



Programul Operațional Competitivitate

CeCBiD-EOSC

Centru Cloud și Big Data pentru participarea la Cloud-ul European pentru Știință Deschisă (CeCBiD-EOSC)

Raport Științific și Tehnic 2.4

Servicii și aplicații informatice pentru suportul activității de modelare și simulare a nanostructurilor complexe

Editor: Mihnea Dulea

Autori: George Alexandru Nemnes
Tudor Mitran
Dragos Ciobanu-Zabet
Amanda Teodora Preda
Nicolae Filipoiu

Versiunea: Finală (1.0)

Data: 27.07.2023

Distributie: Internă

Cod document: CBD_RST-2.4

Rezumat: Proiectul POC CeCBiD-EOSC a fost implementat de către echipa Departamentului Fizică Computațională și Tehnologia Informației din IFIN-HH în perioada 2020-2023. Acest document raportează rezultatele obținute în cadrul Subactivității 2.4 – *“Realizarea de servicii și aplicații informatice pentru suportul activității de modelare și simulare a nanostructurilor complexe”*. Sunt prezentate obiectivele specifice Subactivității 2.4 cu încadrare în obiectivele mai generale ale proiectului. Se descriu apoi rezultatele obținute în direcția dezvoltării și implementării unor tehnici de învățare automată (*machine learning*) pentru predicția proprietăților electronice în materiale nanostructurate. Sunt analizate sisteme finite alcătuite din grafena și nitru de bor, utilizând calcule atomistice în cadrul teoriei funcționale de densitate (*density functional theory, DFT*). Instrumentele computaționale dezvoltate sunt adaptate pentru a fi utilizate în cadrul platformei software dedicate pentru modelarea și simularea nanostructurilor. În final este descrisă interfața de acces la serviciile specializate de proiectare și investigare a nanostructurilor oferite de Centrul de resurse Cloud și Big Data (CCBD).

DREPT DE PROPRIETATE ȘI DECLINAREA RĂSPUNDERII

Acest document conține materiale ale căror drepturi de autor aparțin IFIN-HH și care nu pot fi reproduse sau copiate fără permisiune.

Utilizarea comercială a oricăror informații conținute în acest document poate necesita o licență de la proprietarul informațiilor respective.

Beneficiarul proiectului nu garantează că informațiile conținute în acest raport pot fi utilizate independent în forma în care au fost prezentate, sau că utilizarea informațiilor nu prezintă riscuri, și nu își asumă nicio răspundere pentru pierderile sau daunele suferite de orice persoană care utilizează aceste informații.

Conținutul acestui material nu reprezintă în mod obligatoriu poziția oficială a Uniunii Europene sau a Guvernului României..

Lista reviziilor

Data	Versiunea	Editor/ Autor/ Coautori	Sumarul modificărilor/completărilor principale
06.06.2023	0.1	M. Dulea	Structură document; Prefață; 1.2 Obiective
16.06.2023	0.2	G.A. Nemnes, N.Filipoiu, A. Preda	Rezumat; 1.3 Rezultate preconizate; 2. Rezultate; 3. Concluzii
17.07.2023	0.3	M. Dulea	Restructurare continut; completare 1. Introducere; 2. Implementarea...
24.07.2023	0.4	G.A. Nemnes, N.Filipoiu, A. Preda	Completari la Cap. 3-4. Cap. 5 – Adaptarea codurilor...
25.07.2023	0.5	D. Ciobanu	6. Dezvoltarea interfetei de acces ...
27.07.2027	1.0	M. Dulea	Rezumat; 7. Concluzii; redactarea finala

Prefață

Investițiile în infrastructura de Cloud computing și de Big Data în vederea maximizării potențialului de creștere a economiei digitale europene reprezintă una dintre direcțiile prioritare ale Strategiei Pieței Unice Digitale, care a fost stabilită în 2015 de către Comisia Europeană.¹

Recunoscând capabilitățile de exploatare a fenomenului Big Data oferite de tehnologia Cloud, Comisia a lansat în aprilie 2016 Inițiativa Europeană Cloud², a carei implementare se bazează pe Cloud-ul European pentru Știința Deschisă (*European Open Science Cloud* – EOSC) și pe Infrastructura Europeană de Date (*European Data Infrastructure* – EDI).

În viziunea Comisiei Europene, EOSC trebuie să asigure pentru comunitatea științifică un mediu virtual sigur, deschis, capabil să ofere servicii de stocare, management, analiză, precum și de refolosire a datelor dincolo de frontiere și discipline științifice. În acest cadru, s-a recomandat infrastructurilor europene de cercetare (și în primul rând infrastructurilor ESFRI) să promoveze reutilizarea datelor proprii pentru inovare și în scopuri educaționale prin sprijinirea conectării lor la EOSC.³

Parcursul de Implementare al EOSC⁴ a stabilit liniile de acțiune pentru crearea unei federații pan-europene a infrastructurilor de date pentru cercetare, care să înlocuiască fragmentarea existentă cu soluții eficiente și ușor de utilizat pentru stocarea, găsirea, partajarea și refolosirea datelor. Direcțiile de acțiune propuse pentru implementarea modelului federalizat al EOSC privesc arhitectura sistemului, administrarea datelor, serviciile, accesul și interfețele de acces, regulile de participare, precum și guvernanta. Arhitectura EOSC cuprinde un nucleu federativ, care include resursele partajate ale EOSC, precum și multiple infrastructuri de date federate angajate în furnizarea de servicii către EOSC.

Începând din anul 2018, IFIN-HH a contribuit la implementarea infrastructurii EOSC prin intermediul Departamentului Fizică Computațională și Tehnologia Informației (DFCTI, <https://cc.ifin.ro>), care a participat, în calitate de asociat al coordonatorului, Fundația EGI⁵, la proiectul H2020 EOSC-Hub⁶ (2018-2020), destinat dezvoltării resurselor și serviciilor Cloud inițiale pentru EOSC. Sarcina DFCTI a fost de a furniza, prin intermediul centrului Cloud CLOUDIFIN⁷, resurse pentru susținerea diferitelor comunități de utilizatori, precum și de a contribui cu servicii naționale la catalogul de servicii al EOSC, în conformitate cu regulile de angajare ale proiectului. În acest scop, EOSC-Hub a finanțat activitatea de management și operare a site-ului CLOUDIFIN, asigurând continuitatea furnizării acestor servicii.

Pentru a putea finanța realizarea masei critice de resurse necesară participării la EOSC, DFCTI a propus proiectul CeCBiD-EOSC în cadrul apelului POC 398/2018. Totodată, pentru continuarea implementării serviciilor specifice EOSC, DFCTI participă, începând din 2020, la proiectul H2020 EGI-ACE – „*Advanced Computing for EOSC*” (2020-2023), a cărui misiune este de a asigura servicii EOSC gratuite pentru cercetătorii din toate disciplinele științifice care necesită calcule intensive și Big Data. Astfel, proiectele EGI-ACE și CeCBiD-EOSC acționează complementar, la nivelul EU și, respectiv, național, pentru realizarea strategiei de integrare a infrastructurii de calcul și de date a IFIN-HH în EOSC.

Obiectivul general al proiectului CeCBiD-EOSC este creșterea capacității de cercetare în scopul ridicării nivelului de competitivitate științifică pe plan intern și internațional al IFIN-HH, prin

¹ „A Digital Single Market Strategy for Europe” - COM(2015) 192

² „European Cloud Initiative – Building a competitive data and knowledge economy in Europe” – COM (2016) 178

³ „Long-term sustainability of Research Infrastructures” – SWD(2017) 323

⁴ „Implementation Roadmap for the European Open Science Cloud” - SWD(2018) 83

⁵ Fundația EGI, <https://www.egi.eu/about/egi-foundation/>

⁶ „Integrating and managing services for the EOSC”, <https://www.eosc-hub.eu/>

⁷ Centrul de resurse Cloud al DFCTI, CLOUDIFIN, <http://cloudifin.ifin.ro/>

modernizarea infrastructurii Cloud, extinderea infrastructurii masive de date si realizarea unui centru de date cu performante inalte, care să fie integrat în infrastructura Cloud Europeană pentru Știința Deschisă.⁸

Totodata, proiectul propune o soluție tehnică pentru interconectarea la nivel național, în cadrul unui Cloud federalizat, a centrelor de tip Cloud privat dezvoltate în instituții aparținând sistemului de CDI, capabilă să ofere utilizatorilor acces printr-o interfață unică la resurse și servicii furnizate de aceste centre. Implementarea acestei soluții va eficientiza utilizarea resurselor Cloud de către grupurile de cercetători și va stimula semnificativ cooperarea între specialiștii în tehnologii informatice avansate.

Infrastructura realizată în cadrul proiectului va susține dezvoltarea unor activități de CDI în domeniile sistemelor de calcul paralel și distribuit, învățării automatizate, calculului științific și bioinformaticii, cu aplicații relevante pentru fizica materiei condensate, studiul interacției laser-materie, nanofizică și nanoelectronică.

Obiectivele specifice ale proiectului CeCBiD-EOSC sunt următoarele:

1. Realizarea unui centru performant de resurse Cloud și Big Data prin achiziționarea și instalarea de active corporale și necorporale necesare pentru derularea activităților de CDI prevăzute în proiect.
2. Dezvoltarea și diversificarea serviciilor furnizate la nivel european de către centrul CLOUDIFIN în perspectiva integrării acestuia în EOSC.
3. Realizarea în cadrul centrului de resurse Cloud a unei soluții tehnice capabilă să interconecteze la nivel național, în cadrul unui sistem federalizat, centrele de tip Cloud privat dezvoltate în instituții aparținând sistemului de CDI, capabilă să ofere utilizatorilor acces printr-o interfață unică la resurse și servicii furnizate de către aceste centre.
4. Asigurarea condițiilor de susținere informațională în tehnologie Cloud și Big Data a participării institutului la colaborări internaționale de anvergură, precum și a noilor opțiuni strategice privind angajarea în direcții de cercetare emergente din spațiul științific internațional, cu relevanță socio-economică deosebită.
5. Dezvoltarea și implementarea de servicii și aplicații informatice pentru administrarea și funcționarea centrului de resurse, care utilizează tehnologii Cloud și Big Data pentru: satisfacerea cerințelor IT din faza operațională a proiectului Extreme Light Infrastructure - Nuclear Physics (ELI-NP); modelarea și simularea la nivel molecular a nano- și bio-sistemelor complexe; analiza datelor de secvențiere de nouă generație.
6. Realizarea condițiilor tehnice și asigurarea suportului de specialitate pentru obținerea și/sau îmbunătățirea de către parteneri economici, în special din cadrul Clusterul Tehnologic Magurele (MHTC), a unor produse și servicii în domenii de specializare inteligentă, care vor conduce la creșterea competitivității acestora.
7. Formarea și perfecționarea personalului calificat, precum și transferul de cunoștințe în domeniile Cloud computing și Big Data către personalul științific și tehnic din alte entități ale sistemului de CDI.
8. Diseminarea rezultatelor proiectului prin participarea la conferințe (inter)naționale și publicarea de articole științifice în parteneriat public-privat.

Prin obiectivele sale, proiectul va conduce la extinderea capacității resurselor Cloud și Big Data, precum și la îmbunătățirea calitativă și diversificarea serviciilor de calcul și de analiza de date pe care Centrul de Calcul Avansat din IFIN-HH le va oferi comunității științifice naționale și internaționale, contribuind prin aceasta la dezvoltarea sistemului național de CDI și la creșterea vizibilității la nivel european

Rezultatele prevăzute ale proiectului sunt prezentate mai jos în ordinea specificată în Cererea de Finanțare (RP*i* reprezintă Raportul de Progres nr. *i* în care se descrie realizarea rezultatul respectiv).

⁸ Cerere de Finanțare, proiect CeCBiD-EOSC

Nr.	Rezultat prognozat	Documentul în care a fost raportat
1.	Documentatia de achizitie a serviciilor de consultanta pentru elaborarea documentatiei tehnice necesare echiparii centrului de resurse Cloud si Big Data	RP2
2.	Contract de furnizare a serviciilor de consultanta pentru elaborarea documentatiei tehnice necesare echiparii centrului de resurse Cloud si Big Data	RP2
3.	Documentatie tehnica privind echiparea centrului de resurse Cloud si Big Data	RP3
4.	Documentatia de achizitie a serviciilor de informare si publicitate	RP1
5.	Contract de achizitie servicii de informare si publicitate	RP1
6.	Contracte de achizitie active corporale pentru echiparea centrului de resurse Cloud si Big Data	-
7.	Un comunicat de presa publicat la lansarea proiectului	RP1
8.	Un comunicat de presa publicat la finalizarea proiectului	
9.	Pagina web a proiectului, publicata la adresa http://cecbid-eosc.nipne.ro	RP1
10.	Materiale de informare si publicitate (roll-up-uri, afise, rame afis, mape, banner).	RP1
11.	Sistem de procesare de date achizitionat si instalat. Raport tehnic	-
12.	Sistem de stocare de date achizitionat si instalat. Raport tehnic.	-
13.	Switch pentru interconectarea sistemelor hardware achizitionat si instalat. Raport tehnic.	-
14.	Instalatie de climatizare achizitionata si instalata. Raport tehnic.	-
15.	Sistem UPS achizitionat si instalat. Raport tehnic.	-
16.	Rack-uri pentru gzduirea echipamentelor IT achizitionate si instalate.	-
17.	Tablouri electrice si retea electrica achizitionate si instalate. Raport tehnic.	-
18.	Servicii si aplicatii informatice pentru administrarea si monitorizarea centrului CLOUDIFIN, precum si pentru asigurarea accesului utilizatorilor.	RP1-4
19.	Interfata de acces al utilizatorilor la resurse oferite de centre Cloud multiple. Manual de utilizare.	-
20.	Servicii si aplicatii informatice pentru satisfacerea cerintelor IT din faza operationala initiala a proiectului ELI-NP. Raport stiintific si tehnic	-
21.	Servicii si aplicatii informatice pentru suportul activitatii de modelare si simulare a nanostructurilor complexe (partial)	RP1-4
22.	Servicii si aplicatii informatice pentru suportul activitatii de analiza a datelor de secventiere de noua generatie (partial)	RP1-4
23.	Studiu privind performantele centrului Cloud si Big Data. Manual de management	-

24.	Lucrări și comunicări științifice	RP2-4
25.	Fișe de post	RP1-2
26.	Documente de raportare (rapoarte de activitate / de progres; procese verbale ale întâlnirilor echipei de management a proiectului)	RP1-4
27.	Cereri de plată/rambursare	RP4
28.	Raport de audit final al proiectului.	-

Proiectul CeCBiD-EOSC a demarat la data semnării contractului de finanțare de către Ministerul Educației și Cercetării (19.05.2020), iar finalizarea lui este planificată pentru data de 31.07.2023.

Proiectul CeCBiD-EOSC este cofinanțat din Fondul European de Dezvoltare Regională (FEDR) în baza contractului de finanțare încheiat cu Ministerul Educației și Cercetării în calitate de Organism Intermediar, în numele și pentru Ministerul Fondurilor Europene în calitate de Autoritate de Management.

Cuprins

1. Introducere	11
1.1 CONTEXT GENERAL ȘI NECESITATE	11
1.2 OBIECTIVELE SUBACTIVITĂȚII 2.4	12
1.3 REZULTATE PRECONIZATE.....	12
2. Implementarea mediului de dezvoltare software	13
3. Determinarea proprietatilor electronice in formalismul DFT.....	14
3.1 SISTEMELE GRAFENA - NITRURA DE BOR HEXAGONALA INVESTIGATE	14
3.2 CALCULE DFT HTC PENTRU DETERMINAREA SPECTRULUI ELECTRONIC.....	15
4. Determinarea proprietatilor electronice utilizand tehnici ML.....	16
4.1 DEZVOLTAREA TEHNICILOR DE INVATARE AUTOMATA.....	16
4.2 PREDICTIA FAZEI DE ECHILIBRU SI A PROPRIETATILOR MECANICE IN ALIAJE CU ENTROPIE RIDICATA	20
4.3 PREDICTIA DENSITATII DE SARCINA PENTRU EFICIENTIZAREA BUCLEI DE SELF-CONSISTENTA.....	21
5. Adaptarea codurilor pentru implementare pe platforma software a CCBD	22
6. Dezvoltarea interfetei de acces la platforma de modelare a nanostructurilor...	24
7. Concluzii	28
8. Multumiri	29
ANEXA	30

Lista figurilor

- Fig. 3.1. Structuri tipice CBN pasivate cu hidrogen avand geometrii diferite
- Fig. 3.2: Valorile R^2 pentru antrenarea folosind exemple din clasa s-NFs (a) si h-NFs (b)
- Fig. 4.1: Trasaturi posibile pentru o raza de taiere ce include ordinul $k=3$ ($k \times a_0 = 1.42 \text{ \AA}$).
- Fig. 4.2 Fluxul de lucru pentru determinarea gap-ului electronic
- Fig. 4.3: Reducerea numarului de trasaturi utilizand VT pentru (a) s-NFs si (b) h-Nfs.
- Fig. 4.4: Metode de selectie a trasaturilor: VT, MIT) si RCR.
- Fig. 4.5: Structuri de tip HEA (high entropy alloys) de tip Co-Cr-Fe-Ni-Al in faza FCC si BCC.
- Fig. 4.6: Identificarea fazelor FCC vs. BCC in functie de constanta de retea.
- Fig. 4.7: Maparea densitatii de sarcina cu rețele neuronale echivariante la rotatie
- Fig. 4.8: Reprezentare schematica a unei RE-ANN planare cu doua straturi
- Fig. 6.1: Antetul biroului virtual pentru utilizatorul *perla-pv*
- Fig. 6.2: Biroul virtual pentru utilizatorul autorizat sa foloseasca CBN_ANN
- Fig. 6.3: Energy_gap_prediction.png
- Fig. 6.4: Interfata SIESTA
- Fig. 6.5: Interfata de acces la R-MATRIX
- Fig. 6.6: Interfata R-MATRIX

Rezumat

Scopul livrabilului

Scopul acestui raport este de a prezenta rezultatele obținute în cadrul Subactivității 2.4 a proiectului CeCBiD-EOSC, privind *"Realizarea de servicii și aplicații informatice pentru suportul activității de modelare și simulare a nanostructurilor complexe"*.

Impact

RST2.4 este un livrabil cheie al proiectului, care descrie rezultatele științifice noi obținute în domeniul modelării și simulării nanostructurilor prin metode *ab initio* și de învățare automată, evidențiind impactul atât prin 3 articole științifice publicate, cât și prin implementarea metodelor numerice dezvoltate în cadrul platformei software dedicate.

Conținutul livrabilului

După o scurtă introducere privind contextul general, necesitatea, obiectivele și rezultatele preconizate ale subactivității 2.4, în capitolul 2 se prezintă mediul de dezvoltare software implementat pe infrastructura de calcul. În capitolul următor se descriu sistemele grafen-nitrura de bor investigate și metodologia de calcul a gap-ului electronic în formalismul funcționalei de densitate (DFT). Capitolul 4 este consacrat principalelor rezultate științifice, având în prim-plan metodele de învățare automată dezvoltate, detalii despre implementarea acestora și aplicarea lor la diferite sisteme fizice. În Cap. 5 se prezintă principalele puncte ale adaptării aplicațiilor dezvoltate în vederea implementării acestora pe platforma Cloud a CCBD. Capitolul 6 este dedicat activității de dezvoltare a interfeței web de acces al utilizatorului la platforma de calcul a CCBD.

Concluziile livrabilului

Subactivitatea 2.4 se încheie cu realizarea rezultatului planificat nr. 21 al proiectului - *" 21. Servicii și aplicații informatice pentru suportul activității de modelare și simulare a nanostructurilor complexe. Raport științific și tehnic."*

În cadrul Subactivității 2.4 a fost realizat un flux de lucru (*workflow*) care constă din (i) efectuarea calculelor de tip DFT în regim de high-throughput și (ii) aplicarea tehnicilor de învățare automată (ML) pentru eficientizarea explorării unui număr mare de sisteme candidat. Aceste rezultate s-au concretizat în 3 publicații științifice indexate ISI. De asemenea, instrumentele computaționale dezvoltate au fost adaptate pentru spre a fi pe mai departe utilizate pe platforma software dedicată.

1. Introducere

1.1 Context general și necesitate

Obiectivul specific O5 al proiectului include "Dezvoltarea și implementarea de servicii și aplicații informatice pentru administrarea și funcționarea centrului de resurse, care utilizează tehnologii Cloud și Big Data pentru modelarea și simularea la nivel molecular a nano- și biosistemelor complexe". Acest obiectiv se atinge în cadrul Subactivității 2.4 – "Realizarea de servicii și aplicații informatice pentru suportul activității de modelare și simulare a nanostructurilor complexe".

Modelarea nanodispozitivelor cu proprietăți fizice noi (electronice, optice, mecanice, de conducție electronică și termică) reprezintă o direcție de cercetare de mare actualitate, abordată cu succes și în IFIN-HH, cu aplicații potențiale în multiple domenii de specializare inteligentă.

În cadrul metodelor de modelare *ab initio*, formalismul funcționalei de densitate (*density functional theory* - DFT)⁹ s-a bucurat de un succes deosebit în descrierea proprietăților electronice atât în sisteme anorganice cât și organice, de la semiconductori și aplicații ale acestora în nanotehnologie până la modelarea sistemelor de interes biologic. În prezent, există o multitudine de pachete software care implementează DFT, atât în modul standard cât și prin intermediul unor metode post-DFT, cum ar fi metoda DFT în aproximația GW, metoda DFT cu corecție de termen Hubbard sau metoda funcționalei de densitate dependente de timp, care aduc corecții, în spectrul energetic. Unul din elementele cheie care influențează eficacitatea, dar și acurătatea calculului DFT este reprezentat de setul de elemente de bază; alegerea unui set de funcții de bază localizate poate crește semnificativ eficiența calculului, asigurând o scalare liniară a timpului de calcul cu dimensiunea sistemului.

Metoda DFT a fost implementată în pachetul software SIESTA¹⁰, utilizând orbitali atomi numerici cu suport finit ca elemente de bază, ceea ce permite atât abordarea unor sisteme mari (clusteri atomici, interfețe, metamateriale) cu mii de atomi în celula unitate, cât și o bună acurătate.

Proiectarea computațională a nanodispozitivelor semiconductoare utilizând modelarea DFT pentru descrierea proprietăților electronice necesită frecvent timp de procesor de ordinul zilelor pentru o singură rulare cu pachetul SIESTA și numeroase repetiții ale rularilor, care se efectuează în paralel.

Recent, necesitatea accelerării și automatizării calculului *ab-initio* de modelare a nanostructurilor a condus la soluții inovatoare capabile de implementare în infrastructura CLOUDIFIN, bazate pe tehnici de tip învățare automată¹¹ (*machine learning* – ML), cum sunt *Linear Multiple Regression* MLR, *Decision Tree* DT, *Support Vector Machines* SVM, *Artificial Neural Networks* - ANNs, *Random Forests* RF, etc. Aceste soluții oferă metode alternative, prin care se vizează o descriere mai rapidă și totuși suficient de exactă a proprietăților electronice, chiar și în absența rezolvării explicite a sistemului de ecuații Kohn-Sham, având la bază un set de exemple rezolvate anterior. Aplicații concrete includ descrierea suprafețelor de energie potențială în cadrul simularilor de dinamică moleculară sau evaluarea benzii interzise în noi materiale. Investigarea structurii electronice fără rezolvarea sistemului Kohn-Sham accelerează semnificativ calculele de dinamică moleculară, care presupun evaluări succesive ale parametrilor structurali pentru același sistem.

Aplicarea metodelor ML a avut o contribuție însemnată la prezicerea proprietăților unor noi compuși, bio- sau nanostructuri, pornind de la rezultatele DFT disponibile în baze de date extinse. Utilizând avantajele introduse de SIESTA, se pot efectua în paralel calcule DFT construind un set amplu de exemple destinat ca input pentru tehnicile de tip învățare automată. Prin aceste calcule este vizată corelarea diferitelor proprietăți electronice, optice sau mecanice cu proprietățile structurale ale materialelor respective.

⁹ A.J. Cohen et al, Chem. Rev. 2012, 112, 289–320

¹⁰ Spanish Initiative for Electronic Simulations with Thousands of Atoms (SIESTA), <https://departments.icmab.es/leem/siesta>

¹¹ F. Brockherde et al., Nature Commun. 8, 872 (2017); J. Lee et al., Phys. Rev. B93, 115104 (2016)

1.2 Obiectivele Subactivității 2.4

În cadrul subactivității 2.4 s-a prevăzut realizarea unei platforme software gazduita în infrastructura Cloud, pentru accelerarea și automatizarea calculelor ab-initio de modelare și simulare a nanostructurilor utilizând metode de tip Machine Learning.

Pentru implementarea tehnicilor de tip ML s-a avut în vedere utilizarea unor pachete software *open-source / freeware* specializate, cum sunt TensorFlow, Keras, Theano, Caffe, LIBSVM, FANN.

Realizarea platformei software în cadrul CLOUDIFIN include testarea și optimizarea procedurii de modelare și simulare a nanostructurilor, pentru care s-au prevăzut următorii pași:

1. Implementarea mediului de dezvoltare software, prin instalarea pachetelor *open source* dedicate calculelor de tip *ab initio*: SIESTA, ABINIT, Quantum ESPRESSO. Instalarea pachetelor și bibliotecilor software destinate segmentului de învățare automatizată (TensorFlow, Keras, etc).
2. Efectuarea de calcule folosind teoria funcționalei de densitate (DFT) pentru determinarea proprietăților electronice (de exemplu largimea benzii interzise), optice (de exemplu coeficientul de absorbție) și mecanice (de exemplu modulul de elasticitate) în materiale nanostructurate.
3. Stabilirea parametrilor utilizați ca input în simulările prin intermediul rețelelor neuronale artificiale (ANN): elemente care caracterizează sistemul din punct de vedere structural, compozițional, distribuția speciilor chimice etc.
4. Obținerea de rezultate prin calculele DFT, care vor constitui exemple pentru algoritmi de tip ML, de exemplu pentru antrenarea rețelelor neuronale; se va realiza maparea input-output pentru procedura de tip ML.
5. Testarea algoritmilor ML (e.g. a rețelei neuronale antrenate pe o clasă de sisteme) folosind structuri noi. Predicțiile oferite prin procedurile ML sunt ulterior validate prin calcule DFT, determinându-se astfel atât acuratetea predicției, cât și eficiența soluției de tip ML comparativ cu calculele standard de tip DFT.

1.3 Rezultate preconizate

Rezultatul general planificat pentru această subactivitate în Cererea de Finanțare este 21 - *Servicii și aplicații informatice pentru suportul activității de modelare și simulare a nanostructurilor complexe.*

Acest obiectiv include:

- Determinarea proprietăților electronice în nanomateriale noi de interes actual, utilizând formalismul funcționalei de densitate (DFT);
- Dezvoltarea tehnicilor de aplicare a metodelor de învățare automată pentru modelarea nanostructurilor; antrenarea și testarea rețelelor neurale artificiale.
- Adaptarea codurilor numerice de modelare prin metode de învățare automată în vederea integrării acestora în platforma software dedicată.
- Dezvoltarea și implementarea interfeței de acces la serviciile de modelare și simulare a nanostructurilor asigurate de platforma software gazduita de Centrul de Resurse Cloud și Big Data (CCBD).

2. Implementarea mediului de dezvoltare software

Mediul software necesar pentru dezvoltarea și testarea aplicațiilor de tip ML integrate în platforma de modelare și simulare a nanostructurilor a fost implementat mai întâi pe infrastructura existentă a clusterului CLOUDIFIN, urmând ca ulterior să fie migrat pe echipamentele CCBD, după achiziționarea acestora în cadrul proiectului.

S-au instalat și testat bibliotecile software open source *TensorFlow* și *Keras*, integrate în Python, pentru implementarea rețelelor neurale artificiale (*artificial neural networks*, ANN), precum și alte biblioteci suport Python (de ex. *NumPy*). Au fost efectuate teste ale librăriilor TensorFlow și Keras, punându-se în evidență capacitatea de rulare pe *thread*-uri multiple în infrastructura.

TensorFlow¹² (TF) facilitează definirea și manipularea expresiilor matematice ce implică matrici multidimensionale (tensori). Tensorii circulă printr-un graf de calcul, care reprezintă operațiile și dependențele dintre diferiții tensori și permite procesarea unor baze de date de dimensiuni mari. TF gestionează automat calculele și optimizările și permite atât proiectarea arhitecturii modelului cât și a procesului de antrenare al modelului de învățare automată.

Printre avantajele utilizării bibliotecii TF pentru acest proiect se numără: definirea și configurarea rețelelor neuronale, manipularea datelor de intrare, antrenarea modelelor cu diverși algoritmi de optimizare și evaluarea performanței acestora. TF suportă atât tehnici tradiționale de învățare automată, cât și abordări "*deep learning*", folosind diverse tipuri de modele, cum ar fi rețelele neuronale convoluționale (CNN) și rețelele neuronale recurente (RNN). În plus, TF oferă instrumente pentru calcul distribuit, permițând scalarea modelelor și proceselor de antrenare pe mai multe CPU sau GPU.

Keras¹³ este o interfață de programare a aplicațiilor (API) de nivel înalt pentru rețele neurale, care oferă o gamă largă de straturi predefinite, cum ar fi straturile dense, straturile convoluționale, straturile recurente și altele, care pot fi îmbinate și configurate ușor pentru a crea arhitecturi complexe.

TF se poate integra cu Keras, utilizarea celor două biblioteci fiind esențială pentru construirea, antrenarea și implementarea modelelor de învățare automată abordate în acest proiect.

Este de notat că ambele biblioteci acceptă atât stilul de API secvențial, care permite construirea modelelor prin adăugarea secvențială a straturilor, cât și pe cel funcțional, care oferă mai multă flexibilitate, permițând crearea de modele cu mai multe intrări și ieșiri sau cu straturi comune.

Mediul de dezvoltare software a fost testat în cadrul unui prim studiu privind eficientizarea mapării datelor de intrare pe datele de ieșire vizate pentru predicții, structurile investigate încadrându-se în clasa sistemelor bidimensionale de tip grafena - nitru de bor hexagonală (hBN), având diferite forme geometrice.

În cadrul studiului s-a urmărit selectarea din multitudinea de elemente pentru vectorii de intrare (unde un element reprezintă un quadruplet atomic) a unui set relativ redus care să reproducă consistent valorile de output de interes (în acest caz gap-ul energetic).

Pentru aceasta au fost efectuate în prealabil calcule folosind teoria funcționalei de densitate, utilizând pachetul SIESTA, pe sisteme finite de grafena-hBN cu geometrii de tip pătrat și hexagon, pentru etapa de antrenare fiind considerate 3000 de probe pentru fiecare geometrie, în timp ce pentru validare a fost utilizat un set distinct de 1000 probe.

În urma rezultatelor promitatoare obținute, aceste structuri au fost selectate pentru realizarea studiilor descrise în capitolele următoare.

¹² TensorFlow, <https://www.tensorflow.org/>

¹³ Keras, <https://keras.io/>

3. Determinarea proprietatilor electronice in formalismul DFT

3.1 Sistemele grafena - nitrura de bor hexagonala investigate

Generarea unor intervale energetice interzise in grafena prin insertia de structuri de nitrura de bor hexagonala cu diferite geometrii este un subiect de interes actual in fizica nanodispozitivelor semiconductoare. Pentru un numar relativ mic de ordinul sutelor de atomi, modelarea *ab initio* (DFT) a structurilor optime pentru obtinerea proprietatilor electronice dorite prin simularea diferitelor configuratii posibile necesita un timp indelungat de calcul, ceea ce face necesara dezvoltarea unor tehnici de tip *machine learning* (ML) pentru eficientizarea calculului DFT.

In cadrul proiectului s-a realizat investigatia numerica detaliata a sistemelor bidimensionale grafena - nitrura de bor hexagonala (C-BN) in scopul stabilirii unor proceduri (fluxuri de lucru) pentru identificarea seturilor de trasaturi (*features*) optime in vederea eficientizarii procesului de antrenare/validare/testare, in conditiile mentinerii acuratetei de predictie.

Sistemele C-BN analizate, cu domenii aleatorii si pasivate cu hidrogen, se grupeaza in cele doua clase de "fulgi" de carbon (*carbon nanoflakes* - NF) indicate in Fig. 3.1: cu forma patrata (*square NanoFlakes* - s-NFs) sau hexagonala (*hexagonal NanoFlakes* - h-NFs).

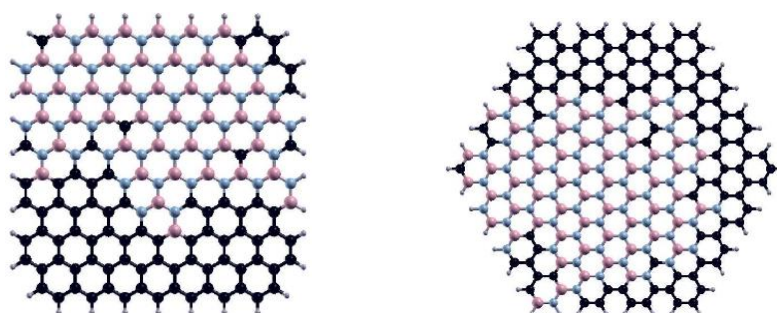


Fig. 3.1: Structuri tipice CBN pasivate cu hidrogen avand geometrii diferite

Acuratetea de predictie creste cu numarul de trasaturi (*features*), care reprezinta grupuri compacte de atomi de diferite dimensiuni (raze de taiere, R_c), alese intr-o prima etapa aleatoriu pentru fiecare dintre cele doua clase de sisteme.

Intr-o etapa preliminara, pregatitoare pentru investigatiile din Cap. 4, s-a analizat transferabilitatea setului de trasaturi intre cele doua clase de sisteme compozitional similare, dar diferite din punct de vedere geometric. Astfel, s-a stabilit un set comun, relativ restrans, de 29 trasaturi si a fost investigata acuratetea de predictie antrenand rețele neurale pe un tip de structuri si testand pe celalalt tip.

Rezultatele indica faptul ca acuratetea de predictie se modifica semnificativ atunci cand antrenarea si testarea se face pe seturi diferite, ceea ce denota faptul ca rețelele neurale utilizate invata proprietati specifice celor doua tipuri de geometrii. Aceste rezultate sunt reproduse in Fig. 3.2, unde se reprezinta valorile calculate ale coeficientului de determinare statistica R^2 .

Din punct de vedere al arhitecturii, rețele neurale utilizate in acest test au avut 3 straturi ascunse, cu 300, 50, respectiv 25 neuroni, avand un singur neuron la iesire si un numar de neuroni egal cu numarul de trasaturi la intrare.

In etapa de antrenare a fost utilizat optimizatorul Adam cu rata de invatare 10^{-5} , iar ponderile pe sinapse au fost initializate cu o distributie aleatorie (uniforma).

Rezultatele obtinute evidentiaza faptul ca rețelele neurale implicate in acest studiu sunt capabile sa capteze informatii care diferentiaza cele doua clase de sisteme.

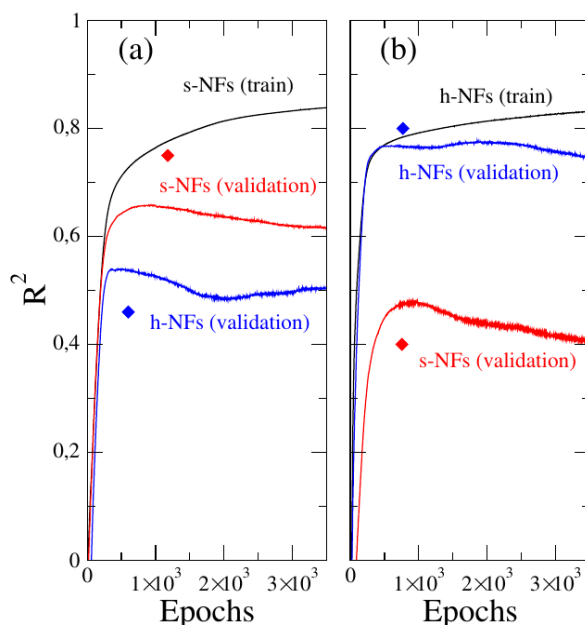


Fig. 3.2: Valorile R^2 pentru antrenarea folosind exemple din clasa s-NFs (a) și h-NFs (b)

3.2 Calcule DFT HTC pentru determinarea spectrului electronic

Intr-o prima faza a studiului, au fost efectuate calcule *ab initio* de tip *high throughput computing* (HTC), utilizand infrastructura CLOUDIFIN, pentru a determina gap-ul energetic din spectrul electronic, care este definit ca diferenta dintre energiile LUMO (*lowest unoccupied molecular orbital*) și HOMO (*highest occupied molecular orbital*) ale structurilor NF:

$$E_{\text{gap}} = E_{\text{LUMO}} - E_{\text{HOMO}}$$

Calcululele DFT au fost implementate folosind pachetul SIESTA. Acesta utilizeaza seturi de baza strict localizate, care permit scalarea liniara a timpului de calcul cu dimensiunea sistemului. Astfel, pachetul SIESTA reprezinta o optiune de calcul eficient pentru un numar mare de sisteme, cu numar relativ mare de atomi, care se preteaza la aplicatii care combina formalismul DFT cu tehnicile de invatare automata.

Implementarea DFT s-a realizat in aproximatia densitatii locale electronice (*Local Density Approximation* – LDA), in parametrizarea Ceperley-Alder¹⁴, folosind setul de baza standard DZP (*double-zeta polarized*)¹⁵ si pseudopotentiale de tip Troullier-Martins¹⁶.

Calcululele DFT indica o distributie a gap-urilor electronice preponderent in intervalul 0-0.3 eV pentru sistemele cu geometrie patrata, in timp ce distributia pentru sistemele hexagonale este mai larga, ajungand la valori de ~ 0.8 eV. La ambele clase de sisteme se observa o crestere statistica a gap-ului cu proportia de nitrura de bor, ceea ce este de asteptat avand in vedere gap-ul mare al materialului ideal (nitrura de bor hexagonala, hBN). La valori mici ale proportiei de hBN se observa insa diferente: in timp ce la sistemele cu geometrie patrata gap-ul scade catre zero, similar sistemului grafena ideal, in cazul geometriei hexagonale, marginile de tip armchair induc un gap finit de ~ 1 eV chiar si in cazul sistemului de grafena.

¹⁴D.M. Ceperley, B.J. Alder, "Ground State of the Electron Gas by a Stochastic Method", Phys. Rev. Lett. 45, 566 (1980).

¹⁵Jose M Soler et al 2002, "The SIESTA method for ab initio order-N materials simulation", J. Phys.: Condens. Matter 14, 2745 (2002)

¹⁶N. Troullier, J.L. Martins, "Efficient pseudopotentials for plane-wave calculations", Phys. Rev. B 43, 1993 (1991)

4. Determinarea proprietatilor electronice utilizand tehnici ML

4.1 Dezvoltarea tehnicilor de invatare automata

In continuare, studiile s-a concentrat pe dezvoltarea tehnicilor ML utilizate pentru investigarea proprietatilor electronice in sistemele de tip nanofulgi de grafena (C-BN) selectate anterior, folosind rezultatele obtinute din calcule de tip *ab initio* descrise in Cap. 3.2. S-au implementat diferite arhitecturi de rețele neuronale artificiale ("artificial neural networks" - ANN) si s-au folosit cele doua biblioteci dedicate, TensorFlow (TF) si Keras integrate in cadrul limbajului de programare Python, care au fost prezentate in Cap. 2.

Primul pas in construirea modelului de invatare automata a fost preprocesarea datelor de intrare obtinute din simularile DFT intr-o forma ideala pentru ANN.

Fiecare atom din nanostructura si vecinii sai de ordinul I formeaza un cvadrupelet atomic care constituie o trasatura din cele 22 posibile. Aceasta distributie a fost ulterior normata, dupa care s-au mai adaugat inca 4 intrari care reprezinta proportiile speciilor atomice H, B, C, N. Stratul de intrare este astfel compus din 26 de neuroni, in timp ce output-ul este un singur neuron – valoarea gap-ului de energie corespunzator fiecarei configuratii calculate initial prin simularile *ab initio*. Pentru procesul de antrenare, *output*-ul a fost de asemenea normat la valoarea de 4 eV.

Pentru fiecare din cele doua clase s-NFs si h-NFs au fost generate in total 3.200 de sisteme/configuratii, care au furnizat 3 seturi de date utilizate pentru: antrenarea ANN (3.000 configuratii), validare/optimizarea modelului (100 de configuratii) si test (100 de configuratii).

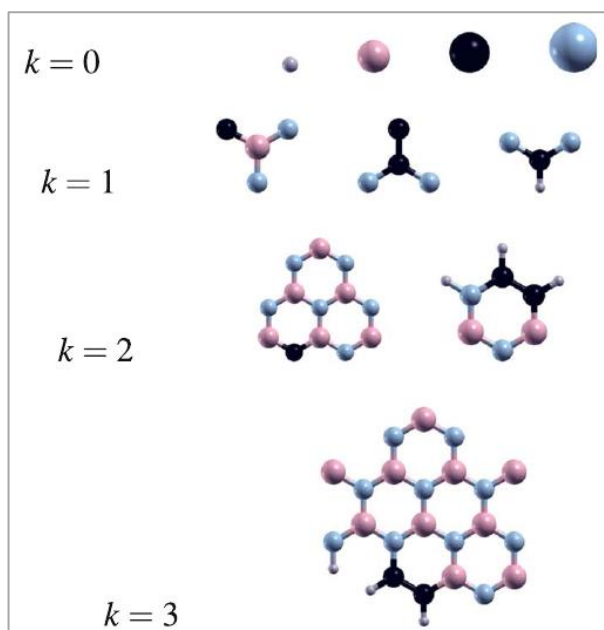


Fig. 4.1: Trasaturi posibile pentru o raza de taiere ce include ordinul $k=3$ ($k \times a_0 = 1.42 \text{ \AA}$).

Complementar metodei propuse care utilizeaza cvadrupeleti atomici si proportiile speciilor, a fost dezvoltata o metoda mai generala pentru identificarea trasaturilor de interes, corelate cu valorile de output, adica cu gap-ul electronic. Aceste trasaturi se obtin folosind o raza de taiere in jurul unui atom din proba, precum in exemplele indicate in Fig. 4.1.

Aceste trasaturi au insa o pondere inegala in determinarea gap-ului electronic, iar numarul lor creste rapid cu raza de taiere, ajungand la ordinul zecilor de mii.

În această situație se impune implementarea unor proceduri adecvate de selecție a trasaturilor, care se poate face în regim nesupervizat sau supervizat. În prima situație, metoda de selecție acționează exclusiv pe datele de intrare, în timp ce în a doua situație este analizată corelația dintre datele de intrare și cele de ieșire.

În consecință, fluxul de lucru pentru determinarea gap-urilor trebuie să includă, pe lângă procedura de identificare a trasaturilor și de contabilizarea a apariției acestora, o procedură de selecție, care poate fi supervizată (în cazul în care sunt utilizate datele de ieșire) sau nesupervizată (în cazul contrar).

Schema fluxului de lucru este reprezentată în Fig. 4.2.¹⁷

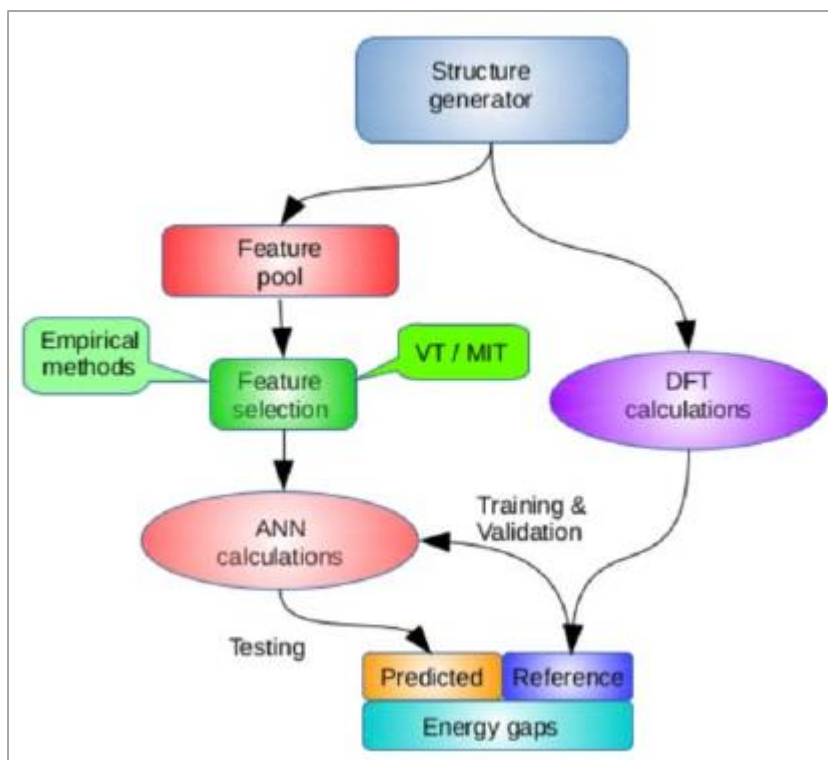


Fig. 4.2 Fluxul de lucru pentru determinarea gap-ului electronic

După generarea structurilor de tip NF, se parcurg două subfluxuri paralele. În primul se determină gap-urile energetice prin calcule în formalismul funcției de densitate descris în Cap.3.2, urmând ca rezultatele obținute să servească în operațiunile de antrenament și validare. În cel de-al doilea se identifică trasaturile, urmând o etapă esențială de selecție, înainte de utilizarea acestora în modelul ANN.

Pentru calculele efectuate pe fiecare din cele două cai se utilizează resurse hardware diferite.

Selecția nesupervizată a fost realizată folosind algoritmul *variance threshold* (VT) din pachetul *scikit-learn*.¹⁸

Reducerea numărului de trasaturi s-a realizat prin eliminarea trasaturilor care se repetă foarte des / foarte rar în setul de antrenare. Vectorul de trasaturi este un vector binar, în care "1" și "0" indică prezența, respectiv absența unei trasaturi.

Varianta distribuției de tip Bernoulli este $p(1-p)$, unde p este probabilitatea de apariție a unei trasaturi. Astfel, pragul (*threshold* - t) selectat este determinat de probabilitatea de apariție a unei trasaturi, $t = p(1-p)$, adică varianta distribuției.

¹⁷ T.L. Mitran, G.A. Nemnes, "Ground state charge density prediction in C-BN nanoflakes using rotation equivariant feature-free artificial neural networks", *Carbon*, 174, 276 (2021)

¹⁸ <https://scikit-learn.org>

Rezultatul procedurii de reducere a numarului de trasaturi este prezentat in Fig. 4.3.

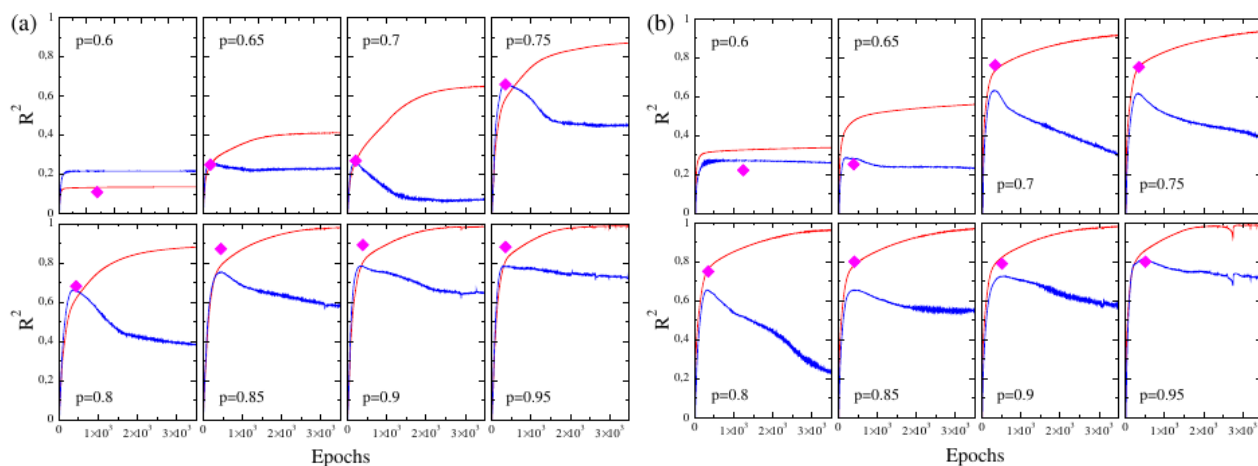


Fig. 4.3: Reducerea numarului de trasaturi utilizand VT pentru
(a) s-NFs si (b) h-NFs.

Pentru ambele sisteme se observa o diminuare vizibila a acuratetei de predictie odata cu micsorarea pragului si, totodata, a numarului de trasaturi.

Totusi, se constata ca aceasta scadere a acuratetei se face relativ lent, ceea ce indica faptul ca reducerea numarului de trasaturi are efecte semnificative doar atunci cand setul de trasaturi este mult diminuat.

In tabelul de mai jos, unde N_f reprezinta numarul de trasaturi rezultat din procedura de selectie de tip *variance threshold*, se poate observa ca aceasta diminuare a acuratetei se face relativ abrupt in jurul valorii $p=0.65-0.7$, care corespunde la un numar de aproximativ 30 de trasaturi. Peste aceasta valoare, se obtine o plafonare a acuratetei (0.75-0.85), pentru un numar de trasaturi cuprins in intervalul 30-100.

Trebuie mentionat ca numarul de trasaturi initial este 204, care sunt alese aleator, respectiv 4 specii atomice si 25x8 trasaturi corespunzatoare celor 8 valori R_c utilizate, care stabilesc dimensiunile clusterilor atomici.

p	0.6	0.65	0.7	0.75	0.8	0.85	0.9	0.95
N_f (s-NFs)	11	25	31	40	46	58	90	95
R_{val}^2 (s-NFs)	0.21	0.24	0.25	0.65	0.65	0.74	0.77	0.78
R_{test}^2 (s-NFs)	0.11	0.25	0.27	0.66	0.68	0.87	0.89	0.88
N_f (h-NFs)	19	28	39	49	56	62	70	86
R_{val}^2 (h-NFs)	0.25	0.28	0.62	0.60	0.64	0.65	0.71	0.80
R_{test}^2 (h-NFs)	0.22	0.25	0.76	0.75	0.80	0.79	0.78	0.80

Tabel 1. Variatia acuratetei de predictie pentru seturile de validare si test

In continuare se are in vedere o procedura supervizata de selectie a trasaturilor bazata pe *mutual information threshold (MIT)*. Aceasta procedura tine cont si de rezultatele de iesire, ceea ce poate conduce la o selectie mai performanta.

Pentru evaluarea metodei MIR este considerat acelasi set de referinta alcatuit din 204 trasaturi. Prin stabilirea unei valori de prag T_{MIR} , este selectat un subset de trasaturi, urmarindu-se maximizarea informatiei mutuale. Pentru valori $T_{\text{MIR}} = 0.01 - 0.7$ se obtin seturi cu un numar de 94, pana la 22 de trasaturi in cazul structurilor de tip s-NF (*square nanoflakes*) si, respectiv, seturi cu un numar de 96, pana la 16 trasaturi in cazul structurilor de tip h-NF (*hexagonal nanoflakes*).

In Fig. 4.4 este prezentata acuratetea de predictie corespunzatoare unei selectii aleatoare, in comparatie cu metoda RCR (*ranking of the cutoff radius*). In metoda RCR sunt selectate

preferential trasaturile urmarind cresterea graduala a razei de taiere. O alegere aleatoare a celor 204 trasaturi, constand in selectia unui numar predefinit (25 trasaturi pentru $k>1$ si, respectiv toate trasaturile pentru $k=0$ si $k=1$) pentru fiecare clasa de trasaturi definite de raza de taiere $R_c^{(k)}$, aduce o crestere relativ lenta a acuratetei de predictie cu numarul de trasaturi N_f , valori optime ($R^2 \sim 0.8$) atingandu-se pentru $N_f \sim 150$.

In contrast, in cazul alegerii preferentiale a trasaturilor rezultatele sunt semnificativ imbunatatite, ceea indica rolul esential jucat de trasaturile de dimensiuni mici, in special cele constituite din cadrupleti atomici. Totusi, desi metoda RCR functioneaza relativ bine, aceasta este adaptata analizei claselor de sisteme in cauza. In general, este de dorit o selectie cu un grad mai mare de generalitate.

Predictiile obtinute folosind metodele de selectie VT si MIT sunt indicate in figura 1(b). Ambele demonstreaza faptul ca se poate obtine o reducere semnificativa a dimensiunii vectorului de trasaturi, prin comparatie cu o selectie aleatoare, mentinand acuratetea de referinta.

De asemenea, s-au investigat utilizarea metodelor VT si MIT in succesiune, rezultatele indicand o reducere de 5-10% in comparatie cu aplicarea metodei VT. Aceasta imbunatatire relativ redusa se poate pune si pe seama faptului ca numarul de trasaturi este relativ mic.

In concluzie, desi setul de trasaturi este mult mai condensat in cazul metodei RCR, metodele VT si MIR pun la dispozitie o selectie consistenta a setului de trasaturi, fiind totodata metode generale, cu larga aplicabilitate, ce pot fi implicate in eficientizarea modelelor de tip retea neurala artificiala cu aplicatii in modelarea nanostructurilor prin intermediul calculelor de tip DFT.

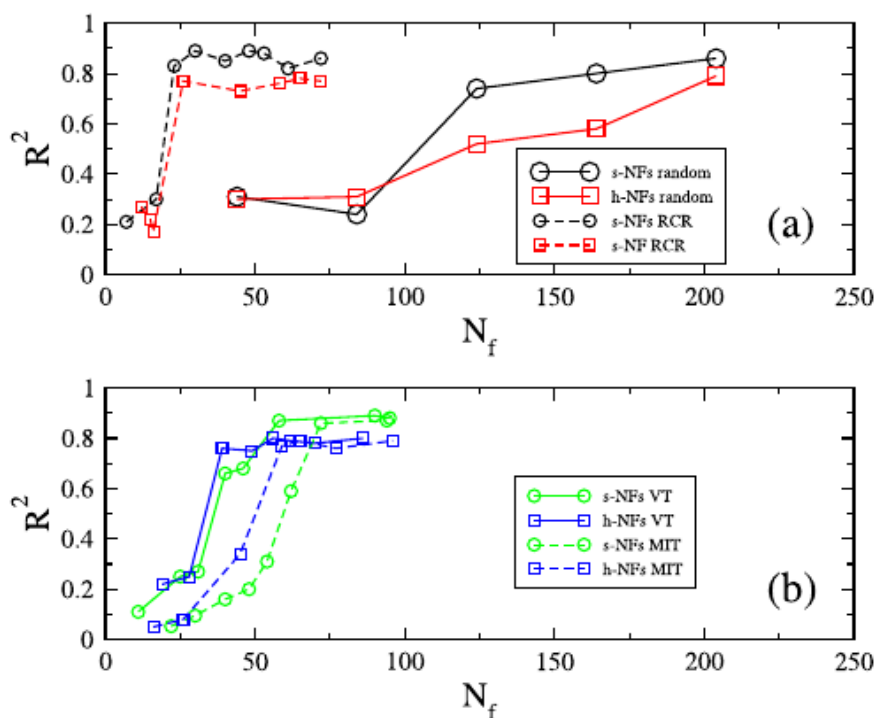


Figura 4.4: Metode de selectie a trasaturilor: VT, MIT) si RCR.

Rezultatele acestui studiu au fost publicate in lucrarea "Feature selection procedures for combined density functional theory—artificial neural network schemes", Physica Scripta 96, 065807 (2021), autori G.A. Nemnes, N. Filipoiu, V. Sipica.

4.2 Predictia fazei de echilibru si a proprietatilor mecanice in aliaje cu entropie ridicata

De asemenea, au fost implementate modele de tip rețele neurale artificiale pentru investigarea proprietatilor electronice si mecanice in sisteme de tip HEA (*high entropy alloys*) [Fig. 4.5], utilizand calcule de tip *ab initio*. Acestea sunt aliaje metalice alcatuite din cel puțin 5 elemente, unele compozitii avand proprietati mecanice, magnetice si magnetocalorice deosebite. Avand in vedere numarul imens de posibilitati in generarea unor sisteme HEA, un calcul exhaustiv nu este posibil.

In acest studiu este analizata capacitatea de predictie a unor marimi de interes in predictia fazei de echilibru: energia totala, energia de coeziune, densitate de stari la nivelul Fermi si relatia stres-deformare. Este ales un set de 3000 sisteme HEA cu 5 elemente (Co,Cr,Fe,Ni,Al) in diferite proportii. Dintre acestea, 2000 de exemple sunt utilizate pentru antrenare, in timp de 500 sunt pentru validarea modelului si 500 pentru testare.

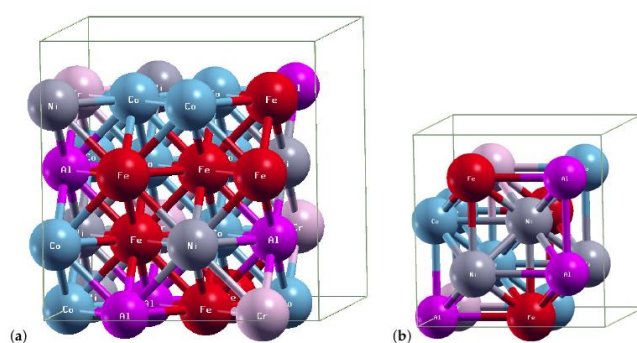


Fig. 4.5: Structuri de tip HEA (high entropy alloys) de tip Co-Cr-Fe-Ni-Al in faza FCC si BCC.

Un rezultat important este predictia fazei de echilibru, FCC vs. BCC, in sisteme HEA [Fig. 4.6]. Pentru a determina faza de echilibru se genereaza sisteme in ambele faze candidate si se modifica treptat constanta de retea urmarindu-se minimul energiei totale. Pentru fiecare dintre cele doua faze candidate, FCC si BCC, se obtin astfel doua minime care se pot compara. In Fig. 4.6 se observa ca modelul ML este capabil sa reproduca energia totala suficient de precis, astfel incat faza de echilibru, FCC, este prezisa cu acuratete.

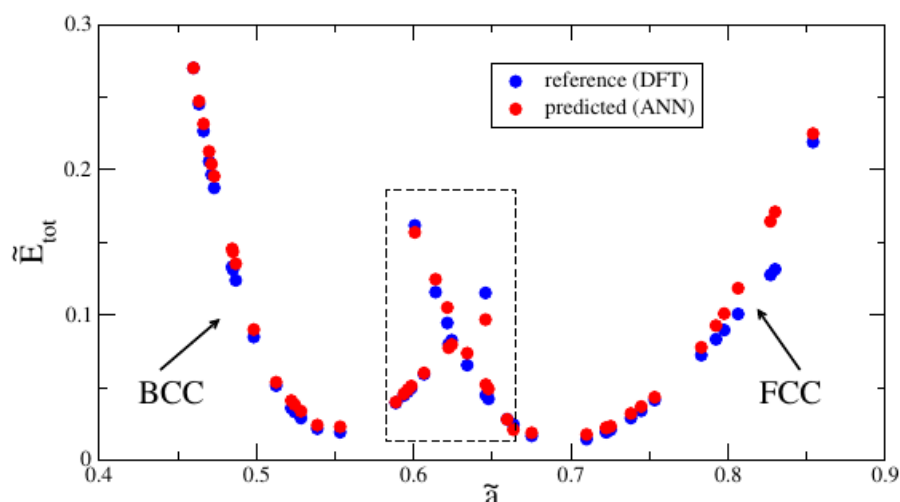


Fig. 4.6: Identificarea fazelor FCC vs. BCC in functie de constanta de retea.

Rezultatele studiului au fost prezentate in lucrarea "Prediction of Equilibrium Phase, Stability and Stress-Strain Properties in Co-Cr-Fe-Ni-Al High Entropy Alloys Using Artificial Neural Networks", *Metals* 10, 1569 (2020), autori N. Filipoiu, G.A. Nemnes.

4.3 Predictia densitatii de sarcina pentru eficientizarea buclei de self-consistenta

Un alt studiu de interes pentru eficientizarea buclei de self-consistenta se bazeaza pe predictia densitatii electronice a starii de echilibru prin metode de invatare automata.

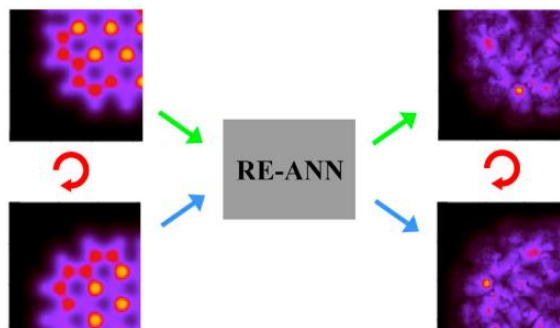


Fig. 4.7: Maparea densitatii de sarcina cu rețele neuronale echivariante la rotatie

In cadrul proiectului au fost dezvoltate urmatoarele aspecte tehnice ale metodei: a) implementarea unei noi arhitecturi de rețele neuronale echivariante la rotatie - RE-ANN (Fig. 4.7, unde in stanga este densitatea de sarcina a atomilor izolati in doua orientari distincte, iar in dreapta – maparea diferentei densitatii de sarcina prezise, pastrand orientarea corecta; b) paralelizarea (prin multithreading) sectiunii de algoritmi raspunzatoare de reprezentarea echivarianta in interiorul rețelelor si optimizarea fluxului de prelucrare a datelor de antrenare a rețelelor si a datelor produse prin predictia rețelelor.

Scopul acestei abordari bazate pe RE-ANN este de a aproxima densitatea de sarcina a starii de echilibru pentru diverse sisteme atomice fara a fi nevoie de efortul numeric impus de simulari clasice *ab initio* DFT. Au fost studiate o serie de structuri atomice heterogene (cu dezordine atomica) de tip nano-fulg alcatuite din carbon si nitrura de bor. RE-ANN au fost antrenate pe un set de 7.1 milioane de valori ale densitatii de sarcina, au fost validate pe un set de 4.8 milioane si, ulterior, testate pe 130 de milioane de valori de test. Datele de intrare reprezinta densitatile de sarcina ale atomilor individuali iar rețelele incearca sa prezica din aceste valori densitatile de sarcina din starea de energie minima pentru rețelele atomice.

Rețele neuronale echivariante au fost dezvoltate prin impunerea unor restrictii de simetrie internă ale conexiunilor dintre neuroni [Fig. 4.8, unde seturile de ponderi care trebuie constranse pentru a pastra echivarianta sunt reprezentate in culori diferite (albastru/rosu)]. Numeric, acest aspect induce un efort computational suplimentar, dar predictiile sunt de o calitate mai buna decat in cazul rețelelor clasice.

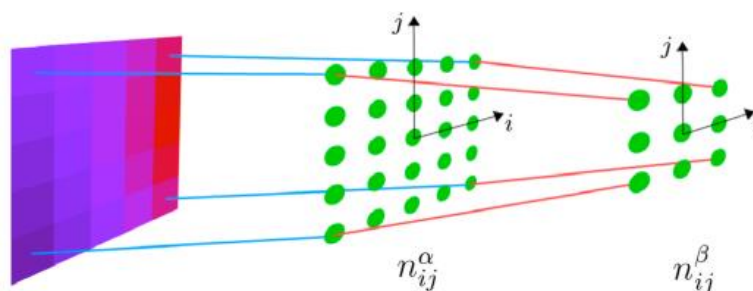


Fig. 4.8 Reprezentare schematica a unei RE-ANN planare cu doua straturi

Rezultatele acestui studiu au fost publicate in articolul "Ground state charge density prediction in C-BN nanoflakes using rotation equivariant feature-free artificial neural networks", Carbon, 174, 276 (2021), autori T.L. Mitran, G.A. Nemnes.

5. Adaptarea codurilor pentru implementare pe platforma software a CCBD

Aplicatiile Python dezvoltate pentru predictia gap-ului electronic au fost adaptate pentru a fi integrate in platforma dedicata serviciilor de modelare a nanostructurilor.

Eficienta codului a fost mai intai testata pentru mai multe tipuri de date de intrare.

Preprocesarea datelor folosite in codurile de invatare automata este un pas esential, in special modul de selectare al trasaturilor ("features") care descriu cel mai bine sistemul fizic de interes.

De exemplu, pentru sistemul de nanofulgi de grafena cu domenii de nitru de bor, datele de intrare pot fi alese intr-o maniera intuitiva ca un vector de 166 de neuroni, unde fiecare neuron reprezinta un singur atom. Desi aceasta abordare duce la rezultate foarte bune pentru reseaua neuronală (coeficienti de determinare de peste 90%), metoda nu este scalabila si nu poate fi aplicata direct pentru sisteme de alte dimensiuni, deoarece intregul process de antrenare si optimizare ar trebui reluat.

In consecinta, a fost pastrata metoda de selectare a trasaturilor descrisa in Cap. 4, in care s-au clasificat fiecare atom si vecinii sai de ordinul I in 22 de clase si s-au adaugat inca 4 neuroni pentru a tine cont de compozitia chimica. Aceasta metoda are avantajul de a fi scalabila, deci algoritmul poate fi utilizat pentru sisteme din aceeasi clasa de diferite dimensiuni. Datele de iesire au ramas neschimbate – gap-urile energetice ale structurilor simulate, normate la valoarea de 4 eV.

Pentru a putea vizualiza rezultatele, aceasta versiune a codului va afisa si un grafic in care se vor reprezenta valorile prezise de model si valorile reale din simularile DFT. De asemenea, va fi afisat coeficientul de determinare calculate pentru datele de test, o metrica de testare a performantei modelului.

Pasul urmator a fost pregatirea codului si rescrierea lui intr-o forma care poate fi integrate pe cloud si care ofera si posibilitatea unei interactiuni intre utilizator si modelul de invatare automata.

Codurile de invatare automata sunt versatile in cadrul unei anumite clase de sisteme, pe care modelul a fost antrenat initial. Pentru a imbunatati gradul de aplicabilitate al programului si pe alte sisteme 2D, este necesara modificarea unor parametrii esentiali pentru reseaua neuronală in etapa de invatare (antrenare), precum: numarul de epoci ("*epochs*"), dimensiunea subsetului de date ("*batch size*"), numarul de neuroni din straturile ascunse ale retelei sau procentul de date de test si validare. Se ofera utilizatorului aplicatiei optiunea de a modifica aceste elemente cheie descrise mai sus si de a analiza modul in care schimbarea acestor descriptori ai retelei influenteaza coeficientul de determinare si implicit exactitatea predictiilor.

Primul parametru pe care il va putea modifica utilizatorul este de numarul de epoci pe care se face antrenarea. Pe parcursul unei epoci, algoritmul parcurge complet setul de date de antrenare si prin "*backpropagation*" se actualizeaza ponderile sinapselor care conecteaza neuronii. Într-o epoca, modelul primește date de intrare, le procesează prin straturile sale și produce predicții. Alegerea numărului potrivit de epoch-uri depinde de diferiți factori, precum complexitatea problemei, dimensiunea setului de date și comportamentul de convergență al modelului.

De asemenea, numărul optim de epoci trebuie ales pentru a obține un echilibru între învățarea corectă și timpul ideal de antrenare. Utilizarea unui număr prea mic de epoci poate duce la un model care nu capteaza correct trasaturile datelor, în timp ce utilizarea unui număr prea mare de epoci poate duce la "*overfitting*" (modelul devine prea specializat pe datele de antrenare și are o performanta redusa pe datele de test). Găsirea echilibrului potrivit și determinarea numărului optim de epoci implică adesea experimentare și monitorizarea performanței modelului pe un set de validare în timpul antrenării. Aceste predictii sunt apoi evaluate prin compararea datelor prezise cu cele de test si este calculat parametrul R^2 pe care il pot vizualiza ca "output" pe platforma.

Un alt parametru esential la care utilizatorul are acces este "*batch size*" (dimensiunea subsetului - lot) si se referă la numărul de exemple din datele de antrenare prelucrate simultan de către model într-o singură iterație. În timpul antrenării, setul de date de antrenament este împărțit în subseturi mai mici, iar fiecare subset conține un număr specific de exemple din datele de antrenare. Prin ajustarea dimensiunii batch-size -ului, utilizatorii pot gestiona consumul de memorie și optimiza viteza de antrenare în funcție de resursele disponibile.

Înțelegerea și ajustarea parametrilor în timpul antrenării are un impact semnificativ asupra performanței modelului. Prin accesul la modificarea numărului de neuroni, epocilor și dimensiunii lotului, utilizatorii pot optimiza și îmbunătăți performanța rețelei neuronale.

Pentru a simplifica etapa de preprocesare a datelor, s-a fost folosit pachetul Python *scikit-learn*. Rețeaua are 14 straturi ascunse, fiecare cu un număr diferit de neuroni, de la 600 în primul strat, până la 25 de neuroni la ultimul.

Funcția de activare optimă a fost "ReLU" (*rectified linear unit*), care este în general potrivită pentru cazurile care au date de intrare pozitive, și am păstrat aceeași funcție de activare în fiecare strat al rețelei.

Parametrul *batch size* a fost setat la 100, iar optimizatorul ales a fost "Adam".

În timpul procesului de antrenare s-a urmărit eroarea patratice medie pentru a determina punctul optim în care algoritmul va face predicții cât mai exacte fără să intre în "*over-fitting*" (când eroarea patratice medie pe setul de validare și cea pe setul de antrenare diverg - eroarea pe setul de validare crește, iar pe setul de antrenare continuă să scadă).

Pentru a măsura gradul de acuratețe al predicțiilor s-a folosit coeficientul de determinare R^2 și s-au obținut valori de peste 85%.

Codul programat este reprodus în ANEXA.

6. Dezvoltarea interfeței de acces la platforma de modelare a nanostructurilor

În ultima parte a Subactivității 2.4 s-a dezvoltat și implementat interfața grafică de acces la platforma software de modelare a nanostructurilor, în care au fost integrate aplicațiile realizate în cadrul proiectului.

Accesul este restricționat. Un utilizator poate folosi serviciile de calcul științific asigurate de CCBD pentru un domeniu științific doar dacă este adăugat de către administratorul CCBD în lista utilizatorilor autorizați să folosească pachetele software respective.

După autentificarea prin interfața realizată în cadrul subactivității 2.1, utilizatorul are acces la biroul sau virtual, care îi prezintă lista serviciilor pentru care are autorizare.

De exemplu, în Fig. 6.1 se prezintă antetul biroului virtual pentru un utilizator care este autorizat să folosească atât serviciile de modelare din domeniul biologiei computaționale (NAMD și TAVERNA) cât și pe cele specifice pentru nanofizică (SIESTA, R-MATRIX, CBN_ANN).

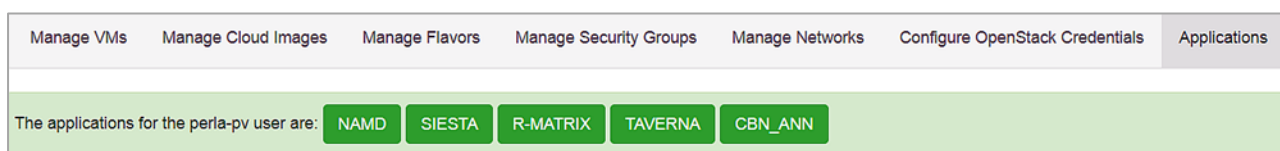


Fig. 6.1: Antetul biroului virtual pentru utilizatorul *perla-pv*

Accesul la aplicația de modelare a nanostructurilor C-BN prin utilizarea rețelelor neurale artificiale (ANN), descrisă în capitolul precedent, se face prin apăsarea butonului "CBN_ANN".

În urma acestei operațiuni se deschide fereastra web a paginii de modelare, reprezentată în Fig. 6.2 de mai jos.

Name	View	Execute	Delete	Download
x.dat				
y.dat				
Energy_gap_prediction.png				
saved_models				
cbn_ann.py				

Fig. 6.2: Biroul virtual pentru utilizatorul autorizat să folosească CBN_ANN

Lansarea aplicației *cbn_ann.py* se realizează prin apăsarea butonului "Execute". Comanda este transmisă către un agent software care rulează pe resursa ce deține acest pachet software.

Pentru aceasta a fost necesara programarea aplicatiei web care trebuie sa transmita mesaje tip text catre resurse.

Astfel, mesajul text de forma "`username|python cbn_ann.py`" este transmis prin protocol MQTT pe topicul "`cloud/req/cloudifin/bchs31/cbn_ann`".

Dupa receptionarea mesajului de catre agentul software, se trece la segmentarea mesajului folosind limitatorul "|". In urma segmentarii sunt generate doua elemente: `username` si comanda care va executa codul de ML pe resursa respectiva. Elementul `username` va fi folosit pt directionarea fisierelor de iesire (`output`) catre utilizatorul care a executat codul de ML.

Astfel, fiecare utilizator poate incarca propriile fisiere de input (`x.dat` si `y.dat` in Fig 6.2), iar fisierele de iesire (`output`) rezultante pot fi accesate doar de catre acesta.

Pe langa butonul de "`Execute`", interfata web mai pune la dispozitie si facilitati de descarcare (`download`), incarcare (`upload`), stergere (`delete`) si de vizualizare a continutului fisierelor.

Dupa procesarea codului de ML sunt generate doua fisiere. Un fisier tip imagine "`Energy_gap_prediction.png`" si un fisier "`cbn_trained_model.h5`", care este trimis in directorul `saved_models` din Fig 6.2.

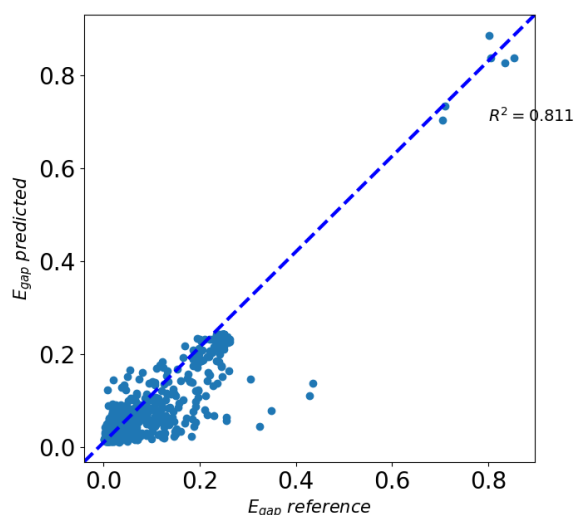


Fig. 6.3: Energy_gap_prediction.png

Pentru **modelarea ab initio** descrisa in Cap. 3, utilizatorul autorizat va apasa butonul "`SIESTA`", deschizand interfata web reprezentata in Fig. 6.4.

Pentru executia acestei aplicatii utilizatorul are posibilitatea sa aleaga atat tipul resursei de calcul - procesare CPU sau procesare GPU, cat si tipul de hardware – server fizic sau virtualizat. In functie de aceasta optiune la apasarea butonului de "`Request Console Access`", mesajul transmis de aplicatia web este directionat catre agentul care ruleaza pe resursa de calcul solicitata.

Cererea este rezolvata in momentul in care pe "`Active Console`" apare in lista aplicatia solicitata si o adresa URL. Acest URL directioneaza utilizatorul catre terminalul de acces al resursei de calcul.

ID	OS Username	Cloud Name	Application	Request Access	Enable	Access Path	Resource	Init Date/Time
20	perla-pv	CLOUDIFIN	siesta	cpu	1	http://cloud-ctrl.nipne.ro:6080/vnc_auto.html?path=%3Ftoken%3D215656a4-7e18-42a1-9c3b-21ba94c3b176	perla-10	2023-03-09 11:53:52.448598
22	perla-pv	CLOUDIFIN	namd	cpu	1	http://cloud-ctrl.nipne.ro:6080/vnc_auto.html?path=%3Ftoken%3D2eac90aa-4e98-45ea-a808-71e6248d99ae	perla-12	2023-03-09 12:50:25.454073
30	perla-pv	CLOUDIFIN	r_matrix	cpu	1	http://194.102.58.37:8000	duals-d.cloudifin.nipne.ro	2023-05-04 14:32:37.590560
31	perla-pv	CLOUDIFIN	namd	gpu	1	http://cloud-ctrl.nipne.ro:6080/vnc_auto.html?path=%3Ftoken%3D2e8d9e776-a004-4088-9ea0-437e5a05f06e	perla-gpu	2023-06-15 09:39:10.541164

Fig. 6.4: Interfata SIESTA

Din motive de securitate, dupa o perioada de timp adresa URL atribuita expira, iar utilizatorul este nevoit sa apese din nou butonul "Request Console Access" daca doreste acces la terminalul resursei de calcul.

Mentionam ca pentru acest tip de aplicatie se folosesc instante virtualizate care sunt deja pornite si gata de lucru in Cloud.

Metoda matricii R (R-Matrix) este utila pentru rezolvarea problemelor de imprastiere in mecanica cuantica, fiind folosita pentru calculul conductantei dispozitivelor cuantice cu transport coerent de sarcina.

Metoda a fost implementata in interfata platformei de servicii a CCBD pentru calculul transmisiei ca functie de energie pentru un potential arbitrar unidimensional.

Prin apasarea butonului "R-MATRIX" din antetul biroului virtual se deschide interfata reprezentata in Fig. 6.5, in care se pot vizualiza instructiunile de utilizare si resursele disponibile aplicatiei.

Aplicatia este instalata si configurata pe o instanta de masina virtuala care se afla initial in modul "pauza".

In momentul in care un utilizator solicita acces la aceasta aplicatie trebuie sa apese pe butonul "Request WEB Access" din Fig. 6.5. Dupa ce este apasat butonul respectiv, instanta de masina virtuala trece in modul "activ".

Dupa trecerea in acest mod, in lista de "Active Console" este afisat si link-ul catre interfata web a aplicatiei, reproducuta in Fig.6.6.

Interfata aplicatiei R-MATRIX ofera posibilitatea utilizatorului a folosi fisiere de input predefinite ca exemple, dar si posibilitatea de a incarca propriile fisiere de input.

The applications for the perla-pv user are: **NAMD** **SIESTA** **R-MATRIX** **TAVERNA**

The R-Matrix method for calculating transmission functions

About R-Matrix simulation

The R-Matrix method is used to solve scattering problems in quantum mechanics. This implementation(*) computes the transmission function for an arbitrary 1D potential as a function of energy. Input: You can submit the input data directly from this page, or you can upload your own input file (.txt/.in). Output: The output is generated as a plain text file. You can download it, send it to your email or see the T(E) plot.

Request WEB Access

Active Console

ID	OS Username	Cloud Name	Application	Request Access	Enable	Access Path	Resource	Init Date/Time
20	perla-pv	CLOUDIFIN	siesta	cpu	1	http://cloud-ctrl.nipne.ro:6080/vnc_auto.html?path=%3Ftoken%3D215956a4-7e18-42a1-9bcb-21ba64cdb176	perla-10	2023-03-09 11:53:52.448596
22	perla-pv	CLOUDIFIN	namd	cpu	1	http://cloud-ctrl.nipne.ro:6080/vnc_auto.html?path=%3Ftoken%3D2eac998aa-4e98-45ea-a808-71e6249d99ae	perla-12	2023-03-09 12:50:25.454073
30	perla-pv	CLOUDIFIN	r_matrix	cpu	1	http://194.102.58.37:8000	dual6-d.cloudifin.nipne.ro	2023-05-04 14:32:37.590560
31	perla-pv	CLOUDIFIN	namd	gpu	1	http://cloud-ctrl.nipne.ro:6080/vnc_auto.html?path=%3Ftoken%3D2e8d9e77d-a004-4088-9ea0-f37b5a06706e	perla-gpu	2023-06-15 09:39:10.541164

Fig. 6.5: Interfața de acces la R-MATRIX

Dupa alegerea fisierului de intrare (exemplu sau definit de utilizator) se apasa butonul "Compute" care lanseaza in executie aplicatia. Fisierul rezultat poate fi descarcat si/sau reprezentat grafic de catre utilizator.

The R-Matrix method for calculating transmission functions

About our R-Matrix simulation

We use the R-Matrix method to solve a scattering problem in quantum mechanics. This demo implementation computes the transmission function for an arbitrary 1D potential as a function of energy. Input: You can submit the input data directly from this page, or you can upload your own input file (.txt/.in) (see below). Output: The output is generated as a plain text file. You can download it, send it to your email or see the T(E) plot.

Input format

The input file should have the following format (please see example file below)

- Effective_mass value, where 'value' represents the effective mass of the electrons.
- Nb value, where 'value' represents the number of basis components.
- Nx value, where 'value' represents the number of points in the scattering region.
- nxl value, where 'value' represents the number of points in the left lead.
- nxr value, where 'value' represents the number of points in the right lead.
- d1 value, where 'value' represents the half length scattering region [m].
- d2 value, where 'value' represents the half length leads + scattering region [m].
- N_energies value, where 'value' represents the number of energies.
- E_min value, where 'value' represents the min energy [eV].
- E_max value, where 'value' represents the max energy [eV].
- BLOCK Potential tokens, just for informing the algorithm where the potential block starts.
- 'Nx' values, representing the potential values [eV].
- END_BLOCK Potential tokens, just for informing the algorithm where the potential block ends.

Example input file format -- A rectangular barrier with length 100 nm and height 0.1 eV.

You can edit the values in order to obtain your own input file. It is important NOT to remove the tokens (Effective_mass, Nb, Nx, etc.). After you are done, press the 'Compute' button below to calculate the transmission. This example takes ~40 s.

```
Effective_mass 0.023
Nb 512
Nx 1024
nrx 1024
nxr 1024
d1 0.5e-7
d2 1.0e-7
N_energies 200
E_min 0.0
E_max 0.2
```

Compute

Or you can upload your own file (.txt/.in). Make sure the file has the above indicated format.

Choose File No file chosen

Compute

Fig. 6.6: Interfața R-MATRIX

7. Concluzii

In cadrul Subactivitatii 2.4 s-au propus si s-au implementat in infrastructura Centrului de resurse Cloud si Big Data (CCBD) doua metodologii de calcul pentru determinarea eficienta a proprietatilor electronice in materiale nanostructurate – simulari *ab initio* si modelare utilizand tehnici de invatare automata (*machine learning*).

Simularile *ab initio* utilizeaza calcule de tip DFT in regim de *high throughput computing*, implementate pe masini virtuale din clusterul CLOUDIFIN. Metodologia propusa este aplicata pentru determinarea gap-urile electronice in sisteme finite de tip *carbon nanoflakes* cu domenii de nitrura de bor hexagonala. De asemenea, sunt investigate fazele de echilibru si proprietatile mecanice in aliaje cu entropie ridicata. Eficientizarea buclei de self-consistentia s-a realizat prin maparea densitii de sarcina in starea fundamentala.

S-a demonstrat ca modelele de retele neurale artificiale (ANN) dezvoltate asigura o buna acuratete de predictie. Sunt evaluate mai multe arhitecturi ANN pentru realizarea regresiiilor (de exemplu pentru predictia gap-ului electronic). Metodele de selectie a trasaturilor (*variance threshold* si *mutual information threshold*) au un caracter general, astfel incat pot fi aplicate si in alte probleme similare.

S-au dezvoltat platforma de servicii asigurate de catre CCBD pentru suportul activității de modelare și simulare a nanostructurilor, precum si interfata de acces la aceasta.

Una dintre metodele ANN propuse a fost implementata pe platforma software a CCBD, astfel incat a devenit disponibila pentru utilizarea de catre comunitatea stiintifica.

Rezultatele noi ale activitatii de cercetare desfasurate in cadrul Subactivitatii 2.4 au fost publicate in trei articole stiintifice:

- N. Filipoiu, G.A. Nemnes, "Prediction of Equilibrium Phase, Stability and Stress-Strain Properties in Co-Cr-Fe-Ni-Al High Entropy Alloys Using Artificial Neural Networks", *Metals* 10, 1569 (2020)
- T.L. Mitran, G.A. Nemnes, "Ground state charge density prediction in C-BN nanoflakes using rotation equivariant feature-free artificial neural networks", *Carbon*, 174, 276 (2021)
- G.A. Nemnes, N. Filipoiu, V. Sipica, "Feature selection procedures for combined density functional theory—artificial neural network schemes", 2021 *Physica Scripta* 96 065807

8. Multumiri

Echipa proiectului ii multumeste dlui Florin Dinu de la Facultatea de Matematica si Informatica a Universitatii din Bucuresti, pentru contributi la realizarea codului de implementare a formalismului matricii R.

ANEXA

```
import numpy as np
from sklearn.model_selection import train_test_split
import tensorflow as tf
import matplotlib.pyplot as plt
from sklearn.metrics import r2_score
import math
import random as rn
import os

tfk = tf.keras
tfkl = tf.keras.layers

x = np.loadtxt("x.dat", dtype=float).reshape(-1,26) #(lines, columns); -1
=> the value is inferred
y = np.loadtxt("y.dat", dtype=float).reshape(-1,1)

x_train, x_test, y_train, y_test = train_test_split(x, y, test_size=1/5,
random_state=45, shuffle=True)
x_train, x_val, y_train, y_val = train_test_split(x_train, y_train,
test_size=1/4, random_state=45)

epochs = 2100

model = tfk.Sequential(
    [
        tfkl.InputLayer(input_shape=(26,)),
        tfkl.Dense(600, activation='relu', name="Hidden_Layer_1",
kernel_initializer='random_uniform', bias_initializer='random_uniform'),
        tfkl.Dense(550, activation="relu", name="Hidden_Layer_2",
kernel_initializer='random_uniform', bias_initializer='random_uniform'),
        tfkl.Dense(500, activation="relu", name="Hidden_Layer_3",
kernel_initializer='random_uniform', bias_initializer='random_uniform'),
        tfkl.Dense(450, activation="relu", name="Hidden_Layer_4",
kernel_initializer='random_uniform', bias_initializer='random_uniform'),
        tfkl.Dense(400, activation="relu", name="Hidden_Layer_5",
kernel_initializer='random_uniform', bias_initializer='random_uniform'),
        tfkl.Dense(350, activation="relu", name="Hidden_Layer_6",
kernel_initializer='random_uniform', bias_initializer='random_uniform'),
```

```
tfkl.Dense(300, activation="relu", name="Hidden_Layer_7",
kernel_initializer='random_uniform', bias_initializer='random_uniform'),
tfkl.Dense(250, activation="relu", name="Hidden_Layer_8",
kernel_initializer='random_uniform', bias_initializer='random_uniform'),
tfkl.Dense(200, activation="relu", name="Hidden_Layer_9",
kernel_initializer='random_uniform', bias_initializer='random_uniform'),
tfkl.Dense(150, activation="relu", name="Hidden_Layer_10",
kernel_initializer='random_uniform', bias_initializer='random_uniform'),
tfkl.Dense(100, activation="relu", name="Hidden_Layer_11",
kernel_initializer='random_uniform', bias_initializer='random_uniform'),
tfkl.Dense(75, activation="relu", name="Hidden_Layer_12",
kernel_initializer='random_uniform', bias_initializer='random_uniform'),
tfkl.Dense(50, activation="relu", name="Hidden_Layer_13",
kernel_initializer='random_uniform', bias_initializer='random_uniform'),
tfkl.Dense(25, activation="relu", name="Hidden_Layer_14",
kernel_initializer='random_uniform', bias_initializer='random_uniform'),
tfkl.Dense(1, name="Output_Layer"),
]
)
```

```
opt = tfk.optimizers.Adam(learning_rate=1e-5, beta_1=0.9, beta_2=0.999,
amsgrad=False)
```

```
model.compile(optimizer= opt, loss="mean_squared_error")
```

```
model.summary()
```

```
history = model.fit(x=x_train, y=y_train, batch_size=100, epochs=epochs,
validation_data=(x_val,y_val), validation_split=0.0,shuffle=False) #train
model
```

```
y_pred = model.predict(x_test)
```

```
r2 = r2_score(y_test, y_pred)
```

```
r2_coeff = math.floor(r2*1000)/1000
```

```
print("R2 coeff: ", r2_coeff)
```

```
plt.rcParams["figure.figsize"] = (7,7)
```

```
plt.scatter(y_test, y_pred)
```

```
plt.text(0.8, 0.7, r'$R^2 = $' + str(r2_coeff), fontsize = 14)
```

```
plt.ylabel(r"$E_{gap} $" + "$predicted$", fontsize = 15)
```

```
plt.xlabel(r"$E_{gap} $" + "$reference$", fontsize = 15)
```

```
plt.xticks(fontsize = 20)
plt.yticks(fontsize = 20)
lineStart = y_pred.min()
lineEnd = y_test.max()
y_lim = plt.ylim()
x_lim = plt.xlim()
plt.plot(x_lim, y_lim, '--', color = 'b', linewidth = 3)
plt.ylim(y_lim)
plt.xlim(x_lim)
plt.savefig("Energy_gap_prediction.png")
#plt.show()

save_dir = os.path.join(os.getcwd(), 'saved_models')
model_name = 'cbn_trained_model.h5'

# Save model and weights
if not os.path.isdir(save_dir):
    os.makedirs(save_dir)
model_path = os.path.join(save_dir, model_name)
model.save(model_path)
print('')
print('Saved trained model at %s ' % model_path)
```