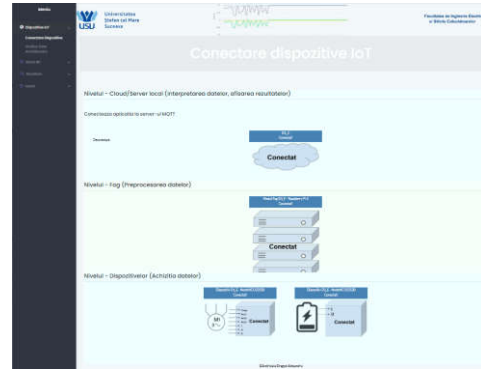


Fig. 37 Interfața *Conectare dispozitive IoT*, status – Conectat, *Flask, Python 3.6*.

Interfața *Vizualizare date de la senzori* – afișează datele primite prin intermediul topicului *date_senzori* al protocolului MQTT, Fig. 38. **Interfața *Vizualizare date din BD*** - Permite afișarea datelor din baza de date în funcție de intervalul introdus de utilizator, Fig. 39.

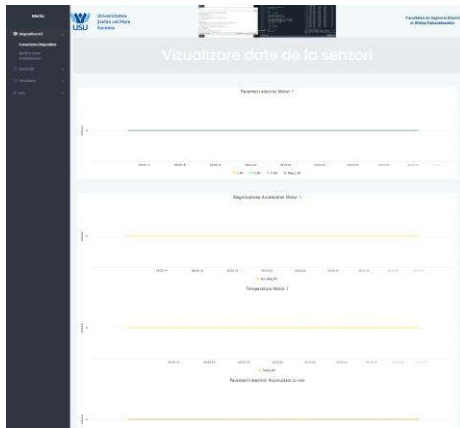


Fig. 38 Interfața *Vizualizare date de la senzori*, *Flask, Python 3.6*.

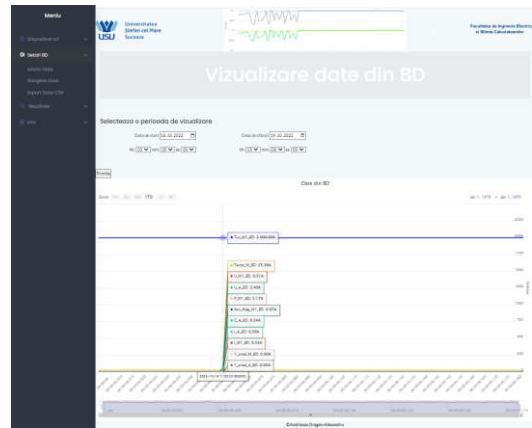


Fig. 39 Interfața *Vizualizare date din BD*, *Flask, Python 3.6*.

Interfața *Ștergere date din BD* - permite ștergerea datelor din baza de date în funcție de intervalul introdus de utilizator Fig. 40. **Interfața *Export date din BD în csv*** Permite exportul datelor din baza de date în format .csv pe un interval introdus de utilizator Fig. 41.

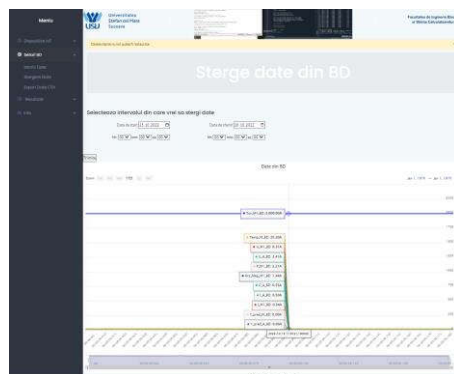


Fig. 40 Interfața *Ștergere date din BD*, *Flask, Python 3.6*.

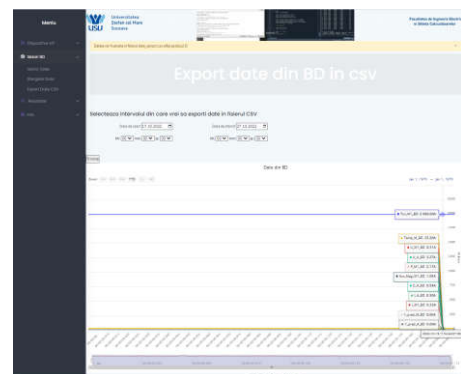


Fig. 41 Interfața *Export date din BD în csv*, *Flask, Python 3.6*.

Interfața *Afișare indici de performanță* - Afișează ciclurile de viață rămase/ scurse ale acumulatorilor de Li-ion, precum și starea de sănătate a motorului electric BLDC (*C₁- Sănătos*,

C2- Elice ciobită, C3- Arbore excentric și C4- Esc defect), Fig. 42. **Interfața Setări date** - Prin intermediul acestei interfețe utilizatorul poate selecta algoritmi utilizați pe cele trei niveluri, nivelul dispozitivelor, nivelul fog și nivelul cloud, Fig. 43.

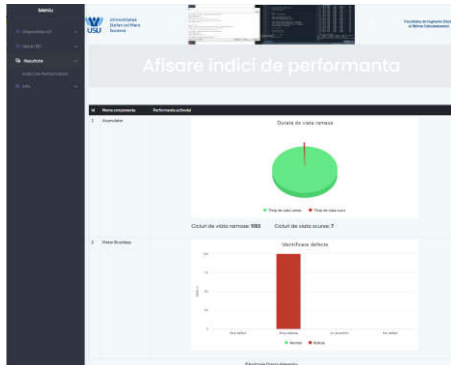


Fig. 42 Interfața Afișare indici de performanță, Flask, Python 3.6.

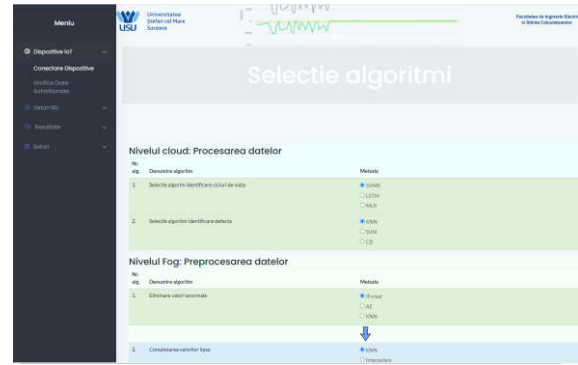


Fig. 43 Interfața Setări date, Flask, Python 3.6.

Operatorii din centrul de mentenanță pot vizualiza starea de sănătate a activelor urmărite și pot lua măsuri de intervenție. Implementarea modulului de *Asistență decizională* (Fig. 30, Fig. 33) a fost lăsată pentru viitoarele dezvoltări.

4.3. Rezultate și discuții

În cadrul acestui capitol, au fost prezentate două variante de sisteme de mentenanță predictivă pentru UAV-uri. Ambele variante au în componență o arhitectură bazată pe FC și diferă prin modul de transmisie a datelor de pe nivelul fog pe nivelul cloud, prima variantă utilizează protocolul LoRa iar cea de-a doua variantă utilizează protocolul WiFi.

După conectarea la topicul de MQTT *date_senzori*, aplicația web analizează datele pentru a extrage, RUL al acumulatorilor de Li-ion și pentru a identifica tipul de defect a motoarelor electrice BLDC, apoi, stochează datele primite în baza de date.

Prin analiza rezultatelor obținute, se mai constată că atunci când utilizăm pentru a transmite date între nivelul de fog și nivelul de cloud LoRa, viteza de comunicație este mult mai scăzută dar datorită distanței mari de acoperire (aprox. 15 [km]), putem face o monitorizare continuă a activelor UAV-ului. LoRa a fost proiectat pentru a transmite date pe distanțe mari la viteze de transmisie mici cu un consum redus de energie.

Prin utilizarea conexiunii WiFi, datorită distanței mici de acoperire operatorul poate vedea starea de sănătate a activelor doar atunci când UAV-ul este aproape de centru de mentenanță, neputând astfel să vadă starea activelor în timp real și deci operatorul nu poate să ia decizii de întreținere în timpul funcționării UAV-ului pentru a preveni eventualele accidente sau daune.

Pentru a demonstra eficacitatea metodelor în determinarea RUL al acumulatorilor de Li-ion, a fost utilizat un acumulator de Li-ion ce a fost încărcat și descărcat de 200 de ori. Algoritmii, SVMR a determinat un număr de 196 de cicluri scurse, RF un număr de 206 de cicluri scurse iar MLR un număr de 223 de cicluri scurse. Prin analiza rezultatelor obținute, se constată că atunci când pentru predicția RUL al acumulatorilor de Li-ion sunt utilizați algoritmi SVMR și RF, predicțiile făcute sunt mai apropiate față de valorile reale.

Pentru a demonstra eficacitatea metodelor de identificare a tipului de defect a motoarelor electrice BLDC, a fost studiată performanța algoritmilor SVM, KNN, NB în clasificarea celor

patru stări (C_1 -Sănătos, C_2 -Elice ciobită, C_3 -Arbore excentric și C_4 -ESC defect) ale motoarelor electrice BLDC. Performanțele algoritmilor utilizați sunt conform Tab. 7.

Modificând simptomele unui defect prin modificarea suprafeței ciobite a elicei, s-a constatat că performanțele metodelor de a clasifica acest defect au scăzut.

4.4. Concluzii

În cadrul acestui capitol, a fost prezentat modul de realizare a două variante de sisteme de mentenanță predictivă ce pot fi utilizate pentru identificarea tipului de defect a motoarelor BLDC și pentru predicția RUL al acumulatorilor de Li-ion din cadrul UAV-urilor acționate electric. Pentru a minimiza cantitatea datelor trimise în cloud și a crește siguranța datelor, ambele variante de sisteme de mentenanță predictivă pentru UAV-uri au o arhitectură bazată pe FC. Preprocesarea datelor pe nivelul fog permite reducerea cantității de date ce trebuie transferată pe nivelul cloud.

Cele două variante de sisteme de mentenanță predictivă concepute, permit identificarea tipului de defect a motoarelor electrice BLDC și estimează RUL al acumulatorilor de Li-ion din cadrul UAV-urilor. Astfel, personalul aflat în serviciul de mentenanță în funcție de rezultatul dat de sistemul de mentenanță predictivă poate înlocui componentele defecte cu scopul de a menține activul în funcțiune.

Identificarea tipului de defect și predicția RUL a componentelor se poate face monitorizând în mod continuu parametri ce ne pot da informații despre starea de sănătate precum și despre tendința de evoluție. Pentru a determina tipul de defect a motoarelor electrice BLDC au fost monitorizate, vibrația, temperatura precum și tensiunea și curentul de alimentare iar pentru predicția RUL al acumulatorilor de Li-ion au fost monitorizate tensiunea și curentul de descărcare.

Pentru a crește fiabilitatea UAV-urilor, pot fi analizate și alte componente care au o probabilitate ridicată de a se defecta.

Cele două variante de sisteme de mentenanță predictivă diferă prin modul în care sunt transmise datele de pe nivelul fog pe nivelul de cloud, prima variantă utilizează protocolul LoRa iar cea de-a doua variantă utilizează protocolul WiFi. Prin utilizarea comunicației LoRa datele pot fi transmise în mod continuu pe distanțe până la 15 [km]. Atunci când este utilizat protocolul WiFi, datele sunt transmise când UAV-ul este aproape de centru de comandă (aproximativ câteva sute de metri) și deci sistemul nu poate identifica starea de sănătate pe parcursul deplasării UAV-ului.

Ținând cont de rezultatele obținute, sistemul de mentenanță predictivă pentru UAV-uri ce utilizează pentru transmisia datelor de pe nivelul fog pe nivelul cloud protocolul LoRa, poate fi utilizat cu succes în situații reale.

5. Concluziile lucrării

Mentenanța predictivă este standardul dorit în contextul „Industrializarea 4.0”, ce poate avea un impact semnificativ asupra programelor de întreținere. Prin adoptarea mentenanței predictive în cadrul industriei, companiile vor beneficia de o serie de îmbunătățiri astfel, opririle accidentale vor fi eliminate, va crește durata de viață a echipamentelor, timpul de oprire a

utilajelor va fi micșorat iar costurile cu mentenanța vor fi reduse. Toate beneficiile enumerate vor crește calitatea produselor și serviciilor.

Sistemele de mentenanță predictivă ce utilizează pentru predicție învățarea automată pot să identifice defectele cu mult timp înaintea metodelor tradiționale. Prin descoperirea defectelor cu mult timp înaintea metodelor tradiționale, costul cu mentenanța va scădea iar timpul de funcționare va fi maximalizat, de asemenea pot fi evitate și eventualele accidente sau daune datorită defectării.

Utilizarea tehnologiei IoT în cadrul mentenanței predictive va aduce beneficii mari. Prin monitorizarea echipamentelor cu ajutorul dispozitivelor IoT dotate cu senzori, conectarea echipamentelor între ele, utilizarea învățării automate pentru a face predicții, personalul calificat va putea efectua activitățile de întreținere doar atunci când este necesar.

Prin utilizarea a unei arhitecturi bazate pe Fog Computing și Edge Computing în cadrul sistemelor de mentenanță predictivă, este minimizată lungimea de bandă necesară transmiterii datelor în cloud iar gradul de confidențialitate a datelor crește datorită faptului că în cloud este trimis doar rezultatul preprocesării datelor și nu toate datele produse de senzori.

Valorile anormale sunt instanțe care sunt departe de valorile așteptate într-un set de date, ele pot fi prezente în orice set de date. Atunci când se încearcă găsirea metodei care să aibă performanța cea mai bună în identificarea valorilor anormale, trebuie să se cerceteze performanțele metodelor candidate pe acel set de date. Performanțele metodelor de identificare a valorilor anormale variază de la un set de date la altul, în funcție de conținutul informațional al datelor.

Orice set de date poate să aibă instanțe lipsă. Atunci când se încearcă găsirea metodei care să aibă performanța cea mai bună în estimarea instanțelor lipsă trebuie să se cerceteze performanțele metodelor candidate pe acel set de date. Performanțele metodelor variază în funcție de conținutul informațional al datelor din setul de date.

Prin predicția RUL al acumulatorilor de Li-ion se poate asigura funcționarea în condiții de siguranță a UAV-urilor. Metoda concepută pentru predicția RUL al acumulatorilor de Li-ion utilizează învățarea automată și presupune etapele: extragerea caracteristicilor, selecția caracteristicilor, crearea modelelor și predicția. În cadrul acestui studiu au fost comparate performanțele mai multor algoritmi de învățare automată precum, SVMR, MLR și RF, în predicția RUL al acumulatorilor de Li-ion, deci au fost create mai multe modele. Pentru antrenarea fiecărui model, au fost utilizate date de antrenament provenite din degradarea a doi acumulatori de Li-ion. Algoritmii SVMR și RF au avut performanțele cele mai bune. Precizia de estimare a RUL al acumulatorilor de Li-ion este ridicată. Capacitatea de generalizare a modelului va crește dacă vor fi utilizate mai multe date de antrenament, provenite din monitorizarea mai multor acumulatori.

Prin identificarea tipului de defect a motoarelor electrice BLDC din componența UAV-urilor se poate preveni eșecul funcțional. Metoda dezvoltată pentru identificarea tipului de defect a motoarelor electrice BLDC de la UAV-uri poate să clasifice defecte precum: elice ciobită, ax excentric și ESC defect (tranzistor MOSFET FDD8896 defect). Pentru a clasifica defectele, metoda utilizează învățarea automată și presupune etapele: extragerea caracteristicilor, selecția caracteristicilor, crearea modelelor și clasificarea. Pentru a mări performanța metodei, au fost comparate performanța modelelor create cu ajutorul clasificatorilor, SVM, KNN și BN, în identificarea tipului de defect a motorului BLDC. Clasificatorul SVM a avut performanțele cele mai bune. Precizia de clasificare este ridicată. Caracteristicile au fost extrase din setul de date generat prin monitorizarea unui motor BLDC

sub influența defectelor enumerate mai sus. Capacitatea de generalizare va crește dacă modelele vor fi antrenate cu mai multe date etichetate ce descriu funcționarea normală și anormală pentru alte turații ale motoarelor BLDC

Cele două variante de sisteme de mentenanță predictivă realizate utilizează o arhitectură bazată pe FC și pot estima RUL al acumulatorilor de Li-ion și clasifica defectele motoarelor BLDC din componența UAV-urilor. Identificarea tipului de defect precum și predicția RUL se face monitorizând continuu parametri ce ne dau informații despre starea de sănătate. Pentru a transmite datele de pe nivelul fog pe nivelul de cloud a fost realizată o comparație privind transmisia cu ajutorul LoRa și WiFi. Prin utilizarea comunicației LoRa datele pot fi transmise în mod continuu pe distanțe până la 15 [km]. Atunci când utilizăm protocolul WiFi viteza de transmisie a datelor este ridicată iar datele sunt transmise când UAV-ul este aproape de centru de comandă (aproximativ câteva sute de metri).

6. Contribuții și direcții de cercetare viitoare

6.1. Contribuții teoretice

- S-au identificat principalele tehnologii pe care se sprijină standardul „Industrializarea 4.0”;
- A fost realizată o analiză comparativă a strategiilor de întreținere ce pot fi utilizate în domeniul industrial, prezentându-se avantajele și dezavantajele utilizării fiecărei strategii de întreținere;
- Analizând sistemele de mentenanță predictivă propuse de cercetători în cadrul lucrărilor de specialitate, s-au identificat elementele componente ale sistemelor de mentenanță predictivă;
- Au fost analizate principalele arhitecturi propuse în domeniul IoT;
- S-au analizat protocoalele de rețea utilizate în IoT;
- Au fost analizate tehnicile de preprocesare a datelor utilizate de sistemele de mentenanță predictivă;
- Prin analiza sistemelor de mentenanță predictivă propuse în cadrul lucrărilor de specialitate, au fost identificați principalii algoritmi de învățare automată ce pot fi utilizați de către sistemele de mentenanță predictivă.

6.2. Contribuții demonstrate pe baza unei cercetări practice

- A fost realizat un stand experimental ce permite studierea procesului de îmbătrânire al acumulatorilor de Li-ion. Automatizând procesul de încărcare, descărcare al acumulatorilor de Li-ion, pe parcursul fiecărui ciclu au fost monitorizați parametri care ne dau informații despre procesul de degradare;
- A fost realizat un stand experimental ce permite monitorizarea funcționării motoarelor BLDC sub influența următoarelor clase de funcționare: C_1 - Sănătos, C_2 - Elice ciobită, C_3 - Arbore excentric și C_4 - ESC defect (tranzistor MOSFET FDD8896 defect). Parametri monitorizați sunt: curentul, tensiune de alimentare, temperatura și accelerația pe cele trei axe;

- A fost realizată o bază de date în care este surprinsă funcționarea motoarelor BLDC sub influența claselor precum: C_1 - *Sănătos*, C_2 - *Elice ciobită*, C_3 - *Arbore excentric și* C_4 - *Esc defect (tranzistor MOSFET FDD8896 defect)*. Pentru monitorizarea motoarelor BLDC au fost utilizate module senzoriale ce au permis monitorizarea parametrilor precum: curent, tensiune de alimentare, temperatura și accelerația pe cele trei axe;
- A fost realizată o bază de date în care este surprins procesul de degradare a capacității de stocare a doi acumulatori de Li-ion. Scăderea capacității de stocare a fost surprinsă monitorizând cu ajutorul modulului senzorial de curent și tensiune INA219 procesul de încărcare/descărcare pe 1200 de cicluri de funcționare a doi acumulatori de Li-ion;
- Utilizând setul de date obținut, din monitorizare funcționării unui motor BLDC și din monitorizarea ciclurilor de încărcare/descărcare a unui acumulator de Li-ion, cu ajutorul limbajului *Python* s-a realizat o analiză comparativă privind performanța principalilor algoritmi de completare a valorilor lipsă ce pot fi utilizați de un sistem de mentenanță predictivă al unui UAV propulsat electric;
- Folosind setul de date obținut, din monitorizare funcționării unui motor BLDC și din monitorizarea ciclurilor de încărcare/descărcare a unui acumulator de Li-ion, cu ajutorul limbajului *Python* s-a realizat o analiză comparativă privind performanța principalilor algoritmi de identificare a valorilor anormale, ce pot fi utilizați de un sistem de mentenanță predictivă a unui UAV propulsat electric;
- A fost dezvoltată o metodă ce poate fi utilizată pentru identificarea tipului de defect a motoarelor BLDC din componența UAV-urilor propulsate electric. Utilizând setul de date obținut din monitorizare funcționării motoarelor BLDC sub influența claselor C_1 - C_4 , în cadrul metodei dezvoltate, a fost analizată cu ajutorul limbajului *Python* performanța a patru modele de învățare automată în clasificarea tipului de defect a motoarelor BLDC din componența UAV-urilor propulsate electric;
- A fost dezvoltată o metodă care poate să fie utilizată pentru a estima RUL al acumulatorilor de Li-ion din componența UAV-urilor propulsate electric. Folosind datele obținute prin monitorizarea ciclurilor de încărcare/descărcare (1200 cicluri) a doi acumulatori de Li-ion, în cadrul metodei dezvoltate, a fost analizată cu ajutorul limbajului *Python* performanța a patru modele de învățare automată în a estima RUL al acumulatorilor de Li-ion din componența UAV-urilor propulsate electric;
- A fost stabilită arhitectura pe care ar trebui să o conțină sistemele de mentenanță predictivă pentru UAV-uri. Cu scopul de a găsi soluția optimă de transmisie a datelor, au fost concepute două scheme bloc ce utilizează pentru transmisia datelor de pe nivelul fog pe nivelul cloud protocoale diferite, prima schemă bloc utilizează LoRa iar cea de-a doua schemă bloc utilizează WiFi;
- Plecând de la schemele bloc concepute, au fost realizate practic cele două variante de sisteme de mentenanță predictivă pentru UAV-urile propulsate electric. Ambele variante de sisteme de mentenanță predictivă au o arhitectură bazată pe FC și folosesc algoritmi de completare a valorilor lipsă, algoritmi pentru identificarea valorilor anormale, metoda dezvoltată pentru identificarea tipului de defect a motoarelor BLDC precum și metoda dezvoltată pentru predicția RUL al acumulatorilor de Li-ion, din capitolului III al tezei de doctorat;

- Pe baza cele două variante de sisteme de mentenanță predictivă pentru UAV-urile propulsate electric, au fost comparate rezultatele obținute pentru transmisia datelor de pe nivelul fog pe nivelul cloud atunci când sunt folosite protocoalele LoRa și WiFi.

6.3. Măsuri și direcții viitoare de cercetare

În cadrul etapei următoare, se vor efectua următoarele implementări și cercetări ce vor avea ca și scop creșterea relevanței și impactul acestei cercetări, precum:

- Se va implementa sistemul de mentenanță predictivă pe un UAV propulsat electric;
- Cu scopul dezvoltării unor metode care să estimeze RUL sau să identifice defectele altor componente, se va monitoriza funcționarea și altor componente care au o fiabilitate mai scăzută;
- Se va încerca transferul tehnologic al sistemului de mentenanță predictivă realizat pentru UAV-uri, prin căutarea unui producător de UAV-uri și convingerea acestuia să investească în implementarea sistemelor de mentenanță predictivă pe UAV-urile produse, pentru a crește siguranța de exploatare a acestora.

6.4. Diseminarea rezultatelor

Contribuțiile proprii dezvoltate în cadrul acestei lucrări au fost diseminate prin publicații și participări la conferințe după cum urmează:

<i>Publicații în cadrul unor jurnale</i>	
1.	Andrioiaia Dragoș-Alexandru , Găitan Vasile-Gheorghiiță, Monitoring the Discharge Voltage of Li-Ion Batteries in the UAV Component Through the Thingspeak IoT Cloud Platform Using LoRa, International Journal of Research in Engineering and Science (IJRES), ISSN: 2320-9364, vol. 11 Nr. 7, pp. 16-19, 2023. (BDI)
2.	Andrioiaia Dragoș-Alexandru , Găitan Vasile-Gheorghiiță, A Specialty Literature Review of the Predictive Maintenance Systems, Journal of Engineering Studies and Research, ISSN 2068-7559, Vol. 29, Nr. 4, pp. 17-23, 2023. (BDI)
3.	Andrioiaia Dragoș-Alexandru , Găitan Vasile-Gheorghiiță, Culea George, Banu Ioan Viorel, Predicting the RUL of Li-Ion Batteries in UAVs Using Machine Learning Techniques, Computers, ISSN 2073-431X, Vol. 13, Nr. 3, 2024. (ISI, F.I.: 2.8)
4.	Andrioiaia Dragoș-Alexandru , Găitan Vasile-Gheorghiiță, Finding fault types of BLDC motors within UAVs using machine learning techniques, Heliyon, ISSN 2405-8440, Vol. 10, Nr. 9, 2024. (ISI, F.I.: 4.0, Q2)
5.	Andrioiaia Dragoș-Alexandru , Găitan Vasile-Gheorghiiță, Pătruț Bogdan, Furdu Iulian, Comparative Performance Analysis of Filling Missing Values Algorithms in PdM Systems of UAV, BRAIN. Broad Research in Artificial Intelligence and Neuroscience, ISSN: 2068-0473, Vol.15, Nr. 2, 2024. (ISI, F.I.: 0.8)
<i>Prezentări la conferințe</i>	
1.	Andrioiaia Dragoș-Alexandru , Găitan Vasile-Gheorghiiță, Culea George, Prediction of BLDC Engine Defects Within UAVS, 17th International Conference of Constructive Design and Technological Optimization in Machine Building Field OPROTEH 2022, Per. 25.05.2022 - 27.05.2022, Org. Universitatea „Vasile Alecsandri” din Bacău; Facultatea de Inginerie, http://oproteh.ub.ro/ , Prezentare orală.

2.	Andrioaia Dragoș-Alexandru , Găitan Vasile-Gheorghită, Comparative analysis of predictive maintenance systems from specialty literature, 18th International Conference of Constructive Design and Technological Optimization in Machine Building Field OPROTEH 2023, Per. 11.05.2023 - 13.05.2023, Org. Universitatea “Vasile Alecsandri” din Bacău; Facultatea de Inginerie, http://oproteh.ub.ro/ , Prezentare orală.
----	---

Bibliografie selectivă

- [1] R. Gouriveau, K. Medjaher, and N. Zerhouni, *From Prognostics and Health Systems Management to Predictive Maintenance 1: Monitoring and Prognostics*. Wiley-ISTE, 2016, p. 163.
- [2] Z. Li, K. Wang, and Y. He, "Industry 4.0 - Potentials for Predictive Maintenance," *Proceedings of the 6th International Workshop of Advanced Manufacturing and Automation*, pp. 42-46, 2016.
- [3] L. Dogaru, "The Main Goals of the Fourth Industrial Revolution. Renewable Energy Perspectives," *Procedia Manufacturing*, vol. 46, pp. 397-401, 2020/01/01/ 2020.
- [4] S. Albishi, B. Soh, A. Ullah, and F. Algarni, "Challenges and Solutions for Applications and Technologies in the Internet of Things," *Procedia Computer Science*, vol. 124, pp. 608-614, 2017/01/01/ 2017.
- [5] P. Sethi and S. R. Sarangi, "Internet of Things: Architectures, Protocols, and Applications," *Journal of Electrical and Computer Engineering*, vol. 2017, p. 9324035, 2017/01/26 2017.
- [6] J. Geismann and E. Bodden, "A systematic literature review of model-driven security engineering for cyber-physical systems," *Journal of Systems and Software*, vol. 169, p. 110697, 2020/11/01/ 2020.
- [7] M. Gushev, "Dew Computing Architecture for Cyber-Physical Systems and IoT," *Internet of Things*, vol. 11, p. 100186, 2020/09/01/ 2020.
- [8] J. Reis and R. Gonçalves, "The Role of Internet of Services (IoS) on Industry 4.0 Through the Service Oriented Architecture (SOA): IFIP WG 5.7 International Conference, APMS 2018, Seoul, Korea, August 26-30, 2018, Proceedings, Part II," 2018, pp. 20-26.
- [9] J. Dalzochio *et al.*, "Machine learning and reasoning for predictive maintenance in Industry 4.0: Current status and challenges," *Computers in Industry*, vol. 123, p. 103298, 2020/12/01/ 2020.
- [10] K. Wang, G. Dai, and L. Guo, "Intelligent Predictive Maintenance (IPdM) for Elevator Service-Through CPS, IOTaS and Data Mining," *International Workshop of Advanced Manufacturing and Automation (IWAMA 2016)*, pp. 1-6, 2016.
- [11] A. Jimenez Cortadi, I. Irigoien, F. Boto, B. Sierra, and G. Rodriguez, "Predictive Maintenance on the Machining Process and Machine Tool," *Applied Sciences*, vol. 10, no. 1, p. 224, 12/27/2019 2019.
- [12] J.-R. Ruiz-Sarmiento, J. Monroy, F.-A. Moreno, C. Galindo, J.-M. Bonelo, and J. Gonzalez-Jimenez, "A predictive model for the maintenance of industrial machinery in the context of industry 4.0," *Engineering Applications of Artificial Intelligence*, vol. 87, p. 103289, 2020/01/01/ 2020.
- [13] I. Analytics. (12/05/2022). *The Top 20 Companies Enabling Predictive Maintenance* [Online]. Available: <https://iot-analytics.com/top-20-companies-enabling-predictive-maintenance/>.
- [14] llumin. (22/02/2022). *Top Predictive Maintenance Software Companies* [Online]. Available: <https://llumin.com/top-predictive-maintenance-software-companies-llu/>.
- [15] SAP. (22/02/2022). *SAP Predictive Maintenance and Service* [Online]. Available: https://help.sap.com/docs/r/p/SAP_Predictive_Maintenance_and_Service.
- [16] Z. Li, Y. Wang, and K.-S. Wang, "Intelligent predictive maintenance for fault diagnosis and prognosis in machine centers: Industry 4.0 scenario," *Advances in Manufacturing*, vol. 5, no. 4, pp. 377-387, 2017/12/01 2017.
- [17] K. Wang, "Intelligent Predictive Maintenance (IPdM) System – Industry 4.0 Scenario," *WIT transactions on engineering sciences*, vol. 113, no. 1, pp. 259-268, 2016.
- [18] J. Wang, Y. Liang, Y. Zheng, R. X. Gao, and F. Zhang, "An integrated fault diagnosis and prognosis approach for predictive maintenance of wind turbine bearing with limited samples," *Renewable Energy*, vol. 145, pp. 642-650, 2020/01/01/ 2020.
- [19] E. Lughofer and M. Sayed Mouchaweh, Edwin Lughofer and M. Sayed-Mouchaweh, Eds. *Predictive Maintenance in Dynamic Systems Advanced Methods, Decision Support Tools and Real-World Applications: Advanced Methods, Decision Support Tools and Real-World Applications*. Predictive Maintenance in Dynamic Systems, 2019, p. 564.

- [20] D. Wu *et al.*, "A fog computing-based framework for process monitoring and prognosis in cyber-manufacturing," *Journal of Manufacturing Systems*, vol. 43, pp. 25-34, 2017/04/01/ 2017.
- [21] G. F. Simone Cirani, Marco Picone, Luca Veltri, *Internet of Things Architectures, Protocols and Standards*. John Wiley & Sons Ltd, 2019.
- [22] A. U. R. Khan, q. f. Hassan, and S. Madani, *Internet of Things: Challenges, Advances, and Applications*. 2018.
- [23] I. Lee, "The Internet of Things for enterprises: An ecosystem, architecture, and IoT service business model," *Internet of Things*, vol. 7, p. 100078, 2019/09/01/ 2019.
- [24] K. Patel, S. Patel, P. Scholar, and C. Salazar, "Internet of Things-IOT: Definition, Characteristics, Architecture, Enabling Technologies, Application & Future Challenges," *International Journal of Engineering Science and Computing*, vol. 6, no. 5, pp. 6122 - 6131 2016.
- [25] D. Liu, H. Liang, X. Zeng, Q. Zhang, Z. Zhang, and M. Li, "Edge Computing Application, Architecture, and Challenges in Ubiquitous Power Internet of Things," (in English), *Frontiers in Energy Research*, Review vol. 10, 2022/02/22 2022.
- [26] I. Sittón-Candanedo, R. Alonso, S. Rodríguez, J. Garcia Coria, and F. De La Prieta, "Edge Computing Architectures in Industry 4.0: A General Survey and Comparison," 2020, pp. 121-131.
- [27] A. M. Rahmani *et al.*, "Exploiting smart e-Health gateways at the edge of healthcare Internet-of-Things: A fog computing approach," *Future Generation Computer Systems*, vol. 78, pp. 641-658, 2018/01/01/ 2018.
- [28] T. Yu, X. Wang, and A. Shami, "A Novel Fog Computing Enabled Temporal Data Reduction Scheme in IoT Systems," presented at the GLOBECOM 2017 - 2017 IEEE Global Communications Conference, 4-8/ 12/ 2017, 2017.
- [29] P. Ferrari *et al.*, "Performance evaluation of full-cloud and edge-cloud architectures for Industrial IoT anomaly detection based on deep learning IEEE," presented at the 2019 II Workshop on Metrology for Industry 4.0 and IoT (MetroInd4.0&IoT), 4-6/06/2019, 2019.
- [30] A. Al-Fuqaha, M. Guizani, M. Mohammadi, M. Aledhari, and M. Ayyash, "Internet of Things: A Survey on Enabling Technologies, Protocols, and Applications," *IEEE Communications Surveys & Tutorials*, vol. 17, no. 4, pp. 2347-2376, 01/07/2015 2015.
- [31] L. Perry, *Internet of Things for Architects*. Packt Publishing, 2018, p. 524.
- [32] M. D. P. Emilio, *Data Acquisition Systems: From Fundamentals to Applied Design*, 1 ed. New York: Springer Publishing Company, Incorporated, 2013, p. 135.
- [33] K. Christos and L. Mihai, "Real-Time Embedded Systems," vol. 1, C. Koulamas, Ed.: MDPI, 2017, p. 190.
- [34] J. Segers, *Analysis Techniques for Racecar Data Acquisition*. SAE, 2010.
- [35] E. Nkechi, D. Bartholomew, C. Obite, and L. Kiwu, "A Monte Carlo Simulation Comparison of Methods of Detecting Outliers in Time Series Data," *Journal of Statistics Applications & Probability*, vol. 3, no. 3, pp. 819-834, 07/25/2022 2022.
- [36] J. Y. Zhang, "Machine Learning With Feature Selection Using Principal Component Analysis for Malware Detection: A Case Study," 2019.
- [37] L. B. Shaheen, Manjula, "A Recapitulation of Imbalanced Data," *International Journal of Innovative Technology and Exploring Engineering (IJITEE)*, vol. 9, no. 3, pp. 452-455, 2020.
- [38] D. V. Mayuri Shelke; Prashant, Shandilya, "Efficient Imbalanced Data Handling Techniques through Undersampling and Oversampling Approach," *International Journal of Innovative Research in Computer and Communication Engineering*, vol. 5, no. 4, pp. 8969-8975, 01/04/2017 2017.
- [39] S. Krishnan and Y. Athavale, "Trends in biomedical signal feature extraction," *Biomedical Signal Processing and Control*, vol. 43, pp. 41-63, 2018/05/01/ 2018.
- [40] M. Stanley and J. Lee, "Sensor Analysis for the Internet of Things," *Synthesis Lectures on Algorithms and Software in Engineering*, vol. 9, no. 1, pp. 1-137, 2018/02/05 2018.
- [41] P. Jahnke, "Machine Learning Approaches for Failure Type Detection and Predictive Maintenance," *Technische Universität Darmstadt*, vol. 19, p. 83, 2015/06/19/ 2020.

- [42] A. Panesar, *Machine Learning and AI for Healthcare: Big Data for Improved Health Outcomes*, 2 ed. United States: Apress, 2019, p. 396.
- [43] R. Shetty, "Predictive Maintenance in the IoT Era," in *Prognostics and Health Management of Electronics: Fundamentals, Machine Learning, and the Internet of Things*: John Wiley and Sons Ltd, 2018, pp. 589-612.
- [44] A. N. Sudeep Tanwar, and Rudra Rameshwar, *Machine Learning in Signal Processing*. CRC Press, 2022, p. 388.
- [45] R. Barga, V. Fontama, and W. H. Tok, *Predictive Analytics with Microsoft Azure Machine Learning 2nd Edition*, 2 ed. Apress, 2015, p. 291.
- [46] S.-J. Kwon, D. Han, J. H. Choi, J.-H. Lim, S.-E. Lee, and J. Kim, "Remaining-useful-life prediction via multiple linear regression and recurrent neural network reflecting degradation information of 20Ah LiNi_xMn_yCo_{1-x-y}O₂ pouch cell," *Journal of Electroanalytical Chemistry*, vol. 858, p. 113729, 2020/02/01/ 2020.
- [47] F. Lu, J. Wu, J. Huang, and X. Qiu, "Aircraft engine degradation prognostics based on logistic regression and novel OS-ELM algorithm," *Aerospace Science and Technology*, vol. 84, pp. 661-671, 2019/01/01/ 2019.
- [48] T. Xia, P. Zhuo, L. Xiao, S. Du, D. Wang, and L. Xi, "Multi-stage fault diagnosis framework for rolling bearing based on OHF Elman AdaBoost-Bagging algorithm," *Neurocomputing*, vol. 433, pp. 237-251, 2021/04/14/ 2021.
- [49] J. Yu, "A selective deep stacked denoising autoencoders ensemble with negative correlation learning for gearbox fault diagnosis," *Computers in Industry*, vol. 108, pp. 62-72, 2019/06/01/ 2019.
- [50] S. Sharma, S. K. Tiwari, and S. Sukhjeet, "Integrated approach based on flexible analytical wavelet transform and permutation entropy for fault detection in rotary machines," *Measurement*, vol. 169, p. 108389, 2021/02/01/ 2021.
- [51] T. W. Rauber, A. L. da Silva Loka, F. d. A. Boldt, A. L. Rodrigues, and F. M. Varejão, "An experimental methodology to evaluate machine learning methods for fault diagnosis based on vibration signals," *Expert Systems with Applications*, vol. 167, p. 114022, 2021/04/01/ 2021.
- [52] Z. An, S. Li, J. Wang, and X. Jiang, "A novel bearing intelligent fault diagnosis framework under time-varying working conditions using recurrent neural network," *ISA Transactions*, vol. 100, pp. 155-170, 2020/05/01/ 2020.
- [53] B. Tarak, Z. Noureddine, and R. Said, "Deep Learning for Fault Diagnosis based on short-time Fourier transform," presented at the 2018 International Conference on Smart Communications in Network Technologies (SaCoNeT), 27-31/10/2018, 2018.
- [54] H. Lüttenberg, C. Bartelheimer, and D. Beverungen, "Designing Predictive Maintenance for Agricultural Machines," in *ECIS*, 2018.
- [55] Y. C. Liang, W. D. Li, X. Lu, and S. Wang, "Fog computing and convolutional neural network enabled prognosis for machining process optimization," *Journal of Manufacturing Systems*, vol. 52, pp. 32-42, 2019/07/01/ 2019.
- [56] G. Scalabrini Sampaio, A. R. d. A. Vallim Filho, L. Santos da Silva, and L. Augusto da Silva, "Prediction of Motor Failure Time Using An Artificial Neural Network," *Sensors*, vol. 19, no. 19, pp. 4342-4359, 2019/8/10/ 2019.
- [57] A. Jimenez-Cortadi, I. Irigoien, F. Boto, B. Sierra, and G. Rodriguez, "Predictive Maintenance on the Machining Process and Machine Tool," *Applied Sciences*, vol. 10, no. 1, p. 224, 12/27/2019 2020.
- [58] U. Shafi, A. Safi, A. R. Shahid, S. Ziauddin, and M. Q. Saleem, "Vehicle Remote Health Monitoring and Prognostic Maintenance System," *Journal of Advanced Transportation*, vol. 2018, p. 8061514, 2018/01/18 2018.
- [59] R. Yang, M. Huang, Q. Lu, and M. Zhong, "Rotating Machinery Fault Diagnosis Using Long-short-term Memory Recurrent Neural Network," *IFAC-PapersOnLine*, vol. 51, no. 24, pp. 228-232, 2018/01/01/ 2018.
- [60] P. Koprinkova-Hristova, M. Hadjiski, L. Doukovska, and S. Beloreshki, "Recurrent Neural Networks for Predictive Maintenance of Mill Fan Systems," *International Journal of Electronics and Telecommunications*, vol. 57, no. 3, 2011/09/01/ 2011.

- [61] T. Pang, T. Yu, and B. Song, "A Bayesian network model for fault diagnosis of a lock mechanism based on degradation data," *Engineering Failure Analysis*, vol. 122, p. 105225, 2021/04/01/ 2021.
- [62] M. Hosseinpour-Zarnaq, M. Omid, and E. Biabani-Aghdam, "Fault diagnosis of tractor auxiliary gearbox using vibration analysis and random forest classifier," *Information Processing in Agriculture*, vol. 9, no. 1, pp. 60-67, 2021/01/23/ 2021.
- [63] V. Gunasegaran and V. Muralidharan, "Fault Diagnosis of Spur Gear System through Decision Tree Algorithm Using Vibration Signal," *Materials Today: Proceedings*, vol. 22, pp. 3232-3239, 2020/01/01/ 2020.
- [64] M. Sayah, D. Guebli, Z. Al Masry, and N. Zerhouni, "Robustness testing framework for RUL prediction Deep LSTM networks," *ISA Transactions*, vol. 113, pp. 28-38, 2020/07/04/ 2020.
- [65] Y. Yu, C. Hu, X. Si, J. Zheng, and J. Zhang, "Averaged Bi-LSTM networks for RUL prognostics with non-life-cycle labeled dataset," *Neurocomputing*, vol. 402, pp. 134-147, 2020/08/18/ 2020.
- [66] L. Xiang, P. Wang, X. Yang, A. Hu, and H. Su, "Fault detection of wind turbine based on SCADA data analysis using CNN and LSTM with attention mechanism," *Measurement*, vol. 175, p. 109094, 2021/04/01/ 2021.
- [67] H. Kai and C. Min, *Big-data analytics for cloud, IoT and cognitive computing*. Wiley, 2017, p. 432.
- [68] Y. Xin-She, *Introduction to Algorithms for Data Mining and Machine Learning*. Academic Press, 2019, p. 188.
- [69] N. Hussain, H. T. Mirza, G. Rasool, I. Hussain, and M. Kaleem, "Spam Review Detection Techniques: A Systematic Literature Review," *Applied Sciences* vol. 9, no. 5, 08/03/2019 2019.
- [70] S. Taheri and M. A. Mammadov, "Learning the naive Bayes classifier with optimization models," in *Int. J. Appl. Math. Comput. Sci.*, 2013.
- [71] L. Florin. (2018, 2022/06/01/). *Inteligență artificială, note de curs* [Online]. Available: <http://florinleon.byethost24.com/>.
- [72] W. Verbeke, B. Baesens, and C. Bravo, *Profit driven business analytics: a practitioner's guide to transforming big data into added value*. New Jersey: John Wiley & Sons, 2018, p. 416.
- [73] P. D. Reddy and L. R. Parvathy, "Prediction Analysis using Random Forest Algorithms to Forecast the Air Pollution Level in a Particular Location," presented at the 2022 3rd International Conference on Smart Electronics and Communication (ICOSEC), Trichy, India, 20-22/10/2022, 2022.
- [74] G.-F. Fan, L.-Z. Zhang, M. Yu, W.-C. Hong, and S.-Q. Dong, "Applications of random forest in multivariable response surface for short-term load forecasting," *International Journal of Electrical Power & Energy Systems*, vol. 139, p. 108073, 2022/07/01/ 2022.
- [75] Z. Fang, X. Yu, and Q. Zeng, "Random forest algorithm-based accurate prediction of chemical toxicity to *Tetrahymena pyriformis*," *Toxicology*, vol. 480, p. 153325, 2022/10/01/ 2022.
- [76] J. Salvador-Meneses, Z. Ruiz-Chavez, and J. Rodríguez, "Compressed kNN: K-Nearest Neighbors with Data Compression," *Entropy*, vol. 21, no. 3, p. 234, 2019/02/28/ 2019.
- [77] K. Maheswari, A. Priya, A. Balamurugan, and S. Ramkumar, "Analyzing student performance factors using KNN algorithm," *Materials Today: Proceedings*, 2021/02/25/ 2021.
- [78] S. Albawi, T. A. Mohammed, and S. Al-Zawi, "Understanding of a convolutional neural network," presented at the 2017 International Conference on Engineering and Technology (ICET), 21-23 Aug. 2017, 2017.
- [79] P. Singh, *Deploy Machine Learning Models to Production*. APREES, 2021, p. 161.
- [80] X.-H. Le, H. V. Ho, G. Lee, and S. Jung, "Application of Long Short-Term Memory (LSTM) Neural Network for Flood Forecasting," *Water*, vol. 11, no. 7, p. 1387, 2019/05/28/ 2019.
- [81] P. Włodarczak, *Machine Learning and its Applications*. India: CRC Press, 2020, p. 199.
- [82] M. Schultz and M. Tropmann-Frick, "Autoencoder Neural Networks versus External Auditors: Detecting Unusual Journal Entries in Financial Statement Audits," presented at the Proceedings of the 53rd Hawaii International Conference on System Sciences, 2020/07/01/, 2020.
- [83] R. Katuwal and P. N. Suganthan, "Stacked autoencoder based deep random vector functional link neural network for classification," *Applied Soft Computing*, vol. 85, p. 105854, 2019/12/01/ 2019.

- [84] X. Feng, G. Ma, S.-F. Su, C. Huang, M. K. Boswell, and P. Xue, "A multi-layer perceptron approach for accelerated wave forecasting in Lake Michigan," *Ocean Engineering*, vol. 211, p. 107526, 2020/09/01/ 2020.
- [85] W. Verbeke, B. Baesens, and C. Bravo, *Profit-Driven Business Analytics A Practitioner's Guide to Transforming Big Data into Added Value*. New Jersey: John Wiley & Sons, Inc., 2018.
- [86] M. Swamynathan, S. Mandal, Ed. *Mastering Machine Learning with Python in Six Steps: A Practical Implementation Guide to Predictive Data Analytics Using Python*. Bangalore, Karnataka, India: Apress, 2017, p. 374.
- [87] A. Smiti, "A critical overview of outlier detection methods," *Computer Science Review*, vol. 38, p. 100306, 2020/11/01/ 2020.
- [88] S. Twumasi-Ankrah, S. K. Appiah, D. Arthur, W. A. Pels, J. K. Afriyie, and D. Nartey, "Comparison of outlier detection techniques in non-stationary time series data," *Global Journal of Pure and Applied Sciences*, vol. 27, no. 1, pp. 55-60, 2021/03/5/ 2021.
- [89] C. A. Peruri Venkata Anusha, Patnala S.R. Chandra Murthy, Ch. Surya Kiran, "Detecting Outliers in High Dimensional Data Sets using Z-Score Methodology," *International Journal of Innovative Technology and Exploring Engineering Regular Issue*, vol. 9, no. 1, pp. 48-53, 2019/11/01/ 2019.
- [90] X. Gong, F. Zhang, T. Lu, and W. You, "Comparative Analysis of three outlier detection methods in univariate data sets," presented at the 2022 3rd International Conference on Electronic Communication and Artificial Intelligence (IWEC AI), Zhuhai, China, 2022/01/14-16/, 2022.
- [91] C. Aggarwal, *Outlier Analysis*, 2 ed. USA: Springer, Cham, 2017, p. 466.
- [92] S. Chaurasia, S. Goyal, and M. Rajput, "Outlier Detection Using Autoencoder Ensembles: A Robust Unsupervised Approach," presented at the 2020 International Conference on Contemporary Computing and Applications (IC3A), Lucknow, India, 2020/02/5-7, 2020.
- [93] S. Ramaswamy, R. Rastogi, and K. Shim, "Efficient Algorithms for Mining Outliers from Large Data Sets," *ACM SIGMOD Record*, vol. 29, no. 2, pp. 427-438, 2000.
- [94] Rahul and R. Katarya, "Impact of Supervised Classification Techniques for the Prediction of Student's Performance," presented at the 2020 Fourth International Conference on I-SMAC (IoT in Social, Mobile, Analytics and Cloud) (I-SMAC), Palladam, India, 2020/11/7-9/, 2020.
- [95] Y. Zhao, "Credit Card Approval Predictions Using Logistic Regression, Linear SVM and Naïve Bayes Classifier," presented at the 2022 International Conference on Machine Learning and Knowledge Engineering (MLKE), Guilin, China, 2022/02/25-27/, 2022.
- [96] M. D. Samad, S. Abrar, and N. Diawara, "Missing value estimation using clustering and deep learning within multiple imputation framework," *Knowledge-Based Systems*, vol. 249, pp. 108968-108980, 2022/08/05/ 2022.
- [97] Y.-P. Chen, C.-H. Huang, Y.-H. Lo, Y.-Y. Chen, and F. Lai, "Combining attention with spectrum to handle missing values on time series data without imputation," *Information Sciences*, vol. 609, pp. 1271-1287, 2022/09/01/ 2022.
- [98] Y. Fu, H. Liao, and L. Lv, "A Comparative Study of Various Methods for Handling Missing Data in UNSODA," *Agriculture*, vol. 11, no. 8, pp. 727-755, 2021/07/30/ 2021.
- [99] A. Shokrzade, M. Ramezani, F. Akhlaghian Tab, and M. Abdulla Mohammad, "A novel extreme learning machine based kNN classification method for dealing with big data," *Expert Systems with Applications*, vol. 183, pp. 115293-115311, 2021/11/30/ 2021.
- [100] J. de-Prado-Gil, C. Palencia, N. Silva-Monteiro, and R. Martínez-García, "To predict the compressive strength of self compacting concrete with recycled aggregates utilizing ensemble machine learning models," *Case Studies in Construction Materials*, vol. 16, pp. e01046-e01063, 2022/06/01/ 2022.
- [101] T. Xu, Z. Peng, and L. Wu, "A novel data-driven method for predicting the circulating capacity of lithium-ion battery under random variable current," *Energy*, vol. 218, p. 119530, 2021/03/01/ 2021.
- [102] J. Wei and C. Chen, "A multi-timescale framework for state monitoring and lifetime prognosis of lithium-ion batteries," *Energy*, vol. 229, p. 120684, 2021/08/15/ 2021.

- [103] Majid Astaneh, Rodolfo Dufo-López, Ramin Roshandela, and J. L. Bernal-Agustin, "A novel lifetime prediction method for lithium-ion batteries in the case of stand-alone renewable energy systems," *International Journal of Electrical Power & Energy Systems*, vol. 103, no. 12, pp. 115–126, 2018/12/ 2018.
- [104] Z. Fei, F. Yang, K.-L. Tsui, L. Li, and Z. Zhang, "Early prediction of battery lifetime via a machine learning based framework," *Energy*, vol. 225, p. 120205, 2021/06/15/ 2021.
- [105] M. S. Hosen, J. Jaguemont, J. Van Mierlo, and M. Bercibar, "Battery lifetime prediction and performance assessment of different modeling approaches," *iScience*, vol. 24, no. 2, p. 102060, 2021/02/19/ 2021.
- [106] S. Li, H. Fang, and B. Shi, "Remaining useful life estimation of Lithium-ion battery based on interacting multiple model particle filter and support vector regression," *Reliability Engineering & System Safety*, vol. 210, p. 107542, 2021/02/01/ 2021.
- [107] K. N. Çerçi and E. Hürdoğan, "Comparative study of multiple linear regression (MLR) and artificial neural network (ANN) techniques to model a solid desiccant wheel," *International Communications in Heat and Mass Transfer*, vol. 116, p. 104713, 2020/07/01/ 2020.
- [108] M. Huang, Q. Wang, R. Jing, W. Lou, Y. Hong, and L. Wang, "Tropical cyclone full track simulation in the western North Pacific based on random forests," *Journal of Wind Engineering and Industrial Aerodynamics*, vol. 228, p. 105119, 2022/09/01/ 2022.
- [109] Y. Yang, "A machine-learning prediction method of lithium-ion battery life based on charge process for different applications," *Applied Energy*, vol. 292, p. 116897, 2021/06/15/ 2021.
- [110] K. A. Severson *et al.*, "Data-driven prediction of battery cycle life before capacity degradation," *Nature Energy*, vol. 4, no. 5, pp. 383-391, 2019/05/01 2019.
- [111] Y. Jiang and W. Song, "Predicting the Cycle Life of Lithium-Ion Batteries Using Data-Driven Machine Learning Based on Discharge Voltage Curves," *Batteries*, vol. 9, no. 8, p. 413, 2023/08/07/ 2023.
- [112] A. Altinors, F. Yol, and O. Yaman, "A sound based method for fault detection with statistical feature extraction in UAV motors," *Applied Acoustics*, vol. 183, p. 108325, 2021/12/01/ 2021.
- [113] A. Benini, F. Ferracuti, A. Moneriù, and S. Radensleben, "Fault detection of a VTOL UAV using Acceleration Measurements," presented at the 2019 18th European Control Conference (ECC), Naples, Italy, 2019/06/28/, 2019.
- [114] J.-y. Lee, W.-t. Lee, S.-h. Ko, and H.-S. Oh, "Fault Classification and Diagnosis of UAV motor Based on Estimated Nonlinear Parameter of Steady-State Model," *International Journal of Mechanical Engineering and Robotics Research*, vol. 10, no. 1, pp. 22-31, 01/01/2021/ 2021.
- [115] O. Yaman, F. Yol, and A. Altinors, "A Fault Detection Method Based on Embedded Feature Extraction and SVM Classification for UAV Motors," *Microprocessors and Microsystems*, vol. 94, p. 104683, 2022/10/01/ 2022.
- [116] S. İlkin, T. H. Gençtürk, F. Kaya Gülağız, H. Özcan, M. A. Altuncu, and S. Şahin, "hybSVM: Bacterial colony optimization algorithm based SVM for malignant melanoma detection," *Engineering Science and Technology, an International Journal*, vol. 24, no. 5, pp. 1059-1071, 2021/10/01/ 2021.
- [117] X. Lv, H. Wang, X. Zhang, Y. Liu, D. Jiang, and B. Wei, "An evolutionary SVM method based on incremental algorithm and simulated indicator diagrams for fault diagnosis in sucker rod pumping systems," *Journal of Petroleum Science and Engineering*, vol. 203, p. 108806, 2021/08/01/ 2021.
- [118] M. Nayfeh, Y. Li, K. A. Shamaileh, V. Devabhaktuni, and N. Kaabouch, "Machine Learning Modeling of GPS Features with Applications to UAV Location Spoofing Detection and Classification," *Computers & Security*, vol. 126, p. 103085, 2023/03/01/ 2023.
- [119] S. Granados-Bolaños, A. Quesada-Román, and G. E. Alvarado, "Low-cost UAV applications in dynamic tropical volcanic landforms," *Journal of Volcanology and Geothermal Research*, vol. 410, p. 107143, 2021/02/01/ 2021.
- [120] D. Mu, C. Yue, and A. Chen, "Are we working on the safety of UAVs? An LDA-based study of UAV safety technology trends," *Safety Science*, vol. 152, p. 105767, 2022/08/01/ 2022.

- [121] M. Ghasri and M. Maghrebi, "Factors affecting unmanned aerial vehicles' safety: A post-occurrence exploratory data analysis of drones' accidents and incidents in Australia," *Safety Science*, vol. 139, p. 105273, 2021/07/01/ 2021.
- [122] P. Killeen, B. Ding, I. Kiringa, and T. Yeap, "IoT-based predictive maintenance for fleet management," *Procedia Computer Science*, vol. 151, pp. 607-613, 2019/01/01/ 2019.
- [123] XAMPP. (2023, 2023/02/02/). *XAMPP Apache + MariaDB + PHP + Perl* [Online]. Available: <https://www.apachefriends.org/>.