

Raport Științific și Tehnic

Identificare:

Contract de finanțare nr. 124/01/04.2020

Contractor: Universitatea de Vest din Timișoara (UVT)

Cod: ERANET-CHISTERA3-DIPET

Etapa 3: 1 Dec 2021 - 30 Noi 2022

Nume etapă: Demonstrarea conceptului

Autori ai raportului și membrii ai proiectului:

Lect. Dr. Gabriel Iuhasz (Co-PI)

Dr. Silviu Panica

Lect. Dr. Marian Neagul

Asist.cercetare Alexandru Munteanu

Director proiect:

Prof. Dr. Dana Petcu (PI)

Pagini web:

<https://dipet.hpc.uvt.ro/>

<https://dipet.eecs.qub.ac.uk/>

Contents

Obiectivele etapei 3	3
Rezumatul Etapei 3	3
Gradul de atingere a rezultatelor estimate	4
Descriere științifică și tehnică	4
GuiFiNet.....	4
Graph data.....	7
Incorporarea grafului	10
Experimente.....	11
Diseminare	14
Buget	15
Concluzii	15

Obiectivele etapei 3

Proiectul DiPET investighează maparea dinamică și transparentă a aplicațiilor de procesare a fluxurilor de date în medii de calcul de tip Fog și Edge și folosește calculul cu trans-precizie cu scopul final de a îmbunătăți aspectele operaționale în termeni de utilizare a resurselor și consum de energie și de a îmbunătăți experiența utilizatorului. Această abordare va crea noi oportunități de îmbunătățire a mai multor aspecte ale sistemelor de tip Fog și Edge, precum creșterea eficienței energetice și a performanței acestor sisteme (se vizează o îmbunătățire a performanței de 2 ori prin trans-precizie și o îmbunătățire a energiei similare). Mai mult, DiPET își propune să simplifice radical programarea, configurarea și desfășurarea procesării fluxului, permițând programatorilor să declare restricții de plasare la resursele dinamice, controlul de planificare și trans-precizie și / sau aproximarea, toate într-o singură specificație concisă. Planificatorul va fi unul în premieră, întrucât în prezent nu există un planificator bun pentru procesarea fluxurilor de date în medii de tip Fog, cu atât mai puțin conștient de trans-precizie, și va oferi transparență de funcționare deopotrivă pentru dezvoltatorii de aplicații și utilizatori. Sistemul DiPET va fi partajat comunității sub licențe open source și distribuit ca software reutilizabil, prin pachete software instalabile. În plus, vor fi dezvoltate cazuri de utilizare, alese dintre domeniile de aplicație relevante, centrate pe utilizator: detectarea intruziunilor în rețea și analiza video distribuită.

Obiectivele principale a etapei 3 sunt:

- experimentarea unor metode/operatori peste care se pot optimiza folosind metode transprecise
- extinderea platformei EDE pe baza rezultatelor obținute din experimente și conectarea acesteia mediului de dezvoltare.

Contributii previzionate se refera la WP1 (operatori streaming cu transprecizie, cu adaptare de precizie numerică, cu alegere pe baze algoritmice, mecanisme pentru acordarea algoritmilor si analiza lor), WP2 (biblioteca de operatori de streaming cu transprecizie), WP3 (monitorizarea aplicației de procesare a fluxului, modelarea performanței fluxului de procesare cu transprecizie, maparea dinamică a resurselor conștiente de transprecizie, integrare), WP4 (instrumente pentru testare, evaluare experimentală și performanță).

Rezumatul Etapei 3

În timpul etapei 3, accentul principal a fost pe munca depusă în WP1, unde echipa UVT este investigatorul principal. Acest pachet de lucru are sarcina de a crea/implementa algoritmi de streaming cu precizie reglabilă. În timpul primei etape, ne-am concentrat pe definirea calității, precum și a valorilor de performanță nefuncționale pentru transprecizie în contextul dispozitivelor Edge/Fog. Aceste cerințe stau la baza activității desfășurate pe parcursul acestei etape. Și anume definirea mai multor componente din Platforma DiPET. Pentru a doua etapă ne-am concentrat pe implementarea/extinderea motorului de detectare a evenimentelor (EDE) creat inițial pentru proiectul de cercetare H2020 ASPIDE. Au fost adăugate și testate

mecanisme de optimizare transparente, cu toate rezultatele pertinente incluse în ultimul raport de referință.

Pentru etapa de referință actuală s-a lucrat cu privire la extinderea metodelor ML pentru a include datele Graph. Acest lucru a fost considerat necesar, deoarece cazul de utilizare GuifiNet pe care au fost testate metodele tranprecise EDE, se pretează în mod inerent acestui tip de structură de date. Aceasta înseamnă că relațiile dintre noduri pot conduce la o mai bună înțelegere și detectare a anomaliilor legate de performanță. În plus, un scenariu de învățare federată este, de asemenea, investigat în mod activ, deoarece protocolul de rutare bmx6 utilizat în GuifiNet este capabil să genereze date grafuri care pot fi utilizate ca pas de preprocesare pentru un scenariu hibrid de învățare centralizat/federat.

Plăcile Nvidia Jetson Nano adăugate infrastructurii UVT în ultima perioadă de raportare au fost instrumentate pentru a fi utilizate în experimente transparente. Suntem interesați să evaluăm cerințele de calcul și de putere ale modelelor antrenate anterior, atât pentru antrenament, cât și pentru inferență.

Gradul de atingere a rezultatelor estimate

Obiectivul științific și tehnic al etapei a fost îndeplinit în totalitate, cum vom prezenta în cele ce urmează.

Descriere științifică și tehnică

GuifiNet

În primul raport științific am descris pe scurt unul dintre cazurile de utilizare utilizate în DIPET care se ocupă de detectarea anomaliilor de performanță într-o rețea mesh. Această rețea mesh se numește GuifiNet¹ ca rezultat al unui proiect început în 2004. În prezent există peste 36.000 de noduri în această rețea. Fiecare nod este utilizat/deținut de un alt utilizator. Aceste noduri sunt conectate prin linii fixe și/sau conexiuni WiFi de mare putere cu un protocol special de rutare numit bmx6. Numărul mare de noduri generează o topologie de rețea destul de complexă.

Pentru experimentele noastre am selectat o subsecțiune de 67 de noduri din această rețea mesh. Figura 1 prezintă topologia actuală a rețelei nodului monitorizat. Este important de subliniat că această topologie este supusă unor modificări constante, deoarece nodurile se pot alătura și părăsi această rețea în momente imprevizibile. De exemplu, 3 noduri fac parte din alte proiecte de cercetare și sunt reconfigurate periodic. Acest lucru este și mai vizibil dacă trasăm nodurile și conexiunea lor sub formă de grafuri, așa cum se arată în Figura 2.

¹ <https://guifi.net/>

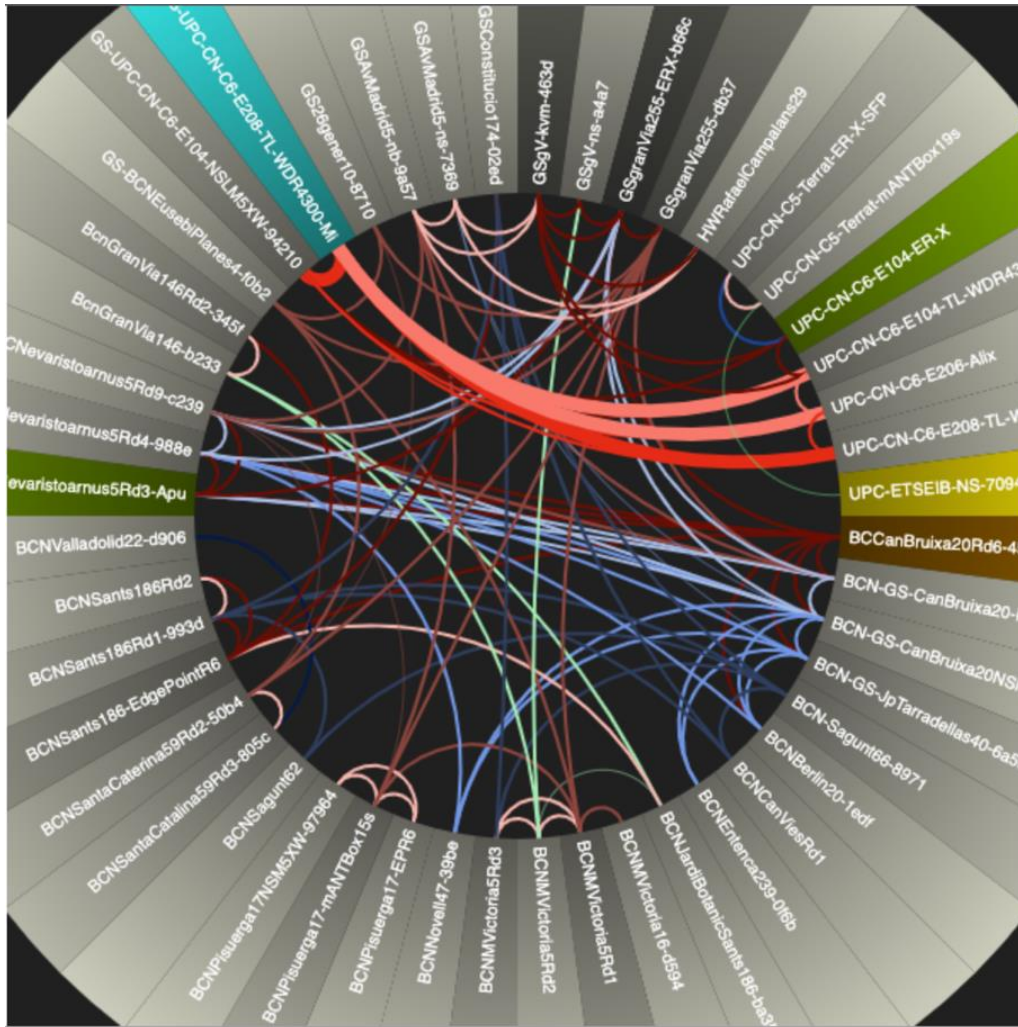


Figura 1 Prezentare generală a topologiei nodurilor de rețea Mesh ²

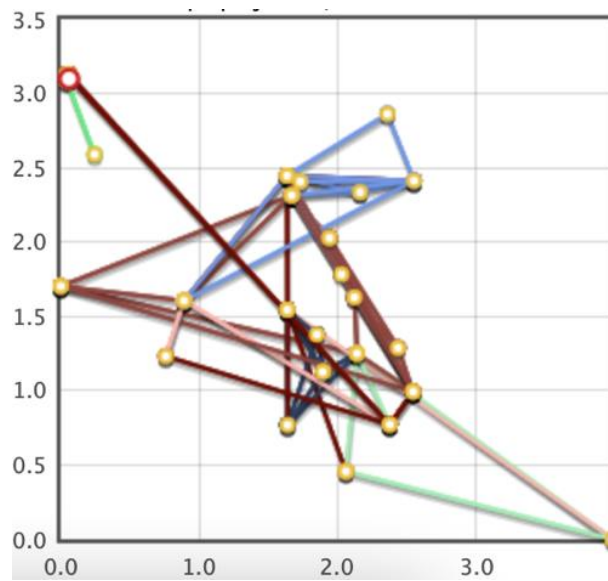


Figura 2 Graful georeferențiat al nodurilor

² <https://dsg.ac.upc.edu/qmpsu/index.php>

Graful prezentat în Figura 2 este georeferențiat, axa X și Y reprezintă kilometri. Aceste date, deși utile pentru vizualizare, produc puține date acționabile pentru detectarea anomaliilor de performanță. Pentru aceasta avem nevoie de date/metrici care detaliază legăturile dintre noduri. Din fericire, avem acces doar la aceste tipuri de metrici din protocolul de rutare bmx6.

În al doilea raport științific am reușit să creăm un sistem de monitorizare care colectează date de sistem de la toate nodurile monitorizate. Setul de date rezultat conținea 1585 de valori unice pe care le-am folosit pentru testarea metodelor de detectare a anomaliilor. În acest set de date avem un rând corespunzător unui minut (un minut fiind perioada de interogare a soluției de monitorizare). Valorile bmx6 utilizate pentru generarea grafurilor sunt colectate la aceeași rată.

Tabel 1 Legături de la nodul 44 la alte noduri

link	name	dst.uid	routes	txRate	rxRate	samples
1	GSgranVia255-db37	66	35	100	100	8505
2	BCCanBruixa20Rd6	24	13	100	100	8504
3	BCN-GS-CanBruixa20-NSM5-e48b	2	3.6	100	100	8504
4	BCNevaristoarnus5Rd4-988e	10	3.2	100	100	8492
5	BCNRadas83Rd1	49	2.5	100	100	8087
6	BCNevaristoarnus5Rd9-c239	47	1.8	100	100	8504
7	UPC-ETSEIB-NS-7094	4	1.1	100	100	8504
8	BCN-GS-CanBruixa20-RKM5-7bbd	43	1	100	100	8503
9	BCNevaristoarnus5-8ed8	62	1	100	100	8504
10	BCNevaristoarnus5Rd3-Apu	15	1	100	100	8504
11	UPC-EETAC-d9f3	33	1	100	100	8500
12	BCNRadas83Rd4	26	1	100	100	8087
13	BCNRadas83-Edge	40	1	100	100	8083

Tabelul 1 arată legăturile de la nodul 44 „BCNevaristoarnus5-8ed7” la alte noduri. Acestea reprezintă interfețe de rețea prezente în nodul 44 care pot fi cu fir sau fără fir. Aceste date pot fi deja folosite pentru a genera o structură grafă care conține legătura dintre noduri, totuși, nu ne spune nimic despre calitatea conexiunii. Din fericire, protocolul de rutare bmx6 calculează și aceste valori de calitate pe care le folosește pentru a direcționa traficul în întreaga rețea.

Figura 3 prezintă o privire de ansamblu asupra metricilor specifice bmx6 disponibile pentru nodul 44. Acum avem toate datele necesare pentru a genera grafuri care descriu mai bine topologia rețelei noastre de plasă și modul în care este reprezentată calitatea legăturilor dintre noduri.

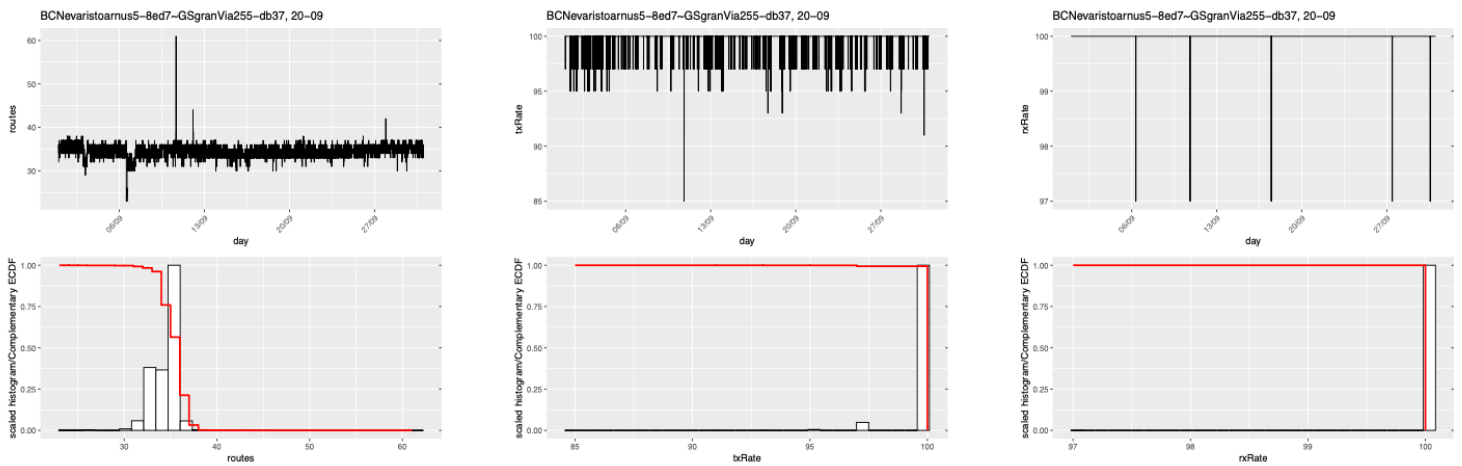


Figura 3 Valorile protocolului de rutare BMX6 pentru nodul 44

Graph data

Valorile detaliate în secțiunea anterioară au fost utilizate pentru a crea grafuri pentru fiecare perioadă de votare. De fapt, am creat un graf global care arată conectivitatea nodurilor și grafuri specifice nodurilor care detaliază calitatea conexiunii fiecărui nod pe baza metricilor bmx6 (un total de 2x67 grafuri pentru fiecare perioadă de sondare). În esență, am creat o serie temporală de grafuri. Figura 4 prezintă un exemplu de graf de conectivitate. Datele inițiale au fost transformate într-o matrice de distanțe care a fost folosită pentru a reprezenta fiecare grafi. Putem vedea mai multe lucruri interesante din acest graf.

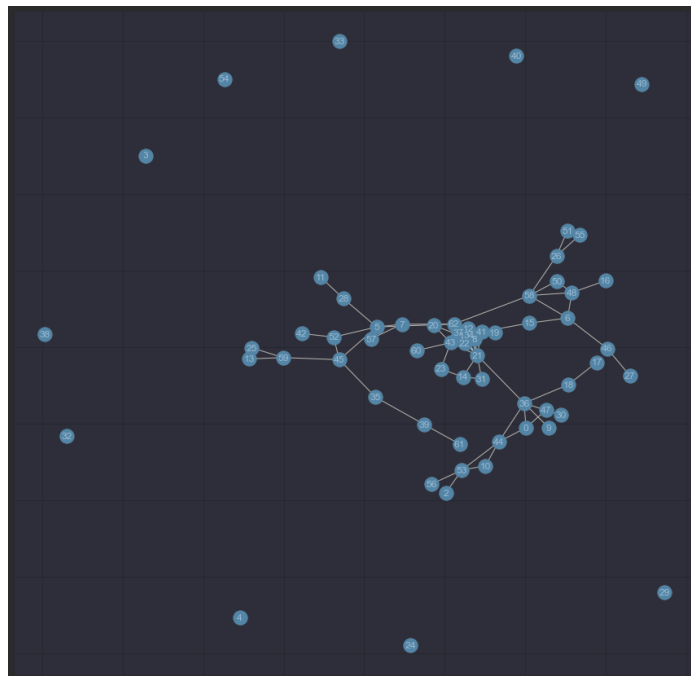


Figura 4 Un exemplu de graf de conectivitate globală

Mai întâi putem vedea topologia generală a rețelei mesh. Acesta poate fi folosit ulterior pentru a detecta modificările conexiunii la rețea. În al doilea rând, putem vedea dacă unele

noduri nu sunt conectate la rețea. În Figura 4 vedem de fapt un exemplu în care mai multe noduri (ID-ul lor este folosit ca etichete de noduri) nu sunt de fapt conectate la rețea.

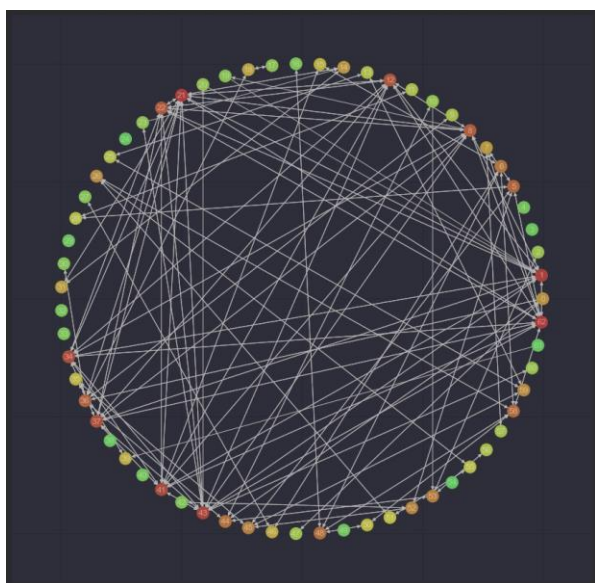
Una dintre cele mai interesante metode de analiză a grafurilor pe care le putem folosi pentru aceste date este ceva numit centralitatea grafului. Aceasta măsoară cât de important este un nod într-un graf. Măsura importanței poate fi calculată din mai multe puncte de vedere, în funcție de ceea ce este relevant în fiecare caz de utilizare. Următoarele paragrafuri vor detalia unele dintre măsurătorile de centralitate care au fost aplicate datelor:

- **Gradul de centralitate:** În cazul grafurilor nedirecționate, gradul este definit ca numărul de conexiuni directe pe care un nod le are cu alte noduri. În cazul grafurilor direcționate avem in-degree (definit ca numărul de conexiuni de la alte noduri la un nod curent) și out-degree (numărul de conexiuni de la un nod la alții). În acest caz, importanța unui nod este proporțională cu gradul sau Guifi-Net propduce grafuri direcțe.
- **Centralitate de apropiere:** se bazează pe distanța geodezică, unde distanța dintre două noduri este definită ca numărul de conexiuni dintre cele două noduri pe calea cea mai scurtă. Practic, centralitatea de apropiere definește importanța unui nod pe cât de aproape este de toate celelalte noduri din grafuri.
- **Centralitate intermediară:** Măsoară importanța unui nod pe baza de câte ori apare pe cea mai scurtă cale între toate perechile de noduri dintr-un graf. Acest lucru ne permite să găsim așa-numitele noduri punte într-un graf, noduri care apar cel mai des în calculele cu cea mai scurtă cale, indicând importanța comunicării între noduri.
- **Centralitatea de vector propriu:** Măsoară importanța unui nod într-un graf în funcție de importanța vecinilor săi. Astfel, dacă un nod este conectat la noduri importante, acesta va avea un scor de centralitate mai mare.

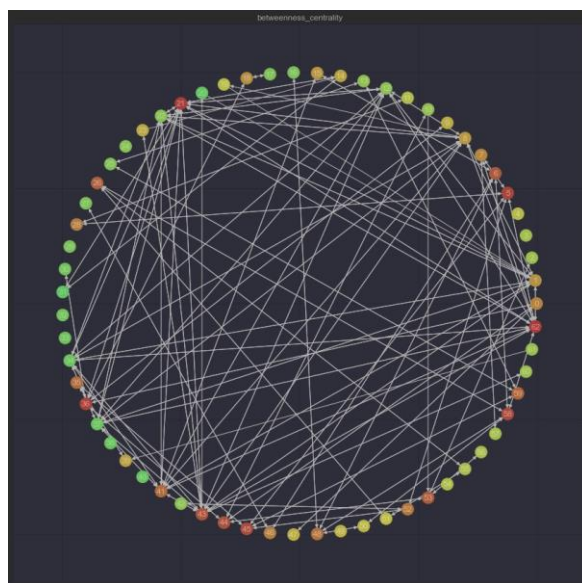
Măsurile de centralitate a listei menționate mai sus nu trebuie considerate ca o listă exhaustivă. De fapt, am calculat și alte măsuri de centralitate, cum ar fi centralitățile Page Rank, Hub și Auth.

Figura 5 arată o comparație între două tipuri de centralități. Graficele sunt reprezentate astfel pentru a fi mai ușor de afișat conexiunile dintre noduri. Este echivalent din punct de vedere topologic cu cel prezentat în Figura 4. Importanța nodului este afișată folosind o scară de culori; nodurile marcate cu roșu sunt mai importante, în timp ce cele marcate cu verde sunt mai puțin importante. Pentru centralitatea de afară prezentată în Figura 5 (a), nodul cu ID 1 este cel mai important, în timp ce în cazul Figurii 5 (b) nodul 62 este cel mai important. Ar trebui să menționăm, de asemenea, că în cazul acestui graf particular prezentat în Figura 5 (a) am ales să reprezentăm măsura centralității Out deoarece măsura centralității In este identică. Acest lucru poate fi explicat că în unele cazuri măsurate există conexiuni unidirecționale, totuși acestea trebuie considerate mai degrabă o excepție decât o regulă.

Când vine vorba de grafuri specifice nodurilor, care detaliază calitatea conexiunii, grafurile arată puțin diferit. Ca și în aceste grafuri, un nod poate fi conectat de mai multe ori cu alte noduri (sunt posibile conexiuni prin cablu sau WiFi la același nod). În mod similar, avem, de asemenea, metrice de calitate legate de conexiunile loop-back, reprezentate ca auto-conexiuni în cadrul unui graf.



(a) Out Centrality



(b) Betweenness Centrality

Figura 5 Comparația dintre centralitatea Out și Betweenness

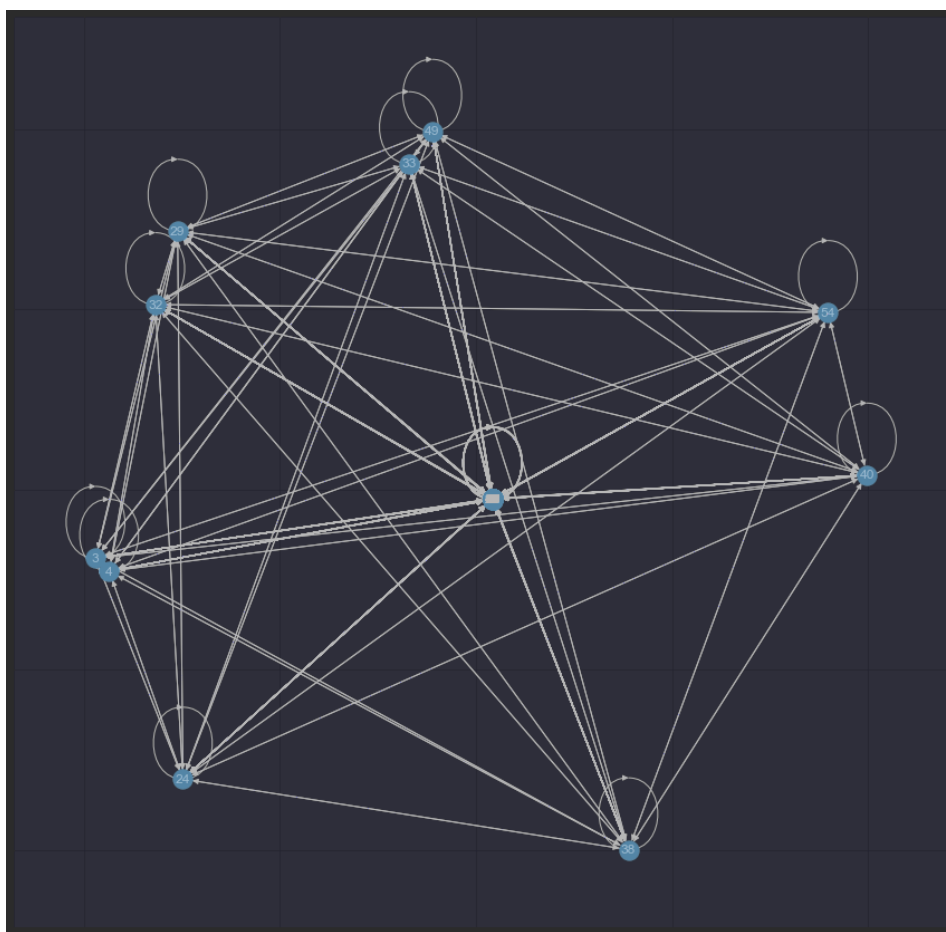


Figura 6 Graf care reprezintă calitatea conectivității nodului

Figura 6 prezintă un exemplu de aceste grafuri. Putem vedea cu ușurință conexiunile multiple dintre noduri, precum și conexiunile loop-back. Ultimul tip de conexiuni sunt irelevante și pot fi ignorate în timpul analizei. Acest lucru nu este universal valabil pentru toate datele din graf, dar a vedea că aceasta arată o vedere a topologiei rețelei, autoconexiunile nu sunt o caracteristică utilă. Măsurile de centralitate pot fi aplicate și acestor tipuri de grafuri.

Informațiile obținute prin analiza datelor grafuri în acest mod pot fi utilizate în diferite moduri. De exemplu, ne uităm la ponderarea valorilor sistemului pe baza măsurilor de centralitate. Operațiunile de inginerie ale acestei caracteristici se bazează pe presupunerea că nodurile grafuri cu anumite măsuri de centralitate vor avea cel mai probabil o influență mai mare în detectarea evenimentelor anormale legate de performanță.

Incorporarea grafului

Odată ce avem datele grafului, mai avem problema cum putem folosi efectiv aceste date în metodele de detectare. Una dintre cele mai populare moduri de a lucra cu grafuri în metodele de detectare bazate pe ML este utilizarea diferitelor metode de încorporare. Aceste metode de încorporare transformă nodurile, muchiile și caracteristicile lor în spațiu vectorial, păstrând în același timp la maximum proprietăți precum structura grafului și informațiile. Algoritmii ML sunt de obicei reglați pentru date continue, de aceea încorporarea este întotdeauna în spațiul vectorial continuu. Prin această definiție putem afirma că această etapă de pre-procesare este legată de reducerea dimensionalității.

Node2Vec³ este una dintre metodele folosite pentru a genera aceste înglobări. Se bazează pe Word2Vec. Folosind acest algoritm, transformăm fiecare nod dintr-un graf într-o încorporare care poate fi folosită de rețelele neuronale pentru a învăța reprezentarea fiecărui nod.

Figura 7 arată rezultatele după graful prezentat în Figura 4, care a fost încorporat folosind Node2Vec. În mod similar, putem folosi Hadamard Embedding (vezi Figura 8 (a)) pentru a încorpora datele de margine. Sau putem folosi Graph2Vec (vezi Figura 8 (b)) care învață să încorporeze subgrafuri ale grafulor. Este ușor de observat că putem folosi mai ușor aceste noi versiuni încorporate a acestor grafuri în diferite metode de detectare a anomaliilor bazate pe ML. Modelele bazate pe Deep Learning sunt potrivite pentru acest lucru.

Ca și în cazul altor tehnici de reducere a dimensionalității, numărul de componente pentru ieșire poate fi definit. În unele situații sunt necesare numere mai mari ale componentelor rezultate pentru a descrie în mod adecvat structurile grafulor inițiale.

³ <https://snap.stanford.edu/node2vec/>

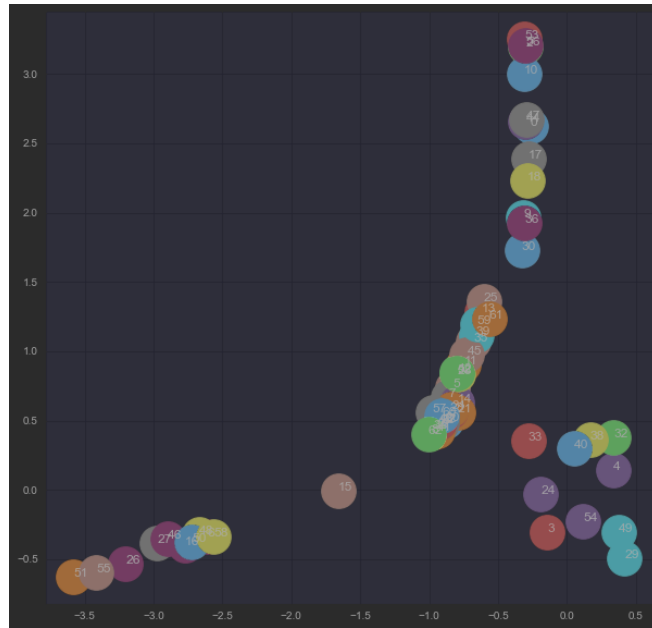
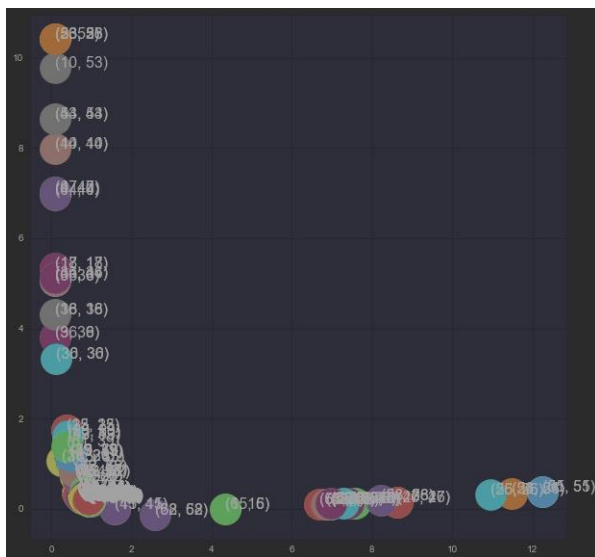
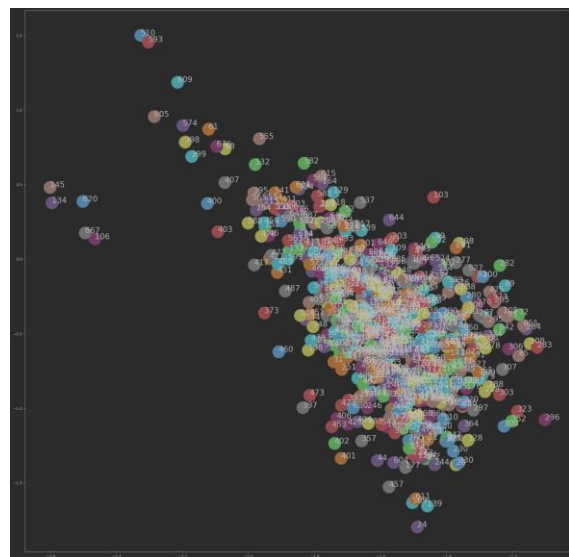


Figura 7 Exemplu de încorporare Node2Vec



(a) Hadamar Embedding



(b) Graph2Vec

Figura 8 Încorporarea exemplilor

Experimente

În timpul acestei etape, majoritatea experimentelor au fost o continuare a reperelor anterioare cu adăugarea târzie a metodelor de detectare a anomaliilor datelor grafuri. În ultima etapă ne-am concentrat pe 3 metode de detectare: Factor aberan local bazat pe cluster (CBLOF), IsolationForest (IF) și Autoencoder variațional (VAE). Pe baza acestor experimente, ne-am îmbunătățit înțelegerea datelor subiacente și am făcut mai multe modificări pașilor de preprocesare. Și anume, am modificat modul în care scalam datele folosind scalatorul MinMax.

Figura 9 prezintă o reprezentare 3D a datelor după PCA cu 3 componente. Evenimentele anormale detectate folosind CBLOF au fost marcate cu roșu. În comparație cu ultimul raport, setul de date rezultat este mai consolidat. Acest lucru este mai vizibil într-o reprezentare 3D decât într-una 2D (vezi Figura 10).

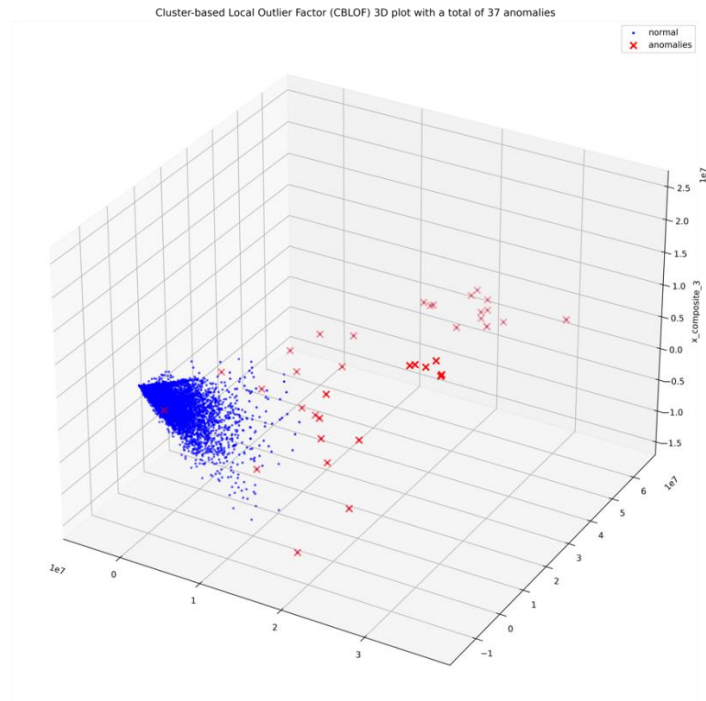


Figura 9 3D Reprezentarea componentelor pentru anomalii CBLOF

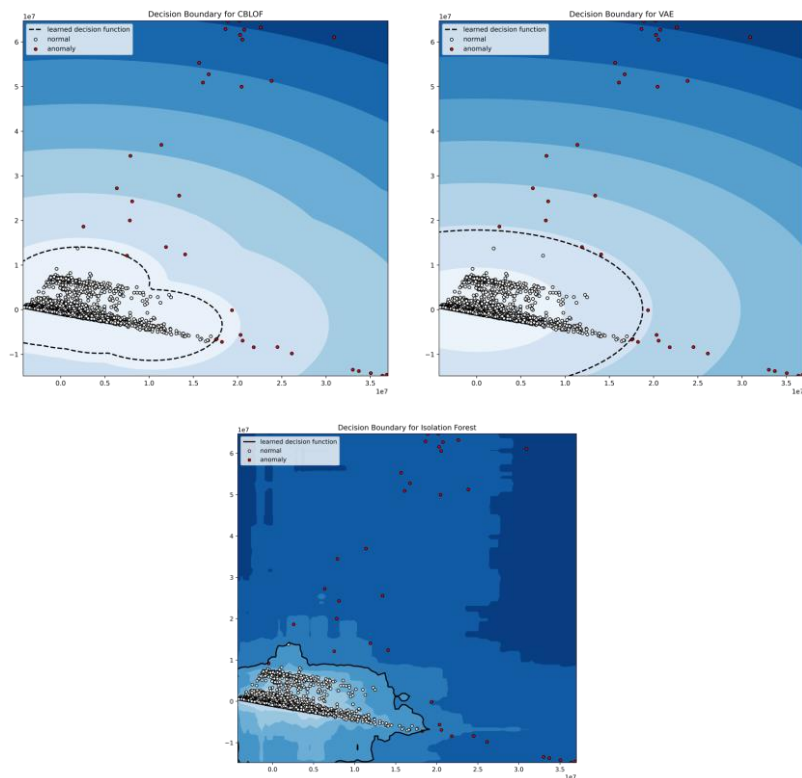


Figura 10 Limitele de decizie privind reprezentarea 2D

Performanța metodelor de detectare este de asemenea crescută, deși nu semnificativ. Figura 10 putem vedea limitele de decizie calculate pentru toate cele 3 metode. După cum era de așteptat, IF dă semne de supraadaptare, limita de decizie încă imită îndeaproape forma punctelor de date dens grupate. Trebuie să remarcăm că această versiune a limitei de decizie este mult mai bună decât cea prezentată în al doilea raport de referință.

Performanța predictivă reală nu este afectată semnificativ, mai degrabă câștigul principal este în timpii de antrenament și de inferență. Am măsurat o scădere de 10-15% a antrenamentului și 2-3% scădere a timpilor de inferență. VAE a fost cel mai puțin afectat, mai ales din cauza supraîncărcării asociate cu instanțierea topologiei DNN.

Pe parcursul ultimei etape am identificat câteva metode de selecție a caracteristicilor care pot fi utilizate pentru o adaptare precisă, în sensul că putem limita cantitatea de date necesare pentru crearea unui model viabil de detectare a anomaliilor ML. Am folosit valorile PCA, Kurtosis, Diferența Absolută medie și Shapely pentru aceste experimente.

Am continuat experimentele noastre în această perioadă de raportare, concentrându-ne pe descrierea completă și cuantificarea efectelor pe care aceste metode le au asupra performanței predictive. Am aplicat mai întâi metodele de reducere a caracteristicilor pe setul de date disponibil. Cele mai multe dintre metodele menționate mai sus pot fi aplicate datelor numai în timp ce valorile Shapley trebuie calculate în raport cu un model predictiv preexistent. Acest fapt este evidențiat și în ultimul raport științific.

Fiecare metodă de reducere a caracteristicilor va da o clasare sau o importanță a caracteristicilor. Aceste date sunt apoi folosite pentru a genera mai multe experimente bazate pe eliminarea caracteristicilor recursive prin adăugarea de caracteristici la setul de date de antrenament după fiecare iterație. De exemplu, antrenamentul inițial se face numai pe caracteristica cea mai bine clasată, la a doua iterație de antrenament adăugăm a doua caracteristică cel mai bine clasată și așa mai departe. Facem acest lucru pentru clasarea tuturor metodelor de reducere a caracteristicilor până când toate caracteristicile sunt adăugate înapoi la setul de date de antrenament.

Figura 11 prezintă rezultatele pentru cele 4 tehnici de reducere a caracteristicilor (PCA, Kurtosis, MAD și Shapely). Fiecare subfigură arată numărul de caracteristici utilizate pentru antrenament pe axa X (de la 0 la 1585) și numărul de evenimente anormale detectate în perioada țintă pe axa Y. Putem vedea o diferență destul de mare de performanță pentru toate cele 3 metode de detectare. VAE este cel mai stabil în cazul PCA, MAD și Shapely, vedem o reducere marcată a eșantionului de date necesar pentru antrenament. Performanțele CBLOF și IF sunt mai neregulate, metoda ulterioară arătând o oarecare variație mare în performanță. Deși putem observa că fiecare metodă ajunge într-un stadiu critic în care performanța lor este mult mai bună la aproximativ 200 de caracteristici.

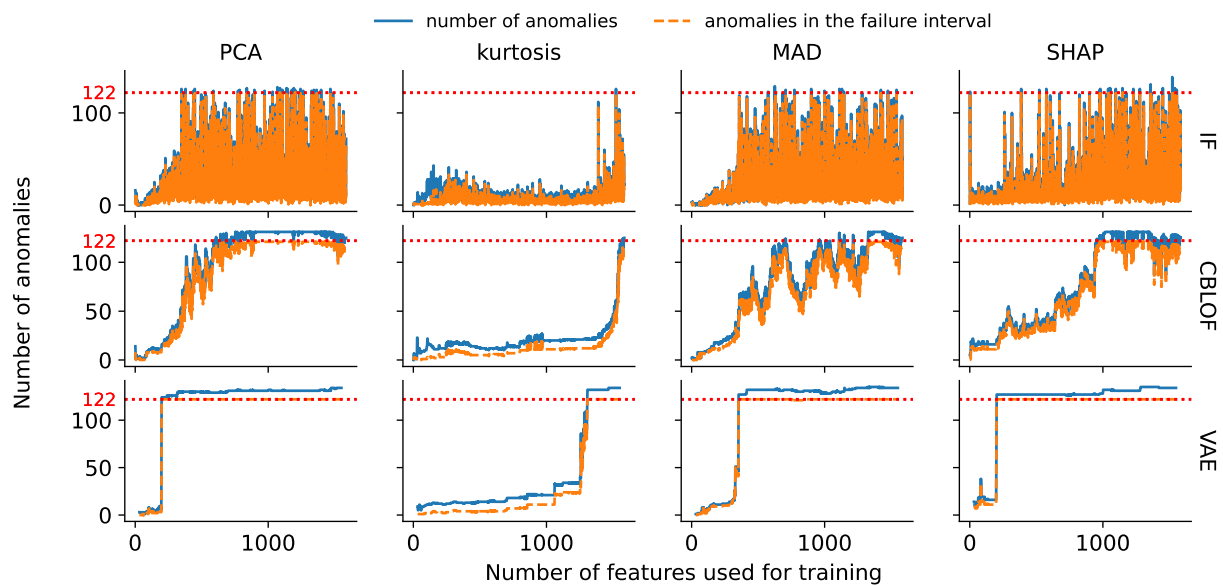


Figura 11 Eliminarea caracteristicilor recursive

Diseminare

1. Articole

In etapa 3 au fost submitte 2 articole:

- Ilorenc Cerda-Adalbert, *Gabriel Iuhasz*, Gabriele Gemmi, "Anomaly Detection for Fault Detection in Wireless Community Networks Using Machine Learning, Computer Communications⁴, Elsevier (Q2 in WoS), în evaluare
- *Gabriel Iuhasz*, *Silviu Panica*, "Cycle Detection and Clustering for Cyber Physical Systems" AINA-2023, International Conference on Advanced Information Networking and Applications (rank B in CORE⁵), submission ID #1570870702

2. Livrabile

Proiectul DIPET este un proiect european colaborativ. Livrabilele la care echipa UVT a contribuit (și disponibile în platforma EVOC, ca anexe la raport) sunt:

- Livrabilul D1.2: First version of transprecise streaming operators (rezultat al WP1), 10.04.2022, autori: *Gabriel Iuhasz* si Hans Vandierendock
- Livrabilul D3.2: Updated monitoring framework and performance model (rezultat al WP3), 31.03.2022, autori: Guillaume Pierre, Alessio Pagliari, *Silviu Panica*, *Gabriel Iuhasz*

⁴ <https://www.sciencedirect.com/journal/computer-communications>

⁵ <http://portal.core.edu.au/conf-ranks/?search=AINA&by=all&source=CORE2021&sort=atitle&page=1>

Buget

Fondurile etapei 3 au fost cheltuite exclusiv pe salarii, regie si audit. NU au fost efectuate deplasari din motive legate de pandemie.

Concluzii

Obiectivele etapei au fost îndeplinite în totalitate. Experimentele descrise mai sus dovedesc validarea conceptelor introduse în proiect.

PI / UVT

Prof. Dr. Dana Petcu 