UNIVERZITET U BEOGRADU

MAŠINSKI FAKULTET

Dušan M. Nedeljković

DETEKCIJA KIBERNETSKIH NAPADA NA SISTEME ZA UPRAVLJANJE PROIZVODNIM RESURSIMA

doktorska disertacija

Beograd, 2023

UNIVERSITY OF BELGRADE

FACULTY OF MECHANICAL ENGINEERING

Dušan M. Nedeljković

DETECTION OF CYBER-ATTACKS ON SYSTEMS FOR MANUFACTURING EQUIPMENT CONTROL

Doctoral Dissertation

Belgrade, 2023

Informacije o mentoru i članovima komisije za odbranu doktorske disertacije

Mentor: dr Živana Jakovljević, redovni profesor
Univerzitet u Beogradu – Mašinski fakultet

Članovi komisije: dr Zoran Miljković, redovni profesor
Univerzitet u Beogradu – Mašinski fakultet

dr Saša Živanović, redovni profesor
Univerzitet u Beogradu – Mašinski fakultet

dr Nikola Slavković, vanredni profesor
Univerzitet u Beogradu – Mašinski fakultet

dr Milica Petrović, vanredni profesor
Univerzitet u Beogradu – Mašinski fakultet

dr Mladen Nikolić, vanredni profesor
Univerzitet u Beogradu – Matematički fakultet

Datum odbrane:

Izjave zahvalnosti

Želeo bih da izrazim zahvalnost svim članovima komisije koji su uzeli učešće u pregledu i oceni ove doktorske disertacije i čiji su komentari i sugestije pomogli da disertacija bude kvalitetnija i sadržajnija.

Neizmernu zahvalnost dugujem svojoj mentorki prof. dr Živani Jakovljević koja me je još tokom izrade Master rada motivisala i zainteresovala za Doktorske studije i koja mi je tokom čitavog njihovog trajanja svojim strpljenjem, nesebičnim komentarima, savetima i sugestijama kako stručnim, tako i životnim, pomogla da dođem do finalne forme doktorske disertacije.

Hvala svim prijateljima i kolegama koji su mi još od prvih studentskih dana pomogli da savladam izazove koji su se nametali. Sa vama je sve lakše i lepše.

Na kraju, beskrajno sam zahvalan mojoj majci Jadranki, ocu Momiru, bratu Stefanu i mojoj verenici Dušici za bezuslovnu ljubav i podršku koju su mi pružili. Hvala vam što ste verovali u mene čak i kada sam sumnjao u sebe.

DETEKCIJA KIBERNETSKIH NAPADA NA SISTEME ZA UPRAVLJANJE PROIZVODNIM RESURSIMA

Sažetak

Integracija kibernetsko-fizičkih sistema (engl. *Cyber-Physical Systems* – CPS) u industrijski internet stvari predstavlja osnovu za prelazak sa centralizovanih na distribuirane sisteme upravljanja u okviru kojih se upravljački zadaci raspodeljuju na različite uređaje, a celokupan zadatak upravljanja realizuje kroz njihov sinhronizovani rad i stalnu razmenu informacija. Sveprisutna komunikacija između elemenata industrijskih sistema upravljanja (engl. *Industrial Control Systems* – ICS) kao i njihovo povezivanje na globalnu mrežu otvaraju prostor za različite kibernetske napade koji pored ekonomskih posledica i katastrofalnih oštećenja opreme mogu imati i negativne uticaje na životnu sredinu i bezbednost na radu.

U fokusu ove doktorske disertacije je problem detekcije kibernetskih napada na komunikacione veze između CPS u okviru sistema za kontinualno upravljanje proizvodnim resursima. U radu je predložena metodologija za kreiranje sistema za detekciju napada koja je zasnovana na principima samonadgledanog učenja i kreiranju autoregresionih modela podataka koji se razmenjuju između uređaja u normalnim uslovima rada (bez napada) korišćenjem različitih tehnika mašinskog učenja. Metodologija vrši automatski izbor svih parametara algoritma za detekciju i uzima u obzir arhitekturu sistema upravljanja i mogućnost implementacije algoritma za detekciju napada na nekom od uređaja u okviru sistema. Može se primeniti kako za sisteme iz kojih je moguće prikupiti dovoljnu količinu podataka, tako i za sisteme za koje je dostupnost podataka ograničena.

Verifikacija razvijenih sistema za detekciju napada sprovedena je na javno dostupnim skupovima podataka i skupovima podataka dobijenim sa eksperimentalnih instalacija koje su razvijene u okviru disertacije. Izvršena je implementacija i eksperimentalna verifikacija sistema za detekciju napada generisanih korišćenjem razvijenih metoda na kreiranoj instalaciji čime su i u realnim uslovima potvrđene postavljene polazne hipoteze.

Ključne reči: sajber bezbednost, industrijski sistemi upravljanja, kibernetsko-fizički sistemi, industrijski internet stvari, kibernetski napadi, sistemi za detekciju napada, mašinsko učenje.
Naučna oblast: Mašinsko inženjerstvo
Uža naučna oblast: Proizvodno mašinstvo
UDK:

DETECTION OF CYBER-ATTACKS ON SYSTEMS FOR MANUFACTURING EQUIPMENT CONTROL

Abstract

The integration of Cyber-Physical Systems (CPS) into the Industrial Internet of Things represents a basis for the transition from centralized to distributed control systems where the control tasks are distributed to different devices, and the overall control task is achieved through their synchronized work and constant information exchange. Ubiquitous communication between elements of Industrial Control Systems (ICS) and their connection to the global network open up space for various cyber-attacks that in addition to economic consequences and catastrophic damage to equipment, can negatively impact the environment and work safety.

This doctoral dissertation focuses on the problem of cyber-attacks detection on communication links between CPS in the continuously controlled manufacturing equipment. The dissertation proposes a methodology for the development of attack detection systems based on self-supervised learning principles and autoregressive models of data that are exchanged between devices in normal operating conditions (without attacks); the models are generated using different machine learning techniques. The methodology automatically selects all parameters of the detection algorithm and takes into account the architecture of the control system and the possibility of implementation of the attack detection algorithm on a certain device within the system. It can be applied not only for the systems from which it is possible to collect a sufficient amount of data but also for the systems where the availability of data is limited.

The developed attack detection systems were verified on publicly available datasets and datasets obtained from experimental installations developed within the dissertation. The implementation and experimental verification of the attack detection system generated using the developed methods was carried out on the created installation, which also confirmed the initial hypotheses in real-world conditions.

Key words: cyber security, industrial control systems, cyber-physical systems, industrial internet of things, cyber-attacks, intrusion detection systems, machine learning.
Scientific field: Mechanical engineering
Scientific subfield: Production engineering
UDC:

Sadržaj

Spisak slika

Spisak tabela

Spisak oznaka

α_i, α_i^*	Lagranževi množioci
ξ_i, ξ_i^*	promenljive
ε	širina margine razdvajanja (SVR parametar)
γ	parametar radijalnog kernela
$\mu_{obučavanje}$, μ_{izbor}	srednja apsolutna greška na skupovima za obučavanje i izbor modela
$\sigma_{obučavanje}, \sigma_{izbor}$	standardna devijacija na skupovima za obučavanje i izbor modela
σ^2	varijansa
$\mathbf{h}$	vektor skrivenog stanja
$\mathbf{c}$	vektor stanja ćelije
$\mathbf{f}$	vektor kapije zaboravljanja
$\mathbf{i}$	vektor ulazne kapije
$\mathbf{g}$	vektor kapije stanja kandidata
$\mathbf{o}$	vektor izlazne kapije
$\mathbf{u}$	vektor kapije za ažuriranje
$\mathbf{r}$	vektor kapije za resetovanje
$\mathbf{W}_{ij}$	matrica težinskih koeficijenata
$\mathbf{b}_i$	*bias* vektor
$\mathbf{x}_i$	ulazna sekvenca 1D signala
$\mathbf{X}_i$	ulazna sekvenca 2D signala
y_i	izlaz (odziv) 1D signala
Y_i	izlaz (odziv) 2D signala
$\hat{y}_i$	estimirana vrednost izlaza 1D signala
$\hat{Y}_i$	estimirana vrednost izlaza 2D signala
b_i	*bias*
v	dužina bafera
T	prag detekcije
z	broj dozvoljenih uzastopnih prekoračenja praga detekcije
n	ukupan broj odbiraka u signalu
C	parametar regularizacije
K	kernel
kr	tip kernel funkcije
c_i	broj blokova/slojeva
f_i	broj filtera u 1D-CNN/2D-CNN/2D-ConvLSTM sloju
u_i	broj jedinica u RNN sloju
fs_i	veličina filtera
p_i	stopa sažimanja u sloju sažimanja
d_i	broj neurona u potpuno povezanom sloju
dr_i	stopa izostavljanja u izostavljajućem sloju
n_x, n_y	dimenzije 2D signala
p_x, p_y	dimenzija prozora detekcije
h	impulsni odziv filtera
m	broj odbiraka u impulsnom odzivu filtera
r	koeficijent polinoma

nv	broj nosećih vektora
bp	broj pozicija prozora detekcije
r_{det}	promenljiva detekcije napada
ln	lažno negativni rezultati
lp	lažno pozitivni rezultati
tn	tačno negativni rezultati
tp	tačno pozitivni rezultati

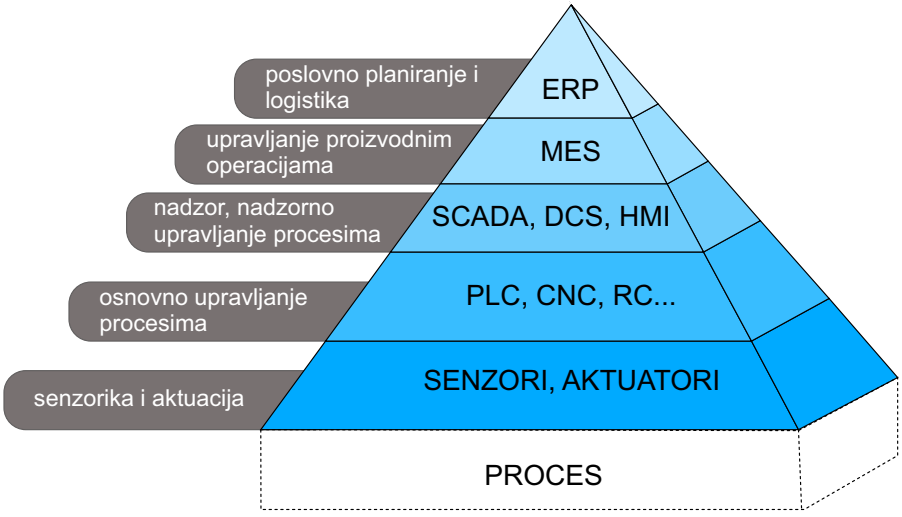
Spisak skraćenica

1D	jednodimenzionalni
2D	dvodimenzionalni
3D	trodimenzionalni
AIC	raspoloživost/integritet/poverljivost (engl. *Availability/Integrity/Confidentiality*)
AUC	oblast ispod krive (engl. *Area Under the Curve*)
CIA	poverljivost/integritet/raspoloživost (engl. *Confidentiality/Integrity/Availability*)
CNN	konvolucione neuronske mreže (engl. *Convolutional Neural Networks*)
CPI	obaveštajni kibernetsko-fizički napadi (engl. *Cyber-Physical Intelligence*)
CPS	kibernetsko-fizički sistemi (engl. *Cyber-Physical Systems*)
DAQ	sistem za akviziciju podataka (engl. *Data Acquisition*)
DCS	distribuirani sistemi upravljanja (engl. *Distributed Control Systems*)
DDoS	distribuirani napad uskraćivanjem pristupa servisu (engl. *Distributed Denial of Service*)
DES	sistemi sa diskretnim događajima (engl. *Discrete Event Systems*)
DMZ	demilitarizovana zona
DNN	duboke neuronske mreže (engl. *Deep Neural Networks*)
DoS	napad uskraćivanjem pristupa servisu (engl. *Denial of Service*)
DWT	diskretna vejvlet transformacija (engl. *Discrete Wavelet Transform*)
EpSP	elektropneumatski sistem za pozicioniranje
ERP	sistem za planiranje resursa (engl. *Enterprise Resource Planning*)
FDI	napad ubrizgavanjem lažnih podataka (engl. *False Data Injection*)
FIR	filter sa konačnim impulsnim odzivom (engl. *Finite Impulse Response*)
FPR	stopa lažno pozitivnih (engl. *False Positive Rate*)
fps	frejmovi u sekundi (engl. *frames per second*)
GAN	generativne suparničke mreže (engl. *Generative Adversarial Networks*)
GRU	rekurentne neuronske mreže sa zatvorenom rekurentnom jedinicom (engl. *Gated Recurrent Unit*)
HIL	hardver u petlji (engl. *Hardware In the Loop*)
ICS	industrijski sistemi upravljanja (engl. *Industrial Control Systems*)
IDS	sistem za detekciju napada (engl. *Intrusion Detection Systems*)
IIoT	industrijski internet stvari (engl. *Industrial Internet of Things*)
IT	informacione tehnologije
LSTM	neuronske mreže sa dugom kratkoročnom memorijom (engl. *Long Short-Term Memory*)
MA	filter pokretnih srednjih vrednosti (engl. *Moving Average*)
MES	sistem za izvršavanje proizvodnje (engl. *Manufacturing Execution System*)
MITM	napad posrednikom (engl. *Man-in-the-Middle*)
ML	mašinsko učenje (engl. *Machine Learning*)
MLP	višeslojni perceptron (engl. *multilayer perceptron*)
MSE	srednja kvadratna greška (engl. *Mean Squared Error*)
OT	operacione tehnologije
PCA	analiza glavnih komponenti (engl. *Principal Component Analysis*)
PLC	programabilni logički kontroler (engl. *Programmable Logic Controller*)
PSNR	maksimalni odnos signal/šum (engl. *Peak Signal to Noise Ratio*)
RBF	kernel koji koristi radijalnu funkciju (engl. *Radial Basis Function*)
ReLU	aktivaciona funkcija ispravljajuće linearne jedinice (engl. *Rectified Linear Unit*)

RNN	rekurentne neuronske mreže (engl. *Recurrent Neural Networks*)
RTOS	operativni sistem namenjen za rad u realnom vremenu (engl. *Real-Time Operating System*)
SAE	složeni autoenkoderi (engl. *Stacked Autoencoders*)
SCADA	sistem za nadzorno upravljanje i prikupljanje podataka (engl. *Supervisory Control and Data Acquisition*)
SCT	teorija nadzornog upravljanja (engl. *Supervisory Control Theory*)
SSA	analiza singularnog spektra (engl. *Singular Spectrum Analysis*)
STFT	kratkotrajna Furijeova transformacija (engl. *Short-Time Fourier Transform*)
SVM	mašine sa nosećim vektorima (engl. *Support Vector Machines*)
SVR	regresija nosećim vektorima (engl. *Support Vector Regression*)
TNR	stopa tačno negativnih (engl. *true negative rate*)
UAE	nepotpuni autoenkoderi (engl. *Undercomplete Autoencoders*)
UPS	sistem za besprekidno napajanje (engl. *Uninterruptable Power Supply*)

1. Uvod

Industrijski sistemi upravljanja (engl. *Industrial Control Systems* – ICS) se sastoje od uređaja (električnih, mehaničkih, hidrauličkih, pneumatskih itd.), mreža za komunikaciju i upravljačkih algoritama koji se koriste za upravljanje radom različitih industrijskih procesa. Neke od najčešćih sfera u kojima se ICS koriste za delimičnu ili potpunu automatizciju upravljačkih zadataka su mašinska, vazduhoplovna, automobilska, farmaceutska, procesna industrija, energetika itd. ICS predstavljaju integralan deo sistema upravljanja preduzećem čija je osnovna hijerarhijska šema standardizovana kroz IEC 62264 [47] i opisana piramidom automatizacije (slika 1). U zavisnosti od nivoa integracije, ICS može obuhvatiti sistem za izvršavanje proizvodnje (engl. *Manufacturing Execution System* – MES), sistem za nadzorno upravljanje i prikupljanje podataka (engl. *Supervisory Control and Data Acquisition* – SCADA), distribuirane sisteme upravljanja (engl. *Distributed Control Systems* – DCS), programabilne logičke kontrolere (engl. *Programmable Logic Controller* – PLC), senzore, aktuatore i sve druge uređaje koji funkcionišu na prva četiri nivoa piramide automatizacije koji se tradicionalno posmatraju kao nivoi operacionih tehnologija (OT). Na petom nivou ove hijerarhije nalazi se sistem za planiranje resursa (engl. *Enterprise Resource Planning* – ERP) koji tradicionalno predstavlja nivo zadužen za implementaciju različitih informacionih tehnologija (IT) u oblasti planiranja proizvodnje, logistike, interakcije sa kooperantima itd.

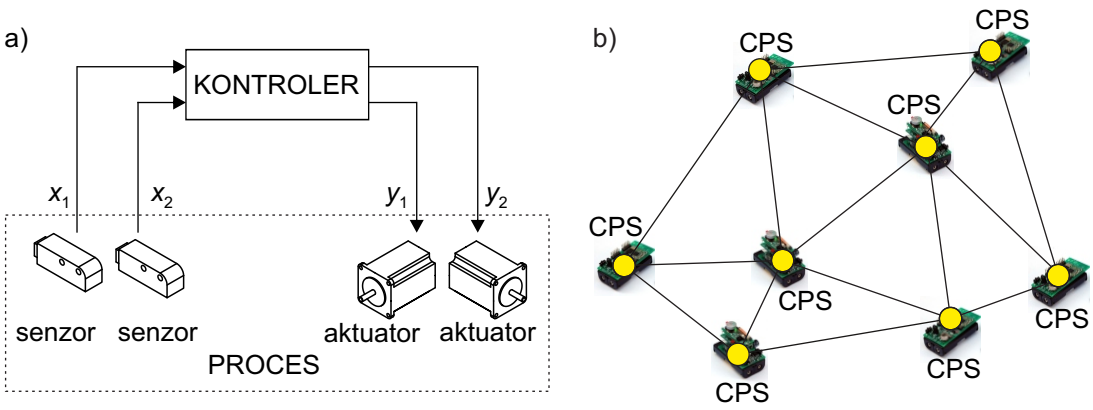


Slika 1: Piramida automatizacije

Današnji zahtevi tržišta u vidu masovne kastomizacije i brzog prilagođavanja proizvodnje različitim vrstama proizvoda uslovljavaju upotrebu novih pristupa i tehnologija u upravljanju proizvodnim sistemima. Odgovor na postavljene zahteve ostvaruje se uvođenjem principa definisanih paradigmom Industrija 4.0 [57] koja podrazumeva integraciju kibernetsko-fizičkih sistema (engl. *Cyber-Physical Systems* – CPS) i na njima zasnovanog Industrijskog interneta stvari (engl. *Industrial Internet of Things* – IIoT) u proizvodne pogone. U okviru ove paradigme, CPS se u proizvodnim pogonima primenjuju na svim nivoima piramide automatizacije, počevši od senzora i aktuatora pa do čitavog proizvodnog sistema [53]. Mehanički uređaji se proširuju lokalnim kontrolerima koji sadrže proračunske i komunikacione module i predstavljaju pametne uređaje – CPS u okviru kojih su fizički proces i proračunske sposobnosti integrisani kroz interakciju u realnom vremenu, a ponašanje sistema u celini definisano njegovim fizičkim i kibernetskim delom [63]. Uz primenu CPS u proizvodnim procesima otvaraju se nove mogućnosti u pogledu modularnosti, fleksibilnosti i rekonfigurabilnosti ICS. Pored toga, visoke proračunske performanse tih komponenti (definisane tipom mikroprocesora), kompaktnost koja

doprinosi brzoj i jednostavnoj ugradnji, kao i fleksibilnost u vidu reprogramiranja, podstiču novi stepen digitalizacije ICS [76]. Poboljšanje performansi koje se pritom ostvaruje, rezultira mnogim današnjim "pametnim" tehnologijama kao što su pametna proizvodnja, pametne električne mreže, pametni transport i pametne zgrade [128].

Implementacija CPS-a u proizvodnom pogonu dovodi do prelaska sa centralizovanih sistema upravljanja struktuiranih kroz piramidu automatizacije (slika 1) na potpuno distribuirane sisteme upravljanja u okviru kojih se upravljački zadaci realizuju kroz koordinisan rad pametnih uređaja gde lokalni kontroleri međusobno razmenjuju relevantne informacije kako bi postigli željeno ponašanje sistema u celini (slika 2b). Pritom, svi elementi hijerarhije automatizacije egzistiraju, ali u smislu funkcionalne hijerarhije raspoređeni su po mreži bez striktne piramidalne segmentacije [136] pri čemu je stroga hijerarhija automatizacije narušena kroz komunikaciju različitih uređaja unutar i preko nesusednih nivoa piramide. Trenutno, CPS se uspešno koriste u proizvodnim pogonima u brojnim aplikacijama, prvenstveno putem pametnih senzora i aktuatora. Ipak, pametni senzori i aktuatori se i dalje često integrišu u ICS na tradicionalan način – povezani su na centralni kontroler (npr. PLC) koji izvršava zadatak upravljanja (slika 2a). Na ovaj način, proračunske mogućnosti CPS-a, njihova modularnost i sposobnost da proizvodne sisteme učine prilagodljivim novim proizvodima nisu u potpunosti iskorišćeni. Postoji nekoliko razloga za to među kojima su inertnost inženjera da implementiraju nove trendove i nedostatak inženjerskih tehnika za projektovanje distribuiranih upravljačkih sistema, odnosno za distribuciju upravljačkih zadataka na pametne uređaje [6]. U skladu sa tim, postoji tendencija da se zadrže postojeće tehnike za dizajn ICS-a koje su praktično testirane i dokazane u mnoštvu primera iz stvarnog sveta. Ipak, zbog nemogućnosti tradicionalnog ICS-a da odgovori na veliku varijabilnost proizvoda, postoji trend da se određeni broj procesa u proizvodnji obavlja ručno što predstavlja korak unazad u pogledu sofisticiranosti procesa, tako da je prelazak na distribuirane sisteme upravljanja neminovan [54].



Slika 2: Arhitektura upravljačkog sistema: a) centralizovana; b) distribuirana

Nezavisno da li se radi o centralizovanoj ili distribuiranoj arhitekturi upravljanja, implementacija CPS i IIoT podrazumeva sveobuhvatnu komunikaciju između umreženih uređaja unutar ICS-a što donosi značajne izazove u oblasti sajber bezbednosti u okviru industrijskih pogona i ostalih kritičnih infrastruktura. Značajan deo ove komunikacije, naročito kada su mobilni uređaji u pitanju, vrši se bežično. Pored toga, kako bi se obezbedio brz odgovor na zahteve tržišta, određeni podaci se direktno iz pogona u realnom vremenu stavljaju na raspolaganje kooperantima pa ICS više nisu izolovana ostrva inherentno otporna na različite kibernetske izazove što potvrđuju uspešno izvedeni napadi na iransko nuklearno postrojenje 2010. godine [91], ukrajinsku električnu mrežu 2015. godine [13], nedavni napad na naftovod u SAD 2021.

godine [117] itd.[1] Iz navedenih primera, evidentan je uticaj kibernetskih napada na ICS, što za posledicu može imati disfunkciju industrijskih sistema i kritičnih infrastruktura, negativno uticati na životnu sredinu i zdravlje ljudi, čak i ugroziti ljudske živote. Da bi se potpuno ili delimično izbegle takve posledice, neophodan je razvoj sistema za sajber bezbednost.

Implementacijom sistema za zaštitu od napada poboljšava se ukupna bezbednost ICS i smanjuje rizik od kibernetskih napada. Ovi sistemi koriste različite tehnike za detekciju, prevenciju i reagovanje na potencijalne napade i ugrožavanje bezbednosti. Sistemi za zaštitu od napada uključuju funkcije kao što su provera transmitovanih podataka, upravljanje pristupom, autentifikacija i autorizacija korisnika, enkripcija podataka, detekcija anomalija itd. Ukupna strategija zaštite od napada predstavlja kombinaciju tehnoloških sistema, sigurnosnih procedura i politika, kao i edukaciju korisnika o sigurnosnim praksama.

Kao jedan od načina zaštite u ICS može se koristiti demilitarizovana zona (DMZ) sa funkcijom posrednika prilikom povezivanja uređaja koji zahtevaju visok stepen bezbednosti na veću nepouzdanu mrežu [45]. Na taj način se spoljašnjim uređajima omogućava pristup opremi u okviru DMZ, ali ne i ostatku mreže. S druge strane, uređaji iz privatne mreže mogu da pristupe DMZ, ali nemaju pristup javnoj mreži. Jedna od predloženih arhitektura bezbedne mreže u okviru ICS bazirana je na piramidi automatizacije (slika 1) i sadrži dve demilitarizovane zone: prvu na prelazu između OT i IT, tj. između MES i ERP kojom se razdvaja mreža na poslovnom nivou od mreže na nivou proizvodnje i druga pozicionirana na nivou preduzeća između ERP i interneta preko koje se ostvaruje povezivanje na globalnu mrežu.

Ipak, DMZ sama po sebi nije dovoljna za zaštitu sistema od različitih kibernetskih napada pogotovo sa konvergencijom i uklanjanjem granica između OT i IT do kojih primena IIoT u okviru Industrije 4.0 dovodi. Na primer, zaobilaženje DMZ dovelo bi napadača u povoljnu poziciju u kojoj bi na raspolaganju imao resurse sadržane u sistemu. Iz tog razloga, uvodi se koncept dubinske odbrane koji je definisan u okviru standarda NIST 800-82 [78] i predstavlja sveobuhvatnu strategiju sajber bezbednosti u okviru ICS koja uključuje primenu više nezavisnih bezbednosnih slojeva za zaštitu od kibernetskih napada. Ideja je da ako se probije jedan bezbednosni sloj, dodatni slojevi i dalje budu u mogućnosti da zaštite sistem od napada. Tehničke mere dubinskih slojeva podrazumevaju primenu fajervolova i sistema za detekciju napada (engl. *Intrusion Detection Systems* – IDS). Fajervol predstavlja prvu liniju odbrane i njegova uloga je da filtrira dolazni saobraćaj na nekom delu mreže i da omogući samo protok podataka koji dolaze od dobronamernih korisnika. Međutim, kako su fajervolovi kreirani na osnovu unapred propisanih pravila, oni ne mogu da pruže potpunu zaštitu od svih vrsta napada pa je upotreba IDS-a neophodna. Upotrebom IDS-a sistemi postaju bezbedni za napade već u fazi njihovog projektovanja, čime se značajno podiže stepen njihove zaštite. IDS se implementiraju na samom uređaju koji prima podatke korišćenjem komunikacionih veza, a njihova uloga je da pravovremeno detektuju napad ako on zaobiđe prethodne linije odbrane i to pre nego što napad postigne željene efekte. Pored toga, neke ključne komponente dubinske odbrane uključuju kontrolu pristupa i segmentaciju mreže. Kontrolom pristupa se osigurava da samo autorizovani korisnici mogu pristupiti osetljivim podacima i elementima sistema, dok se segmentacija mreže koristi za razdvajanje različitih delova mreže kako bi se ograničila potencijalna šteta koja bi mogla da bude uzrokovana dejstvom napada.

Predmet istraživanja ove disertacije predstavljaju sistemi za detekciju kibernetskih napada u okviru industrijskih sistema upravljanja pri čemu su u fokusu istraživanja metode za kreiranje IDS-a za sisteme sa kontinualnim upravljanjem. Ovde je potrebno naglasiti da se iz razloga o kojima će više biti reči u poglavlju 3 pred IDS u okviru ICS postavljaju drugačiji zahtevi u odnosu na IDS u okviru opštih informacionih tehnologija [121]. Prvo, zbog mogućih katastrofalnih posledica, IDS u ICS-u bi trebalo da pokažu nultu toleranciju na kibernetske

[1]Detaljni pregled i analiza kibernetskih napada koji su do sada ostavili najviše uticaja deo su poglavlja 2.

napade. Pored toga, mnoga postojeća rešenja projektovana su u vreme kada su ICS bili izolovani u fizički obezbeđenim okruženjima bez povezivanja na druge mreže (npr. internet) pa je najviše pažnje bilo usmereno na fizičku bezbednost, dok je sajber bezbednost često bila zanemarena [44, 109]. Pojam "bezbednosti prikrivenošću" (engl. *security by obscurity*) koji se često koristio pri projektovanju ICS, podrazumevao je da se u procesu projektovanja ICS principi bezbednosti oslanjaju na pretpostavku da je sistem bezbedan samim tim što je izolovan od spoljašnjeg sveta i što će se sve upravljačke operacije izvoditi na lokalnom nivou [72, 133]. Međutim, učestala komunikacija sastavnih delova ICS-a sa eksternim činiocima, gde neretko i uređaji na najnižem nivou imaju mogućnost direktnog povezivanja na internet, dovela je do situacije da većina napada potiče iz spoljnog sveta. Ovde je potrebno naglasiti da se u okviru ICS i dalje koristi veliki broj ranije uvedenih komunikacionih protokola koji ne mogu brzo i lako biti zamenjeni, a koji ne sadrže ni osnovne sisteme zaštite poput autentifikacije [131]. Na primer, Modbus protokol koji je namenjen serijskoj komunikaciji (najčešće između upravljačkih uređaja poput PLC-a) na fizičkom sloju koristi RS232 i RS485 protokole i inicijalno ne uključuje nikakve aspekte sajber bezbednosti.

Funkcionalnost ICS-a ostvaruje se korišćenjem namenskih softvera koji se implementiraju direktno na uređaj ili posredstvom operativnih sistema. Veliki broj programskih rešenja i/ili operativnih sistema koji se pritom koristi ne razmatraju nikakve ili uključuju samo osnovne aspekte sajber bezbednosti. Na primer, rad upravljačkih uređaja poput PLC-a često je uslovljen vremenskim ograničenjima pa se obično u te svrhe koriste operativni sistemi namenjeni za rad u realnom vremenu (engl. *Real-Time Operating System* – RTOS). Jedan od nedostataka ovih sistema predstavlja delimično ili potpuno odsustvo algoritama za upravljanje pristupom, tako da svi korisnici u tom slučaju imaju potpune privilegije u vidu raspoloživih informacija i resursa što povećava prostor za moguće napade [125]. Posebnu klasu softvera predstavljaju softveri koji se direktno pokreću na kontrolerima poput PLC-a i mikrokontrolera gde potencijalni napad može neposredno uticati na rad senzora i aktuatora. Pored toga, u ovo razmatranje spadaju i programi sa viših nivoa namenjeni za zadatke upravljanja i nadzora kontrolera kao delova CPS, koji se pokreću na bazi operativnih sistema opšte namene.

U okviru ove doktorske disertacije biće istraživani IDS koji su nezavisni od komunikacionih protokola i implementiraju se u okviru centralnog kontrolera ili na samim pametnim uređajima na prva dva nivoa piramide automatizacije. Posebna pažnja će biti usmerena na arhitekturu upravljačkih sistema i mogućnost implementacije IDS-a na odgovarajuće hardverske komponente u okviru nje, pri čemu će biti razmatrane specifičnosti sistema sa centralizovanim i distribuiranim upravljanjem pre svega u smislu proračunskih sposobnosti uređaja koji se koriste u okviru njih. Jedan od ključnih ciljeva pri projektovanju IDS-a koji se primenjuju na uređajima za upravljanje, jeste očuvanje potpune funkcionalnosti industrijskih sistema upravljanja pre svega uzimajući u obzir neophodnost njihovog rada u realnom vremenu i kašnjenja koje implementacija IDS-a inherentno uzrokuje.

1.1. Cilj istraživanja

Ova doktorska disertacija predstavlja rezultate istraživanja u oblasti sajber bezbednosti proizvodnih sistema i njen naučni cilj je razvoj metodologije za kreiranje algoritama koji predstavljaju osnove sistema za detekciju kibernetskih napada u okviru industrijskih sistema upravljanja. Osnovni ciljevi istraživanja su:

- Razvoj metoda za generisanje modela podataka koji se razmenjuju između uređaja u normalnim uslovima rada (bez napada) i to kako za sisteme iz kojih je moguće prikupiti dovoljnu količinu podataka tako i za sisteme iz kojih je količina podataka koja se može prikupiti ograničena;

- Razvoj metoda zasnovanih na principima samonadgledanog učenja za kreiranje sistema za detekciju napada na komunikacione veze između CPS u okviru sistema za kontinualno upravljanje proizvodnim resursima uzimajući u obzir arhitekturu sistema upravljanja i mogućnost implementacije algoritma za detekciju napada na nekom od uređaja u okviru njega;
- Razvoj metode za automatski izbor algoritma (uključujući i sve njegove parametre) za detekciju napada na komunikacione veze između CPS koji će se efikasno koristiti u procesu detekcije napada u realnom sistemu;
- Verifikacija razvijenih metoda na javno dostupnim skupovima podataka i skupovima podataka dobijenih sa eksperimentalne instalacije;
- Implementacija i eksperimentalna verifikacija sistema za detekciju napada generisanih korišćenjem razvijenih metoda na kreiranoj laboratorijskoj instalaciji.

1.2. Polazne hipoteze

Na osnovu znanja i veština akumuliranih u dosadašnjim istraživanjima i detaljne analize relevantnih literaturnih izvora u oblasti detekcije kibernetskih napada na industrijske sisteme upravljanja s jedne i u oblastima obrade signala i mašinskog učenja s druge strane, a uzimajući u obzir arhitekturu upravljačkih sistema i trenutne potrebe industrije za sistemima za detekciju kibernetskih napada, definišu se sledeće polazne hipoteze u okviru ove disertacije:

- **Prva hipoteza**: Autoregresioni modeli podataka čija se komunikacija vrši između elemenata sistema upravljanja distribuiranih na pametne uređaje, a koji su kreirani korišćenjem tehnika mašinskog i dubokog učenja, mogu predstavljati osnovu za kreiranje sistema za detekciju napada na komunikacione veze u okviru sistema upravljanja proizvodnim resursima.
- **Druga hipoteza**: Iz familije autoregresionih modela podataka čija se komunikacija vrši između elemenata sistema upravljanja moguće je automatski odabrati model kao i sve ostale parametre algoritma za detekciju napada zasnovanog na tom modelu koji će se efikasno koristiti u procesu detekcije napada u realnom sistemu uzimajući u obzir arhitekturu sistema upravljanja.
- **Treća hipoteza**: Primenom algoritma za detekciju napada zasnovanih na mašinskom i dubokom učenju moguće je u realnom vremenu u okviru proračunski i energetski ograničenih kibernetsko-fizičkih sistema postići visok nivo detekcije napada ne izazivajući pritom kašnjenja koje će ugroziti funkcionalnost sistema.

1.3. Struktura rada

Doktorska disertacija je struktuirana u osam osnovnih poglavlja u kojima su predstavljena istraživanja i doprinos izabranoj naučnoj oblasti. Pored navedenih osam, u devetom poglavlju dat je i spisak korišćene literature.

Uvodno poglavlje kroz predmet i cilj istraživanja opisuje problematiku kojom se disertacija bavi, a koja je vezana za detekciju kibernetskih napada na sisteme za upravljanje proizvodnim resursima. Pored toga, u ovom poglavlju navedene su polazne hipoteze koje su tokom istraživanja i dokazane.

U drugom poglavlju analizirani su postojeći načini podele kibernetskih napada nakon čega je predložen novi model klasifikacije napada u ICS. Takođe, navedeni su neki od najpoznatijih kibernetskih napada koji su se desili u prošlosti i diskutovani su njihovi uticaji.

U trećem poglavlju izvršena je analiza relevantnih postojećih metoda za detekciju kibernetskih napada sa fokusom na tehnike zasnovane na podacima. Pored toga, analizirani su neki od najpoznatijih javno dostupnih skupova podataka koji su korišćeni za evaluaciju performansi metoda za detekciju kibernetskih napada. Na kraju poglavlja data je lista standarda iz oblasti sajber bezbednosti sa osvrtom na aspekte i podoblasti bezbednosti koje su u standardima razmatrane.

Četvrto poglavlje kroz dve osnovne faze prikazuje razvoj metodologije za kreiranje algoritama za detekciju kibernetskih napada. Prva faza odnosi se na oflajn generisanje algoritma za detekciju napada i u okviru poglavlja su opisani koraci ove faze. Nakon opisa procesa pretprocesiranja signala, predloženi su opšti oblici arhitektura korišćenih u oflajn procesu generisanja algoritama, definisani su kriterijumi za izbor odgovarajućeg modela zasnovanog na mašinskom učenju (engl. *Machine Learning* – ML), kao i postupak automatskog određivanja vrednosti praga detekcije. Takođe, u ovom delu opisane su tehnike zasnovane na mašinskom učenju koje su korišćene za modeliranje transmitovanih podataka. Druga faza odnosi se na predloženi algoritam onlajn detekcije kibernetskih napada koji se zasniva na predikciji izabranog modela i automatski određenoj vrednosti praga detekcije.

U petom poglavlju prezentovani su rezultati primene razvijenih algoritama za detekciju kibernetskih napada. Verifikacija razvijenih algoritama izvršena je na osnovu dve studije slučaja. Prva studija slučaja urađena je na javno dostupnom skupu podataka koji je široko zastupljen prilikom verifikacije metoda za detekciju napada što je iskorišćeno za komparativnu analizu postignutih performansi detekcije napada sa rezultatima prikazanim u relevantnim postojećim istraživanjima. Za potrebe druge studije slučaja korišćen je skup podataka dobijen iz eksperimentalne instalacije koja je kreirana u okviru doktorske disertacije. U okviru ove studije slučaja prikazana je implementacija razvijenih algoritama za detekciju napada na kontrolere pametnih uređaja i izvršena je validacija algoritama na eksperimentalnoj instalaciji.

Šesto poglavlje odnosi se na rešavanje problema nedovoljne količine podataka. U ovom poglavlju primenom metode za proširivanje skupa podataka generisani su podaci koji su kasnije u cilju verifikacije upotrebljeni u postupku kreiranja algoritama za detekciju kibernetskih napada.

U sedmom poglavlju predstavljena je procedura kreiranja i evaluacije algoritama namenjenih za detekciju kibernetskih napada na sekvence 2D signala. Prilikom razvoja modela ispitana je mogućnost primene arhitektura zasnovanih na različitim ML tehnikama. Performanse razvijenih algoritama za detekciju napada testirane su na sekvencama 2D signala koje su kreirane u okviru ove doktorske disertacije.

Na kraju, u osmom poglavlju izvršena je diskusija rezultata primene algoritama za detekciju kibernetskih napada razvijenih u okviru ove disertacije. Nakon analize postignutih rezultata, definisane su prednosti i nedostaci predložene metodologije i izneti relevantni zaključci kao i smernice za budući rad i istraživanje.

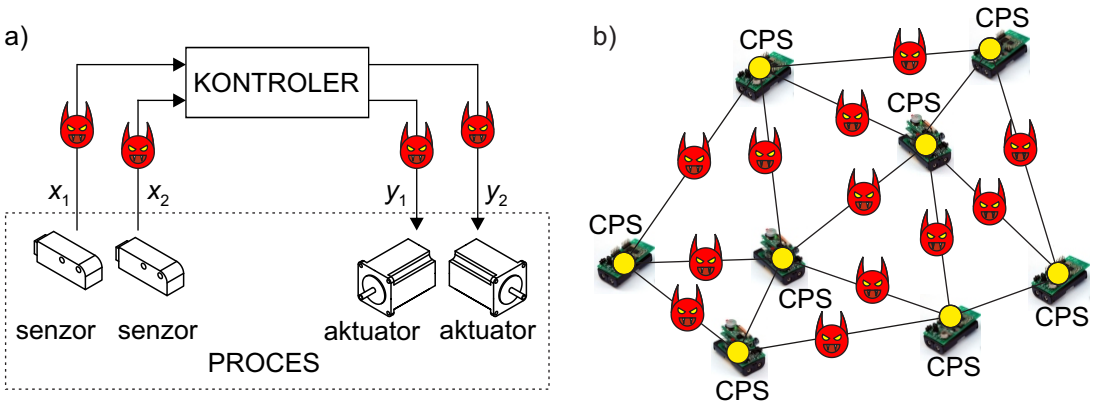
2. Analiza i klasifikacija postojećih vrsta kibernetskih napada

U okviru ovog poglavlja izvršena je analiza postojećih načina podele kibernetskih napada nakon čega je predložen novi model klasifikacije napada u ICS. Pored toga, navedeni su i neki od najpoznatijih uspešno izvedenih napada na ICS.

Upravljačka petlja ICS-a koristi kontrolere (PLC, mikrokontroler i sl.), senzore i aktuatore za upravljanje fizičkim procesima. Nezavisno od arhitekture upravljačkog sistema, uz primenu pametnih uređaja i IIoT, funkcionisanje ICS-a zasniva se na sveobuhvatnoj komunikaciji koja omogućava razmenu podataka u realnom vremenu i koordinaciju između različitih komponenti sistema.

U slučaju centralizovane arhitekture upravljačkog sistema (slika 3a), kontroler interpretira senzorske signale x_k i na osnovu implementiranog algoritma upravljanja šalje odgovarajuće komande y_k aktuatoru koji izvršavanjem odgovarajuće akcije zatvara upravljačku petlju. Kao tačke učestale komunikacije, veze senzor/kontroler i kontroler/aktuator predstavljaju osetljiva mesta za različite kibernetske napade.

S druge strane, kod distribuiranog sistema upravljanja učestala razmena podataka izvršava se komunikacionim linijama između lokalnih kontrolera u okviru pametnih uređaja i predstavlja mesto potencijalnih kibernetskih napada od strane različitih protivnika (slika 3b). Kako senzori i aktuatori neposredno učestvuju u izvođenju procesa, kompromitovanje izmerene vrednosti senzora i/ili upravljačke komande poslate aktuatoru može prouzrokovati direktan uticaj na ostale delove sistema, njegove performanse, bezbednost čoveka i okoline.



Slika 3: Upravljački sistemi sa napadima na komunikacione linije: a) centralizovana arhitektura; b) distribuirana arhitektura

Sa slike 3 jasno je uočljivo da se kao posledica sveobuhvatne komunikacije stvara veliki broj tačaka za potencijalno dejstvo kibernetskih napada. U nekim slučajevima dovoljno je kompromitovanje samo jedne komunikacione linije kako bi se postigla disfunkcija celog sistema.

2.1. Klasifikacija kibernetskih napada

Klasifikacija kibernetskih napada može biti od velikog značaja za razumevanje načina njihovog delovanja i uticaja na sisteme, što se može iskoristiti prilikom projektovanja mehanizama za detekciju napada. Pored toga, na osnovu definisanih klasa napada mogu se kreirati određene akcije koje se izvršavaju u trenutku njihove detekcije i na taj način smanjiti ili potpuno izbeći posledice koje napad može izazvati.

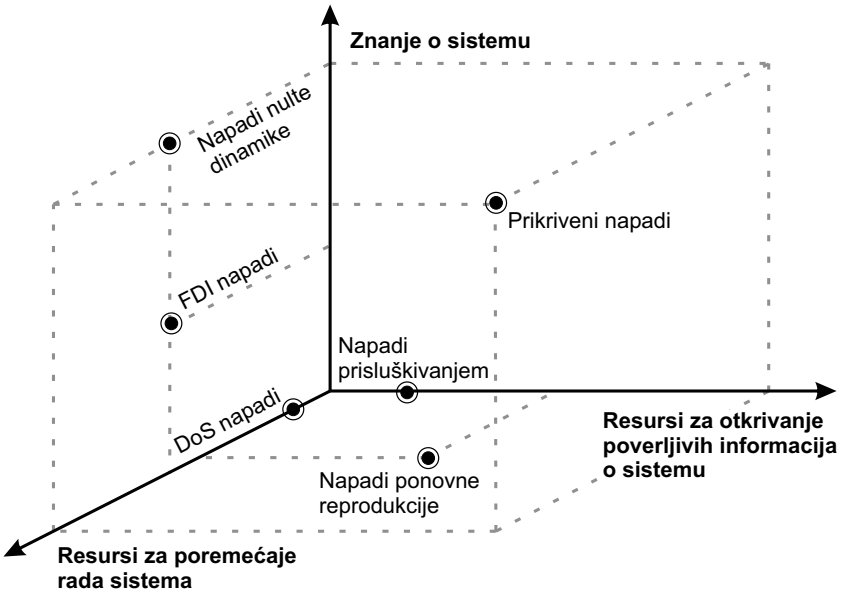
Jedna od prvih klasifikacija napada na CPS data je u okviru [23] gde su napadi podeljeni u tri osnovne grupe: 1) napadi uskraćivanjem pristupa servisu (engl. *Denial of Service* – DoS), 2)

napadi ponovne reprodukcije (engl. *replay*) i 3) napadi obmanom (engl. *deception*). Takođe, u okviru navedenog rada za date tipove napada definisan je i opšti matematički model gde su DoS napadi opisani kao $\overline{y}_k \in \emptyset$, što podrazumeva da je slanje podataka y_k neuspešno, gde y_k i $\overline{y}_k$ predstavljaju poslate i primljene podatke. S druge strane, napadi ponovne reprodukcije opisani su sa $y_k \in Y_k$, gde je Y_k skup prethodno transmitovanih podataka, dok su napadi obmanom definisani sa $\overline{y}_k = y_k + y_k^a$, gde y_k^a predstavlja izmenu transmitovanih podataka prouzrokovanu napadom.

U [25] napadi na pametne mreže za napajanje energijom (engl. *smart grids*) u zavisnosti od lokacije njihovog delovanja podeljeni su u tri grupe: 1) napadi na IT sloj, 2) napadi na sloj komunikacije i 3) napadi na energetski sloj. Detaljnija klasifikacija napada izvršena je pojedinačno za svaku od tri grupe. Kako su u fokusu ove disertacije napadi na ICS, najveća pažnja u ovoj podeli biće usmerena na napade na energetski sloj. U okviru tog sloja napadi se lokacijski dele na: a) napade na upravljačke stanice i b) napade na opremu (komponente). Dalje, u napade na upravljačke stanice svrstavaju se dva tipa napada ubrizgavanjem lažnih podataka (engl. *False Data Injection* – FDI): DC-FDI i AC-FDI, u zavisnosti da li napadnuti sistemi koriste jednosmernu (DC) ili naizmeničnu (AC) struju.

Klasifikacija napada na upravljački sistem CPS predstavljena je i u [22] gde su napadi svrstani u tri različite kategorije: 1) DoS, 2) Napad degradacijom servisa (engl. *Service degradation*) i 3) Obaveštajni kibernetsko-fizički napadi (engl. *Cyber-Physical Intelligence* – CPI). Fokus ovog rada bio je, između ostalog, i na DoS napadima pa je iz tog razloga ovaj tip napada dodatno podeljen u tri grupe: a) DoS dodavanje smetnji (uvođenje dodatnog kašnjenja na komunikacionim linijama između kontrolera i senzora/aktuatora), b) DoS gubitak podataka (potpuno se onemogućava protok podataka komunikacionim linijama između kontrolera i senzora/aktuatora) i c) DoS-FDI (lažni podaci se šalju kontroleru na način kao da ih dobija sa senzora, odnosno aktuatoru na način kao da ih dobija od kontrolera).

Različiti tipovi napada na CPS okarakterisani su u [114] kroz trodimenzionalni prostor u kome ose predstavljaju znanje o sistemu, resurse za poremećaje rada sistema (npr. narušavanje integriteta i/ili dostupnosti podataka) i resurse za otkrivanje poverljivih informacija o sistemu. Pozicija napada u okviru definisanog prostora određena je nivoom informacija koje su potrebne za njegovo uspešno izvršavanje, kao i količinom raspoloživih resursa. Dovoljno znanje o sistemu može omogućiti napadaču da kreira kompleksnije napade koji su teži za detekciju, što kasnije može prouzrokovati veće posledice na rad sistema. Posedovanje resursa za otkrivanje poverljivih informacija o sistemu omogućava da se tokom napada dobiju potrebne informacije, ali korišćenjem samo ovih resursa ne može se ugroziti funkcionalnost sistema. S druge strane, upotrebom resursa za poremećaje rada sistema, moguće je tokom dejstva napada uticati na funkcionalnost delova sistema i sistema u celini. Shodno opisanim svojstvima predloženog trodimenzionalnog prostora, predstavljeno je ukupno šest tipova napada: 1) DoS, 2) FDI, 3) napadi ponovne reprodukcije, 4) napadi nulte dinamike (engl. *zero dynamics*), 5) napadi prisluškivanjem (engl. *eavesdropping*) i 6) prikriveni (engl. *covert*) napadi (slika 4). Napad nulte dinamike zahteva visok nivo znanja o sistemu i ima za cilj da manipuliše dinamikom ciljanog sistema na takav način da njegovo ponašanje izgleda normalno, ali je zapravo kompromitovano. U slučaju ovog napada, kada napadač dobije pristup ciljanom sistemu, on pokušava da manipuliše unutrašnjom dinamikom sistema, umesto da menja vrednosti ulaza ili izlaza. Na primer, modifikacija povratne sprege upravljačkog sistema može da izazove oscilovanja koja bi dalje prouzrokovala ulazak sistema u nestabilno stanje.



Slika 4: Trodimenzionalni prostor napada [114]

Sa slike 4 može se uočiti da, u poređenju sa ostalim tipovima napada, prikriveni napad uzima najveće vrednosti po sve tri ose što znači da ga je najteže detektovati, njegovim dejstvom se mogu prikupiti velike količine korisnih informacija o sistemu i prouzrokovati najveće posledice na sistem. Međutim, za uspešno sprovođenje ovog napada potrebno je i najviše ulaznih informacija o sistemu koje često nisu na raspolaganju napadaču.

Prema [22], proces komunikacije između pametnih uređaja može biti ugrožen na tri osnovna načina: izazivanjem smetnji poput uvođenja šuma, presretanjem, tj. gubitkom poslatih paketa kao nosača informacija i ubrizgavanjem lažnih podataka.

Iako navedene analize obuhvataju neke od najčešće korišćenih vrsta napada, čini se da su predloženi pristupi usko vezani za određenu klasu sistema. Stoga je u nastavku data sveobuhvatna podela kibernetskih napada u ICS (slika 5) koja na najvišem nivou obuhvata tri osnovne klase [55, 131]:

1. **Napadi uskraćivanjem pristupa servisu – DoS napadi** (engl. *Denial of Service*);
2. **Obaveštajni kibernetsko-fizički napadi – CPI napadi** (engl. *Cyber-Physical Intelligence*);
3. **Napadi obmanom** (engl. *deception*).

DoS napadi koriste različite mehanizme da bi trajno ili privremeno ugrozili raspoloživost podataka. Oni nisu prikriveni, ali ukoliko deluju u pravom trenutku mogu prouzrokovati značajne posledice. DoS napadi mogu često biti pogrešno okarakterisani kao problemi u funkcionisanju mreže. Ovi napadi obuhvataju [93]:

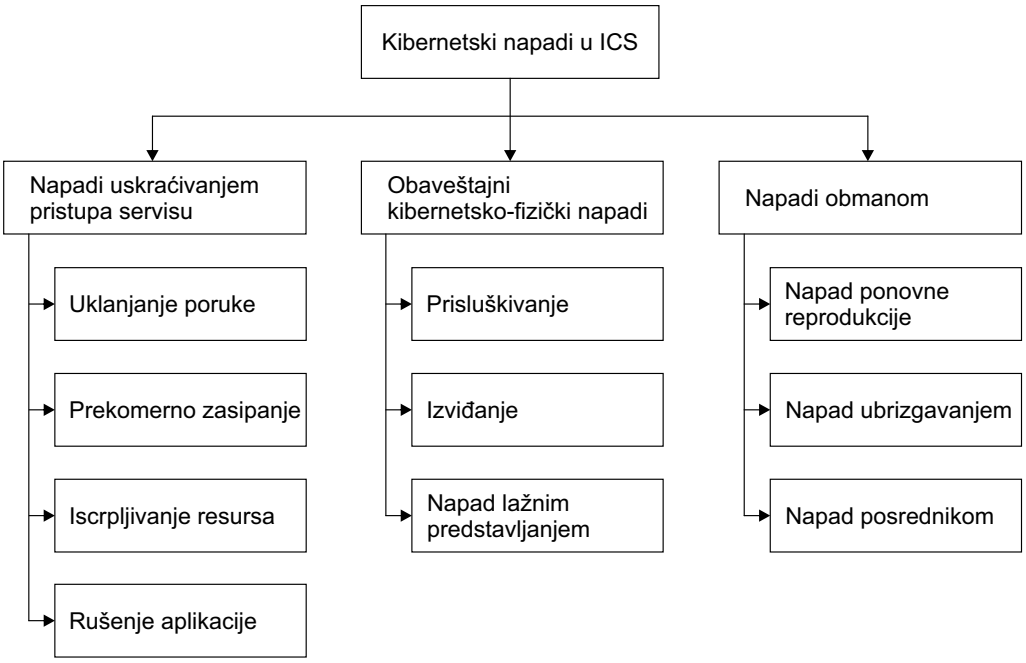
- **Uklanjanje poruke** (engl. *message removal*) gde napadač direktno sprečava dolazak poruke na željeno odredište. Iako napadač može izabrati da ukloni određene poruke, za pokretanje ovog napada nije neophodno kreiranje zlonamerne poruke sa specifičnim sadržajem [129];
- **Prekomerno zasipanje** (engl. *flooding*) gde napadač kontinualno šalje veliki broj poruka koji nadmašuje dostupni protok mreže i/ili poruke sa velikim brojem zahteva koje server ne može ispuniti, čime se drugim korisnicima blokira pristup mreži;

- **Iscrpljivanje resursa** – u okviru ovog napada šalje se preveliki broj validnih poruka (svaka komanda definisana porukom zauzima po jedan dostupni resurs) tako da se zauzmu svi raspoloživi resursi i onemogući pristup drugim korisnicima;
- **Rušenje aplikacije** izazvano je slanjem specifične poruke koja onemogućava rad aplikacije.

Napadi obmanom predstavljaju perfidne i najbolje skrivene napade koji mogu izazvati najgore posledice na rad sistema. Mogu biti u različitim oblicima kao što su:

- **Napad ponovne reprodukcije** (engl. *Replay*) – beleži podatke u jednom periodu rada sistema i ponovo ih reprodukuje u drugom;
- **Napad ubrizgavanjem** (engl. *Injection*) – ubrizgava izmenjene/pogrešne podatke na komunikacionu liniju;
- **Napad posrednikom** (engl. *Man-in-the-Middle* – MITM) – preuzima kontrolu nad komunikacijom između uređaja, kreira podatke i šalje ih između uređaja prema svom nahođenju.

U ovim napadima protivnici pokušavaju na različite načine da ostanu neprimetni i postignu željeni negativni efekat na performanse sistema koristeći dostupne alate za izmenu podataka.



Slika 5: Taksonomija kibernetskih napada u ICS

Konačno, obaveštajni napadi narušavaju poverljivost podataka kroz prikupljanje informacija o sistemu i identifikaciju rada sistema. Ovi napadi po pravilu prethode napadima čije je izvršavanje uslovljeno informacijama o sistemu koji napadaju (pre svega napadi obmanom), a mogu se klasifikovati u sledeće tri vrste:

- **Prisluškivanje** (engl. *Eavesdropping*) se koristi da presretne i preuzme podatke koji se razmenjuju unutar mreže i na taj način narušava poverljivost mreže;
- **Izviđanje** (engl. *Reconnaissance*) može biti aktivno i pasivno. Aktivnim izviđanjem se ostvaruje interakcija sa ciljanim sistemom kako bi se otkrile potencijalne vulnerabilnosti, dok se kod pasivnog pristupa isti cilj pokušava postići samo osmatranjem (bez interakcije sa sistemom) [21];

- **Napad lažnim predstavljanjem** (engl. *Spoofing*) je napad u kojem se napadač koristeći lažne podatke uspešno predstavlja kao neko drugi (kome je sistem dostupan) i tako obezbeđuje pristup komunikacionim vezama.

Razmatranjem osnovnih klasa napada (DoS napada, CPI napada i napada obmanom) mogu se uočiti razlike u vidu posledica koje se njihovim dejstvom prouzrokuju. Iako izazvane posledice zavise od konkretnog napada, načina njegovog izvođenja, nivoa zaštite sistema koji se napada itd, može se reći da su u opštem slučaju napadi obmanom najopasniji. Ova tvrdnja proističe iz činjenice da napadi obmanom direktno utiču (ubrizgavanjem, izmenom vrednosti itd.) na podatke koji često imaju krucijalni značaj za funkcionisanje sistema. Pored toga, u poređenju sa ostalim klasama, napadi obmanom smatraju se najbolje skrivenim napadima. Iako su u stanju da izazovu značajne posledice na rad sistema, DoS napadi se zbog svoje neprikrivenosti mogu smatrati manje opasnim u odnosu na napade obmanom. S druge strane, CPI napadi važe za najmanje opasne, jer po pravilu ne ostvaruju direktan uticaj na sistem. Ipak, u slučajevima kada je dejstvo jednog napada (npr. napada obmanom) uslovljeno informacijama o sistemu koje su dobijene primenom nekog drugog napada (npr. CPI napada), teško je jasno definisati njihov pojedinačni udeo u izazvanim posledicama.

2.2. Pregled realizovanih kibernetskih napada na industrijske sisteme upravljanja

Kibernetski napadi predstavljaju jednu od najznačajnijih i najučestalijih pretnji za rad savremenih ICS. U nastavku je hronološki navedeno nekoliko kibernetskih napada čije su posledice otvorile nova pitanja zaštite ICS i rapidno ubrzale razvoj u oblasti sajber bezbednosti u industrijskim sistemima upravljanja.

Stuxnet napad [91] koji se desio 2010. godine smatra se prekretnicom u sajber bezbednosti za ICS imajući u vidu da je nakon njega došlo do značajnih ulaganja i intenziviranja istraživanja u ovoj oblasti. Nakon ovog napada javnost je postala svesna potencijalne štete koju kibernetski napadi na industrijska postrojenja mogu izazvati. *Stuxnet* napad je lansiran na iransko postrojenje za obogaćivanje uranijuma sa ciljem da se zaustavi razvoj nuklearnog programa u ovoj državi. Naime, malver (engl. *malware*) koji je unet u sistem putem USB fleš memorije je preko upravljačkog sistema zasnovanog na Profinet protokolu izazvao poremećaj na rotoru nuklearne centrifuge promenom njegove brzine i radnog pritiska iznad predviđenih granica. Uticaj napada prouzrokovao je istovremeno disfunkciju više sistema namenjenih za izvođenje centrifugalnih procesa i na praktičan način ukazao na mogućnost napada na ICS i razmere posledica koje ovi napadi mogu imati.

BlackEnergy napad (verzija 1.0 pokrenuta oko 2007. godine) omogućava različite tipove napada kako bi se ugrozio rad ciljanog sistema [58]. Ovaj napad se može svrstati u grupu obaveštajnih i najčešće se koristi za ciljano sprovođenje distribuirane verzije napada uskraćivanjem pristupa servisu (engl. *Distributed Denial of Service* – DDoS), kao i za pokretanje drugih napada izviđanjem. Ubrzo nakon nastanka ovaj napad je prilagođen kako bi se lakše pristupilo SCADA sistemima u ICS, najčešće u okviru energetskih postrojenja.

Jedna od najpoznatijih primena *BlackEnergy* napada je napad na ukrajinsku električnu mrežu koji se desio 2015. godine [13] uvođenjem poremećaja u sistem za distribuciju električne energije. Malver koji je putem imejla pušten u sistem obezbedio je pristup celokupnoj mreži za distribuciju električne energije i isključivanje glavnih prekidača. Pored toga, pokrenuto je i izvršavanje DoS napada koji je obustavio servis telefonske mreže i sistem za besprekidno napajanje (engl. *Uninterruptable Power Supply* – UPS). Posledice napada ogledale su se kroz nestanak struje u tri provincije kojima je obuhvaćeno oko 225.000 korisnika.

Kibernetski napad na sistem za preradu vode desio se 2021. godine u Floridi (SAD) [118]. Nakon što je preko softvera za udaljeni pristup uspostavljena komunikaciona veza sa upra-

vljačkim sistemom, postepeno je povećavana koncentracija natrijum hidroksida koji se koristi u procesu prerade vode. Kako je povećanje procenta ove supstance (čak 111 puta u odnosu na predviđenu količinu – sa 100 na 11.100 jedinica) primećeno, privremeno je zaustavljen proces isporuke prerađene vode. U suprotnom, povišena koncentracija natrijum hidroksida direktno bi mogla uticati na zdravlje ljudi koji tu vodu koriste.

Naftovod SAD bio je meta jednog od najvećih kibernetskih napada koji se desio u 2021. godini [117]. Napad je otpočeo preuzimanjem 100 GB podataka u roku od 2 sata u kojima su bile sadržane bitne informacije o napadnutom sistemu, a koje su poslužile daljem razvoju napada. Uticaj napada prouzrokovao je potpuno isključivanje naftovoda kojim je u regularnim uslovima rada transportovano najmanje 2,5 miliona barela nafte dnevno. Rad čitavog postrojenja bio je obustavljen 6 dana, da bi nakon pokretanja bilo potrebno još 3 dana za dostizanje njegove potpune funkcionalnosti. Samo u cilju dekripcije podataka koju su kasnije napadači obezbedili utrošeno je oko 4,4 miliona dolara.

Iz navedenih primera lako je uočiti različite posledice koje mogu izazvati kibernetski napadi. Jedna od njih je disfunkcija pojedinih uređaja ili sistema u celini što prouzrokuje smanjenje raspoloživih kapaciteta, odnosno potpuno obustavljanje rada napadnutog sistema. Ovakvi kibernetski napadi po pravilu ostavljaju značajne ekonomske posledice. Čak i kada je sistem funkcionalan, promena vrednosti pojedinih parametara izvan predviđenih granica može uticati na zdravlje ljudi, kao u slučaju napada na sistem za preradu vode na Floridi. Pored toga, potencijalne katastrofe izazvane dejstvom kibernetskih napada mogu direktno ugroziti bezbednost ljudi na šta je na eklatantan način ukazao *Stuxnet* napad.

3. Pregled i analiza postojećih modela za detekciju kibernetskih napada

Kao što je prethodno navedeno u poglavlju 1, razvoj i primena sistema za detekciju kibernetskih napada umnogome doprinosi podizanju stepena zaštite i poboljšanju ukupne bezbednosti ICS-a. U ovom poglavlju predstavljene su specifičnosti zahteva koji se postavljaju pred sisteme za zaštitu od kibernetskih napada u ICS u odnosu na opšte IT i izvršen je pregled aktuelnih istraživanja vezanih za razvoj metoda za detekciju kibernetskih napada u okviru ICS. Takođe, izvršena je i analiza najpoznatijih javno dostupnih skupova podataka koji se koriste za generisanje sistema za detekciju napada u ICS i za testiranje njihovih performansi. Konačno, postojeći standardi iz oblasti sajber bezbednosti navedeni su i analizirani na kraju ovog poglavlja.

3.1. Zahtevi sajber bezbednosti za opšte IT i ICS

Sajber bezbednost u opštim informacionim tehnologijama bila je predmet značajnog broja istraživanja na osnovu kojih su razvijena i implementirana zaštitna rešenja koja se regularno primenjuju u svakodnevnom životu i radu. S druge strane, kao što je navedeno u prethodnim poglavljima, sajber bezbednost u okviru ICS predstavlja relativno novu istraživačku oblast koja je postala aktuelna sa ukidanjem granica izmedju OT i IT, a intenzivirana je nakon *Stuxnet* napada. Nezavisno od toga da li se razmatraju opšte IT ili ICS, prilikom razmene podataka pred sistem sajber bezbednosti postavljaju se tri osnovna zahteva: poverljivost, integritet i raspoloživost podataka [6, 109]. **Poverljivost** podrazumeva da su podaci dostupni samo korisnicima kojima su namenjeni. **Integritet** podataka postiže se osiguravanjem da primaoci dobijaju podatke koji nisu izmenjeni u toku prenosa i od pošiljalaca koji su ih zaista poslali, dok **raspoloživost** ukazuje na to da podaci moraju biti dostupni korisnicima u odgovarajućim vremenskim rokovima.

Iako su navedeni zahtevi bezbednosti podataka isti u slučaju opštih IT i ICS, njihov pojedinačni značaj se suštinski razlikuje. Osnovna razlika potiče od prirode sistema koji se razmatraju, odnosno resursa kojim oni upravljaju. Naime, u slučaju ICS glavni resurs predstavljaju fizički uređaji, dok je kod opštih IT sistema fokus usmeren na podatke. U skladu sa tim, potencijalne posledice kompromitovanja sistema su suštinski različite pa se kod opštih IT sistema uglavnom svode na ekonomske posledice, dok kod ICS može doći i do trajnih oštećenja opreme, ugrožavanja životne sredine, zdravlja pa čak i života ljudi.

Različiti zahtevi za sajber bezbednost u ova dva tipa sistema proizilaze iz njihovih karakteristika. Upravljački zadaci u ICS se uglavnom izvršavaju u realnom vremenu pa je za adekvatno funkcionisanje ovih sistema neophodna pravovremena raspoloživost podataka i ostvarivanje brzog odziva. Dakle, u slučaju ICS-a krucijalni zahtev je da podaci budu brzo raspoloživi primaocu, ali je povoljna okolnost da je za ove sisteme specifično da se po pravilu vrši prenos manje količine podataka. S druge strane, raspoloživost podataka u realnom vremenu je kod opštih IT manje značajna jer se u najvećem broju slučajeva kašnjenje podataka može tolerisati, ali se pred ove sisteme postavljaju zahtevi većih protoka podataka u odnosu na ICS. Takođe, kod opštih IT poverljivost i integritet podataka su najznačajniji naročito u internet bankarstvu, trgovini i sličnim aplikacijama, dok je kod ICS poverljivost podataka manjeg prioriteta u odnosu na njihovu raspoloživost imajući u vidu da je zaštita opreme, a samim tim i bezbednost, zdravlje na radu i zaštita životne sredine kruciajlna.

Još jedna od razlika između opštih IT i ICS ogleda se u načinu na koji je moguće implementirati i ažurirati sisteme zaštite. Naime, u okviru ICS upravljački resursi moraju uvek biti raspoloživi pa strategija prekida rada sistema (poput restartovanja u slučaju opštih IT) nije

moguća *ad hoc*, već ovakvi prekidi moraju biti unapred planirani. Vrlo značajna razlika između opštih IT i ICS, koja ima izuzetno važan uticaj na implementirane sisteme sajber bezbednosti, jesu i hardverski resursi. Naime, proračunske sposobnosti ovih resursa u okviru ICS su značajno manje nego kod opštih IT. Pored toga, kao što je u uvodu navedeno, u ICS se često koriste stari komunikacioni protokoli koji se iz ekonomskih ili tehničkih razloga ne mogu brzo zameniti, a u okviru kojih se po pravilu ne mogu implementirati savremeni sistemi zaštite. Načelno, životni vek komponenata u okviru ICS je duži (minimum 10-15 godina, a vrlo često i duže) nego kod opštih IT (3-5 godina) što utiče i na mogućnost implementacije složenih zaštitnih mehanizama. Iz izloženih razloga unapređenje softvera dodavanjem bezbednosnih mehanizama, što je česta praksa u opštim IT, kod ICS je retko moguća. Iz navedenog može se zaključiti da su zahtevi kod opštih IT sistema poređani (počevši od najbitnijeg) na sledeći način: poverljivost/integritet/raspoloživost – CIA (engl. *Confidentiality/Integrity/Availability*), dok se kod ICS taj redosled menja na: raspoloživost/integritet/poverljivost – AIC (engl. *Availability/Integrity/Confidentiality*).

Shodno navedenim zahtevima sajber bezbednosti koji se postavljaju pred opšte IT i ICS, može se zaključiti da postojeći IDS koji se primenjuju u opštim IT nisu odgovarajući za detekciju kibernetskih napada u ICS pa je neophodno kreiranje IDS koji pre svega garantuju raspoloživost podataka u realnom vremenu. Ključni uslovi za ispunjavanje ovog zahteva podrazumevaju da kompleksnost IDS-a odgovara proračunskim resursima uređaja na koje se implementira, kao i da se IDS može primeniti u okviru postojećih komunikacionih protokola u okviru ICS.

3.2. Postojeće metode za detekciju kibernetskih napada

IDS u opštem slučaju mogu biti zasnovani na otkrivanju zloupotreba, detekciji anomalija ili na hibridnim metodama. Tehnike otkrivanja zloupotrebe koriste bazu znanja koja sadrži unapred poznate napade. Ovim pristupom porede se trenutne aktivnosti sa napadima sadržanim u bazi znanja i napad se otkriva kada se primeti aktivnost koja u potpunosti odgovara napadu definisanom u bazi. Primena ovih tehnika obezbeđuje efikasno otkrivanje napada sa malim procentom lažno pozitivnih rezultata [34]. Međutim, bilo koji napad koji se ne nalazi u bazi znanja (uključujući i male varijacije napada sadržanih u bazi) biće okarakterisan kao normalno ponašanje. Iz tog razloga, IDS zasnovani na ovom pristupu zahtevaju stalno ažuriranje baze znanja i nisu pogodni za otkrivanje novih i sofisticiranih napada, što je jedan od ključnih zahteva IDS u okviru ICS imajući u vidu neophodnost nulte tolerancije na napade.

S druge strane, pristup detekcije anomalija zasnovan je na modelu normalnog ponašanja sistema i kod ovog pristupa **napad se otkriva kao odstupanje između modeliranog i ostvarenog rada sistema**. Pojam normalnog ponašanja sistema podrazumeva da sistem funkcioniše u predviđenim uslovima bez dejstva i uticaja bilo koje vrste kibernetskih napada. Granica između normalnog i ponašanja sistema pod dejstvom napada često nije jednoznačno i precizno određena. Na primer, dejstvo napada ne mora uzrokovati značajna odstupanja u odnosu na minimalne i maksimalne vrednosti podataka u normalnim uslovima rada, već se može ogledati kroz promenu dinamike unutar tih granica. Takođe, uz neophodna znanja o sistemu, napadači često projektuju napade sa tendencijom da njihovo dejstvo na sistem bude slično normalnom ponašanju, ali da istovremeno prouzrokuju značajne posledice. Iz navedenih razloga, modeliranje normalnog ponašanja kao sastavnog dela projektovanja IDS-a predstavlja izazovan zadatak. Zbog mogućih katastrofalnih posledica, prilikom kreiranja IDS-a razmatraju se najgori scenariji koji uključuju generisanje inovativnih napada. U tom kontekstu, detekcija anomalija predstavlja izabranu tehniku koja će biti korišćena u okviru ove disertacije.

Detekcija anomalija može biti zasnovana na mreži ili na domaćinu (engl. *host*) [11]. Detekcija anomalija zasnovana na mreži analizira mrežni saobraćaj kako bi se identifikovali obrasci koji odstupaju od normalnog ponašanja. Pored sadržaja koji se prenosi preko mreže, u analizu su

često uključene karakteristike paketa (npr. struktura i broj bitova), IP adrese i portovi sa kojih se vrši slanje. Dobra strana ovog pristupa ogleda se u tome što je primenom samo jednog IDS-a često moguće obuhvatiti mrežu velikih razmera. Takođe, implementacija IDS-a zasnovanog na mreži po pravilu je relativno jednostavna. Međutim, kada su u pitanju velike mreže u kojima je prisutna učestala razmena podataka, javlja se problem analize i obrade svih paketa u realnom vremenu [29]. Takođe, mnogi komunikacioni protokoli implementirani u ICS ne uzimaju u obzir sajber bezbednost što znatno ograničava mogućnost primene IDS-a zasnovanog na mreži.

S druge strane, detekcijom anomalija koja je zasnovana na domaćinu nadgleda se ponašanje pojedinačnog uređaja i upozorava se sistem ako se dese aktivnosti koje u određenoj meri ne odgovaraju uobičajenom ponašanju sistema [75]. Pozicija na kojoj se implementiraju algoritmi za detekciju anomalija zasnovanih na domaćinu daje mogućnost da budu detektovani napadi koji su prethodno prošli stepene zaštite poput fajervola, DMZ ili IDS-a zasnovanih na mreži.

Iz navedenih razloga u okviru ove disertacije razmatraće se detekcija napada zasnovana na domaćinu, a ne na mreži. Shodno tome, u nastavku je prikazan pregled postojećih metoda koji spadaju u ovu vrstu tehnika za detekciju anomalija. U zavisnosti od pristupa koji se koristi, metode za detekciju anomalija u ICS mogu se svrstati u dve osnovne klase: 1) metode zasnovane na dizajnu i 2) metode zasnovane na podacima.

3.2.1. Metode za detekciju anomalija zasnovane na dizajnu

Metode za detekciju anomalija zasnovane na dizajnu podrazumevaju matematički formalizovan model ponašanja sistema u normalnim uslovima rada. Ovi pristupi su takođe poznati i kao pristupi zasnovani na specifikaciji [101].

Iako su vršena određena istraživanja i u oblasti kontinualnih sistema upravljanja o čemu će kasnije biti reči, metode za detekciju napada zasnovane na dizajnu se prevashodno primenjuju kod sistema sa diskretnim događajima (engl. *Discrete Event Systems* – DES). Za kreiranje modela sistema sa diskretnim događajima korišćene su brojne tehnike, a jedna od najzastupljenijih je teorija nadzornog upravljanja (engl. *Supervisory Control Theory* – SCT).

SCT u osnovi uključuje modeliranje supervizora koji predstavlja upravljački entitet čiji je cilj da koordinira ponašanjem postrojenja ili sistema koji se takođe može modelirati korišćenjem SCT. Supervizor je na osnovu modela ponašanja postrojenja projektovan kao automat i opisuje moguća stanja postrojenja i prelaze između njih. U kontekstu SCT, postrojenje se odnosi na fizički ili logički sistem kojim se upravlja. To može biti bilo koji sistem koji ima konačan skup stanja čija se promena ostvaruje kao rezultat diskretnih događaja [116].

Pristup za modeliranje i detekciju napada obmanom na upravljačke komande aktuatora i očitavanja senzora u udaljenim postrojenjima u okviru DES predstavljen je u [10], gde se stanje sistema pod dejstvom napada modelira korišćenjem konačnih automata i SCT, dok se pristupi razvijeni za identifikaciju greške u DES koriste za detekciju napada i sprečavanje dostizanja neželjenih stanja sistema. Sličan pristup je za napade posrednikom na senzorske signale prikazan u okviru [65]. Ista grupa autora je u okviru [67] predložila strategiju odbrane za napade iz [10] i [65] zasnovanu na SCT. U sklopu istraživanja prikazanog u [67] napadi su podeljeni u dve grupe u zavisnosti od toga da li ih je moguće otkriti i pokazano je da je u određenim slučajevima moguće izbeći posledice napada iako ga nije moguće otkriti. Pored toga, u [66] predloženi su mehanizmi zasnovani na SCT za implementaciju bezbednosnih modula za napade iz [65]. U ovom pristupu, bezbednosni modul nakon detekcije napada obaveštava supervizora koji onemogućava sve upravljive događaje kako bi se izbegla neželjena stanja sistema.

Inteligentni napadači koji imaju predznanje o upravljačkom podsistemu DES koji napadaju i koji imaju mogućnost proizvoljne izmene senzorskih signala u okviru ovog sistema modelirani su u [110] korišćenjem SCT i predložena je metoda za generisanje supervizora robusnih na ove napade. U [123] definisani su potrebni i dovoljni uslovi za detekciju višestrukih napada obmanom gde se dostupni događaji zamenjuju nizom kompromitovanih događaja. Ovaj pristup

zasnovan je na SCT i zahteva od supervizora da izabere i primeni odgovarajući jezik (skup ograničenja i dozvoljenih ponašanja) na osnovu merenja senzora koja su kompromitovana od strane jednog (ne zna se kog) od nekoliko potencijalnih napadača. Ograničenje pristupa iz [110, 123] ogleda se u primeni samo jednog robusnog supervizora koji je namenjen određenom napadu što utiče na efikasnost detekcije u slučajevima kada se pojavljuju višestruki napadi. Kako bi se ovo ograničenje prevazišlo, u okviru [71] predložen je pristup zasnovan na teoriji igara i konačnim automatima kojim su spojeni različiti supervizori (uključujući i supervizore iz [110, 123]) kreirani za detekciju napada na senzore. Na taj način se izdvajanjem odgovarajućih supervizora mogu detektovati različiti napadi.

Prikriveni napadi i napadi ponovne reprodukcije u DES koji imaju pristup svim senzorima i aktuatorima modelirani su primenom SCT u [33]. Takođe, u ovom radu predložen je metod za detekciju zasnovan na permutaciji ulaza i izlaza kontrolera na strani postrojenja i supervizora, kao i poređenju ostvarenog i očekivanog ponašanja sistema. U [32] razmatrani su napadi koji u potpunosti preuzimaju kontrolu nad postrojenjem za određeni vremenski period, dok su u [35, 135] predložene metode za projektovanje prikrivenih napada u takvim sistemima.

Analizom navedenih istraživanja sa aspekta arhitekture upravljačkog sistema, može se primetiti da u pristupima poput [123, 126] supervizor izvršava centralizovano upravljanje. Za razliku od ovih pristupa, u [52] koristeći SCT formalizam modeliraju se napadi ubrizgavanjem i/ili uklanjanjem podataka na komunikacione linije između lokalnih kontrolera u okviru distribuiranog sistema upravljanja. U ovom istraživanju predstavljen je metod za detekciju modeliranih napada, kao i pristup za identifikaciju komunikacionih veza na koje je neophodno implementirati mehanizme za detekciju kako bi se izbegle eventualne posledice na sistem.

Pored metoda na bazi konačnih automata i SCT [71, 110, 123], razvijeni su i pristupi koji za modeliranje postrojenja i/ili supervizora koriste Petrijeve mreže. Na primer, istraživanje prikazano u [134] bavi se senzorskim napadima na postrojenje koje je modelirano kao ograničena Petrijeva mreža, uzimajući u obzir određena dinamička svojstva mreže. U ovom slučaju supervizor se projektuje tako da ne može dostići zabranjena stanja, odnosno obezbeđuje se da Petrijeva mreža bude "*živa*" i pod dejstvom napada. Takođe, u [127] predloženi su različiti supervizori bazirani na Petrijevim mrežama za detekciju napada na senzore i aktuatore koji su predstavljeni u [10].

Pored značajnih rezultata koji su ostvareni u projektovanju IDS zasnovanih na dizajnu za sisteme sa diskretnim događajima, ovaj tip IDS razmatran je i u slučaju kontinualnih sistema upravljanja [97, 115]. Ipak, primena ovih rešenja ograničena je zbog kompleksnosti procesa, kao i zbog grubih pretpostavki koje moraju biti uzete u obzir kako bi bilo moguće kreirati odgovarajući analitički model.

Ograničenja metoda zasnovanih na dizajnu diskutovana su u [120] kroz uporednu analizu ove klase metoda i metoda zasnovanih na podacima u postupku generisanja invarijanti posmatranog sistema. U ovom kontekstu, invarijante predstavljaju pravila/uslove na osnovu kojih se vrši detekcija napada. Između ostalog, u okviru ove analize ističe se neophodnost primene tehnika baziranih na podacima bar u delovima sistema sa kontinualnom prirodom, kao i potreba za automatizacijom procesa kreiranja čak i jednostavnih invarijanti kada se radi od sistemima koji sadrže relativno veliki broj senzora i aktuatora.

3.2.2. Metode za detekciju anomalija zasnovane na podacima

Kod pristupa zasnovanih na podacima ponašanje sistema se modelira uzimajući u obzir podatke prikupljene tokom njegovog rada. Kao i kod metoda zasnovanih na dizajnu, cilj svake od tehnika je da se na osnovu odstupanja između modeliranih i primljenih podataka detektuje napad neposredno nakon njegove pojave. Poslednjih godina pristupi zasnovani na podacima bili su predmet brojnih istraživanja u oblasti sajber bezbednosti gde su se kao pogodan izbor za razvoj IDS nametnule tehnike mašinskog učenja. Tehnike razvoja IDS zasnovane na mašinskom

učenju mogu se svrstati u tri osnovne kategorije u zavisnosti od podataka koji se koriste prilikom generisanja modela: 1) nadgledano učenje (engl. *supervised learning*), 2) nenadgledano učenje (engl. *unsupervised learning*) i 3) samonadgledano učenje (engl. *self-supervised learning*) [28, 75].

Kod **nadgledanog učenja** model se generiše na osnovu podataka snimljenih u normalnom režimu rada i u toku napada, pri čemu je za svaki podatak poznato da li je snimljen tokom normalnog rada sistema ili tokom napada, kao i koja vrsta napada je primenjena tokom prikupljanja konkretnog podatka. Ovaj pristup ima dva značajna ograničenja od kojih se prvo odnosi na nedostatak adekvatnih skupova podataka sa označenim napadima, dok se drugo tiče sposobnosti generalizacije kreiranog modela koji kasnije uglavnom nije u mogućnosti da detektuje napade na kojima nije bio obučavan [75].

Kod **nenadgledanog učenja** model se takođe kreira na osnovu podataka snimljenih u normalnom režimu rada i u toku napada, ali podaci nisu označeni i nije poznato da li su prikupljeni tokom normalnog rada sistema ili ne. Kod ovog pristupa algoritam samostalno utvrđuje zakonitosti koje se nalaze u podacima i procesom klasterovanja (klasifikacije bez nadzora) zaključuje da li se radi o normalnom ponašanju sistema ili anomaliji.

Značajno ograničenje koje karakteriše navedena dva pristupa učenja jeste to što je često nemoguće prikupiti dovoljnu količinu podataka iz realnog sistema pod dejstvom napada. Rešenje ovog problema postignuto je kroz **samonadgledano** učenje koje podrazumeva obučavanje modela isključivo koristeći podatke prikupljene tokom normalnog režima rada sistema (bez napada). Generisani model opisuje normalno funkcionisanje sistema, a detekcija napada se izvodi poređenjem modelirane i ostvarene vrednosti. Model kreiran na ovaj način nije predodređen za detekciju samo nekih vrsta napada pa su njegove sposobnosti generalizacije po pravilu bolje u poređenju sa modelima kreiranim po principu nadgledanog ili nenadgledanog učenja. Pristup samonadgledanog učenja je najzastupljeniji u zadacima detekcije napada, delom zbog toga što je podatke u normalnom režimu rada sistema u opštem slučaju lakše prikupiti.

Tehnike mašinskog učenja dale su značajan doprinos u kreiranju mehanizama za detekciju napada. Međutim, ograničavajući faktor za širu primenu ove klase IDS-a predstavlja nedovoljan broj javno dostupnih skupova podataka na osnovu kojih je moguće kreirati mehanizme za detekciju i testirati njihove performanse. Iz navedenih razloga se u sledećem odeljku vrši pregled javno dostupnih skupova podataka za kreiranje IDS-a u okviru ICS.

3.2.2.1. Skupovi podataka kreirani za razvoj metoda za detekciju napada

Idealan slučaj razvoja IDS-a zasnovanih na podacima predstavljao bi scenario gde bi se podaci u neograničenim količinama prikupljali direktno iz realnih ICS, a nakon toga i testirali u okviru ovih sistema. Međutim, kako realni sistemi često izvršavaju bitne operacije poput proizvodnje električne energije, prečišćavanja vode itd, navedeni pristup je teško izvodljiv iz više razloga. Na primer, implementacija napada na ICS, iako u istraživačke svrhe, može dovesti do prekida procesa proizvodnje ili oštećenja uređaja što dalje može prouzrokovati značajne ekonomske posledice. Pored toga, zaštita i poverljivost podataka igraju bitnu ulogu pa realni podaci mogu dovesti na određeni način do krađe intelektualne svojine, kao i prikazivanja osetljivosti sistema na pojedine napade što kasnije može biti iskorišćeno prilikom kreiranja napada na taj sistem [19].

Postavke za testiranje se prema funkcionalnim elementima od kojih se sastoje mogu svrstati u grupe fizičkih, virtuelnih i hibridnih [19]. Grupu fizičkih postavki predstavljaju eksperimentalne instalacije koje u cilju konfigurisanja mrežnih i fizičkih slojeva koriste prava hardverska i softverska rešenja. Na ovaj način mogu se dobiti realni podaci, uključujući i kašnjenja koja su inherentno prisutna u industrijskim sistemima. Pored toga, moguće je posmatrati i uticaj različitih smetnji i napada na pojedinačne uređaje. Međutim, razvoj fizičkih instalacija nekada iziskuje mnogo vremena i sredstava, a u primerima kritičnih procesa (npr. nuklearnih postrojenja) otvaraju se pitanja bezbednosti njihove realizacije i upotrebe. U navedenim razlozima

ogledaju se prednosti primene virtuelnih postavki, često koristeći koncept simulacije sa hardverom u petlji (engl. *Hardware In the Loop* – HIL) gde se fizički procesi simuliraju najčešće putem namenskih softvera [74]. Pored toga, ovakav pristup obezbeđuje veću fleksibilnost i mogućnost brze izmene postavke i parametara sistema. Ipak, postavlja se pitanje validnosti podataka koji su generisani korišćenjem simulacije, posebno kada se radi o kompleksnim procesima koje nije lako simulirati. Kao kompromisno rešenje između fizičkih i virtuelnih postavki javlja se hibridni pristup u kojem se rad pojedinih uređaja simulira, dok se ostali uređaji fizički implementiraju. Na ovaj način mogu se izbeći fizički oblici uređaja čija nabavka zahteva velika sredstva, odnosno uređaja čiji rad može izazvati određene posledice po sistem, okruženje i/ili čoveka. Kako hibridne postavke sadrže i fizičke i virtuelne komponente, neretko dolazi do preklapanja između dva pristupa. Na primer, zamenom najmanje jednog uređaja u virtuelnom obliku njegovim fizičkim modelom, virtuelni model postaje hibridni [130].

Shodno navedenim prednostima i nedostacima različitih tipova ispitnih postavki, predložena su četiri ključna zahteva[2] koje postavka treba da ispunjava [130]. Prvi zahtev tiče se validnosti podataka, odnosno poklapanja sa podacima dobijenim iz realnog ICS-a. Ponovljivost, kao druga karakteristika, u ovom kontekstu posmatra se kroz mogućnost da se performanse različitih mehanizama za detekciju napada porede pod istim eksperimentalnim uslovima; jasno je da je ispunjenje ove karakteristike lakše postići u slučaju virtuelnih postavki. Treću karakteristiku predstavlja tačnost merenja, pri čemu je neophodno voditi računa da implementacija sistema za akviziciju ne ugrozi funkcionalnost sistema. Pored toga, posebna pažnja usmerena je ka odabiru pozicije senzora i njihovoj međusobnoj sinhronizaciji. Poslednji zahtev koji se postavlja pred postavke za testiranje jeste bezbednost izvođenja procesa koja može biti ugrožena prilikom testiranja napada na sistem, pogotovo kada se radi o kritičnim procesima koji uključuju npr. hemijske reakcije.

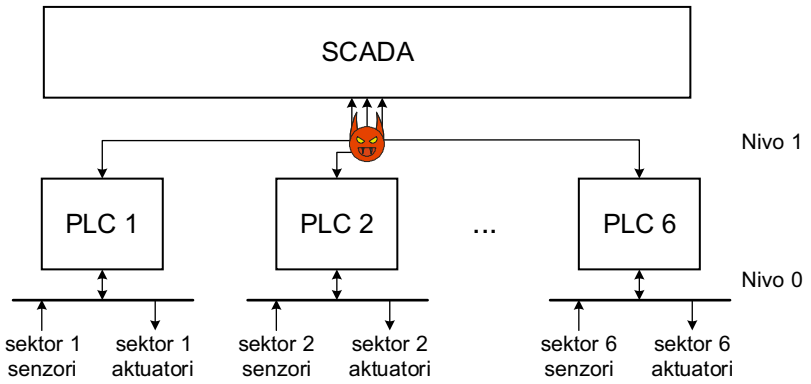
Ispitne postavke korišćene za kreiranje javno dostupnih skupova podataka u manjoj ili većoj meri zadovoljavaju navedene zahteve. U nastavku su ukratko predstavljene četiri ispitne postavke sa fokusom na skupove podataka koji su njihovom upotrebom generisani. Odabir skupova podataka izvršen je na osnovu učestalosti njihove primene u istraživanjima vezanim za razvoj metoda za detekciju napada.

SWaT skup podataka

SWaT (engl. *Secure Water Treatment*) postrojenje predstavlja skaliranu fabriku za preradu vode koja ima mogućnost da proizvede 5 galona prečišćene vode u minuti [36]. Ovo postrojenje sastoji se od šest linijski povezanih sektora (P1-P6) gde celokupan proces počinje skladištenjem sirove vode u rezervoaru (P1) koja u nastavku prolazi kroz postupak inicijalne provere (P2) gde se u zavisnosti od kvaliteta vode dodaju određena hemijska sredstva. Sektori P3 i P4 namenjeni su uklanjanju nepoželjnih materijala i dehlorisanju vode. U sektoru P5 unapređuje se kvalitet vode kroz uklanjanje neorganskih nečistoća, dok se u poslednjem sektoru voda skladišti i prosleđuje za dalju distribuciju.

Svakom sektoru dodeljen je PLC koji je povezan sa SCADA sistemom preko nivoa 1 (slika 6). U okviru sistema na odgovarajuće PLC-ove povezano je ukupno 25 senzora i 26 aktuatora (nivo 0 - slika 6). Senzori su podeljeni u četiri različite klase u zavisnosti od toga da li mere protok, nivo tečnosti, pritisak ili koncentraciju hemijskih sredstava. S druge strane, aktuatorski signali pokreću pumpe ili ventile.

[2]Navedeni zahtevi nisu sortirani prema prioritetu važnosti.

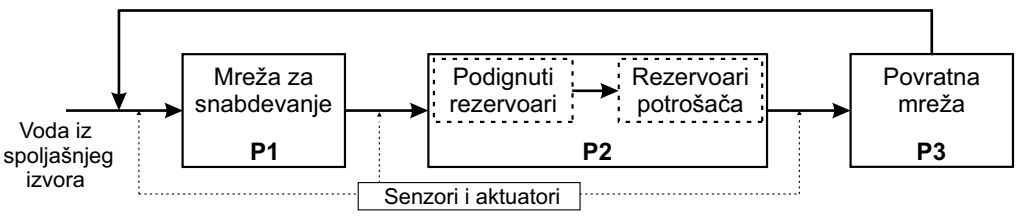


Slika 6: Arhitektura upravljačkog sistema SWaT postrojenja sa naznačenim tačkama napada – prilagođeno iz [36]

Akvizicija podataka iz sistema trajala je 11 dana, pri čemu je prvih 7 dana sistem funkcionisao u normalnim uslovima, a tokom preostala 4 dana bio je izložen različitim napadima. Svi podaci sa senzora i aktuatora odabirani su frekvencijom od 1 Hz. Pre početka prikupljanja podataka sistem je potpuno ispražnjen, nakon čega je bilo potrebno 6 sati za stabilizaciju nivoa vode u različitim rezervoarima. Tokom poslednja četiri dana pokrenuto je ukupno 36 napada na komunikacione linije između PLC-ova i SCADA sistema, koji su napadali istovremeno jedan ili više uređaja (senzora/aktuatora) i/ili sektora. Napadi su imali različito trajanje i različite posledice na dinamiku sistema, zahtevajući različito vreme za stabilizaciju sistema. U okviru ovog skupa podataka svaki signal senzora/aktuatora u normalnim uslovima rada i pod dejstvom napada sadrži 496.800 i 449.919 odbiraka, tim redom.

WADI skup podataka

WADI (engl. *Water Distribution testbed*) predstavlja skaliranu verziju postrojenja za distribuciju vode sa maksimalnim protokom od 10 galona (oko 38 litara) filtrirane vode u minuti [2]. Ovo postrojenje projektovano je u okviru iste institucije kao i SWaT postrojenje i predstavlja nastavak istraživanja u oblasti sajber bezbednosti. Između ostalog, jedan od ciljeva ovog postrojenja jeste utvrđivanje efekata i posledica koje napadi na jedan sistem (SWaT) prouzrokuju na drugom sistemu (WADI). Treba napomenuti da su ova dva sistema međusobno povezana. Funkcionalnost WADI postrojenja ostvaruje se kroz tri linijski povezana sektora (P1-P3) kojima su dodeljeni sopstveni skupovi PLC-ova (slika 7). Sektor P1 sadrži dva rezervoara za sirovu vodu kao i senzore za praćenje nivoa vode. Voda na ulazu u P1 dobija se iz SWaT postrojenja, gradske mreže za snabdevanje vodom ili povratnom spregom iz sektora P3. Za održavanje odgovarajućeg kvaliteta vode instaliran je sistem za doziranje hemikalija kao i senzori za merenje parametara kvaliteta vode. P2 se sastoji iz dva podignuta rezervoara i šest rezervoara potrošača. Sirova voda dovodi se u podignute rezervoare koji na osnovu unapred postavljenog režima potražnje snabdevaju vodom rezervoare potrošača. Kada rezervoari potrošača ispune svoje potrebe, voda se odvodi u sektor povratne mreže (P3) koja je takođe opremljena rezervoarom. Ako se izuzme razlika u broju sektora, arhitektura WADI upravljačkog sistema ista je kao u slučaju SWaT postrojenja (slika 6).

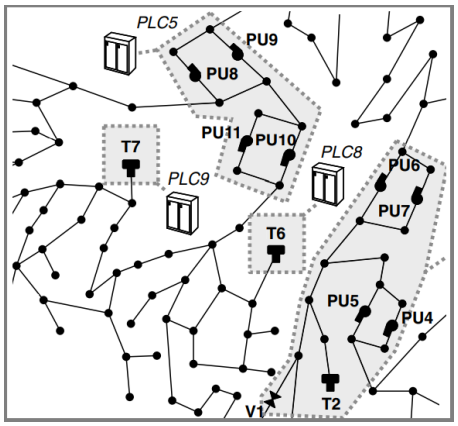


Slika 7: WADI proces distribucije vode – prilagođeno iz [2]

Komunikacija između senzora, aktuatora i PLC-ova ostvaruje se putem Eternet protokola ili GPRS (engl. *General Packet Radio Services*) veza; odabir žičane ili bežične komunikacije vrši se promenom stanja ručnog prekidača. Po svom originalnom dizajnu WADI ne raspolaže nikakvim mehanizmima sajber bezbednosti. Postrojenje sadrži ukupno 123 uređaja (senzora i aktuatora) na kojima je sprovedena akvizicija podataka. Prikupljene podatke sa 33 uređaja čine binarne vrednosti, dok su podaci sa ostalih uređaja realni. Prikupljanje podataka trajalo je ukupno 16 dana, od čega je prvih 14 dana sistem radio u ustaljenom režimu rada (bez napada) dok je tokom preostala 2 dana lansirano ukupno 15 napada (minimalno i maksimalno trajanje napada je 1,27 i 29 minuta). Na kraju, akvizicija podataka rezultirala je sa ukupno 1.221.372 odbiraka.

BATADAL skup podataka

BATADAL (engl. *Battle of Attack Detection Algorithms*) skup podataka prvobitno je nastao za takmičenje [113] u okviru koga su poređene performanse tehnika za detekciju napada. Ovaj skup podataka generisan je simulacijom[3] procesa *C-Town* koji predstavlja mrežu za distribuciju vode srednje veličine i sadrži 429 cevi, 388 spojeva, 7 rezervoara za skladištenje vode, 11 pumpi (raspoređenih na 5 stanica), 5 ventila i jedan rezervoar. Upravljačka logika ostvaruje se korišćenjem 9 PLC-ova koji na osnovu informacija sa senzora upravljaju radom pumpi i ventila (slika 8). PU, T i V na slici 8 označavaju pumpe, rezervoare i ventile, tim redom. SCADA sistem se koristi za koordinaciju i nadgledanje celog procesa kao i za prikupljanje informacija sa PLC-ova.



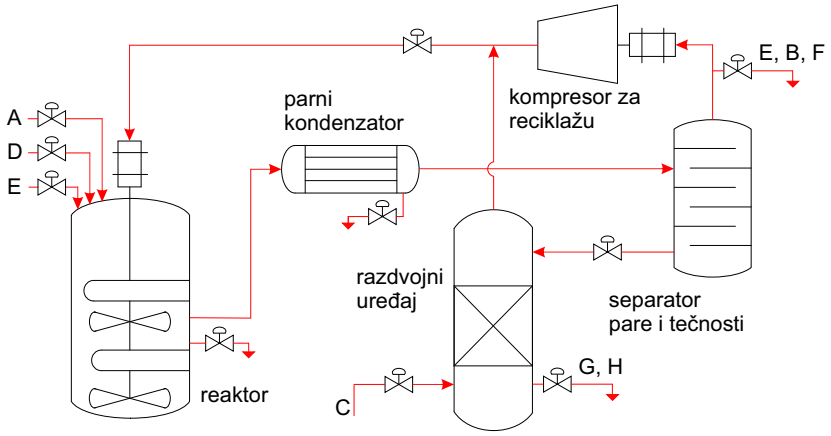
Slika 8: Deo procesa *C-Town* – prilagođeno iz [113]

Ceo skup podataka podeljen je na tri dela gde prvi deo sadrži podatke koji predstavljaju rezultate simulacije rada sistema u normalnim uslovima tokom 365 dana, dok drugi i treći deo (snimljeni tokom 6, odnosno 3 meseca) sadrže 14 napada (svaki po 7 napada) različitog oblika, intenziteta i trajanja. U svakom od tri navedena dela na 60 minuta vrše se očitavanja vrednosti sa 43 senzora koji mere nivo vode u rezervoaru (7 signala), protok (24 signala), kao i ulazni i izlazni pritisak za jedan aktivirani ventil i sve stanice (12 signala) što je rezultiralo sa ukupno 12.938 odbiraka. Svi signali sem stanja ventila i pumpi (binarni signali) sadrže realne vrednosti. Kreirani napadi obmanom imali su direktan uticaj na senzore i aktuatore. Konkretno, napadi ponovne reprodukcije menjali su podatke za dati sat podacima koji su snimljeni tokom istog sata prethodnog dana. Trajanje napada variralo je u opsegu od 24 do 110 časova (isto toliko odbiraka). Treba napomenuti da je prilikom takmičenja BATADAL učesnicima dato vreme početka i kraja samo jednog napada, dok su ove informacije za ostalih 13 napada bile nepoznate.

[3]Simulacija je kreirana u softverskom paketu *Matlab* korišćenjem biblioteke *epanetCPA*.

Tennessee Eastman skup podataka

Tennessee Eastman skup podataka [69] kreiran je simulacijom stvarnog hemijskog proizvodnog procesa koji je opisan u [70]. Kako obuhvata mnoge tipične pojave iz stvarnih hemijskih procesa koje se mogu opisati pomoću prikupljenih podataka, ovaj sistem se često koristi za razvoj i testiranje različitih tehnika sajber bezbednosti. Sistem se sastoji od pet glavnih delova: reaktora, parnog kondenzatora, separatora pare i tečnosti, kompresora za reciklažu i razdvojnog uređaja, kao i od osam dodatnih komponenti koji doprinose izvršavanju hemijskog procesa (slika 9). Slovne oznake (A, B, C...) predstavljaju ulazne/izlazne hemijske supstance.



Slika 9: Uprošćeni prikaz *Tennessee Eastman* procesa

Nezavisni skupovi simuliranih podataka generisani su tokom izvršavanja procesa u normalnim uslovima (bez napada) i pod dejstvom ukupno 21 napada, gde je u jednom trenutku bio aktivan samo jedan napad. Pritom, scenariji napada su podrazumevali ubrizgavanje lažnih podataka i uvođenje šuma (napadi obmanom), kao i uskraćivanje pristupa servisu (DoS napadi). Odabiranje vrednosti vršeno je na svaka 3 minuta. Svaki od kreiranih skupova (sa i bez napada) dalje je podeljen na skupove za obučavanje i testiranje. U skupovima za obučavanje, skup podataka koji karakteriše normalne uslove rada sadrži 500 odbiraka, dok svaki skup koji opisuje sistem pod dejstvom napada sadrži 480 odbiraka. Oba skupa podataka za testiranje sastoje se od 960 odbiraka. Razmatrane su ukupno 52 promenljive koje opisuju rad 41 senzora i 11 aktuatora. Prikupljeni podaci sa svih uređaja predstavljeni su realnim vrednostima.

3.2.2.2. Analiza postojećih metoda za detekciju napada zasnovanih na podacima

Korišćenjem prethodno razmatranih skupova podataka kreirani su sistemi za detekciju napada zasnovani na podacima koji će biti analizirani u nastavku. Sveobuhvatnost SWaT skupa podataka i činjenica da su podaci prikupljeni iz skalirane verzije realnog postrojenja doprinele su tome da je ovaj skup podataka najčešće korišćen za testiranje performansi sistema za detekciju napada. Treba naglasiti da je pojava SWaT skupa podataka dala značajan zamah ovoj oblasti i ubrzala razvoj mehanizama za detekciju napada.

Kao što je već navedeno, tehnike za razvoj IDS-a zasnovane na mašinskom učenju mogu biti nadgledane, nenadgledane i samonadgledane. U nastavku ovog odeljka biće razmatrane postojeće metode za detekciju napada koje su zasnovane na ovim pristupima.

Tehnike zasnovane na nadgledanom učenju

Iako je efikasnost IDS-a kreiranih na bazi nadgledanog učenja upitna kada je reč o napadima koji nisu korišćeni prilikom obučavanja, u određenim istraživanjima upotrebljen je ovaj pristup generisanja IDS-a. U [99] predstavljen je metod zasnovan na klasifikaciji konvolucionim neuronskim mrežama. Redukcija broja obeležja na osnovu kojih se vrši klasifikacija izvršena je

primenom analize glavnih komponenti (engl. *Principal Component Analysis* – PCA) i hipergrafova. Naime, implementacijom PCA pronađene su i zanemarene nulte sopstvene vrednosti, nakon čega je primenom hipergrafova izvršena identifikacija značajnih obeležja i uklanjanje redudantnih vrednosti. Na ovaj način značajno je smanjeno vreme potrebno za kreiranje modela. Za evaluaciju performansi predloženog metoda korišćeni su signali sa senzora i aktuatora iz SWaT skupa podataka.

Sistem za detekciju napada predložen u [1] sastoji se od četiri osnovna koraka i kombinuje više različitih tehnika. Cilj prva dva koraka jeste da se primenom definisanih pravila i statističkih alata pojedinačno prate merenja svakog senzora i detektuju značajna odstupanja u poređenju sa uobičajenim vrednostima. U trećem koraku primenom potpuno povezanih neuronskih mreža modelira se svaki signal posebno, a detekcija napada se izvodi na osnovu razlike modelirane i stvarne vrednosti. Prag detekcije u ovom slučaju definisan je kao proizvod najveće greške dobijene u procesu obučavanja i ručno postavljenog parametra. U poslednjem koraku korišćenjem PCA izdvajaju se obeležja i klasifikuju na osnovu varijanse na klase normalnog ponašanja i napada. BATADAL skup podataka korišćen je u cilju evaluacije predložene metode.

Hibridni metod prikazan u [102] koristi kombinaciju konvolucionih neuronskih mreža (engl. *Convolutional Neural Networks* – CNN) i rekurentnih neuronskih mreža (engl. *Recurrent Neural Networks* – RNN), konkretno neuronske mreže sa dugom kratkoročnom memorijom (engl. *Long Short-Term Memory* – LSTM). U ovom slučaju korišćenjem CNN-a vrši se klasifikacija karakteristika ekstrahovanih pomoću vejvlet transfromacije za detekciju tzv. neskrivenih (engl. *non-stealthy*) napada, dok se LSTM koristi za detekciju skrivenih (engl. *stealthy*) napada. Nakon primene vejvlet transformacije svakom odbirku ručno je dodeljen odgovarajući izlaz u zavisnosti od toga da li se radi o normalnom ponašanju sistema ili napadu.

Sistem za detekciju napada prikazan u [20] zasnovan je na klasifikaciji podataka sa senzora primenom njihove logičke analize. Na osnovu izdvojenih karakteristika može se izvršiti klasifikacija na stabilno i nestabilno stanje što označava rad sistema u normalnim uslovima i pod dejstvom napada, tim redom. Pored trenutnog dejstva napada, klasifikacija nestabilnog stanja u ovom slučaju može biti izazvana i zaostalim uticajem napada (koji se prethodno desio) na sistem.

U cilju detekcije kibernetskih napada, metod nadgledanog učenja predložen u [3] primenjuje tehnike složenih autoenkodera (engl. *Stacked Autoencoders* – SAE), dubokih neuronskih mreža (engl. *Deep Neural Networks* – DNN), kao i klasifikatore na bazi stabla odlučivanja (engl. *Decision Tree*). Pored testiranja performansi detekcije napada na signalima iz SWaT skupa podataka, robusnost predloženog pristupa ispitana je kroz mogućnost detekcije napada koristeći model kreiran na osnovu neuravnoteženih skupova podataka gde je količina podataka u normalnom radu značajno veća od količine podataka usled dejstva napada.

Tehnike zasnovane na samonadgledanom učenju

ICS u normalnim uslovima rada pruža mogućnost prikupljanja velike količine podataka pa su IDS na bazi samonadgledanog učenja u istraživanjima privukle značajnu pažnju. Kako veliki broj metoda zasnovanih na samonadgledanom učenju koristi autoregresiju za kreiranje modela ponašanja sistema u normalnim uslovima, pre pregleda tehnika zasnovanih na samonadgledanom učenju pogodno je bliže pojasniti autoregresiju. Naime, autoregresija predstavlja tehniku kojom se predviđaju trenutne vrednosti podataka na osnovu njihovih prethodnih vrednosti. Model kreiran na taj način definisan je koeficijentima koji opisuju korelaciju između trenutnih i prethodnih vrednosti. Na primer, kod autoregresionog modela reda v trenutna vrednost modelira se kao funkcija v prethodnih vrednosti.

U zavisnosti od podataka koje koriste za kreiranje modela autoregresione metode mogu biti univarijatne i multivarijatne. Univarijatne metode modeliraju ponašanje jedne promenljive (npr. senzorskog signala), dok se multivarijatnim pristupom kreira jedan model koji opisuje ko-

relacije između više promenljivih (npr. više senzorskih i/ili aktuatorskih signala). Kod metoda zasnovanih na samonadgledanom učenju vrednost signala koja je procenjena na osnovu dobijenog modela, poredi se sa vrednošću dobijenom komunikacionom linijom i napad se detektuje ukoliko razlika pređe odgovarajući prag.

S obzirom na to da su signali koji se dobijaju sa senzora, odnosno koji se šalju ka aktuatoru vremenske sekvence, LSTM predstavlja jednu od prvih i često primenjenih arhitektura za modeliranje ponašanja sistema u normalnim uslovima [30, 37, 46, 98]. U [37] predložen je multivarijatni pristup koji pored detekcije napada ima mogućnost identifikacije napadnutog uređaja. U okviru ovog istraživanja za kreiranje modela i testiranje performansi detekcije napada u razmatranje su uključeni signali samo iz prvog sektora SWaT postrojenja koji obuhvata ukupno 5 uređaja (2 senzora i 3 aktuatora). Prag detekcije sadrži donju i gornju graničnu vrednost koje su definisane na osnovu kumulativne sume razlike između predikcije i stvarne vrednosti uključujući i jedan hiperparametar koji se ručno podešava. U multivarijatnom pristupu iz [46] korišćeni su svi signali iz SWaT skupa podataka gde je pored LSTM modela razmatran i pristup zasnovan na mašinama sa nosećim vektorima (engl. *Support Vector Machines* – SVM). Prag detekcije u ovom slučaju definisan je po principu pokušaja i greške gde se vrednost praga menja u zavisnosti od postignutih rezultata detekcije.

Još jedan multivarijatni pristup koji za modeliranje normalnog ponašanja koristi LSTM mreže predstavljen je u [30]. Najbolja arhitektura modela izabrana je na osnovu rezultata detekcije sprovedene na *Tennessee Eastman* skupu podataka. Za razliku od izbora arhitekture modela gde su korišćeni i podaci sa napadima, prag detekcije izračunat je samo na osnovu podataka prikupljenih tokom normalnog ponašanja sistema.

Efikasnost jednodimenzionalnih (1D) konvolucionih neuronskih mreža u zadacima detekcije kibernetskih napada prikazana je u [59] gde je za modeliranje ponašanja sistema korišćen multivarijatni pristup. Jedan deo ovog istraživanja bio je usmeren i na primenu LSTM neuronskih mreža, kao i na kombinaciju LSTM i CNN arhitektura. Performanse detekcije ovim pristupom ispitane su na SWaT skupu podataka gde je prag detekcije određen na osnovu signala sa napadima.

Nastavak istraživanja iz [59] prikazan je u [60] gde je pored CNN-a primenjena i tehnika nepotpunih autoenkodera (engl. *Undercomplete Autoencoders* – UAE). U procesu generisanja modela, osim signala senzora i aktuatora u vremenskom domenu, na ulazu su korišćene i njihove karakteristike iz frekventnog domena. U fazi pretprocesiranja podataka korišćena je PCA tehnika kako bi se izdvojile odgovarajuće karakteristike iz podataka i na taj način obezbedila bolja predikcija modela, dok je vrednost praga detekcije definisana na isti način kao u [59]. Prilikom evaluacije performansi iz SWaT skupa podataka izostavljeno je ukupno 15 senzora zbog značajne razlike između podataka u normalnom radu sistema i podataka prikupljenih pod dejstvom napada. Pored SWaT skupa podataka razmatrani su i BATADAL i WADI skupovi.

Opšta metodologija za detekciju napada koja obuhvata pet osnovnih tačaka predložena je u [98] gde je kao primer za projektovanje modela normalnog ponašanja izabran LSTM. Iz SWaT skupa podataka koji je korišćen za testiranje performansi izostavljeno je ukupno 15 signala iz istog razloga kao u [60]. Takođe, u okviru ovog istraživanja, prag detekcije je odabran istim postupkom kao u [59].

Kompozitni oblik autoenkodera korišćen je za detekciju kibernetskih napada u [124]. Za razliku od uobičajenog načina primene autoenkodera gde se predikcija i rekonstrukcija ulaznog signala izvode odvojeno korišćenjem enkodera i dekodera, u ovom pristupu se navedeni procesi izvode korišćenjem celog autoenkodera. Za model autoenkodera izabrana je LSTM arhitektura, dok je vrednost praga detekcije određena maksimalnom razlikom između stvarne i estimirane vrednosti na skupu podataka za obučavanje. Ukupno 45 aktuatorskih i senzorskih signala iz SWaT skupa podataka korišćeno je za evaluaciju performansi ovog pristupa.

Višeslojni perceptron primenjen je za projektovanje IDS-a u [100] gde su razmatrani samo

napadi koji su direktno uticali na prvi sektor SWaT skupa podataka. U okviru ovog istraživanja uveden je kriterijum za odabir najbolje arhitekture zasnovan na *Theil*-ovoj U-statističkoj meri koja se računa kao relativna razlika između stvarne vrednosti i predikcije. Svakom razmatranom uređaju dodeljen je odgovarajući model ponašanja gde je broj potpuno povezanih slojeva varirao u opsegu od 1 do 3. Proces odabira najbolje arhitekture optimizovan je algoritmom mrežne pretrage. Prag detekcije određen je po principu pokušaja i greške gde su gornja i donja granična vrednost izračunate na bazi kumulativne sume.

Izbor optimalne DNN arhitekture izvršen je i u [104] gde su u tu svrhu korišćeni genetički algoritmi. Kriterijum korišćen za izbor modela uključuje podatke snimljene u normalnim uslovima rada i pod dejstvom napada. Pritom, razmatrani su različiti tipovi neuronskih mreža poput višeslojnog perceptrona (engl. *multilayer perceptron* – MLP), CNN, RNN, gde je potpuno povezana neuronska mreža sa 4 sloja izabrana kao optimalna. Za detekciju napada na SWaT skupu podataka korišćen je prag detekcije zasnovan na srednjoj vrednosti razlike između stvarnih i estimiranih podataka na delu podataka namenjenih za obučavanje.

Potpuno povezana neuronska mreža dala je najbolje rezultate i u analizi sprovedenoj u [132], koja je između ostalog obuhvatila različite RNN i CNN arhitekture. Prilikom testiranja mehanizma za detekciju iz SWaT skupa podataka izostavljeni su signali sa 6 uređaja dok su razmatrani signali u zavisnosti od njihove prirode okarakterisani kao kontinualne ili kombinacija kontinualne i diskretne promenljive. Glavnu razliku između ove dve vrste promenljivih predstavlja funkcija cilja pa se prilikom obučavanja kontinualnih promenljivih koristi funkcija cilja srednje kvadratne greške, dok se na drugu vrstu promenljivih primenjuje kombinacija funkcija srednje kvadratne greške i binarne unakrsne entropije (engl. *binary cross-entropy*).

U [4] prikazan je univarijatni pristup za detekciju napada nazvan PASAD (engl. *Process-Aware Stealthy Attack Detection*) koji koristi analizu singularnog spektra. Ideja primene ove tehnike jeste razdvajanje signala na vremenske serije manje dužine i pronalaženje njihovih redukovanih reprezentacija dekompozicijom singularnih vrednosti. Vrednost praga detekcije napada određena je na osnovu podataka za validaciju (deo skupa podataka za obučavanje modela) i konstante čija je vrednost ručno definisana. Kako bi se testirale performanse razvijenog mehanizma za detekciju, u sklopu istraživanja kreirani su napadi različitog nivoa prikrivenosti. Pored toga, za eksperimentalnu evaluaciju korišćeni su i javno dostupni skupovi podataka, konkretno *Tennessee Eastman* i dva signala prikupljena iz SWaT postrojenja.

Generativne suparničke mreže (engl. *Generative Adversarial Networks* – GAN) pokazale su se kao efikasne u zadacima detekcije kibernetskih napada [64, 90]. U multivarijatnom pristupu predloženom u [64] generator i diskriminator projektovani su na bazi LSTM tipa neuronske mreže gde generator pokušava da kreira podatke koji odgovaraju raspodeli originalnih podataka i šalje ih diskriminatoru koji ima zadatak da utvrdi da li se radi o originalnim ili lažnim podacima. Na osnovu njihove interakcije definiše se vrednost praga detekcije koji se kasnije koristi za detekciju napada. S druge strane, u [90] predložen je bidirekcioni tip GAN-a koji kombinuje RNN i CNN. U ovom pristupu, uslov za detekciju napada zasniva se na statističkom parametru energetske distance koji koristi odstupanje između dve raspodele podataka (originalne i raspodele podataka koju kreira generator) tako što računa energiju potrebnu da se od jedne raspodele dođe do druge. Efikasnost pristupa iz [64, 90] testirana je na SWaT skupu podataka, dok je u [64] korišćen i WADI skup podataka.

Hibridne metode poput metode predstavljene u [27] kombinuju različite tehnike sa ciljem da nadomeste nedostatke pojedinačnih tehnika. Naime, u [27] predložen je hibridni metod koji za izdvajanje karakteristika iz podataka koristi CNN autoenkodere, dok je za detekciju napada izabrana tehnika izolovanih šuma (engl. *isolation forests*). Obučavanje modela izvodi se pojedinačno gde se prilikom kreiranja modela izolovanih šuma kao ulaz koriste izdvojene karakteristike dobijene modelom autoenkodera. Nastavak ovog istraživanja [28] doveo je do primene dvostrukih izolovanih šuma, gde je prvi model kreiran na osnovu sirovih podataka, dok

su za obučavanje drugog modela podaci pretprocesirani PCA tehnikom. U oba pristupa [27, 28] detekcija napada ostvaruje se poređenjem vrednosti predloženog skora (određen srednjom dužinom svih stabala) i ručno definisane vrednosti praga detekcije. Verifikacija predloženih mehanizama za detekciju napada sprovedena je na SWaT skupu podataka, dok su performanse pristupa iz [28] testirane i na WADI skupu podataka.

Metoda za detekciju napada koja kombinuje potpuno povezanu neuronsku mrežu i SVM predstavljena je u [7]. Naime, prilikom kreiranja modela ponašanja vrši se obučavanje neuronske mreže gde je izabrana funkcija cilja zasnovana na SVM. Performanse kreiranog modela utvrđene su na SWaT skupu podataka, pri čemu je detekcija napada direktno zavisila od ručno postavljene vrednosti praga detekcije.

Takođe, multivarijatni pristup predložen u [68] može se posmatrati kao kombinacija mašinskog učenja i modela zasnovanog na dizajnu. Naime, modeliranje ponašanja uređaja ostvaruje se primenom grafičke metode nazvane TABOR i vremenskih automata (engl. *time automata*), dok se za definisanje korelacija između signala različitih uređaja koriste Bajesove mreže. Istraživanje je sprovedeno na prvih 5 od 6 sektora u SWaT skupu podataka gde je pored detekcije zadatak mehanizma bio da lokalizuje napad, odnosno da pronađe uređaj koji je napadnut.

3.2.2.3. Diskusija analiziranih metoda

Analizom razmatranih sistema za detekciju napada može se primetiti da je za evaluaciju performansi SWaT skup podataka pronašao primenu u najvećem broju istraživanja (tabela 1). Jednu od glavnih prednosti SWaT i WADI skupova podataka predstavlja to što su podaci prikupljeni iz realne instalacije, dok su podaci u okviru BATADAL i *Tennessee Eastman* skupova generisani korišćenjem simulacije. Pored toga, potencijalni razlog za odabir SWaT skupa podataka u odnosu na WADI predstavlja sveobuhvatnost SWaT postrojenja, kao i zavisnost WADI od SWaT postrojenja.

Stoga, kako bi se performanse metodologije predložene u ovoj doktorskoj disertaciji uporedile sa što većim brojem postojećih istraživanja, SWaT skup podataka odabran je kao odgovarajući.

Tabela 1: Zastupljnenost skupova podataka u relevantnim istraživanjima

Skup podataka	Istraživanje
SWaT	[3],[4],[7],[20],[27],[28],[37],[46],[59],[60],[64], [68],[90],[98],[99],[100],[102],[104],[124],[132]
WADI	[28],[60],[64]
BATADAL	[1],[60]
Tennessee Eastman	[4],[30]

Izuzimajući istraživanja [37, 100] koja prilikom detekcije razmatraju samo napade čija se meta nalazi unutar prvog sektora, [59, 68] gde se svaki sektor posmatra zasebno i [4] gde su u razmatranje uključena samo dva signala, u većini istraživačkih radova navedenih u tabeli 1 kreiran je multivarijatni IDS koji detektuje anomalije koristeći istovremeno signale pod napadima svih senzora i aktuatora. Međutim, kada se posmatra arhitektura upravljačkog sistema sa slike 6, postavlja se pitanje mogućnosti praktične primene razvijenih IDS na realnoj eksperimentalnoj instalaciji jer nije jasno na kom uređaju bi IDS mogao biti implementiran. Naime, shodno opisanom upravljačkom sistemu (slika 6) jedino SCADA ima na raspolaganju informacije sa svih senzora i aktuatora. S druge strane, napadi na komunikacione linije pokrenuti su na nivou 1 [36], tako da pozicija dejstva napada čini da SCADA nema uvid u napade na signale koji su poslati ka aktuatorima preko odgovarajućih PLC-ova - ovi napadi se dešavaju na komunikaciji između SCADA-e i PLC-a i signali sa napadom su dostupni samo PLC-u i aktuatoru. Dalje, PLC ima na raspolaganju samo signale koji su na njega povezani, a ne i signale sa drugih

senzora i aktuatora. Imajući u vidu navedeno, u sistemu ne postoji uređaj koji ima uvid u sve senzorske i aktuatorske signale pod dejstvom napada tako da je nejasno na kom uređaju bi IDS zasnovan na multivarijatnom pristupu koji koristi sve senzorske i aktuatorske signale bio primenjen.

Potrebno je istaći da napadi na aktuatore imaju uticaj i na ostatak sistema, što dalje dovodi do promene senzorskih podataka kako u sektoru napadnutog aktuatora, tako i u drugim sektorima. Stoga, u okviru ove disertacije polazi se od pretpostavke da je detekciju napada moguće vršiti analizom samo senzorskih signala bez upotrebe signala sa aktuatora.

Primenom većine razmatranih metoda nadgledanog učenja [3, 20, 99] postignuti su bolji rezultati u poređenju sa metodama samonadgledanog učenja. Međutim, metode sa nadgledanim učenjem razvijene su korišćenjem malog broja scenarija napada i stoga ne pokrivaju mnoštvo mogućih napada koji mogu dovesti sistem u neželjeno stanje. Pored toga, ove metode su testirane na istim ili vrlo sličnim napadima koji su korišćeni u procesu razvoja samih metoda. Iz navedenih razloga stvaraju se određena predubeđenja koja kasnije mogu imati posledice u pogledu loših svojstava generalizacije, a samim tim i upitnim performansama prilikom detekcije napada koji nisu bili uključeni u proces generisanja modela.

Značajan nedostatak velikog broja razvijenih metoda predstavlja način određivanja vrednosti praga detekcije. Naime, u navedenim istraživanjima vrednost praga detekcije utvrđena je ručno (često po principu pokušaja i greške) koristeći poznate napade koji se nalaze u signalima pa se postavlja pitanje primenljivosti ovih metoda za detekciju napada koji nisu korišćeni u razmatranom skupu podataka. Na primer, u [4, 7, 46, 59, 60, 98] prag detekcije definisan je po principu pokušaja i greške tako da maksimizira performanse detekcije poznatih napada. S druge strane, u [37, 100] prag detekcije je definisan donjom i gornjom granicom koje uključuju kumulativnu sumu, kao i ručno podesive parametre, dok je u [27, 28] ova vrednost određena srednjom dužinom svih stabala.

3.3. Standardi iz oblasti sajber bezbednosti

Potpuna automatizacija i povezanost na svim nivoima piramide automatizacije, kao i određeni nivo autonomnosti industrijskih procesa postignuti implementacijom koncepta Industrija 4.0 doveli su do potrebe za uvođenjem standarda koji podržavaju i definišu osnovne aspekte industrijskih procesa uključujući IT, ICS, procese umrežavanja itd. Standardi u opštem slučaju eksplicitno deklarišu zahteve za ispunjavanje definisanih normi, obično u vidu minimalnih uslova koje je neophodno ispuniti. Pored toga, specifikacija propisana standardima za industrijske komponente i softvere nudi proizvođačima, kupcima i svim drugim uključenim stranama konzistentan jezik što pojednostavljuje procese kupovine, integracije sistema, održavanja itd. Tako se standardima iz oblasti sajber bezbednosti predlažu smernice za rešavanje pitanja poput bezbednosti informacija i bezbedne komunikacije na svim nivoima. Uzimajući u obzir kompleksnost i raznovrsnost sistema na koje se primenjuju standardi sajber bezbednosti, razvijen je širok spektar standarda iz ove oblasti. Neki od standarda koji su privukli najviše pažnje prikazani su u tabeli 2, gde su navedene i podoblasti koje su u datim standardima razmatrane.

Tabela 2: Standardi iz oblasti sajber bezbednosti

Naziv	**Podoblast**
IEC 62443 [48]	sajber bezbednost
ISO/IEC 27001 [49]	bezbednost informacija
NIST 800-53: rev 4-5 [77, 79]	bezbednost i upravljanje informacionim sistemima i organizacijom
NIST 800-82: rev 2 [78]	bezbednost ICS
IISF	bezbednost industrijskog interneta

Iako pojedini standardi obuhvataju veliki deo aspekata sajber bezbednosti, neretko se javlja potreba da se u jednom sistemu primeni više standarda što dovodi do pojma hibridnih standarda. Ovakav pristup služi kako bi se umanjili nedostaci primene samo jednog standarda, ali kompleksnost koja se pritom stvara može predstavljati ograničavajući faktor. Iz tog razloga, da bi se dobila funkcionalna celina, izabrani standardi treba da poseduju mogućnost usaglašavanja kako na delovima koji se odnose na poslovne, tako i na delovima koji se odnose na tehničke zadatke, a sve u cilju zadovoljavanja adekvatnog nivoa bezbednosti. Pre kombinovanja dve metode, potrebno je kroz komparativnu analizu ustanoviti prednosti i nedostatke koje sadrže razmatrani standardi i na osnovu tih zaključaka graditi hibridni model [40].

Kako su u fokusu ove disertacije pitanja vezana za sajber bezbednost u ICS, standardi koji najviše odgovaraju ovim aspektima su NIST 800-82: rev 2 i IEC 62443. Serija standarda IEC 62443 koncipirana je kroz 4 dela, gde se prvi deo bavi pojmovima, konceptima i modelima u opštem slučaju, dok se u drugom delu razmatraju pravila i procedure sajber bezbednosti na poslovnom nivou (nivoi 4 i 5 sa slike 1). Treći deo usmeren je na ICS kao celinu, dok četvrti deo razmatra njegove sastavne komponente. S druge strane, NIST 800-82: rev 2 je orijentisan direktno na ICS gde se između ostalog razmatra topologija ICS, definišu kritične tačke sistema, a navedeni su i potrebni koraci za procenu i upravljanje rizikom. Pored toga, ovim standardom obuhvaćeni su i drugi značajni izazovi u oblasti sajber bezbednosti poput anomalija i napada, a shodno tome predložene su i smernice za izbegavanje ili ublažavanje posledica koje napadi prouzrokuju.

4. Razvoj metodologije za kreiranje algoritama za detekciju kibernetskih napada

Analizom postojećih pristupa za detekciju napada u poglavlju 3 uočeno je više nedostataka koji mogu značajno uticati na performanse detekcije kibernetskih napada. Pojedini nedostaci ogledaju se u nemogućnosti kreiranja modela ponašanja za kompleksnije sisteme kada su u pitanju metode zasnovane na dizajnu, loše performanse prilikom detekcije napada na kojima nije vršeno obučavanje za metode zasnovane na nadgledanom učenju, kao i loša svojstva generalizacije metoda koja su izazvana načinom izbora pojedinih parametara poput praga detekcije ili kriterijumima za odabir odgovarajućeg modela. Stoga, neophodan je razvoj nove metodologije kako bi se rešili navedeni nedostaci i time povećao ukupan nivo bezbednosti ICS-a.

Metodologija za razvoj IDS-a razvijena u okviru ove disertacije i predstavljena u ovom poglavlju bazirana je na primeni tehnika mašinskog i dubokog učenja, koristi podatke prikupljene tokom normalnog rada sistema (bez napada) i spada u klasu metoda zasnovanih na samonadgledanom učenju. Za razliku od analiziranih pristupa, ova metodologija ne predlaže jedinstven model mašinskog/dubokog učenja koji će odgovarati ponašanju različitih sistema, već je kreirana da pronađe odgovarajuću arhitekturu sa relativno malim brojem obučavajućih parametara koja na adekvatan način modelira razmatrani signal. Ovde se kreće od pretpostavke da će manji broj obučavajućih parametara za rezultat imati manju računsku složenost modela i izazvati manje kašnjenje u onlajn detekciji napada. Zahtevi rada u realnom vremenu koji se postavljaju pred IDS direktno utiču na bezbednost sistema, gde pravovremene akcije uslovljene odlukama definisanih algoritmom detekcije predstavljaju ključni faktor u umanjivanju ili potpunom izbegavanju uticaja napada na sistem. Iz tog razloga, neophodno je optimizovati računsku složenost IDS-a kako bi se u što većoj meri redukovalo kašnjenje prouzrokovano njihovom primenom, imajući u vidu da su uređaji na koje se implementira IDS energetski i proračunski ograničeni. Konkretno, predložena metodologija sprovodi polumrežnu (engl. *semi-grid*) pretragu parametara koji definišu specifičnu arhitekturu sa ciljem da se pronađe model koji ima dobre performanse na podacima za obučavanje. Pored toga, na osnovu podataka za obučavanje, metodologija automatski određuje vrednost praga detekcije, što predstavlja jedan od najosetljivijih parametara prilikom kreiranja IDS-a baziranog na tehnikama mašinskog/dubokog učenja. Treba napomenuti da je metodologija kreirana da detektuje različite tipove napada na komunikacione veze koji su detaljnije opisani u poglavlju 2.

Prilikom projektovanja metodologije za detekciju napada vođeno je računa o arhitekturi upravljačkog sistema, kao i o mogućim tačkama napada kako bi implementacija razvijenog IDS-a bila izvodljiva na određenom uređaju u upravljačkoj mreži. Naime, u obzir se uzimaju informacije koje su dostupne uređaju na kojem se IDS implementira. Na primer, u slučaju sistema sličnog sistemu prezentovanom na slici 2a, IDS može biti implementiran na kontroleru i/ili na pametnom senzoru i aktuatoru. Ukoliko se napadi izvršavaju na komunikacionim linijama između PLC-a i aktuatora u primeru sa slike 2a, tada su podaci namenjeni aktuatoru u toku napada dostupni samo napadnutom aktuatoru (kontroler je jedino svestan podataka koje je poslao i nema informacije o eventualnim izmenama podataka). S druge strane, podaci sa senzora u toku napada dostupni su jedino kontroleru, a senzor ih nema na raspolaganju. Stoga, u ovom primeru, IDS implementiran na kontroler može biti baziran na podacima sa senzora, dok se podaci sa aktuatora mogu iskoristiti jedino za implementaciju IDS-a na samom aktuatoru. Uzimajući u obzir navedena ograničenja u vidu dostupnosti podataka, da bi implementacija IDS-a na kontroler bila moguća, u okviru ove doktorske disertacije predlaže se metodologija koja je zasnovana na primeni jednostruke (univarijatne) regresije gde se estimacija trenutne vrednosti signala (npr. senzorskih) izvodi na osnovu njegovih prethodnih vrednosti.

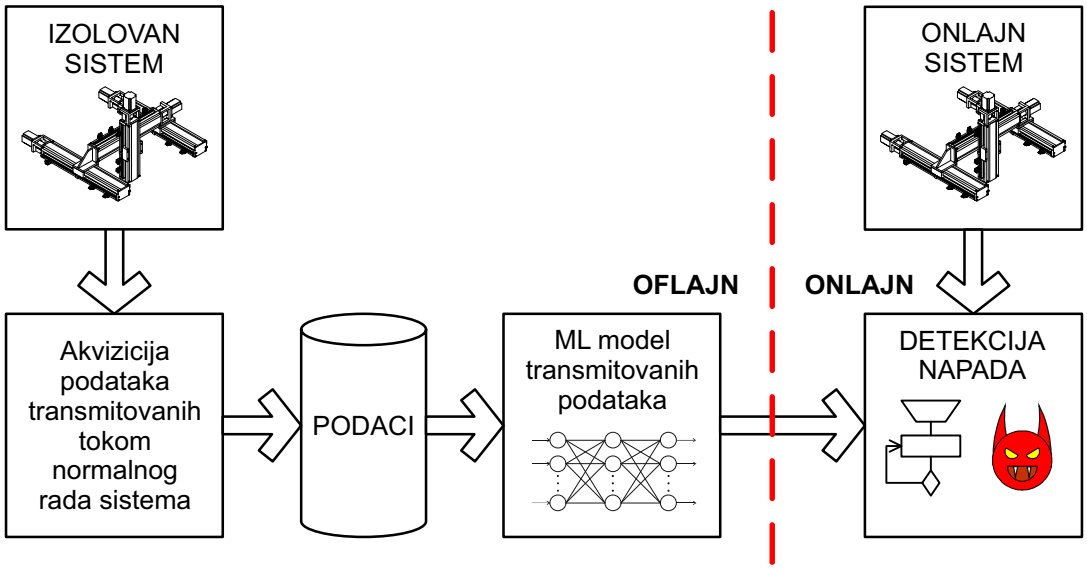
4.1. Osnovne faze metodologije za razvoj algoritama za detekciju kibernetskih napada

Predložena metodologija za kreiranje algoritama za detekciju kibernetskih napada zasnovana je na podacima i sastoji se od dve glavne faze [84]:

1. Oflajn faza – generisanje modela korišćenjem tehnika mašinskog učenja;
2. Onlajn detekcija kibernetskih napada.

Tokom prve faze kreira se model transmitovanih signala kroz sledeća tri koraka (slika 10):

1. Sistem radi u izolovanim uslovima, bez povezivanja na globalnu mrežu, čime se isključuje mogućnost za pojavu kibernetskih napada;
2. Tokom rada sistema u izolovanim uslovima vrši se akvizicija transmitovanih podataka;
3. Podaci prikupljeni tokom normalnog rada (bez pojave napada) koriste se za generisanje ML modela transmitovanih podataka.



Slika 10: Opšta postavka kreirane metodologije za razvoj algoritma za detekciju kibernetskih napada u ICS

Kao što je navedeno u poglavlju 1, današnji ICS ne predstavljaju više izolovana ostrva gde se informacije i podaci zadržavaju na lokalnom nivou, već se zbog potreba tržišta interakcija sa kooperantima izvršava u realnom vremenu čime se otvaraju različite mogućnosti za ugrožavanje bezbednosti ICS-a. Kako bi se u najvećoj meri redukovala mogućnost za potencijalne kibernetske napade, u procesu akvizicije podataka koji se kasnije koriste za generisanje modela normalnog ponašanja, sistem se potpuno izoluje od spoljašnjeg sveta. Na taj način postižu se bezbedni i kontrolisani uslovi za funkcionisanje ICS-a tokom prikupljanja podataka na osnovu kojih će kasnije biti razvijen IDS.

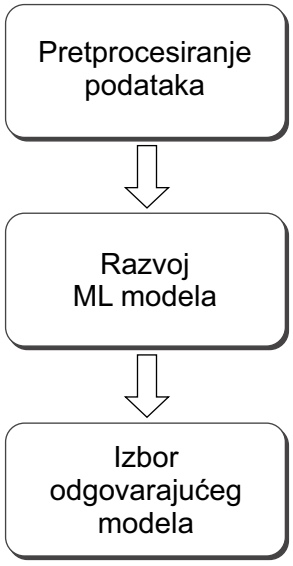
Predložena metodologija zasnovana je na primeni autoregresije transmitovanih signala. Naime, kao što je u prethodnom odeljku naglašeno, modeliranje signala obavlja se korišćenjem univarijatne regresije, gde se vrednost svakog signala u datom trenutku procenjuje na osnovu njegovih vrednosti u prethodnim trenucima. Predložena postavka može se lako proširiti na multivarijatnu regresiju gde se više signala istovremeno modelira koristeći jedan model. Kako se u slučaju univarijatnog pristupa ne razmatraju i nisu poznate korelacije između različitih

signala, u postupku generisanja modela nameću se strožiji zahtevi za IDS algoritam u odnosu na multivarijatni pristup. Međutim, primenom univarijatnog pristupa olakšava se implementacija IDS-a jer se razvijeni algoritmi mogu lako implementirati na prijemnom uređaju i izvršiti po prijemu svakog odbirka signala, a razvijeni modeli se mogu upotrebiti i u slučaju da se algoritmi originalno implementirani na jednom upravljačkom uređaju distribuiraju na veći broj uređaja. Još jedan od razloga zbog kojih je realna implementacija univarijatnog pristupa na uređaje koji čine upravljački sistem znatno lakša, leži u činjenici da se kod ovog pristupa ne javljaju problemi vezani za različite brzine odabiranja i različita kašnjenja signala koja mogu biti izazvana prethodnom obradom i prenosom podataka sa senzora i aktuatora. Takođe, univarijatni pristup obezbeđuje prilično jednostavnu alokaciju algoritama na uređaje, što se može sprovesti i na distribuiranim sistemima upravljanja poput sistema sa slike 2b.

Tokom onlajn faze za detekciju napada koriste se svi regresioni modeli razvijeni u oflajn fazi. Imajući u vidu da su svi senzori i aktuatori koji upravljaju radom nekog procesa međusobno povezani upravo kroz dati proces, tokom rada sistema moguće je napad usmeren ka jednom uređaju detektovati na nekom drugom uređaju. Na primer, napad na aktuator može biti detektovan na signalu senzora ukoliko je imao uticaj na upravljani proces; ukoliko napad nije imao uticaja na proces, onda je njegova efektivnost izostala. U skladu sa navedenim, čak i ako na pojedinim komunikacionim vezama iz različitih razloga, uključujući i karakteristike uređaja, nije moguće implementirati IDS, ne znači da napadi na te komunikacione veze neće biti otkriveni u nekom drugom elementu sistema upravljanja.

4.2. Oflajn generisanje modela

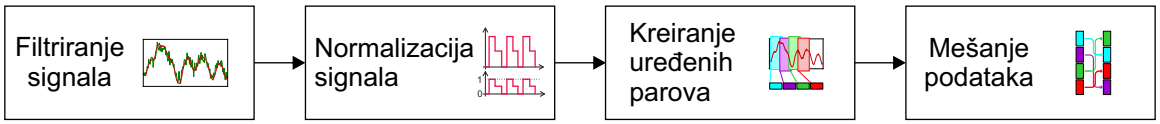
Na najvišem nivou, oflajn algoritam za generisanje ML modela (slika 11) sastoji se od tri faze: 1) pretprocesiranje podataka (signala), 2) razvoj univarijatnog autoregresionog modela zasnovanog na mašinskom učenju i 3) izbor odgovarajućeg modela i podešavanje vrednosti praga za detekciju napada. U daljem tekstu se detaljnije razmatraju ove faze.



Slika 11: Oflajn generisanje modela – osnovne faze

4.2.1. Pretprocesiranje podataka

Kako su za primenu tehnika zasnovanih na mašinskom učenju, a posebno DNN tehnika, podaci od krucijalnog značaja, njihovo pretprocesiranje predstavlja bitan korak u cilju unapređenja performansi algoritma. Pretprocesiranje podataka izvodi se primenom različitih tehnika poput normalizacije, filtriranja, segmentacije, balansiranja (uravnoteženja) skupa podataka itd. U okviru predložene metodologije faza pretprocesiranja sadrži četiri osnovna koraka: filtriranje signala, normalizaciju signala, kreiranje uređenih parova i mešanje podataka (slika 12).

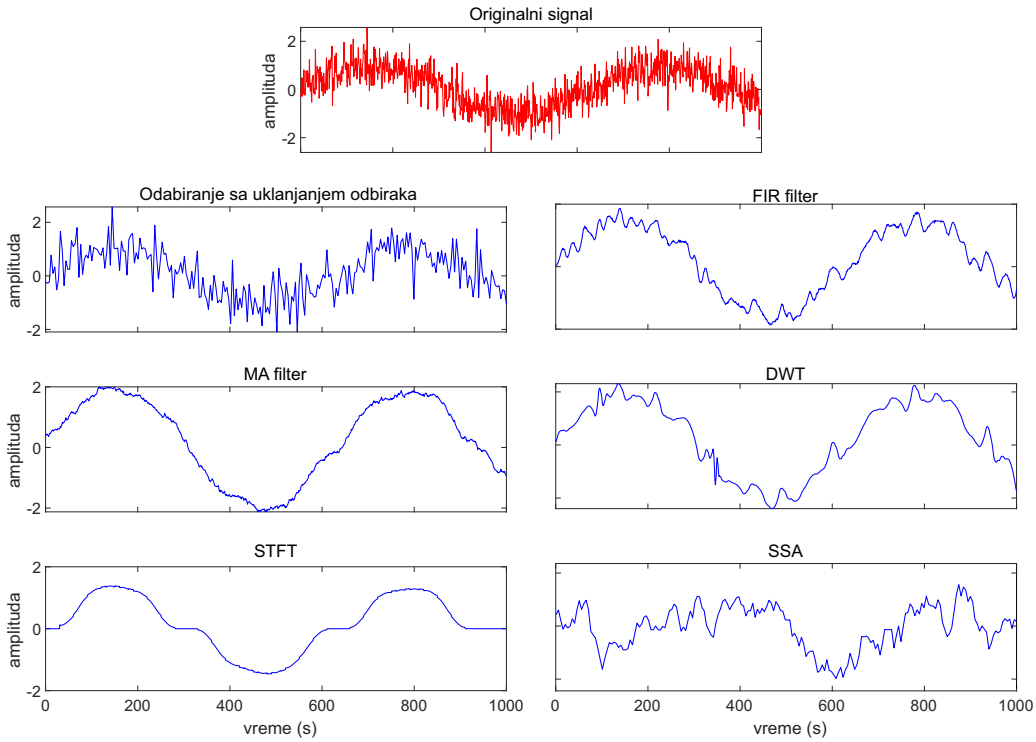


Slika 12: Faza 1 – Pretprocesiranje podataka

U radu sa podacima koji su prikupljeni iz realnih sistema, npr. iz ICS, neizostavni su šumovi visokih frekvencija čija pojava često prouzrokuje lošije performanse sistema u kojem se dati podaci koriste. Iz tog razloga, jedan od koraka u procesu obrade podataka jeste uklanjanje visokofrekventnog šuma iz snimljenog signala.

U radovima razmatranim u poglavlju 3 korišćene su različite tehnike u ove svrhe. Na primer, u [60] iz spektra signala dobijenog kratkotrajnom Furijeovom transformacijom (engl. *Short-Time Fourier Transform* – STFT) isključuju se delovi sa niskom energijom koji pored ostalih spektralnih karakteristika sadrže visokofrekventni šum. U [68] korišćen je jednostavan filter pokretnih srednjih vrednosti (engl. *Moving Average* – MA) koji spada u grupu filtera sa konačnim impulsnim odzivom (engl. *Finite Impulse Response* – FIR) gde svi koeficijenti pomerajućeg prozora imaju istu (srednju) vrednost. U cilju izdvajanja određenih frekvencija iz signala, u [102] pored smanjenja frekvencije odabiranja (engl. *down-sampling*) upotrebljena je diskretna vejvlet transformacija (engl. *Discrete Wavelet Transform – DWT*) koja u osnovi koristi FIR filtere. U [61] izvršeno je odabiranje sa uklanjanjem odbiraka (engl. *undersampling*) kako bi se iz signala isključio visokofrekventni sadržaj.

Kako bi se odabrao pristup za pretprocesiranje signala u okviru ove doktorske disertacije upoređeni su rezultati primene ukupno šest tehnika koje su često zastupljene u relevantnim istraživanjima: 1) FIR filteri, 2) Odabiranje sa uklanjanjem odbiraka, 3) DWT, 4) MA filter, 5) Analiza singularnog spektra (engl. *Singular Spectrum Analysis* – SSA) i 6) STFT. Analiza navedenih tehnika obuhvatala je i varijaciju vrednosti odgovarajućih parametara. Kod tehnike odabiranja sa uklanjanjem odbiraka uklanjan je svaki četvrti odbirak, FIR filter je projektovan korišćenjem 16 koeficijenata, dok su vrednosti MA filtera izračunate na osnovu 16 odbiraka. Kod DWT izabran je *Daubechies* vejvlet 4. reda (db4) kao i 4 nivoa dekompozicije signala. Pored toga, STFT je podrazumevala prozor od ukupno 100 odbiraka sa faktorom preklapanja od 0,7. Na kraju, SSA model definisan je na osnovu prozora od 200 odbiraka i 10 sopstvenih vrednosti korišćenih za rekonstrukciju signala. Izabrane tehnike primenjene su na proizvoljnom signalu dužine 1000 odbiraka koji uključuje nasumično generisani šum (slika 13).



Slika 13: Uporedna analiza tehnika za pretprocesiranje signala

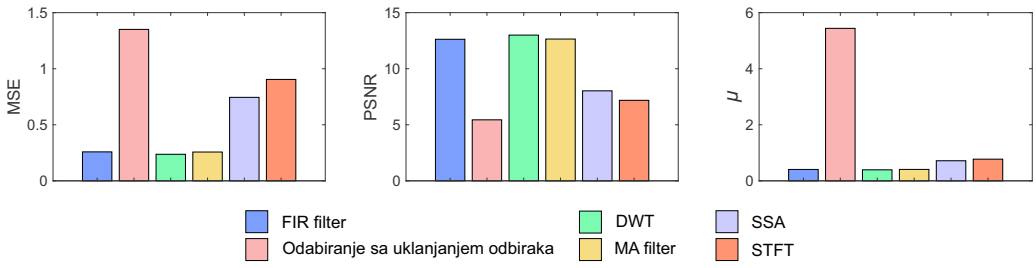
Sa slike 13 može se primetiti da je visokofrekventni šum delimično ili potpuno uklonjen iz signala primenom svih tehnika izuzev tehnike odabiranja sa uklanjanjem odbiraka. Ipak, za procenu performansi ovih tehnika neophodno je uvesti određene metrike. Kao metrike koje se često koriste u ove svrhe izabrane su srednja kvadratna greška (engl. *Mean Squared Error* – MSE), maksimalni odnos signal/šum (engl. *Peak Signal to Noise Ratio* – PSNR) i srednja apsolutna greška μ [112]. Opšti izrazi za izračunavanje MSE, PSNR i μ mogu se zapisati na sledeći način:

$$MSE = \frac{1}{n}\sum_{i=1}^{n}(x_i - \tilde{x}_i)^2 \tag{1}$$

$$PSNR = 10 \cdot log_{10}\left(\frac{max(x_i)^2}{MSE}\right) \tag{2}$$

$$\mu = \frac{1}{n}\sum_{i=1}^{n}|x_i - \tilde{x}_i| \tag{3}$$

gde x_i i $\tilde{x}_i$ predstavljaju vrednost i-tog odbirka originalnog i filtriranog signala, dok je n ukupan broj odbiraka u signalu. Iz izraza (1), (2) i (3) jasno je da su bolji filteri karakterisani manjom vrednošću MSE i μ, odnosno većom vrednošću PSNR.



Slika 14: Poređenje vrednosti MSE, PSNR i μ prilikom pretprocesiranja signala sa slike 13 korišćenjem različitih tehnika

Kao najefikasnije tehnike u ovom primeru mogu se izdvojiti FIR filteri, DWT i MA filteri čijom su primenom ostvareni približno isti rezultati. Suprotno tome, najlošije performanse postignute su tehnikom odabiranja sa uklanjanjem odbiraka, dok se korišćenje SSA i STFT pokazalo kao nedovoljno dobro.

Odabir jedne od tri tehnike koje su dale najbolje rezultate, sproveden je na osnovu dalje analize koja između ostalog uključuje i kašnjenja koje primena ovih tehnika prouzrokuje. Naime, kako pravovremena detekcija napada u realnom vremenu može ublažiti/sprečiti posledice na sistem, čoveka i/ili životnu sredinu, teži se da kašnjenje prouzrokovano implementacijom tehnika za detekciju bude što je moguće manje. Generalno, kao tehnika najveće računske složenosti izdvaja se DWT. Na primer, za uklanjanje visokofrekventnog šuma iz signala sa slike 13 primena DWT 4. reda sa db4 vejvletom zahteva 64 odbirka, dok taj broj u slučaju FIR filtera primenjenog na istom signalu iznosi svega 16. FIR filtere, uključujući i MA filtere karakteriše relativno niska računska složenost i jednostavna implementacija. U slučaju namenski generisanog FIR filtera promenom vrednosti parametara moguće je postići željeni frekventni odziv ili fazni pomeraj filtera čime se može podešavati stepen filtriranja signala. S druge strane, sva podešavanja vezana za MA filtere svode se na promenu dužine pomerajućeg prozora što je u pojedinim slučajevima nedovoljno za definisanje adekvatnog propusnog opsega.

Iz navedenih razloga, u okviru predložene metodologije za detekciju kibernetskih napada, za uklanjanje sadržaja visokih frekvencija odabran je FIR filter. FIR filter u svom odzivu sadrži konačan broj odbiraka i implementira se korišćenjem operacije konvolucije. U okviru ove doktorske disertacije za svaki signal se upotrebljava namenski kreiran FIR filter koji zavisi od konkretnih spektralnih karakteristika signala. Prilikom kreiranja filtera, pored izbora adekvatnog propusnog opsega, uzima se u obzir i kašnjenje koje nastaje njegovom implementacijom, a koje se može smanjiti uvođenjem što manjeg broja odbiraka u impulsnom odzivu filtera (m). Za određivanje vrednosti koeficijenata u impulsnom odzivu filtera korišćen je Parks-Meklelanov algoritam [96].

Nakon otklanjanja šuma iz signala, u fazi pretprocesiranja sprovodi se normalizacija njegovom maksimalnom vrednošću, čime se postiže bolje razumevanje i jednostavnije tumačenje pojedinih karakteristika sadržanih u podacima. Takođe, ovim korakom ostvaruje se kompatibilnost sa algoritmima mašinskog učenja, s obzirom na to da neke tehnike zahtevaju da podaci budu normalizovani kako bi se postigle optimalne performanse prilikom obučavanja modela.

Kao što je prethodno rečeno, predloženi pristup zasniva se na autoregresiji pa se trenutna vrednost signala x_i estimira na osnovu sekvence od v prethodnih vrednosti. Ulazni oblik podataka u ML algoritam za obučavanje definisan je preko obučavajućih parova na sledeći način:

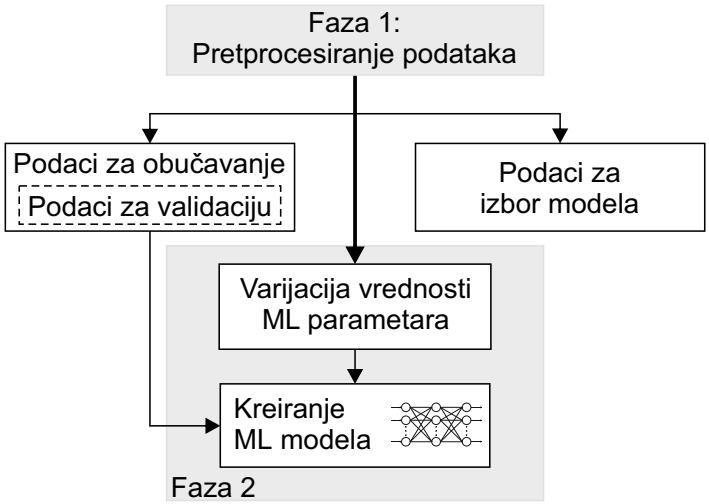
$$(\mathbf{x}_i, y_i) \in [([x_1, ..., x_v], x_{v+1}), ([x_2, ..., x_{v+1}], x_{v+2}), ..., ([x_{i-v}, ..., x_{i-1}], x_i), ..., ([x_{n-v}, ..., x_{n-1}], x_n)] \tag{4}$$

gde je $\mathbf{x}_i = [x_{i-v}, ..., x_{i-1}]$ ulazna sekvenca, dok $y_i = x_i$ predstavlja odgovarajući odziv.

Poslednji korak koji se izvršava pre procesa obučavanja modela je mešanje podataka (engl. *shuffle*). Ovim postupkom obezbeđuje se da podaci namenjeni za obučavanje/validaciju/izbor modela pripadaju istoj raspodeli, čime se prilikom generisanja modela izbegava predubeđenje određenim delom signala. Kako se mešanje podataka izvodi nakon kreiranja uređenih parova (4), zavisnosti u signalu ostaju sačuvane u ulaznim sekvencama $\mathbf{x}_i = [x_{i-v}, ..., x_{i-1}]$. Ovaj princip mešanja opravdano je primeniti kada se koriste tehnike bez memorije (kao što je CNN) kod kojih međusobni raspored obučavajućih parova nije bitan. S druge strane, u tehnikama sa memorijom poput RNN, redosled obučavajućih parova je krucijalan i mešanje podataka se ne vrši jer bi dovelo do gubitka zavisnosti na osnovu kojih se kasnije generišu modeli.

4.2.2. Razvoj ML modela

Nakon navedenih postupaka pretprocesiranja podataka, skup podataka je neophodno podeliti na podskupove za obučavanje/validaciju/izbor modela (slika 15). Analizom dosadašnjih istraživanja utvrđeno je da su u najvećem broju slučajeva najbolje performanse ML modela ostvarene kada je korišćena podela u kojoj je 80% ukupne količine podataka namenjeno za obučavanje, a ostalih 20% za izbor modela [56]. Stoga, ovaj odnos je izabran u daljem postupku kreiranja i izbora ML modela. Podaci za validaciju predstavljaju sastavni deo skupa za obučavanje i u ovom slučaju obuhvataju 10% od originalnog skupa podataka.



Slika 15: Faza 2 – Razvoj ML modela

Cilj razvoja ML modela jeste da se na osnovu ulaznih podataka pronađu karakteristike i relacije čime se postiže dublje razumevanje ponašanja sistema. Kreirani model ponašanja koristi se potom za predikciju narednih vrednosti. Izbor odgovarajuće tehnike za kreiranje ML modela zavisi od više faktora poput oblika i prirode podataka, proračunskih resursa koji su na raspolaganju itd. U nastavku su predstavljene razmatrane tehnike za modeliranje transmitovanih podataka.

4.2.2.1. ML tehnike za modeliranje transmitovanih podataka

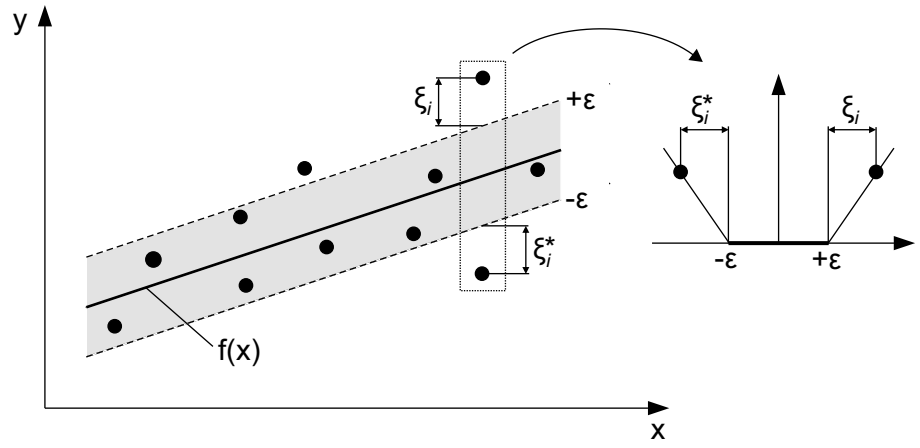
Za modeliranje transmitovanih podataka biće ispitana mogućnost korišćenja ukupno tri klase tehnika mašinskog učenja: 1) Regresija nosećim vektorima [87, 88], 2) Konvolucione neuronske mreže [84] i 3) Rekurentne neuronske mreže [82]. Struktura i princip funkcionisanja svake od razmatranih tehnika biće opisana u nastavku.

Regresija nosećim vektorima

Mašine sa nosećim vektorima (SVM) predstavljaju algoritam mašinskog učenja koji se koristi u zadacima klasifikacije linearno i nelinearno razdvojivih podataka [122]. Osnovna ideja SVM-a jeste da pronađe hiperravan koja najbolje razdvaja vektore obeležja koji reprezentuju podatke u različite klase. Hiperravan se bira tako da maksimizira marginu između dve klase koja je definisana kao rastojanje između hipperravni i najbližih vektora obeležja iz svake klase.

Regresija nosećim vektorima (engl. *Support Vector Regression* – SVR) koristi isti osnovni princip kao SVM, ali umesto pronalaženja hiperravni koja razdvaja različite klase, pokušava da pronađe hiperravan koja odgovara tačkama podataka uz minimalnu grešku. Ova hiperravan aproksimira podatke sa maksimalnim odstupanjem jednakim ε, gde je ε parametar SVR. Prilikom određivanja hiperravni, greške koje su manje od ε se ne uzimaju u obzir i iz tog razloga, uvodi se ε-neosetljiva (engl. ε-*insensitive*) funkcija gubitka kojom se u procesu pronalaženja

hiperravni dozvoljavaju odstupanja u definisanim granicama (slika 16). Podešavanje vrednosti parametra ε koji predstavlja širinu margine svodi se na pronalaženje kompromisnog rešenja između kompleksnosti i generalizacije modela.



Slika 16: Određivanje hiperravni u okviru ε-SVR [106]

Neka je dat skup podataka za obučavanje $\{(\mathbf{x}_1, y_1), (\mathbf{x}_2, y_2), ..., (\mathbf{x}_l, y_l)\}$ gde je $\mathbf{x}_i, i \in [1, l]$ vektor ulaznih promenljivih, dok y_i predstavlja odgovarajući odziv. Cilj ε-neosetljive SVR (ε-SVR) je da pronađe funkciju $f(\mathbf{x}_i)$ koja je udaljena maksimalno za vrednost ε od y_i, a istovremeno teži da bude ravna što je više moguće. Pritom, greške koje su manje od vrednosti ε se ne razmatraju. Funkcija $f(\mathbf{x})$ je opisana sa:

$$f(\mathbf{x}) = \langle \mathbf{w}, \mathbf{x} \rangle + b \tag{5}$$

Ravnost funkcije postiže se kada je $\mathbf{w}$ malo pa se problem pronalaženja parametara hiperravni svodi na sledeći problem minimizacije:

$$\begin{aligned} &\text{minimizirati } \frac{1}{2}||\mathbf{w}||^2 \\ &\text{prema } |y_i - \langle \mathbf{w}, \mathbf{x}_i \rangle - b| \leq \varepsilon \end{aligned} \tag{6}$$

čime se minimizira norma $\mathbf{w}$, a podaci se zadržavaju unutar hiper-cilindra prečnika ε koji je formiran oko funkcije $f(\mathbf{x})$. Izraz (6) podrazumeva aproksimaciju svih parova $(\mathbf{x}_i, y_i)$ sa ε preciznošću. Kako se ovaj slučaj dešava izuzetno retko u realnosti, odstupanja se mogu definisati uvođenjem nenegativnih promenljivih ξ_i, ξ_i^* (slika 16) pa izraz (6) dobija sledeći oblik [106]:

$$\begin{aligned} &\text{minimizirati } \frac{1}{2}||\mathbf{w}||^2 + C\sum_{i=1}^{l}(\xi_i + \xi_i^*) \\ &\text{prema } \begin{cases} y_i - \langle \mathbf{w}, \mathbf{x}_i \rangle - b \leq \varepsilon + \xi_i \\ \langle \mathbf{w}, \mathbf{x}_i \rangle + b - y_i \leq \varepsilon + \xi_i^* \end{cases} \end{aligned} \tag{7}$$

gde je sa C označen parametar regularizacije kojim se ostvaruje kompromis između ravnosti funkcije i broja tačaka koje se nalaze izvan ε cilindra. Problem optimizacije (7) može se zapisati u sledećoj formi [106]:

$$\begin{aligned} &\text{minimizirati } \begin{cases} -\frac{1}{2}\sum_{i,j=1}^{l}(\alpha_i - \alpha_i^*)(\alpha_j - \alpha_j^*)\langle \mathbf{x}_i, \mathbf{x}_j\rangle \\ -\varepsilon\sum_{i=1}^{l}(\alpha_i - \alpha_i^*) + \sum_{i=1}^{l} y_i(\alpha_i - \alpha_i^*) \end{cases} \\ &\text{prema } \sum_{i=1}^{l}(\alpha_i - \alpha_i^*) = 0, \text{ gde } \alpha_i, \alpha_i^* \in [0, C] \end{aligned} \tag{8}$$

gde α_i i α_i^* predstavljaju Lagranževe množioce koji imaju nenulte vrednosti ukoliko je uslov $|f(\mathbf{x}_i) - y| \geq \varepsilon$ ispunjen. U suprotnom, ukoliko se vektori nalaze unutar ε cilindra, α_i i α_i^* nestaju (jednaki su nuli). Vektori koji zadovoljavaju uslov nenultih vrednosti α_i i α_i^* nazivaju se noseći vektori (engl. *support vectors*). Rešenje problema opisanog u (7) dato je sa:

$$\mathbf{w} = \sum_{nv}(\alpha_i - \alpha_i^*)\mathbf{x}_i \tag{9}$$

gde nv predstavlja broj nosećih vektora, a $f(\mathbf{x})$ postaje:

$$f(\mathbf{x}) = \sum_{nv}(\alpha_i - \alpha_i^*)\langle \mathbf{x}_i, \mathbf{x}\rangle + b \tag{10}$$

ε-SVR se može proširiti na nelinearnu regresiju preslikavanjem ulaznih vektora u višedimenzionalne prostore u okviru kojih se primenjuje linearna regresija. Kako su u (8) ulazni vektori za obučavanje prisutni jedino u skalarnom proizvodu, višedimenzionalni prostor se može implicitno definisati pri čemu je dovoljno znati skalarni proizvod u tom prostoru, tj. nije neophodno vršiti eksplicitnu transformaciju podataka. Skalarni proizvod se može definisati korišćenjem kernela $K(\mathbf{x}, \mathbf{x}_i)$ – funkcije koja zadovoljava uslove Merserove teoreme [103]. Upotrebom kernela u izrazu (8) dobija se:

$$\begin{aligned} &\text{minimizirati } \begin{cases} -\frac{1}{2}\sum_{i,j=1}^{l}(\alpha_i - \alpha_i^*)(\alpha_j - \alpha_j^*)K(\mathbf{x}_i, \mathbf{x}_j) \\ -\varepsilon\sum_{i=1}^{l}(\alpha_i - \alpha_i^*) + \sum_{i=1}^{l} y_i(\alpha_i - \alpha_i^*) \end{cases} \\ &\text{prema } \sum_{i=1}^{l}(\alpha_i - \alpha_i^*) = 0, \text{ gde } \alpha_i, \alpha_i^* \in [0, C] \end{aligned} \tag{11}$$

Sada, funkcija $f(\mathbf{x})$ postaje:

$$f(\mathbf{x}) = \sum_{nv}(\alpha_i - \alpha_i^*)K\left(\mathbf{x}_i, \mathbf{x}_j\right) + b \tag{12}$$

Neki od najčešće korišćenih kernela su radijalni, polinomalni, vejvlet, sigmoidalni kernel itd. Izbor kernela zavisi od aplikacije u kojoj se primenjuje. Kernel koji koristi radijalnu funkciju (engl. *Radial Basis Function* – RBF) definisan je sa:

$$K\left(\mathbf{x}, \mathbf{x}_i\right) = \exp\left\{-\gamma|\mathbf{x} - \mathbf{x}_i|^2\right\}, \text{gde je } \gamma = \frac{1}{2\sigma^2} \tag{13}$$

gde σ^2 predstavlja varijansu. S druge strane, polinomalni kernel određen je na sledeći način:

$$K\left(\mathbf{x}, \mathbf{x}_i\right) = \left[(\mathbf{x} \cdot \mathbf{x}_i) + r\right]^d \tag{14}$$

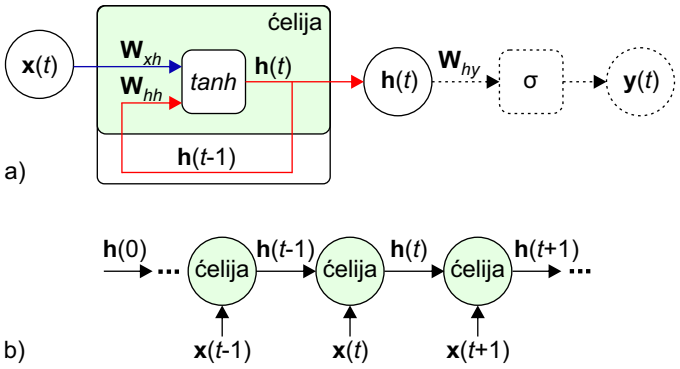
gde je r koeficijent polinoma, dok parametar d predstavlja stepen polinoma.

Rekurentne neuronske mreže

Rekurentne neuronske mreže predstavljaju klasu neuronskih mreža u kojima izlazni vektor $\mathbf{y}(t)$ zavisi ne samo od trenutnog ulaza $\mathbf{x}(t)$, već i od sekvence prethodnih ulaza $\mathbf{x}(t-1)$, $\mathbf{x}(t-2), ...$ reprezentovanih kroz vektor prethodnog skrivenog stanja $\mathbf{h}(t-1)$. Obuhvatajući prethodnu istoriju podataka, RNN uči i razume njihovu sekvencijalnu prirodu. Na taj način, RNN ciklično računa izlaz za svaki element sekvence. U okviru ove doktorske disertacije razmatrana su tri tipa RNN: 1) Jednostavne RNN (engl. *Simple RNN*), 2) Neuronske mreže sa dugom kratkoročnom memorijom – LSTM i 3) Rekurentne neuronske mreže sa zatvorenom rekurentnom jedinicom (engl. *Gated Recurrent Unit* – GRU).

Jednostavne RNN

Jednostavne RNN, poznate još i kao Elmanove mreže [26], predstavljaju potpuno povezanu neuronsku mrežu u okviru koje su skriveni slojevi zasnovani na ćelijama sa povratnom spregom. Petlja u okviru ćelije zadržava vektor skrivenog stanja iz prethodnog odbirka $\mathbf{h}(t-1)$ i dodaje novi ulazni vektor, kao što je prikazano na slici 17a. Na ovaj način, uključujući $\mathbf{h}(t-1)$, vektor skrivenog stanja $\mathbf{h}(t)$ u trenutku t zavisi od sekvence svih prethodnih vrednosti vektora ulaza $\mathbf{x}$, što se može videti na rastavljenom prikazu ćelije (slika 17b). U poređenju sa ostalim tipovima RNN, jednostavne RNN imaju topologiju najsličniju arhitekturi regularnih neuronskih mreža.



Slika 17: Arhitektura jednostavne RNN: a) skriveni sloj; b) rastavljeni prikaz ćelije

Početna vrednost vektora skrivenog stanja (obično postavljena na 0) označena je sa $\mathbf{h}(0)$. Vektor skrivenog stanja $\mathbf{h}(t)$ u trenutku t određen je na sledeći način:

$$\mathbf{h}(t) = tanh\left(\mathbf{W}_{xh}\mathbf{x}(t) + \mathbf{W}_{hh}\mathbf{h}(t-1) + \mathbf{b}_h\right) \tag{15}$$

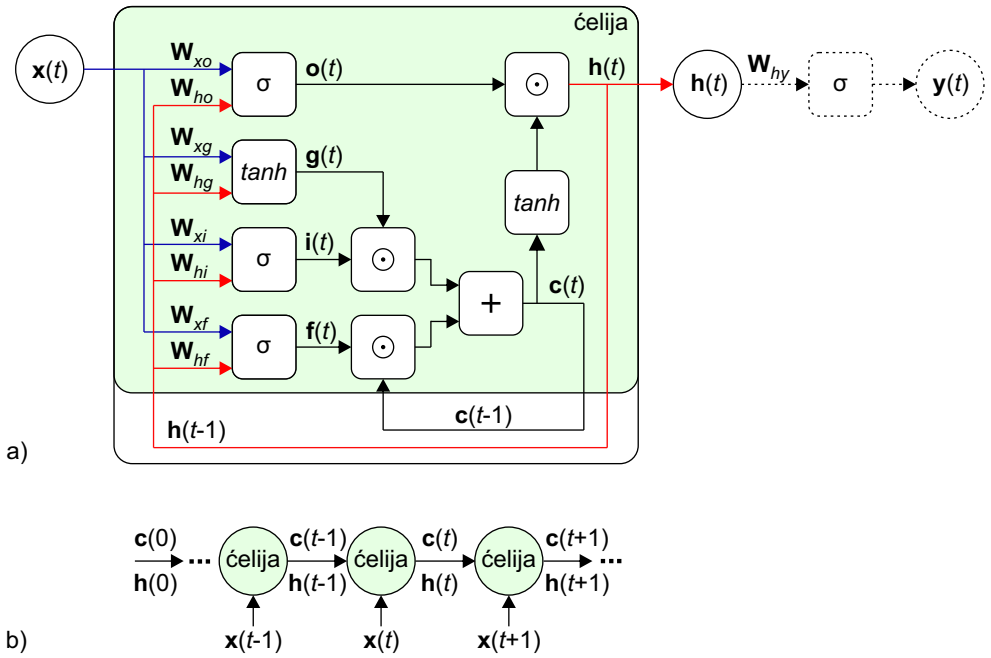
gde je $\mathbf{b}_h$ *bias* vektor, *tanh* predstavlja aktivacionu funkciju hiperboličkog tangensa, dok su sa $\mathbf{W}_{xh}$ i $\mathbf{W}_{hh}$ označene matrica ulaza i matrica skrivenog stanja, tim redom. Vrednost izlaza iz mreže $\mathbf{y}(t)$ izračunava se kao:

$$\mathbf{y}(t) = \sigma\left(\mathbf{W}_{hy}\mathbf{h}(t) + \mathbf{b}_y\right) \tag{16}$$

gde σ predstavlja aktivacionu funkciju, $\mathbf{W}_{hy}$ je matrica izlaza, dok $\mathbf{b}_y$ označava *bias* vektor.

Neuronske mreže sa dugom kratkoročnom memorijom – LSTM

LSTM je razvijen kako bi se rešio problem nestajanja/eksplozije gradijenta koji prilikom obučavanja jednostavne RNN otežava/usporava proces konvergencije [42]. Ova pojava nastaje kao posledica rekurentnog množenja vektora skrivenog stanja $\mathbf{h}$ sa matricom skrivenog sloja $\mathbf{W}_{hh}$, kao što je prikazano u jednačini (15), a za posledicu ima kratkoročno pamćenje jednostavnih RNN. Struktura LSTM sačinjena je od rekurentno povezanih ćelija koje se nazivaju memorijski blokovi (slika 18).



Slika 18: LSTM arhitektura: a) memorijski blok skrivenog sloja (ćelija); b) rastavljeni prikaz ćelije

Vektor stanja $\mathbf{c}$ sadržan u okviru LSTM ćelije memoriše važne informacije tokom rada mreže. Vrednost vektora stanja ćelije $\mathbf{c}(t)$ u trenutku t određena je sa ukupno tri kapije (engl. *gates*):

- Kapija zaboravljanja (engl. *forget gate*) – $\mathbf{f}$;
- Ulazna kapija (engl. *input gate*) – $\mathbf{i}$;
- Kapija stanja kandidata (engl. *candidate state gate*) – $\mathbf{g}$.

Cilj kapije zaboravljanja ($\mathbf{f}$) jeste da se na osnovu prethodnog skrivenog stanja $\mathbf{h}(t-1)$ i trenutnog ulaza $\mathbf{x}(t)$ odrede i izostave irelevantni podaci iz prethodnog stanja ćelije $\mathbf{c}(t-1)$. S druge strane, ulazna kapija ($\mathbf{i}$) određuje koje podatke iz kapije stanja kandidata – $\mathbf{g}(t)$ treba zadržati za sledeće unutrašnje stanje ćelije – $\mathbf{c}(t)$. Vrednost $\mathbf{c}(t)$ je opisana sledećom formulom:

$$\mathbf{c}(t) = \mathbf{f}(t) \odot \mathbf{c}(t-1) + \mathbf{i}(t) \odot \mathbf{g}(t) \tag{17}$$

gde $\odot$ označava Adamarov[4] proizvod (proizvod po elementima matrice). Vrednosti odgovarajućih kapija izračunavaju se na sledeći način:

$$\mathbf{i}(t) = \sigma\left(\mathbf{W}_{xi}\mathbf{x}(t) + \mathbf{W}_{hi}\mathbf{h}(t-1) + \mathbf{b}_i\right) \tag{18}$$

$$\mathbf{f}(t) = \sigma\left(\mathbf{W}_{xf}\mathbf{x}(t) + \mathbf{W}_{hf}\mathbf{h}(t-1) + \mathbf{b}_f\right) \tag{19}$$

$$\mathbf{g}(t) = tanh\left(\mathbf{W}_{xg}\mathbf{x}(t) + \mathbf{W}_{hg}\mathbf{h}(t-1) + \mathbf{b}_g\right) \tag{20}$$

gde $\mathbf{W}_\_$ i $\mathbf{b}_\_$ označavaju odgovarajuće matrice težinskih koeficijenata i *bias*-e, tim redom. Iz jednačina (18), (19) i (20) može se primetiti da ulazna kapija i kapija zaboravljanja koriste sigmoidnu aktivacionu funkciju (σ) koja je u opštem obliku definisana sa $\sigma(x) = 1/(1+e^{-x})$, dok funkcija hiperboličkog tangensa ($tanh$) figuriše u slučaju kapije stanja kandidata. U slučaju sigmoidne funkcije σ izlaz je definisan u opsegu $[0, 1]$, dok se u slučaju funkcije hiperboličkog tangensa dobija vrednost iz opsega $[-1, 1]$.

[4]Francuski matematičar Jacques Hadamard (1865-1963).

Vektor skrivenog stanja koji predstavlja izlaz iz LSTM ćelije određen je sledećom formulom:

$$\mathbf{h}(t) = \mathbf{o}(t) \odot tanh\left(\mathbf{c}(t)\right) \tag{21}$$

gde je vrednost vektora izlazne kapije $\mathbf{o}(t)$ u trenutku t definisana sa:

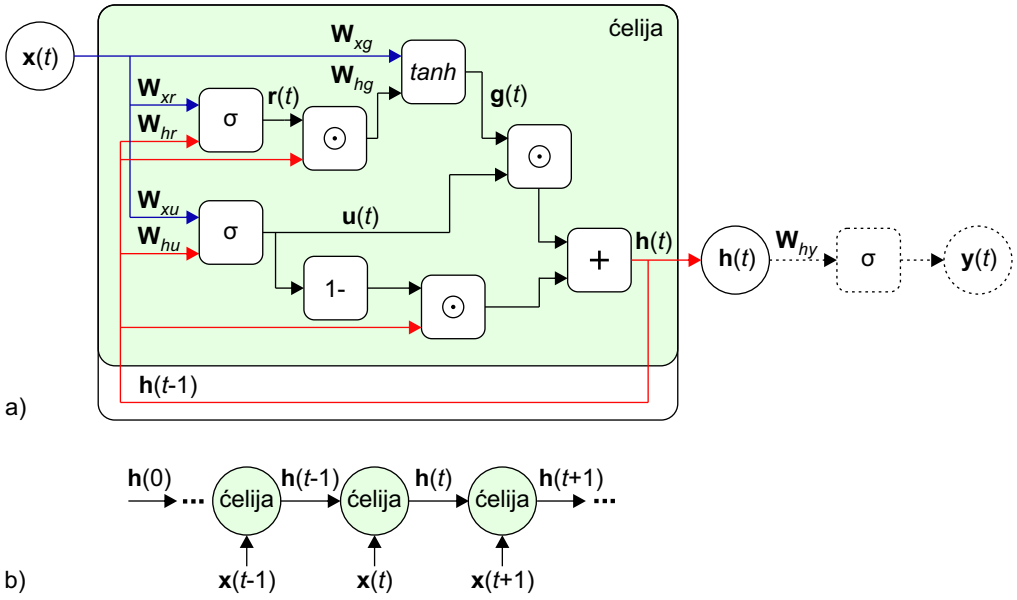
$$\mathbf{o}(t) = \sigma\left(\mathbf{W}_{xo}\mathbf{x}(t) + \mathbf{W}_{ho}\mathbf{h}(t-1) + \mathbf{b}_o\right) \tag{22}$$

Izlaz iz LSTM mreže određen je na sledeći način:

$$\mathbf{y}(t) = \sigma\left(\mathbf{W}_{hy}\mathbf{h}(t) + \mathbf{b}_y\right) \tag{23}$$

Rekurentne neuronske mreže sa zatvorenom rekurentnom jedinicom – GRU

GRU predstavlja relativno novi tip RNN koji je nastao kao rezultat pokušaja da se pojednostavi složena struktura LSTM-a. Jednostavnost u strukturi GRU (slika 19) obezbeđuje manju računsku složenost i pogodnosti prilikom implementacije u poređenju sa LSTM mrežama. S druge strane, to može dovesti i do manje tačnosti predikcije. Po sličnom principu kao kod LSTM-a, određivanje relevantnih podataka iz vektora prethodnog skrivenog stanja koji će biti zadržani ostvaruje se korišćenjem ćelija. Međutim, u ovom slučaju unutrašnje stanje ćelije nije eksplicitno definisano.



Slika 19: GRU arhitektura: a) memorijski blok skrivenog sloja (ćelija); b) rastavljeni prikaz ćelije

Ćelija kod GRU mreže sastoji se od dve kapije: kapije za ažuriranje (engl. *update gate*) – **u** i kapije za resetovanje (engl. *reset gate*) – **r**. Kapija za ažuriranje može se opisati kao kombinacija kapije zaboravljanja i ulazne kapije iz LSTM-a, čiji je cilj da istovremeno određuje koje podatke treba ukloniti, a koje dodati vektoru skrivenog stanja **h**. Vektor kapije ažuriranja $\mathbf{u}(t)$ u trenutku t određen je sa [14, 16]:

$$\mathbf{u}(t) = \sigma\left(\mathbf{W}_{xu}\mathbf{x}(t) + \mathbf{W}_{hu}\mathbf{h}(t-1) + \mathbf{b}_u\right) \tag{24}$$

S druge strane, koji deo vektora skrivenog stanja treba izostaviti, određuje se pomoću kapije

za resetovanje ($\mathbf{r}$) na sledeći način:

$$\mathbf{r}(t) = \sigma\left(\mathbf{W}_{xr}\mathbf{x}(t) + \mathbf{W}_{hr}\mathbf{h}(t-1) + \mathbf{b}_r\right) \tag{25}$$

Kao i u strukturi LSTM mreža, GRU sadrži aktivaciju kandidata $\mathbf{g}(t)$:

$$\mathbf{g}(t) = tanh\left(\mathbf{W}_{xg}\mathbf{x}(t) + \mathbf{W}_{hg}\mathbf{h}(t-1) \odot \mathbf{r}(t) + \mathbf{b}_g\right) \tag{26}$$

Izlaz iz GRU ćelije u trenutku t predstavlja vektor skrivenog stanja $\mathbf{h}(t)$ definisan sa:

$$\mathbf{h}(t) = \mathbf{h}(t-1) \odot (1 - \mathbf{u}(t)) + \mathbf{g}(t) \odot \mathbf{u}(t) \tag{27}$$

Na kraju, izlaz iz GRU mreže u trenutku t određen je na isti način kao u slučaju jednostavnih RNN i LSTM:

$$\mathbf{y}(t) = \sigma\left(\mathbf{W}_{hy}\mathbf{h}(t) + \mathbf{b}_y\right) \tag{28}$$

Prilikom obučavanja RNN, posebno kada je reč o kompleksnim arhitekturama, RNN modeli neretko postaju skloni preobučavanju. Kako bi se ovaj problem prevazišao koristi se izostavljajući sloj (engl. *dropout layer*) kojim se tokom obučavanja nasumično izostavljaju pojedini neuroni (njihove vrednosti se postavljaju na nulu) [107]. Na taj način sprečava se da se model previše oslanja na bilo koje obeležje iz podataka što često predstavlja uzrok preobučavanja. Ovaj sloj se po pravilu postavlja između dva sloja u neuronskoj mreži gde se procenat izostavljenih neurona (u odnosu na ukupan broj neurona) definiše parametrom koji se naziva stopa izostavljanja.

Konvolucione neuronske mreže

Konvolucione neuronske mreže predstavljaju vrstu neuronskih mreža koje primenjuju konvoluciju – operaciju često korišćenu u filtriranju signala u okviru digitalne obrade signala, ekstrahovanju frekvencija i/ili vremenskih karakteristika iz signala, kao i u procesima analize i digitalne obrade slike [38, 62]. Konvolucija dve funkcije ($f(t)$ i $g(t)$) definisana je na sledeći način:

$$(f * g)(t) = \int_{-\infty}^{+\infty} f(\tau)g(t-\tau)\,d\tau \tag{29}$$

gde je $f(t)$ funkcija koja opisuje ulaz, funkcija $g(t)$ predstavlja filter/kernel, dok $*$ označava operator konvolucije. Za jednodimenzionalne signale (1D) dobijene iz linearnih vremenski invarijantnih kauzalnih sistema, diskretni oblik konvolucije određen je sa [12]:

$$(f * g)(k) = \sum_{k=0}^{m} f(k)g(m-k) \tag{30}$$

Funkcije f i g definisane su u trenucima k, gde m označava ukupan broj odbiraka u filteru, tj. veličinu filtera.

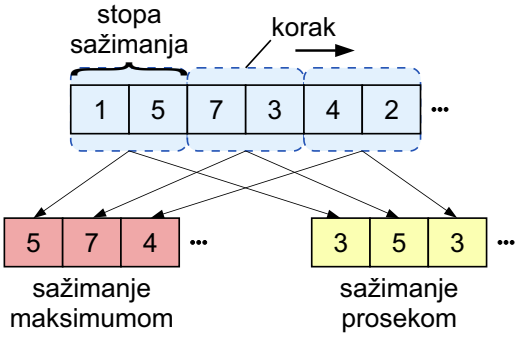
Neuronska mreža se može okarakterisati kao CNN ukoliko u bar jednom svom sloju primenjuje operaciju konvolucije umesto matričnog množenja; takav sloj naziva se konvolucioni sloj [38]. Izlaz h_{ij}^l iz konvolucionog sloja l dobija se konvolucijom između filtera i ulaznih podataka (izlaz iz sloja $l-1$):

$$h_{ij}^l = a(\mathbf{w}_i^l * \mathbf{x}_j^l + b_i^l) \tag{31}$$

gde $\mathbf{x}_j^l$ predstavlja j-ti deo ulaznog vektora sloja l na koji se primenjuje operacija konvolucije, dok $\mathbf{w}_i^l$ i b_i^l označavaju vektor težinskih koeficijenata i *bias* i-tog filtera sloja l, tim redom. Slično kao i kod ostalih slojeva u neuronskim mrežama, izlaz iz konvolucionog sloja prolazi kroz aktivacionu funkciju $a(\cdot)$ čime se izvršavaju nelinearne transformacije i na taj način prepoznaju

i izdvajaju korisne karakteristike iz podataka. Broj i veličina filtera u svakom sloju predstavljaju ulazne parametre koji definišu arhitekturu sloja, dok se vrednost koeficijenata filtera određuje u procesu obučavanja neuronske mreže. Korišćenje *bias*-a je opciono, gde se njegova vrednost (u slučaju kada je uvršten) dobija tokom obučavanja.

U okviru CNN konvolucioni sloj je često praćen slojem sažimanja (engl. *pooling layer*), ispravljajućim slojem (engl. *flattening layer*) i potpuno povezanim slojem (engl. *fully connected layer*). Sloj sažimanja redukuje broj odbiraka dobijenih na izlazu prethodnog sloja, čime se smanjuje broj obučavajućih parametara CNN. Stopa sažimanja označava broj odbiraka u skupu na kom se sprovodi sažimanje, tj. broj odbiraka koji će biti sažeti u jedan, nakon čega se vrši pomeranje prozora za vrednost koraka i tako definiše novi skup (slika 20). Sažimanje može da se vrši odabirom odbirka koji ima maksimalnu vrednost (engl. *max pooling*) iz prethodno definisanog skupa, dok se ostali odbirci odbacuju (slika 20). Pored toga, koristi se i sažimanje prosekom (engl. *average pooling*) gde se preuzima prosečna vrednost odbiraka prisutnih u definisanom skupu, a svi elementi skupa se odbacuju.



Slika 20: Sloj sažimanja: sažimanje maksimumom i prosekom (stopa sažimanja i korak imaju vrednost 2)

Ispravljajući sloj transformiše ulazne podatke bilo kog oblika u format vektora. Obično se koristi da prilagodi postojeći format podataka za dalji rad sa potpuno povezanim slojevima.

4.2.2.2. Određivanje arhitektura i varijacija ML parametara

Nezavisno od izbora tehnike, cilj razvoja ML modela je da se na osnovu podataka za obučavanje dobije što tačnija predikcija narednih vrednosti. U procesu obučavanja modela na osnovu vrednosti ulaznih podataka prilagođavaju se vrednosti internih (obučavajućih) parametara. Ovaj postupak sprovodi se iterativno sa konačnim brojem iteracija i poznat je pod nazivom optimizacija parametara. Ideja je da se kroz svaku narednu iteraciju u procesu obučavanja poboljšava sposobnost modela da izvrši što tačniju predikciju (smanjuje se vrednost greške), iako se u ovom postupku može desiti da se greška sa porastom broja iteracija povećava. Broj potrebnih iteracija po pravilu zavisi od složenosti problema i veličine skupa podataka.

Za pokretanje procesa obučavanja modela neophodno je prethodno definisati njegovu arhitekturu. Arhitektura modela predstavlja strukturu njegovih sastavnih delova i određuje na koji način će model obraditi ulazne podatke i generisati predikciju. Arhitektura ML modela definisana je hiperparametrima koji predstavljaju vrednosti postavljene pre početka procesa obučavanja. Na primer, pojedini hiperparametri definišu broj slojeva, broj skrivenih jedinica u sloju, tip aktivacione funkcije, parametre regularizacije itd.

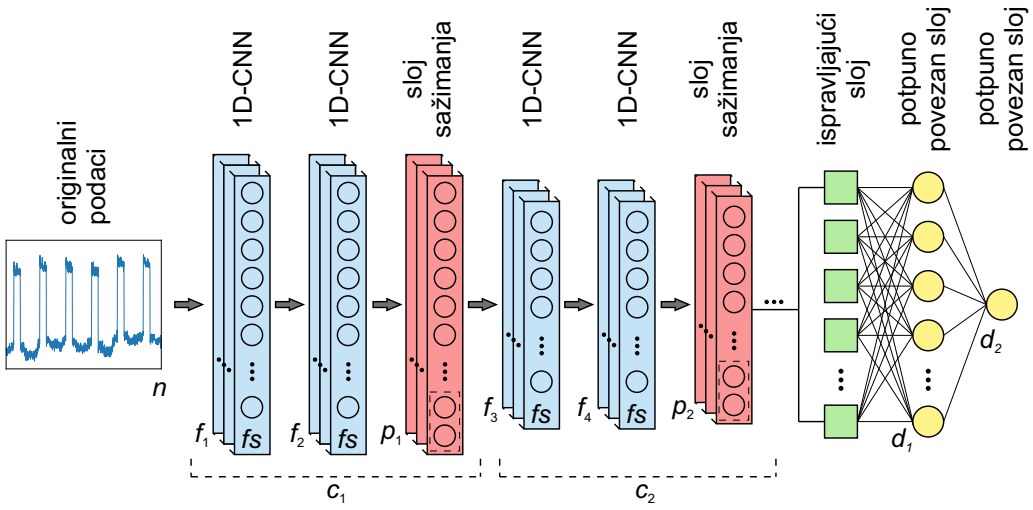
Kako svaku tehniku karakteriše relativno veliki broj hiperparametara, varijacija vrednosti svih hiperparametara predstavlja vremenski zahtevan proces. Pored toga, obučavanje modela često iziskuje značajne vremenske resurse što dodatno produžava proces razvoja ML modela. Stoga, u okviru ove doktorske disertacije proces razvoja ML modela skraćuje se predlaganjem opštih arhitektura čime je delimično određena pozicija i broj slojeva unutar modela.

Kako SVR predstavlja algoritam koji se temelji na istoj arhitekturi, različiti SVR modeli dobijaju se varijacijom vrednosti hiperparametara. Naime, SVR parametri koji će biti varirani u procesu generisanja modela su:

- ε – širina margine razdvajanja;
- kr – tip kernel funkcije;
- C – parametar regularizacije.

S druge strane, u slučaju CNN, predložena je opšta arhitektura prikazana na slici 21. Arhitektura počinje sa c konvolucionih blokova $c_i, i \in \{1, ...c\}$, gde je jedan blok sastavljen od dva konvoluciona sloja, nakon čega se pojavljuje sloj sažimanja. Izlaz iz poslednjeg sloja sažimanja predstavlja ujedno i ulaz u ispravljajući sloj kojim se prilagođava ulazni format podataka za potpuno povezani sloj (sa d_1 neurona) koji sledi. Na kraju mreže, korišćen je potpuno povezani (izlazni) sloj čiji broj neurona d_2 odgovara broju izlaznih parametara; ukoliko se radi o univarijatnoj autoregresiji (relacija (4)), onda je $d_2 = 1$. Na slici 21, $f_i, i \in \{1, ..., 2c\}$ predstavlja broj filtera u konvolucionom sloju i, dok je fs_i veličina filtera u konvolucionom sloju i. U sloju sažimanja, parametar koji se podešava jeste stopa sažimanja označena sa $p_i, i \in \{1, ..., c\}$. Pritom, sažimanje u sloju sažimanja i vrši se izdvajanjem maksimalne vrednosti (engl. *max pooling*) iz skupa dužine p_i, pri čemu je i vrednost koraka jednaka p_i. Za kreiranje CNN modela na bazi opisane arhitekture biće varirane vrednosti sledećih parametara:

- c – broj konvolucionih blokova;
- f_i – broj filtera u konvolucionom sloju i;
- fs_i – veličina filtera u konvolucionom sloju i;
- p_i – stopa sažimanja u sloju sažimanja i;
- d_1 – broj neurona u prvom potpuno povezanom sloju.

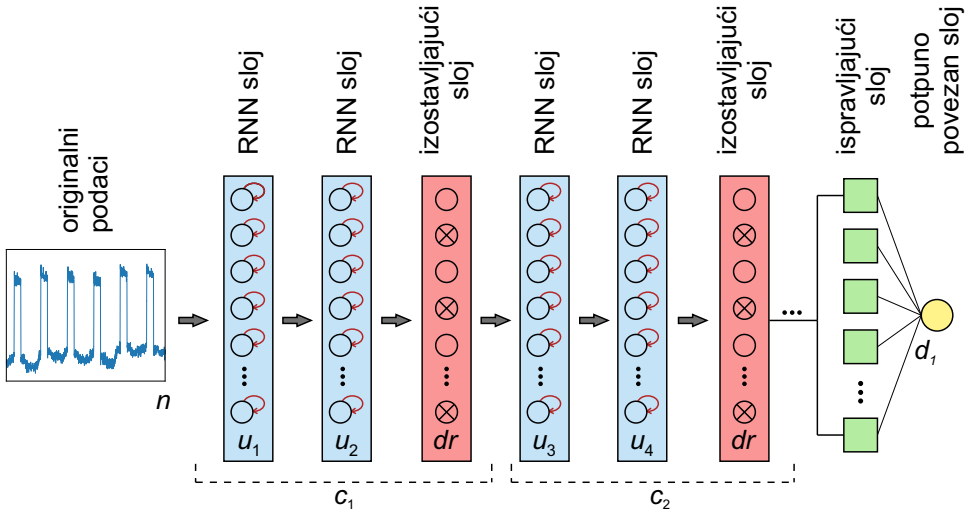


Slika 21: Opšta arhitektura CNN

Slično prikazanoj CNN arhitekturi na slici 21, RNN (važi za sve tri vrste) arhitektura sadrži c blokova, koji su u ovom slučaju sastavljeni od dva rekurentna sloja, nakon čega se primenjuje izostavljajući sloj (slika 22). Dalje, izlaz iz poslednjeg izostavljajućeg sloja predstavlja ulaz u ispravljajući sloj. Za razliku od CNN arhitekture, RNN arhitektura ima samo jedan potpuno

povezani (izlazni) sloj, čiji je broj neurona d_1 takođe određen brojem izlaznih parametara. Broj jedinica u svakom RNN sloju predstavljen je sa $u_i, i \in \{1, ..., 2c\}$, dok je sa dr označena stopa izostavljanja u izostavljajućim slojevima. Generisanje jedinstvenih RNN modela biće izvršeno varijacijom vrednosti sledećih parametara:

- c – broj rekurentnih blokova;
- u_i – broj jedinica u rekurentnom sloju i;
- dr – stopa izostavljanja u izostavljajućim slojevima.

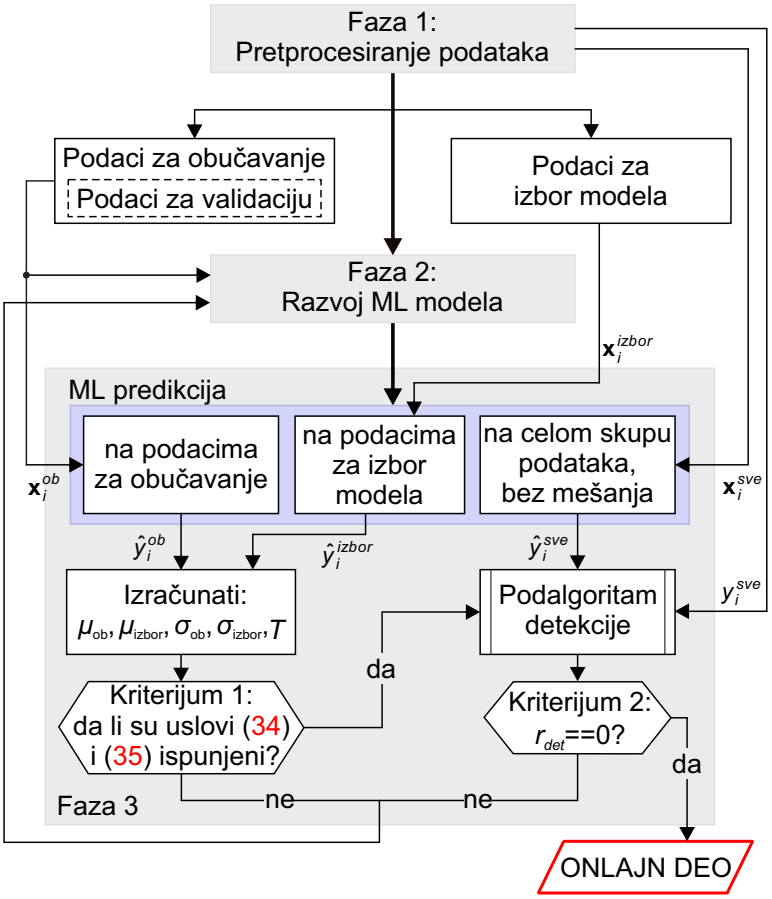


Slika 22: Opšta arhitektura RNN

Treba napomenuti da su se predložene arhitekture pokazale kao najbolje, ali da razvijena metodologija za detekciju kibernetskih napada nije ograničena na korišćenje isključivo ovih arhitektura. Još jedan značajan parametar koji ima uticaj na performanse metode i direktno utiče na kašnjenje prepoznavanja napada jeste broj prethodnih vrednosti signala koji se uzima pri autoregresiji – v. Ovaj parametar zapravo predstavlja dužinu ulaznog vektora $\mathbf{x}_i$ i on je polazna osnova u kreiranju arhitekture modela bilo da se radi o SVR, RNN ili CNN. U daljem tekstu će se v nazivati dužina bafera.

4.2.3. Izbor odgovarajućeg modela

Kao što je ranije navedeno, pronalaženje odgovarajuće arhitekture ML modela je od krucijalnog značaja za postizanje željenih performansi. Pogrešan odabir arhitekture može dovesti do loših performansi, uključujući nisku tačnost, poteškoće u optimizaciji modela i preobučenost/nedovoljnu obučenost (engl. *overfitting/underfitting*) modela. U nekim slučajevima to može rezultirati previše jednostavnim modelom koji nije u mogućnosti da na pravi način opiše kompleksne zavisnosti u podacima, ali i previše složenim modelom koji ne može generalizovati i primeniti naučeno na novim podacima. Stoga, jedno od ključnih razmatranja prilikom kreiranja arhitekture ML modela predstavlja pronalaženje kompromisa između složenosti modela i performansi generalizacije. U cilju ostvarivanja dobrih performansi ML modela, što predstavlja jedan od ključnih preduslova za uspešnu detekciju kibernetskih napada, u nastavku se opisuje postupak izbora odgovarajuće arhitekture ML modela (slika 23).



Slika 23: Faza 3 – Izbor odgovarajućeg modela i izračunavanje vrednosti praga za detekciju napada

U postupku kreiranja modela polazi se od modela sa najmanjim brojem obučavajućih parametara. Takav pristup prihvaćen je u cilju odabira odgovarajućeg modela sa najmanjim brojem parametara čime se ostvaruje najniža računska složenost, a samim tim i najmanje kašnjenje prilikom implementacije što je bitno prilikom aplikacija koje se izvode u realnom vremenu. Ovo razmatranje dodatno dobija na značaju kada se uzmu u obzir energetska i proračunska ograničenja uređaja na koje se IDS implementira.

Kada se ML model kreira, prosleđuje se u proces selekcije gde se utvrđuje da li zadovoljava prethodno definisane kriterijume. U ovom procesu, važno je imati jasno definisane kriterijume čije ispunjavanje obezbeđuje željeni nivo performansi. U zavisnosti od konkretne aplikacije, odnosno zahteva koje ML model treba da ispuni, kriterijumi za odabir optimalnog modela mogu uzimati u obzir različite statističke karakteristike, kompleksnost modela, procenu efikasnosti i/ili robusnosti modela itd.

Uvođenje više kriterijuma u proces izbora optimalnog modela može se sprovesti na nekoliko različitih načina. Jedan od načina jeste dodeljivanje prioriteta svakom kriterijumu gde će se konačna odluka donositi njihovim sastavljanjem u jedan hibridni kriterijum [24]. Ovakav pristup zahteva dodeljivanje težinske vrednosti svakom kriterijumu koja definiše njegov udeo u konačnoj odluci i predstavlja novi hiperparametar čiju je vrednost neophodno podesiti. Određivanje vrednosti ovih hiperparametara često se izvodi ručnim putem što dodatno usložnjava proces izbora odgovarajućeg ML modela. S druge strane, u slučaju definisanja međusobno nezavisnih kriterijuma, ML model koji ispuni sve zadate uslove bira se kao odgovarajući. Nezavisno od broja kriterijuma i njihovih međusobnih korelacija, kada se proces izbora ML modela stavi u kontekst samonadgledanog učenja, jasno je da definisani kriterijumi mogu uključivati samo podatke snimljene tokom normalnog rada sistema, a ne i podatke kada se sistem nalazio pod dejstvom napada.

U okviru predložene metodologije, u procesu selekcije modela učestvuju dva kriterijuma koja su međusobno nezavisna:

1. Kriterijum kojim se na osnovu statističkih parametara utvrđuje da model nije sklon preobučavanju ili nedovoljnom obučavanju;

2. Kriterijum kojim se utvrđuje robusnost modela na poremećaje koji nastaju kao posledica vibracija, mehaničkih svojstava uređaja, različitih uticaja radnog okruženja itd.

Za potrebe prvog kriterijuma koji je zasnovan na statističkim karakteristikama odstupanja između modelirane i snimljene (stvarne) vrednosti, korišćenjem kreiranog ML modela vrši se predikcija na podacima za obučavanje i na podacima za izbor modela. Statističke karakteristike korišćene prilikom uspostavljanja prvog kriterijuma su srednja vrednost (μ) i standardna devijacija (σ). Srednja vrednost i standardna devijacija odstupanja između modelirane i stvarne vrednosti izračunavaju se na sledeći način:

$$\mu_k = \frac{1}{n_k - v} \sum_{i=v+1}^{n_k} \left| y_i^k - \hat{y}_i^k \right|, \; k = \{obučavanje, izbor\} \tag{32}$$

$$\sigma_k = \sqrt{\frac{1}{n_k - v} \sum_{i=v+1}^{n_k} \left(\left| y_i^k - \hat{y}_i^k \right| - \mu_k \right)^2}, \; k = \{obučavanje, izbor\} \tag{33}$$

gde n_k predstavlja dužinu skupova za obučavanje/izbor modela, dok y_i^k i $\hat{y}_i^k$ označavaju stvarnu i estimiranu vrednost i-tog odbirka, tim redom. Uslovi koji definišu prvi kriterijum su:

$$\frac{\mu_{izbor} - \mu_{obučavanje}}{\mu_{izbor}} \cdot 100 < 0.5 \tag{34}$$

$$\frac{\sigma_{izbor} - \sigma_{obučavanje}}{\sigma_{izbor}} \cdot 100 < 0.5 \tag{35}$$

Ovim uslovima utvrđuju se razlike između prikupljenih podataka i njihove predikcije u izdvojenim skupovima podataka za obučavanje i izbor modela. Relativne vrednosti razlike utvrđene su na osnovu statističkih parametara – srednje vrednosti (34) i standardne devijacije (35). Da bi se potvrdilo da razmatrani model nije sklon preobučavanju ili nedovoljnom obučavanju, relativne vrednosti razlika u izrazima (34) i (35) ne smeju preći vrednost 0,5% gde je granična vrednost utvrđena empirijskim putem.

Na osnovu srednje vrednosti i standardne devijacije razlike između snimljenih i estimiranih vrednosti na skupu podataka za izbor modela izračunava se vrednost praga detekcije što predstavlja jedan od najosetljivijih parametara prilikom kreiranja IDS-a baziranog na podacima. Naime, vrednost praga detekcije T definisana je kao suma srednje vrednosti i trostruke standardne devijacije odstupanja između snimljenih i estimiranih vrednosti na skupu podataka za izbor modela, uključujući na taj način 99,73% odstupanja:

$$T = |\mu_{izbor}| + 3\sigma_{izbor} \tag{36}$$

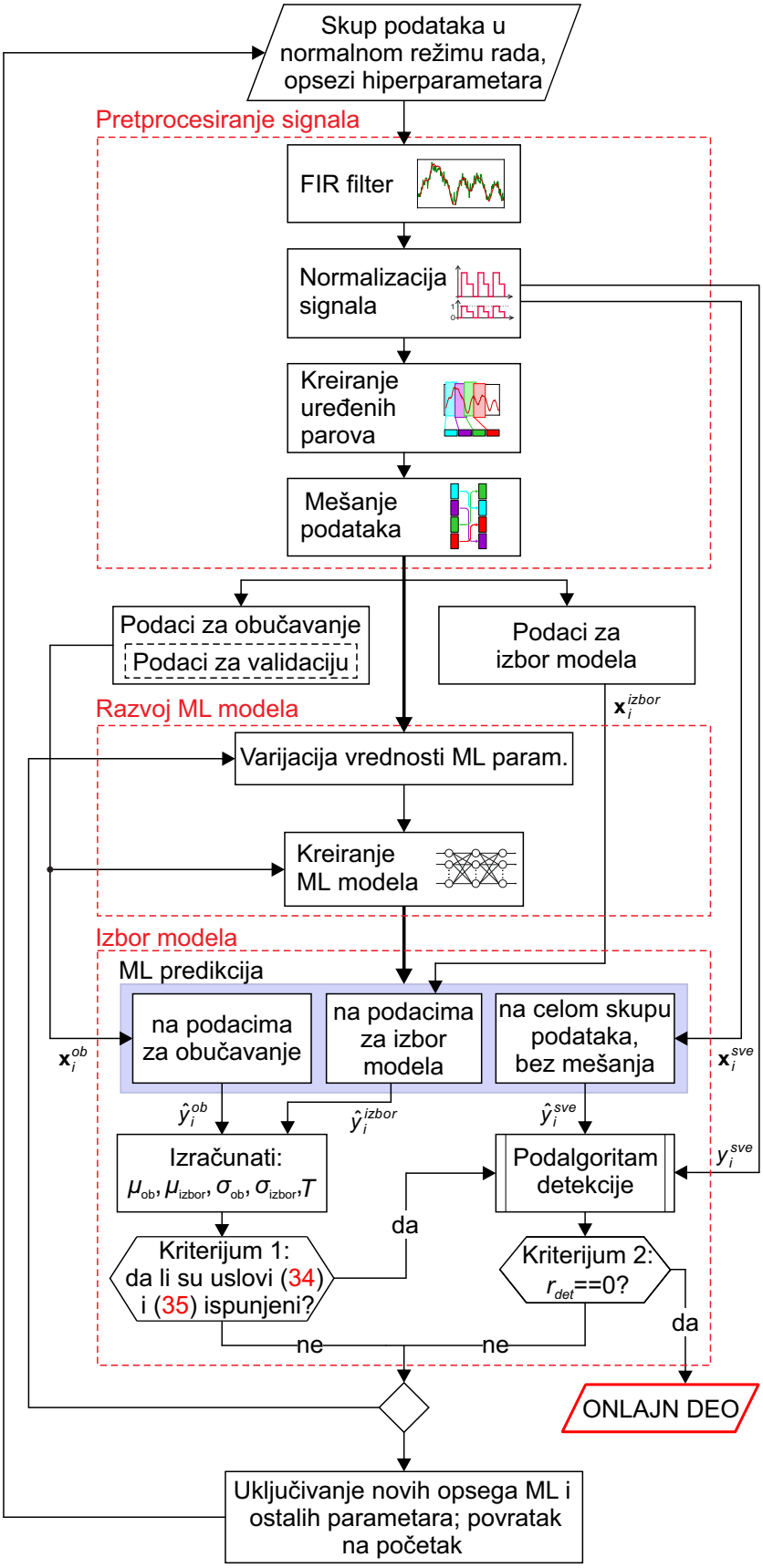
Kao što se može primetiti iz izraza (36), prag detekcije je isključivo funkcija podataka i ne predstavlja jedan od parametara koje treba na bilo koji način podešavati u procesu generisanja ML modela. Za razliku od alternativnih multivarijatnih pristupa gde se često koristi jedna vrednost praga detekcije za sve signale, u predloženom pristupu za svaki signal se na osnovu podataka za izbor modela i dobijene estimacije izračunava posebna vrednost praga detekcije. Kako različite signale karakteriše i različita dinamika, samim tim i odziv usled dejstva kibernetskih napada, univerzalna vrednost praga detekcije često nije odgovarajuća za sve signale. Stoga, u predloženom pristupu prag detekcije signala dobija se korišćenjem statističkih

karakteristika odstupanja isključivo između njegovih stvarnih i estimiranih vrednosti bez uticaja i bilo kakve vrste predubeđenja nastalih od strane drugih signala.

Kako u normalnom radu (bez napada) u okviru ICS postoje poremećaji koji nastaju kao posledica mehaničkih svojstava uređaja, vibracija itd, neophodno je stvoriti uslove koji obezbeđuju da oni neće biti okarakterisani kao napadi [101]. Iz tog razloga teži se robusnom IDS što se u slučaju predloženog pristupa postiže uvođenjem drugog kriterijuma. U ovom kriterijumu postupak onlajn detekcije napada (predstavljen u nastavku) primenjen je na originalne podatke – ceo skup podataka pre mešanja. Naime, korišćenjem ML modela koji predstavlja izlaz iz faze 2 oflajn algoritma sprovodi se estimacija na celom skupu podataka. Na osnovu vrednosti originalnih podataka y_i^{sve}, njihove estimacije $\hat{y}_i^{sve}$ i praga detekcije T, kao i uslova koji su definisani postupkom onlajn detekcije proverava se da li će neki deo podataka prikupljenih u toku normalnog rada biti okarakterisan kao napad. Ukoliko se pritom ne detektuje nijedan napad na skupovima za obučavanje, validaciju i izbor modela, model se bira kao odgovarajući i zajedno sa pragom detekcije prosleđuje se u onlajn deo za detekciju napada.

S druge strane, ukoliko u procesu selekcije modela ne postoji nijedan model koji ispunjava uslove oba kriterijuma, ceo proces (kreiranje i odabir modela) se ponavlja, ali sa novim (dodatim) vrednostima ML parametara i dužine bafera (v). Na primer, za slučaj CNN-a, ukoliko je u postupku generisanja modela utvrđeno da dolazi do preobučavanja, logično je da broj slojeva ili broj filtera u sloju bude smanjen i/ili da stepen sažimanja u slojevima sažimanja bude povećan. Kada se dodavanjem novih vrednosti parametara povećava i kompleksnost modela, to se uvek sprovodi u granicama koje garantuju zadovoljavajuće performanse u skladu sa proračunskim sposobnostima uređaja na koje se IDS implementira.

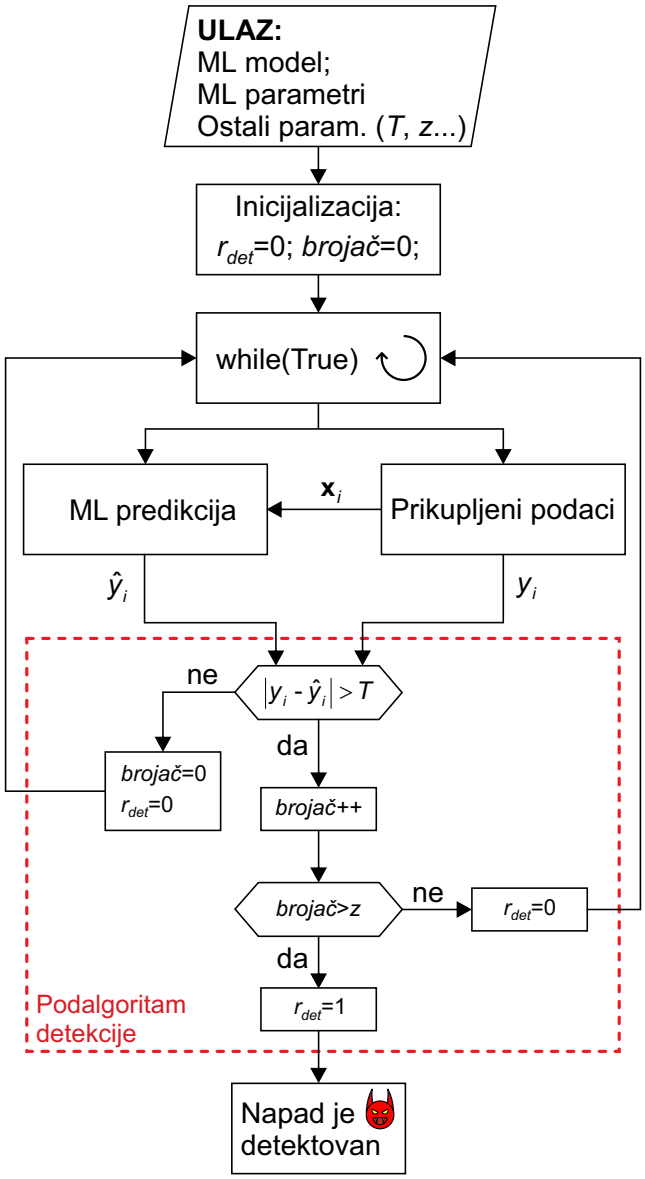
Celokupan oflajn deo predložene metodologije za detekciju kibernetskih napada prikazan je na slici 24. Pored prethodno opisanih faza za pretprocesiranje podataka, razvoj i izbor odgovarajućeg ML modela kao i njihovih međusobnih korelacija, slika 24 prikazuje i uključivanje novih opsega ML i ostalih parametara.



Slika 24: Postavka oflajn dela metode za generisanje i odabir modela

4.3. Onlajn detekcija napada

ML model koji je u prethodnoj fazi zadovoljio postavljene kriterijume i odabran je kao odgovarajući predstavlja jedan od ulaza u onlajn algoritam za detekciju napada koji je baziran na prikupljenim i estimiranim vrednostima signala (slika 25). Pored ML modela, parametri koji se dovode na ulaz u algoritam za detekciju napada su vrednost praga detekcije T i broj dozvoljenih uzastopnih prekoračenja praga – z.



Slika 25: Onlajn detekcija napada

Na osnovu ML modela i niza prethodnih vrednosti $\mathbf{x}_i$ izvršava se predikcija trenutne vrednosti $\hat{y}_i$ koja zajedno sa primljenom (stvarnom) vrednošću y_i predstavlja deo ulaza u podalgoritam detekcije napada (slika 25). Uvođenje podalgoritma kao sastavnog dela algoritma sa slike 25 i promenljive r_{det} koja figuriše unutar njega, izvršeno je isključivo u cilju jasnijeg predstavljanja metodologije u odeljcima 4.2.3 i 7.1.3. U okviru podalgoritma proverava se da li apsolutno odstupanje između primljene vrednosti i predikcije premašuje prag detekcije T. Ukoliko je odstupanje između y_i i $\hat{y}_i$ veće od T za z uzastopno primljenih vrednosti signala smatra se da je na sistem izvršen napad. Detekcija napada se ne vrši na osnovu prvog detektovanog prekoračenja praga kako bi se obezbedila robusnost algoritma na eventualne trenutne

netipične[5] vrednosti. Dalji tok algoritma za detekciju napada zavisi od izlaza iz podalgoritma, gde se u prvom slučaju nastavlja rad sistema kroz beskonačnu petlju, dok se u slučaju kada je napad detektovan algoritam usmerava ka izlazu. Informacija o detekciji napada najčešće inicira sprovođenje prethodno definisanih upravljačkih akcija u cilju sprečavanja posledica koje napad može izazvati.

Za potrebe podalgoritma kreirana je promenljiva *brojač* koja označava trenutni broj uzastopnih prekoračenja praga detekcije, kao i izlazna promenljiva r_{det} čija vrednost ukazuje da li je napad detektovan ili ne. Inicijalna vrednost obe promenljive postavljena je na 0. Svaki put kada apsolutna razlika između y_i i $\hat{y}_i$ prekorači vrednost praga detekcije T, *brojač* uvećava svoju vrednost, dok se u suprotnom njegova vrednost vraća na 0. Kada *brojač* premaši broj dozvoljenih uzastopnih prekoračenja z napad je detektovan i promenljiva r_{det} dobija vrednost 1.

[5]Vrednosti koje nisu karakteristične za ponašanje sistema u normalnom radu (bez napada).

5. Detekcija kibernetskih napada u ICS

U ovom poglavlju prikazani su rezultati primene metodologije za kreiranje algoritama za detekciju kibernetskih napada na ICS koja je predstavljena u prethodnom poglavlju, kao i rezultati primene kreiranih algoritama u detekciji napada na izabrane sisteme. Najpre, za svaku tehniku opisanu u poglavlju 4 definisani su opsezi vrednosti hiperparametara koji će biti korišćeni u procesu kreiranja ML modela. Nakon toga, predstavljene su dve studije slučaja reprezentovane kroz dva skupa podataka koji su korišćeni za evaluaciju razvijenih metoda:

1. SWaT (engl. *Secure Water Treatment*) [51] skup podataka;
2. Skup podataka dobijen iz Elektropneumatskog sistema za pozicioniranje koji je kreiran u okviru ove disertacije [88].

Analizirani su rezultati detekcije napada dobijeni primenom različitih tehnika mašinskog učenja razmatranih u poglavlju 4 kako bi se odredila najpogodnija tehnika čiji će rezultati kasnije biti upoređeni sa postojećim pristupima predloženim od strane drugih autora, a koji su analizirani u poglavlju 3.

Prateći proceduru iz poglavlja 4, tokom oflajn generisanja modela, vrednosti pojedinih hiperparametara su fiksirane, dok su vrednosti ostalih hiperparametara varirane u određenom opsegu. Fiksiranje pojedinih hiperparametara proisteklo je iz potrebe da se proces kreiranja modela učini manje vremenski zahtevnim, tj. da se smanji broj mogućih kombinacija. Pritom, vrednosti hiperparametara fiksirane su u dva empirijski određena slučaja:

1. Kada su određenom vrednošću hiperparametra postignuti odgovarajući rezultati, a dalja promena njegove vrednosti je prouzrokovala lošije rezultate ili je bila iz nekog drugog razloga neprihvatljiva (npr. velika kompleksnost ML modela);
2. Kada promena vrednosti hiperparametra nije izazvala nikakvu ili je izazvala zanemarljivo malu promenu rezultata.

U slučaju SVR modela varirane su dužina bafera v i širina margine razdvajanja ε (tabela 3). Pored toga, različiti SVR modeli definisani su i sa različitim vrstama kernela: radijalna funkcija, polinomalni i linearni kernel. Iako su prilikom kreiranja ML modela korišćeni svi tipovi kernela navedeni u tabeli 3, često se na osnovu prirode i dinamike signala koji se razmatra može unapred pretpostaviti koji je kernel odgovarajući i uzimanjem samo ovog kernela u razmatranje smanjiti vreme potrebno za definisanje pogodnog modela. Vrednost parametra regularizacije C nije varirana kao hiperparametar, već je određivana u postupku generisanja svakog pojedinačnog modela na osnovu najmanje vrednosti greške prilikom primene unakrsne validacije (engl. *cross validation*).

Tabela 3: Varijacija vrednosti hiperparametara – SVR

Parametar	**Oznaka**	**Vrednost**
širina margine razdvajanja	ε	0,001, 0,002, 0,005, 0,01, 0,05, 0,1, 0,2, 0,5, 1
tip kernel funkcije	*kr*	RBF, *polinomalni*, *linearni*
dužina bafera	v	2, 4, 8, 16, 32

Kada se posmatraju opšti oblici korišćenih CNN i RNN arhitektura između njih se mogu uočiti određene sličnosti poput konvolucionih i rekurentnih blokova, tako da neki od parametara u okviru ovih DNN mogu biti zajedno razmatrani. U slučaju CNN arhitektura može se primetiti da se za kreiranje modela koriste dva ili tri konvoluciona bloka (tabela 4). Pored

toga, razmotrene su i CNN arhitekture koje obuhvataju samo jedan konvolucioni blok, ali se ti slučajevi prema predloženim kriterijumima nisu pokazali kao odgovarajući. S druge strane, kod RNN broj rekurentnih blokova je variran u opsegu 1-3. U pojedinim slučajevima pokazalo se da je c=1 bilo dovoljno za generisanje odgovarajućeg RNN modela, iako se i za ovaj tip neuronske mreže najčešće c=2 ispostavilo kao optimalno (tabela 5).

Tabela 4: Varijacija vrednosti hiperparametara – CNN

Parametar	**Oznaka**	**Vrednost**
dužina bafera	v	16, 32, 64
broj konvolucionih blokova	c	2, 3
stepen sažimanja u 1. sloju sažimanja	p_1	2
stepen sažimanja u 2. sloju sažimanja	p_2	2
stepen sažimanja u 3. sloju sažimanja	p_3	2
broj neurona u 1. potpuno povezanom sloju	d_1	30, 40, 50
veličina filtera	fs	2, 3, 4
broj filtera u 1. CNN sloju	f_1	4, 8, 16
broj filtera u 2. CNN sloju	f_2	8, 16, 32
broj filtera u 3. CNN sloju	f_3	8, 16, 32
broj filtera u 4. CNN sloju	f_4	16, 32, 64
broj filtera u 5. CNN sloju	f_5	16, 32, 64
broj filtera u 6. CNN sloju	f_6	32, 64, 128

Tabela 5: Varijacija vrednosti hiperparametara – RNN

Parametar	**Oznaka**	**Vrednost**
dužina bafera	v	2, 4, 8, 16, 32
broj rekurentnih blokova	c	1, 2, 3
stopa izostavljanja u izostavljajućem sloju	dr	0, 0,05, 0,1
broj jedinica u 1. RNN sloju	u_1	4, 8, 16, 32
broj jedinica u 2. RNN sloju	u_2	4, 8, 16, 32
broj jedinica u 3. RNN sloju	u_3	8, 16, 32, 64
broj jedinica u 4. RNN sloju	u_4	8, 16, 32, 64
broj jedinica u 5. RNN sloju	u_5	16, 32, 64, 128
broj jedinica u 6. RNN sloju	u_6	16, 32, 64, 128

Specificirani skupovi vrednosti za broj filtera f_i u CNN sloju i i broj jedinica u_i u RNN sloju i, $i \in \{1, \dots 6\}$ određeni su empirijskim putem. Manje vrednosti f_i i u_i od predloženih dovodile su do nedovoljno dobrih modela (nisu zadovoljavali definisane kriterijume za izbor modela), dok je njihovo dalje uvećavanje rezultiralo velikim brojem obučavajućih parametara. Kada je redukcija broja obučavajućih parametara u pitanju, stepen sažimanja $p_i, i \in \{1, \dots 3\}$ u slojevima sažimanja u okviru CNN-a postavljen je na 2. Veće vrednosti stepena sažimanja prouzrokovale su preveliku redukciju, što je dovodilo do gubitka značajnih informacija sadržanih unutar podataka.

U svim konvolucionim i rekurentnim slojevima korišćena je aktivaciona funkcija ispravljajuće linearne jedinice (engl. *Rectified Linear Unit* – ReLU) definisana sa $a(x)$=$max(0, x)$. Prilikom obučavanja CNN/RNN modela korišćena je funkcija cilja srednje kvadratne greške, pri čemu je za optimizaciju izabran *Adam* optimizator sa parametrom učenja α=0,001. Gubici obučavanja i validacije ostali su gotovo nepromenjeni nakon 5 epoha, stoga je taj broj epoha izabran kao optimalan. Kako je stohastika sastavni deo obučavanja neuronskih mreža i njen uticaj je neizbežan, generisanje modela svake arhitekture ponovljeno je tri puta.

Nezavisno od korišćene tehnike, kao optimalni broj dozvoljenih uzastopnih prekoračenja praga detekcije usvojen je z=15. Kreiranje RNN i CNN modela realizovano je korišćenjem *Python* programskog jezika u *Spyder* okruženju i upotrebom *Tensorflow v2.3.0* platforme, koja je orijentisana na oblast mašinskog učenja. Za generisanje SVR modela korišćena je funkcija *fitrsvm* sadržana u okviru softverskog paketa *Matlab R2022b*. Stanica koja je korišćena za proračun sastoji se od procesora *Intel i7-10750H*, 16GB RAM i grafičke karte *GeForce GTX 1650 Ti*.

Parametri za procenu performansi IDS-a

Da bi se izvršila uporedna analiza performansi algoritama za detekciju napada neophodno je usvojiti odgovarajuću vrstu metrike. U okviru ove doktorske disertacije odabir metrike sproveden je na osnovu količine dostupnih podataka i metrika koje su korišćene u postojećim istraživanjima analiziranim u poglavlju 3. Naime, kako bi uporednom analizom bio obuhvaćen najveći broj relevantnih istraživanja, metrike koje su odabrane su:

1. F_1 skor;
2. *Tačnost*;
3. Stopa lažno pozitivnih (engl. *false positive rate – FPR*) rezultata.

Pre detaljnijeg opisa svake metrike, neophodno je u kontekstu detekcije kibernetskih napada definisati parametre na kojima su predložene metrike zasnovane:

- *tp* – tačno pozitivni (engl. *true positive*) rezultati predstavljaju uspešno detektovane napade;
- *tn* – tačno negativni (engl. *true negatives*) rezultati odnose se na pravilno okarakterisano ponašanje sistema u normalnom radu;
- *lp* – lažno pozitivni (engl. *false positive*) rezultati označavaju da je ponašanje sistema bez prisustva napada okarakterisano kao napad;
- *ln* – lažno negativni (engl. *false negatives*) rezultati označavaju slučajeve gde je napad prisutan, ali nije detektovan.

F_1 skor baziran je na *preciznosti* (engl. *precision*) i *odzivu* (engl. *recall*), a izračunava se na sledeći način:

$$F_1 = 2 \cdot \frac{preciznost \cdot odziv}{preciznost + odziv} \tag{37}$$

gde su *preciznost* i *odziv* (poznat i kao stopa tačno pozitivnih (engl. *true positive rate*)) definisani izrazima:

$$preciznost = \frac{tp}{tp + lp} \tag{38}$$

$$odziv = \frac{tp}{tp + ln} \tag{39}$$

Iz izraza (38) može se primetiti da *preciznost* meri udeo tačno pozitivnih rezultata u ukupnom broju slučajeva koji su okarakterisani kao pozitivni. S druge strane, *odziv* (39) meri udeo tačno pozitivnih rezultata od svih stvarno pozitivnih instanci u skupu podataka.

F_1 skor se može računati na dva načina – po događaju i po odbirku. Kod F_1 skora po događaju za svaki uspešno detektovan napad *tp* uvećava svoju vrednost za 1, bez obzira na vreme trajanja napada, dužinu prelaznog perioda i broj odbiraka koji se detektuju kao napad

nakon njegovog delovanja. Na sličan način, *lp* uvećava vrednost za 1 kada se pojavi grupa uzastopnih odbiraka pri normalnom funkcionisanju sistema koja je okarakterisana kao napad, dok *ln* predstavlja broj napada koji nisu detektovani, ne uzimajući u razmatranje vreme njihovog trajanja. S druge strane, F_1 skor po odbirku posmatra svaki odbirak posebno pa shodno tome prilikom proračuna uzima u obzir trajanje napada. Naime, koristeći ovaj pristup svaki detektovan odbirak tokom napada dodaje se *tp*, odbirak koji nije detektovan tokom napada dodaje se *ln*, dok se svaki odbirak okarakterisan kao napad tokom normalnog rada sistema dodaje *lp*. Nedostatak ovog pristupa ogleda se u tome da se odbirci koji pripadaju prelaznom periodu nakon napada, čija je vrednost uslovljena dinamikom sistema, pogrešno smatraju lažno pozitivnim. U slučaju IDS-a u ICS, F_1 skor po događaju je adekvatnija metrika u odnosu na F_1 skor po odbirku. Naime F_1 skor po događaju bolje oslikava prirodu rada sistema jer će sistem upravljanja zaustaviti rad postrojenja ili primeniti odgovarajuću logiku čim se napad detektuje nedozvoljavajući njegov dalji razvoj i generisanje daljih signala nakon detekcije napada.

Tačnost predstavlja statističku karakteristiku kojom se naglašava sposobnost IDS-a da uspešno detektuje napade, kao i da prepozna ponašanje sistema u normalnom radu. Ona predstavlja udeo tačno okarakterisanih instanci (*tp* i *tn*) u njihovom ukupnom broju. Za korišćenje ove metrike neophodno je znati vrednosti sva četiri parametra (*tp*, *tn*, *lp* i *ln*) [8]:

$$tačnost = \frac{tp + tn}{tp + tn + lp + ln} \tag{40}$$

FPR metrikom pokazuje se tendencija IDS-a da generiše lažno pozitivne rezultate pa je u idealnom slučaju vrednost ovog parametra jednaka nuli [8]:

$$FPR = \frac{lp}{lp + tn} \tag{41}$$

Pored predstavljenih metrika koje će biti korišćene u nastavku, postoje i metrike koje su široko zastupljene u relevantnim istraživanjima, ali iz različitih razloga nisu izabrane kao odgovarajuće. Jedna od takvih metrika je AUC (engl. *Area Under the Curve*) koja je definisana kao površina ispod ROC (engl. *Receiver Operating Characteristic*) krive koja prikazuje *odziv* u odnosu na *FPR* za različite vrednosti praga detekcije. Kako se u pristupu koji je predložen u ovoj doktorskoj disertaciji za jedan signal koristi jedna vrednost praga detekcije, nije moguće formirati ROC krivu, odnosno nije moguće izračunati AUC.

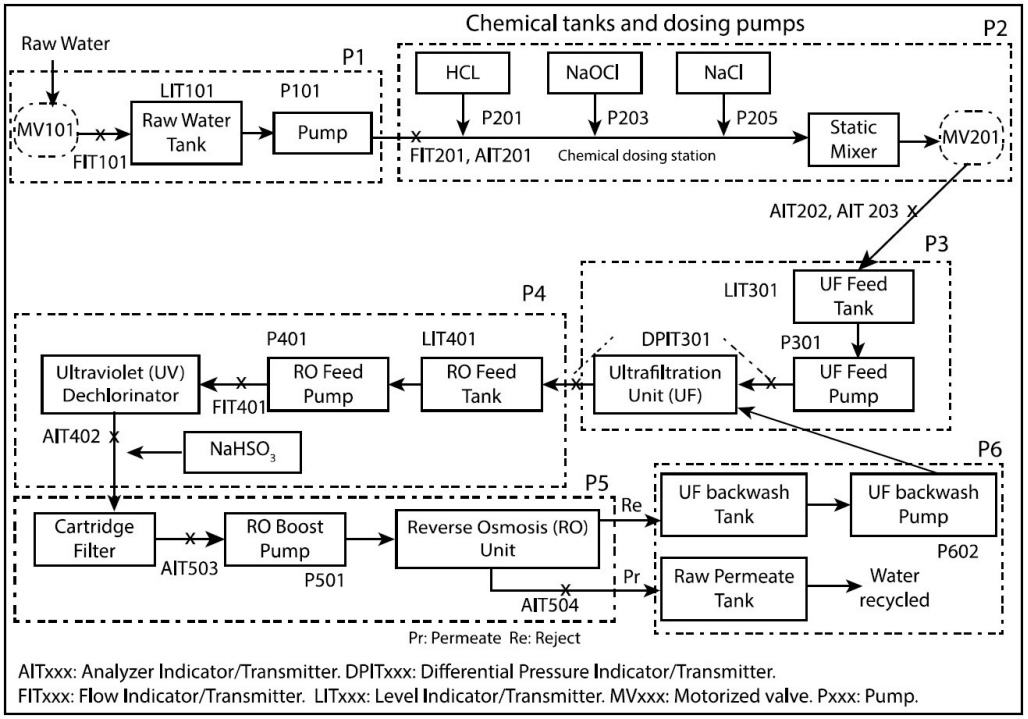
Metrika *specifičnost* ili stopa tačno negativnih (engl. *true negative rate – TNR*) koja se izračunava kao $tn/(tn + lp)$, naglašava značaj lažno pozitivnih rezultata. Kada se uporedi izraz za *specifičnost* i izraz (41) kojim je definisan *FPR*, može se zaključiti da je *specifičnost*=1-*FPR*. Stoga, ova metrika nije uvrštena u dalje razmatranje, jer u odnosu na *FPR* ne donosi nikakve dodatne informacije o performansama IDS-a.

Geometrijska sredina (engl. *geometric mean – G-mean*) predstavlja metriku koja se definiše kao proizvod *odziva* i *specifičnosti* i često se koristi u slučajevima kada klase unutar skupa podataka nisu balansirane [8]. Vrednost *odziva* uključena je u razmatranje kroz F_1 skor, dok se *specifičnost* može izraziti preko *FPR*-a. Iz ovog razloga, parametar *geometrijske sredine* nije prepoznat kao neophodan za dalju analizu i poređenje rezultata sa postojećim pristupima.

5.1. Studija slučaja 1 - SWaT skup podataka

Kao što je već napomenuto u odeljku 3.2.2.1, SWaT predstavlja funkcionalnu skaliranu fabriku za preradu vode osmišljenu i kreiranu na Univerzitetu za tehnologiju i projektovanje u Singapuru (engl. *Singapore University of Technology and Design*) koja poseduje mogućnost proizvodnje 5 galona (oko 19 litara) prečišćene vode u minuti. Glavni cilj ovog postrojenja jeste prikupljanje podataka iz realnog sistema koji se dalje mogu koristiti u različitim istraživanjima,

a najviše su namenjeni istraživanjima u oblasti sajber bezbednosti [36]. Proces prečišćavanja vode podeljen je u šest serijski povezanih sektora P1-P6 (slika 26) gde je svakom sektoru dodeljen jedan PLC. Voda se iz spoljašnjeg izvora dovodi u prvi sektor gde se skladišti u rezervoaru. U drugom sektoru vrši se procena kvaliteta vode, kao i početni koraci tretiranja vode (dodavanje hemijskih supstanci) ukoliko vrednosti kvalitativnih parametara vode nisu u predviđenim granicama. Sledeći korak predstavlja uklanjanje eventualnih nepoželjnih materijala iz vode finim filterskim membranama u sektoru 3. Nakon filtriranja vode u P3, u sektoru P4 uklanja se preostala količina hlora (koji je dodat u P2) korišćenjem ultraljubičastih lampi. Pretposlednji sektor (P5) ima za cilj uklanjanje neorganskih nečistoća iz vode što ujedno predstavlja i poslednji korak tretiranja vode. Tako prerađena voda dovodi se u sektor P6 odakle se dalje prosleđuje u mrežu za snabdevanje ili se vraća u prvi sektor na ponovno tretiranje.

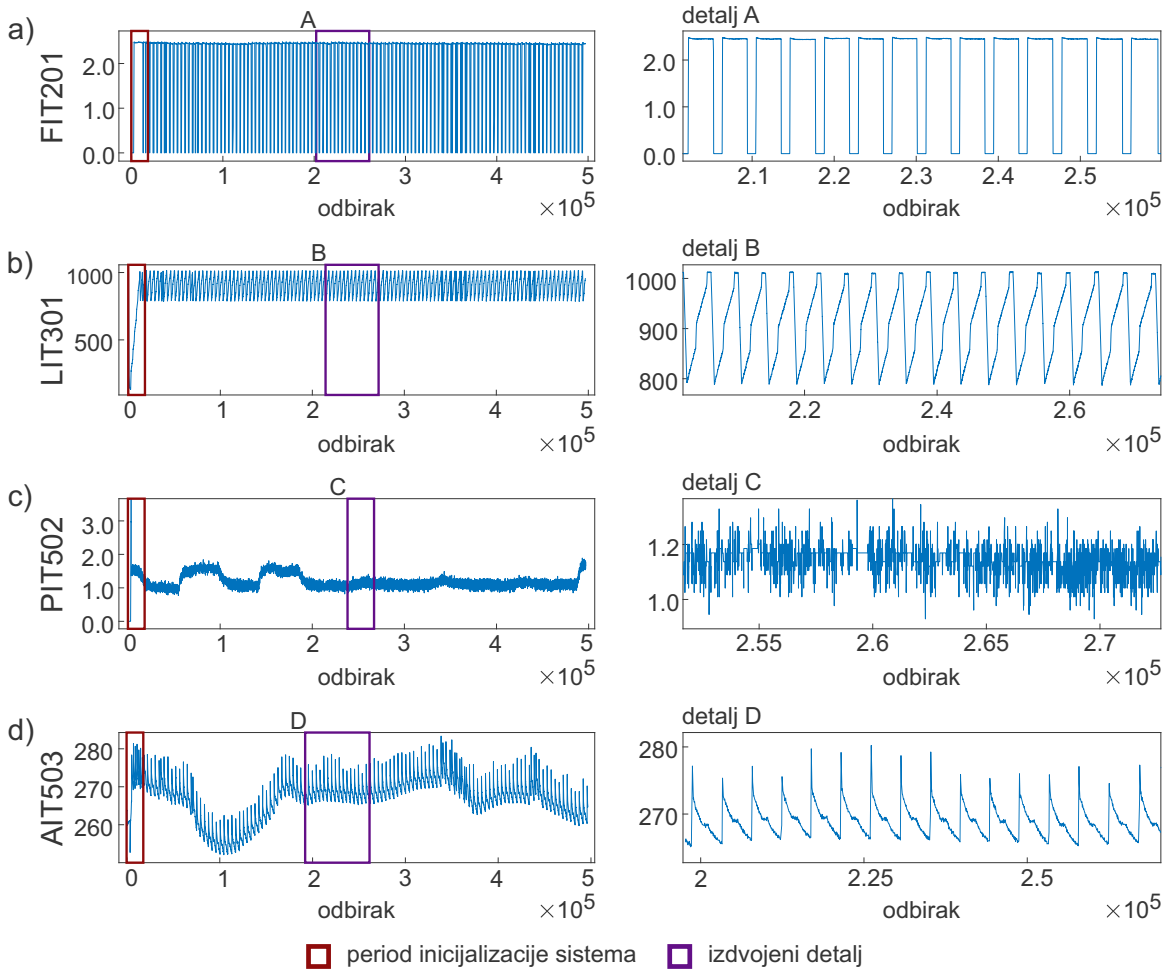


Slika 26: SWaT – postavka sistema [36]

U postrojenju postoji ukupno 25 senzora i 26 aktuatora koji su povezani žičanom ili bežičnom vezom na odgovarajući PLC. Na taj način postiže se očitavanje vrednosti sa senzora, odnosno slanje upravljačkih akcija ka aktuatorima. Na sledećem (višem) nivou svi PLC-ovi u postrojenju povezani su sa SCADA sistemom. Senzori su podeljeni u četiri različite klase u zavisnosti od veličine koju mere: protok (FIT), nivo tečnosti (LIT), pritisak (PIT) i hemijska svojstva (AIT). Na slici 27 prikazan je po jedan signal za svaku klasu senzora. Pored prikaza celih signala izdvojeni su i detalji kako bi se jasnije uočila cikličnost procesa sprovedenih u SWaT postrojenju. Svaka grupa senzora u SWaT skupu podataka pruža jedinstvene informacije o različitim aspektima procesa prečišćavanja vode. Signali koje generišu pokazuju specifičnu dinamiku i karakteristike, kao što su varijacije protoka, promene nivoa, fluktuacije pritiska ili merenja hemijskih svojstava. Analiza i razumevanje ove dinamike signala može pomoći kako u detekciji napada, tako i u optimizaciji performansi procesa i obezbeđivanju pouzdanog i efikasnog rada sistema.

Signali sa senzora FIT predstavljaju vrednosti protoka i njihova dinamika može otkriti način na koji se brzina vode u sistemu menja. Ove promene mogu biti izazvane različitim radnim uslovima kao što su promene u radu pumpe ili ventila. Na signalu sa slike 27a mogu se primetiti periodične oscilacije kao i nagle promene vrednosti protoka koje su uslovljene

režimom rada sistema. Signali sa senzora nivoa kao što su LIT senzori prikazuju promene u visini tečnosti unutar rezervoara. Ovi signali obezbeđuju praćenje statusa nivoa rezervoara i pružaju uvid u dostupnost i upotrebu vode u sistemu. Dinamika signala sa slike 27b pokazuje da se nivo vode u rezervoaru u dva vremenska segmenta postepeno povećava, nakon čega se naglo smanjuje. Signali dobijeni sa PIT senzora predstavljaju izmerene vrednosti pritiska. Kao i u slučaju merenja protoka, varijacije u pritisku najčešće su rezultat promene u radu pumpe ili ventila. Kada se poredi sa ostalim signalima sa slike 27, signal iz ove grupe (slika 27c) karakterisan je učestalijim oscilacijama oko određene vrednosti bez jasno vidljivih intervala koji se ciklično ponavljaju. Senzori iz klase AIT mere koncentraciju određenih hemijskih supstanci (npr. HCl, NaCl i NaOCl) u vodi. Na signalima sa AIT senzora mogu se primetiti intervali gde se vrednost ciklično povećava i smanjuje (detalj D sa slike 27d). Međutim, kada se posmatra bilo koji ceo signal iz AIT klase senzora koji je snimljen tokom normalnog rada SWaT postrojenja, lako je uočiti da vrednosti ne variraju uvek oko približno istih vrednosti kao što je to slučaj kod signala sa FIT i LIT senzora.



Slika 27: SWaT – primeri signala sa različitih vrsta senzora koji mere: a) protok (FIT); b) nivo tečnosti (LIT); c) pritisak (PIT); d) hemijska svojstva (AIT)

Akvizicija podataka sa svih senzora i aktuatora sprovedena je tokom 11 dana sa frekvencijom odabiranja od 1 Hz, uključujući dva različita scenarija: 1) funkcionisanje sistema odvija se u normalnim uslovima rada (bez napada) – prvih 7 dana i 2) sistem je pod dejstvom napada – poslednja 4 dana. Pre početka akvizicije podataka sistem je bio potpuno ispražnjen, nakon čega je bilo potrebno 6 časova kako bi se stabilizovao nivo vode u različitim rezervoarima unutar postrojenja. Na slici 27 crvenim pravougaonicima prikazan je interval koji predstavlja inicijalizaciju sistema gde se jasno može uočiti razlika između ovog perioda i ustaljenog režima

rada sistema koji je usledio nakon inicijalizacije.

Tokom poslednja 4 dana kreirano je ukupno 36 napada koji su prema hronološkom redosledu delovanja označeni rednim brojevima od 1-41 (napadi 5, 9, 12, 15 i 18 su izostavljeni kao neuspeli). U tabeli 6 za svaki od 36 napada navedene su oznake uređaja čiji su signali direktno napadnuti. Može se primetiti da je većina napada imala kao ciljnu tačku signal sa jednog uređaja, dok su neki napadi direktno delovali na signale sa dva (napadi 21, 24, 25...) ili maksimalno tri (napadi 22, 23, 29 i 30) uređaja.

Tabela 6: SWaT skup podataka – svi napadi sa ciljnim uređajima [36]

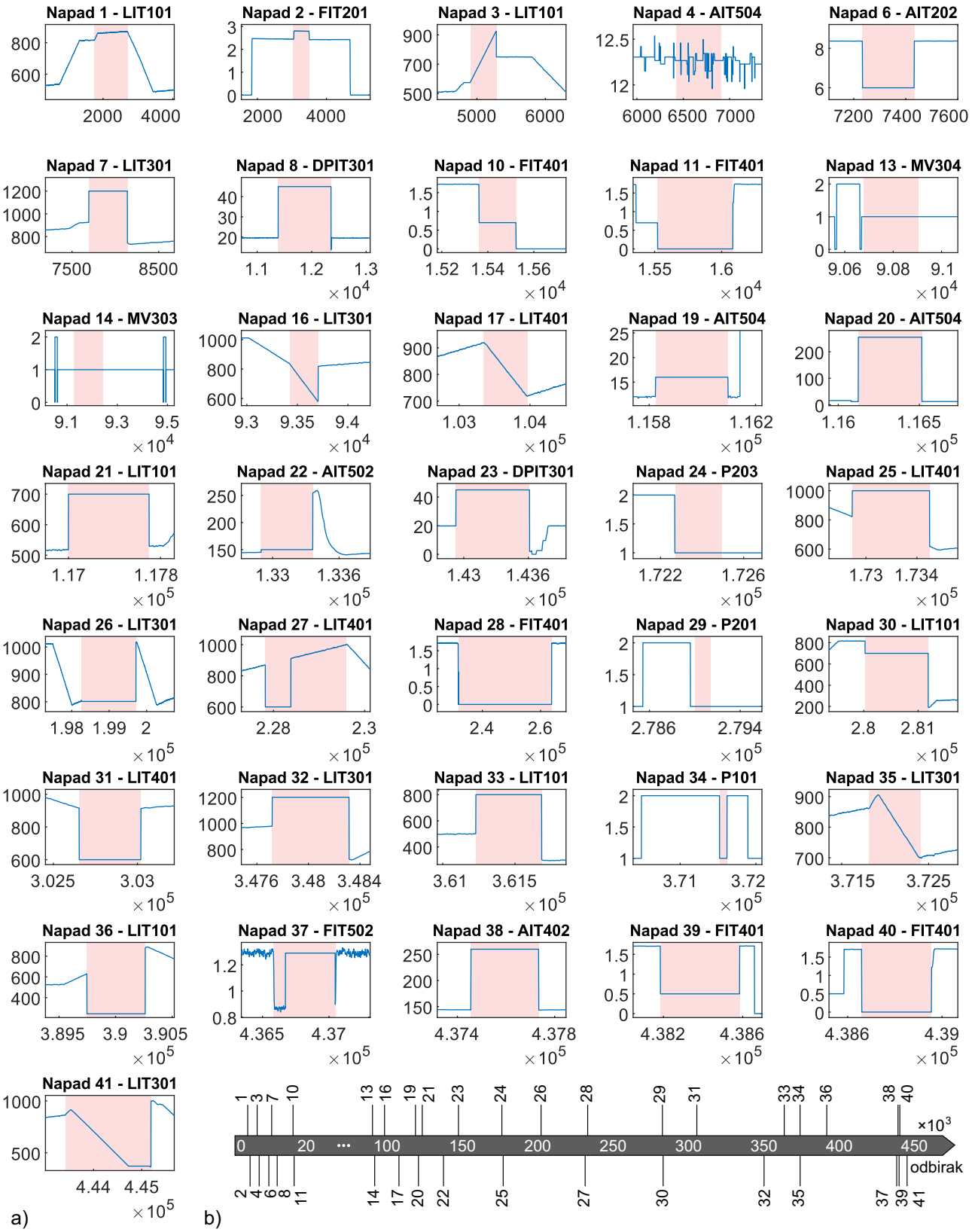
Napad	**Napadnuti uređaj**	**Napad**	**Napadnuti uređaj**	**Napad**	**Napadnuti uređaj**
1	MV101	17	MV303	30	LIT101, P101, MV201
2	P102	19	AIT504	31	LIT401
3	LIT101	20	AIT504	32	LIT301
4	MV504	21	MV101, LIT101	33	LIT101
6	AIT202	22	UV401, AIT502, P501	34	P101
7	LIT301	23	P602, DPIT301, MV302	35	P101, P102
8	DPIT301	24	P203, P205	36	LIT101
10	FIT401	25	LIT401, P401	37	P501, FIT502
11	FIT401	26	P101, LIT301	38	AIT402, AIT502
13	MV304	27	P302, LIT401	39	FIT401, AIT502
14	MV303	28	P302	40	FIT401
16	LIT301	29	P201, P203, P205	41	LIT301

Na slici 28a prikazani su napadi na signalima sa uređaja na kojima je njihov uticaj bio najuočljiviji. Plavom bojom prikazan je signal, dok crveni pravougaonici označavaju vremenski period dejstva napada. Pored neuspelih napada, postoje i napadi koji su nakon delovanja ostavili zanemarljiv (napadi 4, 24 i 34) ili nisu prouzrokovali nikakav (napadi 13, 14 i 29) uticaj na sistem [51] što se može uočiti i na slici 28a. Hronološki prikaz dejstva svih napada tokom poslednja 4 dana predstavljen je vremenskom osom na slici 28b.

Kada se posmatra njihova lokacija, napadi se mogu podeliti u klase u zavisnosti od toga da li napad deluje na jedan ili više uređaja u okviru jednog ili više sektora. Kreirani napadi su okarakterisani različitom dinamikom, dužinom trajanja, intenzitetom, a shodno tome i uticajem na delove sistema ili sistem u celini, kao i vremenom potrebnim za stabilizaciju. U slučaju pojedinih napada koji su delovali u nizu, sistem nije imao dovoljno vremena da se stabilizuje između dva napada.

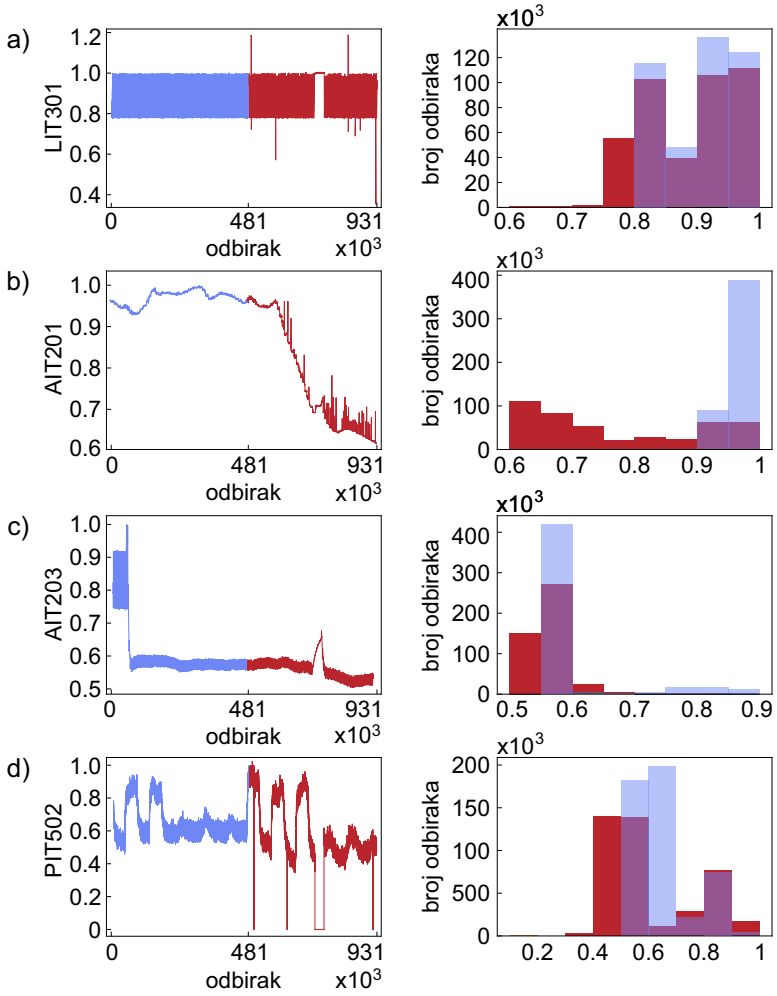
Neki od napada (npr. napadi 3, 16 i 41) predstavljaju linearno povećanje/smanjenje vrednosti signala, što prouzrokuje nedostatak ili prelivanje vode u rezervoarima. Druga grupa napada poput napada 6, 19, 28, 31 (navedeni su samo neki) fiksiraju vrednost signala, dok bi te vrednosti signala u suprotnom varirale. Napadi kojima su cilj aktuatori zadržavaju aktivno/neaktivno stanje aktuatora duži ili kraći vremenski period nego što je inicijalno zadato upravljačkim komandama. Na primer, napad 26 zadržava pumpu P101 uključenom 24 min, iako bi trebalo da bude isključena tokom tog vremenskog perioda. S druge strane, uticaj napada 35 je isključivanje pumpi P101 i P102 na 8 min, iako bi tad trebalo da budu uključene. Uticaji ovakvih napada dovode do smanjenja/prekoračenja nivoa tečnosti u rezervoarima, oštećenja uređaja, promene količine hemijskih supstanci koje dalje utiču i na promenu kvaliteta prečišćene vode itd. Detalji vezani za pojedinačni uticaj svih napada mogu se pronaći u [51].

Kada se posmatra arhitektura SWaT sistema (slika 6), jasno je da se IDS može implementirati ili na PLC-ovima ili na SCADA-i. Tokom kreiranja algoritama za detekciju napada u razmatranje se uzimaju samo signali dobijeni sa senzora. Naime, ovde se polazi od hipoteze da će uspešni napadi na aktuatore imati uticaj na rad postrojenja i samim tim će biti vidljivi na senzorskim signalima.



Slika 28: Napadi u SWaT skupu podataka: a) pojedinačni prikaz napada; b) vremenska osa napada

Ovde treba napomenuti da se, kao rezultat postavke eksperimenta, rad nekih od senzora (AIT201, AIT203 i PIT502) nije stabilizovao nakon određenih napada, što ih je onemogućilo da nakon uticaja napada nastave da funkcionišu u skladu sa svojom kalibracijom i dostignu opsege vrednosti tipične za normalan rad. Signali sa navedenih senzora u vremenskom domenu, kao i histogram raspodele podataka u normalnom režimu rada (prvih sedam dana) i tokom rada pod uticajem napada (poslednja četiri dana) prikazani su na slici 29b-d. Nasuprot tome, na slici 29a prikazan je primer signala sa senzora LIT301 gde je nakon delovanja napada uspostavljeno stabilno stanje što se najbolje može uočiti stepenom preklapanja podataka na datom histogramu.



Slika 29: Signali u vremenskom domenu i histogrami raspodele podataka prikupljenih tokom normalnog rada sistema (plava) i tokom uticaja napada (crvena): a) LIT301; b) AIT201; c) AIT203; d) PIT502

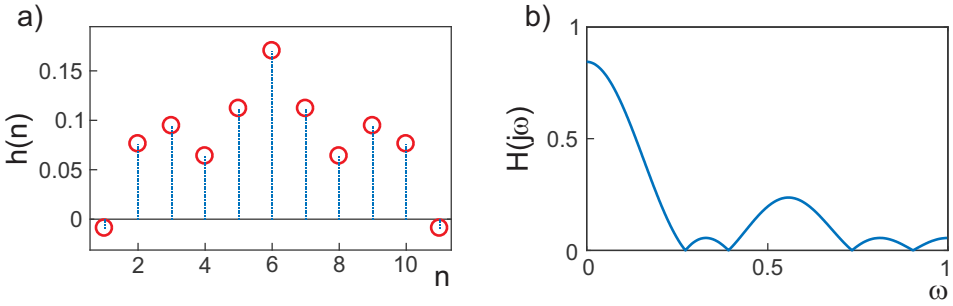
Kako bi se izbegli pogrešni zaključci koji mogu nastati usled neodgovarajuće postavke eksperimenta i neslaganja između signala prikupljenih pre i nakon napada, signali dobijeni sa senzora AIT201, AIT203 i PIT502 izostavljeni su iz daljeg razmatranja. Ovaj problem uočili su i drugi autori u prethodnim istraživanjima koja su uključivala u razmatranje SWaT skup podataka, kao na primer [60] i [132] gde su izostavljeni signali sa čak 15 senzora. Stoga, razvoj algoritama za detekciju kibernetskih napada u okviru ove doktorske disertacije sproveden je na osnovu signala sa 22 senzora, a signali sa senzora AIT201, AIT203 i PIT502 su izostavljeni.

5.1.1. Primena razmatranih ML tehnika za kreiranje IDS-a

U okviru ovog odeljka biće izvršena uporedna analiza algoritama za detekciju napada kreiranih u ovoj disertaciji na osnovu predložene metodologije, a korišćenjem razmatranih tehnika

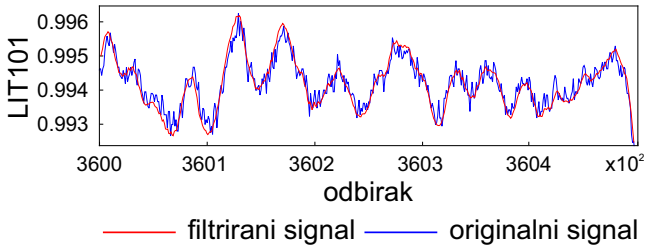
mašinskog učenja (SVR, tri tipa RNN i CNN). Poređenje performansi detekcije napada biće izvršeno na pet različitih signala sa senzora iz SWaT skupa podataka: LIT101, FIT201, LIT401, PIT501 i FIT601. Signali su birani tako da što više različitih napada bude uključeno u proces detekcije. Pored toga, kako bi se izbegla potencijalna predubeđenja, u razmatranje su uključeni senzori tri različite vrste, čime je ispitana mogućnost modeliranja signala različitih dinamika. Izabrani senzori sadržani su u pet različitih sektora.

Oflajn generisanje modela izvršeno je na podacima koji su prikupljeni tokom prvih 7 dana (sistem je funkcionisao u normalnim uslovima rada), gde su odbirci generisani tokom punjenja sistema i pre stabilizacije (u prvih šest sati) izostavljeni iz skupa za obučavanje. Prvi korak u oflajn procesu generisanja modela je pretprocesiranje signala. Iz tog razloga, za potrebe razmatranih signala iz SWaT skupa podataka, koristeći Parks-Meklelanov algoritam [96] razvijen je niskopropusni FIR filter sa propusnim opsegom $[0; 0{,}1\pi]$, nepropusnim opsegom $[0{,}25\pi; \pi]$ i prelaznim regionom između. Kako bi kašnjenje prouzrokovano uvođenjem filtera bilo što manje, kreiran je filter sa 11 koeficijenata čiji je impulsni odziv definisan sa $h(n)$=[-0,0091; 0,0761; 0,0944; 0,0638; 0,1119; 0,1702; 0,1119; 0,0638; 0,0944; 0,0761; -0,0091] (slika 30a). Kreirani filter u frekventnom domenu prikazan je na slici 30b.



Slika 30: Niskopropusni filter korišćen za pretprocesiranje razmatranih signala u okviru SWaT skupa podataka: a) impulsni odziv; b) frekventni domen

Primena razvijenog FIR filtera prikazana je na delu signala dobijenog sa senzora LIT101 (slika 31).



Slika 31: Primena razvijenog FIR filtera na signalu sa senzora LIT101

Predloženom metodologijom (slika 24) za svaki od pet razmatranih signala varijacijom parametara navedenih u tabelama 3, 4 i 5 kreirano je po pet različitih modela (SVR, tri tipa RNN i CNN) što je rezultiralo sa ukupno 25 ML modela.

Arhitekture svih modela, kao i broj obučavajućih parametara (broj nosećih vektora za SVR modele) prikazani su u tabeli 7. U slučaju SVR, arhitektura modela definisana je dužinom bafera v, širinom margine razdvajanja ε, tipom kernela i parametrom regularizacije C. Kod SVR modela dužina bafera varira u opsegu od v=2 za LIT101 i PIT501 do v=8 za FIT201 i FIT601. Odabir odgovarajućeg kernela bio je uslovljen dinamikom signala pa je za signale karakterisane naglim i pretežno linearnim promenama (FIT201 i FIT601) korišćen *linearni* tip kernela, dok je u slučaju ostalih senzora primenjen RBF kernel.

Arhitekture RNN/CNN definisane su brojem blokova, brojem jedinica/filtera u rekurentnim, odnosno konvolucionim slojevima u_i i $f_i, i \in \{1, \dots 6\}$ i dužinom bafera v što dalje definiše broj obučavajućih parametara. Pored toga, RNN modeli definisani su i stopom izostavljanja dr. Može se primetiti da svi RNN i CNN modeli sadrže po 2 bloka koji su definisani opštim arhitekturama (slike 21 i 22), odnosno po 4 rekurentna/konvoluciona sloja. Dužina bafera v za sve RNN i CNN modele iznosila je 16. Broj neurona u prvom potpuno povezanom sloju za sve CNN modele je d_1=30, dok je veličina filtera u konvolucionim slojevima fs=2. Izabrana vrednost stope izostavljanja za sve RNN modele bila je dr=0, osim za model jednostavne RNN u slučaju senzora LIT401 i LSTM model za senzor FIT601 gde je stopa izostavljanja bila dr=0,05.

Tabela 7: Arhitekture kreiranih modela za pet signala iz SWaT skupa podataka

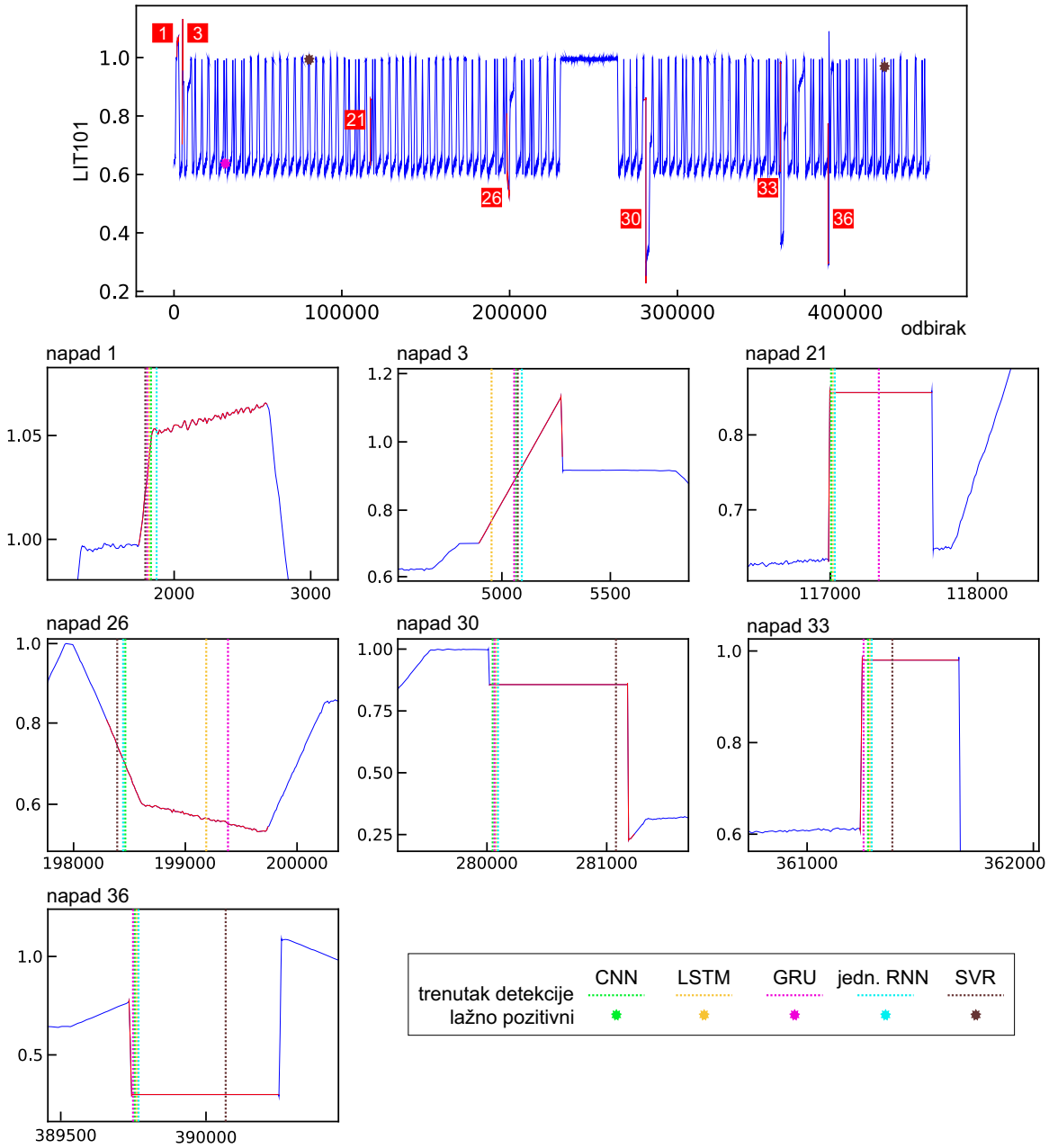
Tehnika \ Signal		LIT101	FIT201	LIT401	PIT501	FIT601
SVR	v, ε, kr, C	v=2, ε=0,005 kr=RBF C=0,160	v=8, ε=0,002 kr = *linearni* C=1	v=4, ε=0,01 kr=RBF C=0,065	v=2, ε=0,001 kr=RBF C=0,011	v=8, ε=0,0005 kr = *linearni* C=1
	br. nos. vekt.	316	792	172	4.042	791
jednostavne RNN	u_1-u_2-u_3-u_4, v, dr	4-4-8-8, v=16, dr=0	4-4-8-8, v=16, dr=0	4-4-32-16, v=16, dr=0,05	4-4-8-8, v=16, dr=0	4-4-16-16, v=16, dr=0
	br. param.	309	309	2.045	309	941
LSTM	u_1-u_2-u_3-u_4, v, dr	4-4-8-8, v=16, dr=0	4-4-8-8, v=16, dr=0	4-4-8-8, v=16, dr=0	4-4-8-8, v=16, dr=0	16-32-8-32, v=16, dr=0,05
	br. param.	1.209	1.209	1.209	1.209	14.017
GRU	u_1-u_2-u_3-u_4, v, dr	4-4-8-8, v=16, dr=0	4-4-8-8, v=16, dr=0	4-4-8-8, v=16, dr=0	4-4-8-8 v=16, dr=0	4-4-8-64 v=16, dr=0
	br. param.	981	981	981	981	14.813
CNN	f_1-f_2-f_3-f_4, v	4-8-8-16, v=16	4-8-8-16, v=16	4-8-8-16, v=16	4-8-8-16, v=16	4-16-16-16, v=16
	br. param.	2.473	2.473	2.473	2.473	3.193

Iz tabele 7 može se primetiti da modeli na bazi jednostavne RNN za tri od pet senzora imaju najmanje parametara. S druge strane, kreirani CNN modeli za tri od pet senzora sadrže najviše obučavajućih parametara, ali je u pogledu implementacije modela u realnom vremenu taj broj i dalje relativno mali. Kada se posmatra računska složenost u pogledu specifičnog signala, modeli sa najviše parametara dobijeni su za senzor FIT601 (izuzev SVR pristupa gde model za PIT501 sadrži najviše nosećih vektora).

Na slikama 32-36 prikazane su performanse detekcije napada na signalima sa senzora LIT101, FIT201, LIT401, PIT501 i FIT601. Na ovim slikama signal bez napada prikazan je plavom, dok je period dejstva napada označen crvenom bojom. Trenuci detekcije napada označeni su vertikalnim isprekidanim linijama različitih boja (dodeljene boje za svaku tehniku označene su na slikama). Markerima različitih boja (raspored boja kao prilikom obeležavanja trenutaka detekcije) označeni su lažno pozitivni rezultati.

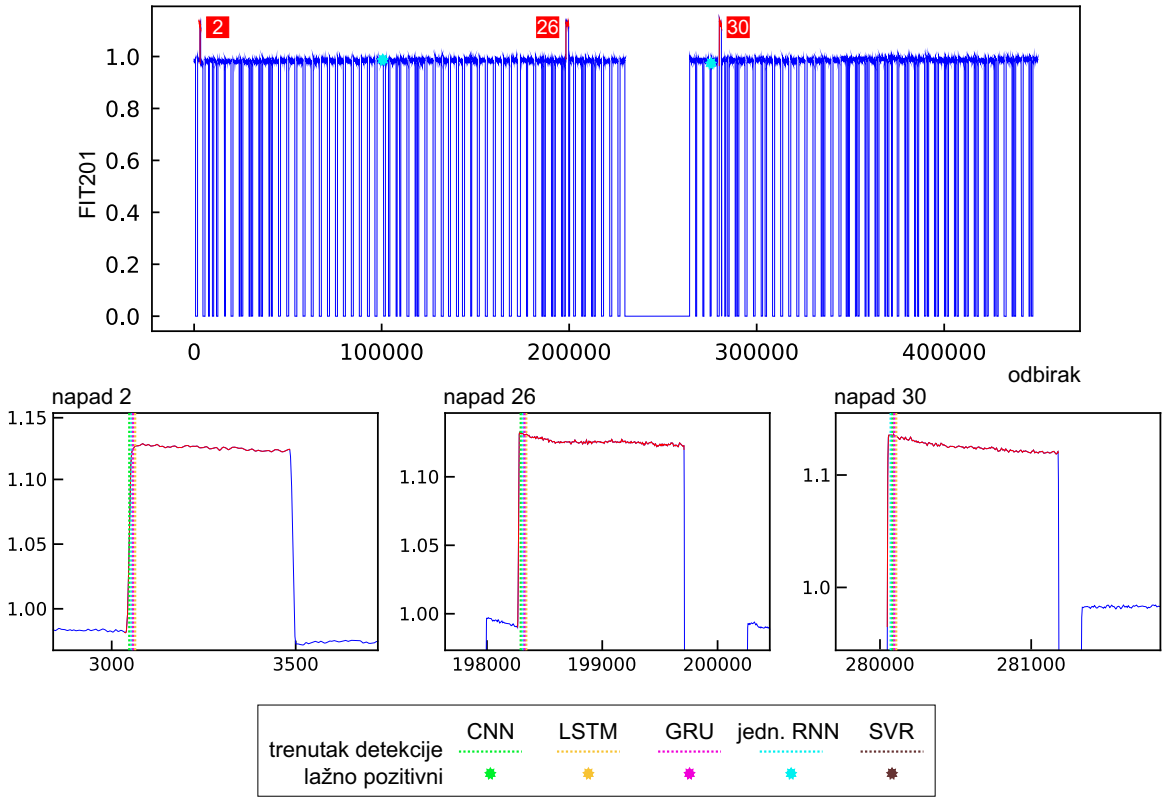
U slučaju signala sa senzora LIT101 korišćenjem DNN tehnika detektovano je pet napada koji direktno ugrožavaju ovaj uređaj: napadi 3, 21, 30, 33 i 36 (slika 32). Osim što direktno utiče na rad senzora LIT101, napad 21 za ciljnu tačku ima i aktuator MV101, dok je napad 30 usmeren i na P101 i MV201. Pored toga, detektovani su i napadi 1 i 26 koji su direktno usmereni na MV101, odnosno P101 i LIT301 jer su ostavili posledice na rad sistema koje su se reflektovale na signale sa LIT101. Primenom SVR modela, izostala je detekcija napada 21, dok je preostalih 6 navedenih napada detektovano. Lažno pozitivni rezultati dobijeni su primenom GRU i SVR modela.

Kada se porede trenuci detekcije korišćenjem različitih modela, primetna su kašnjenja SVR modela u slučaju napada 30, 33 i 36, LSTM modela kod napada 21 i 26 i GRU modela za napad 26. S druge strane, primenom LSTM modela najbrže se detektuje napad 3.



Slika 32: Detekcija napada na signalu sa senzora LIT101

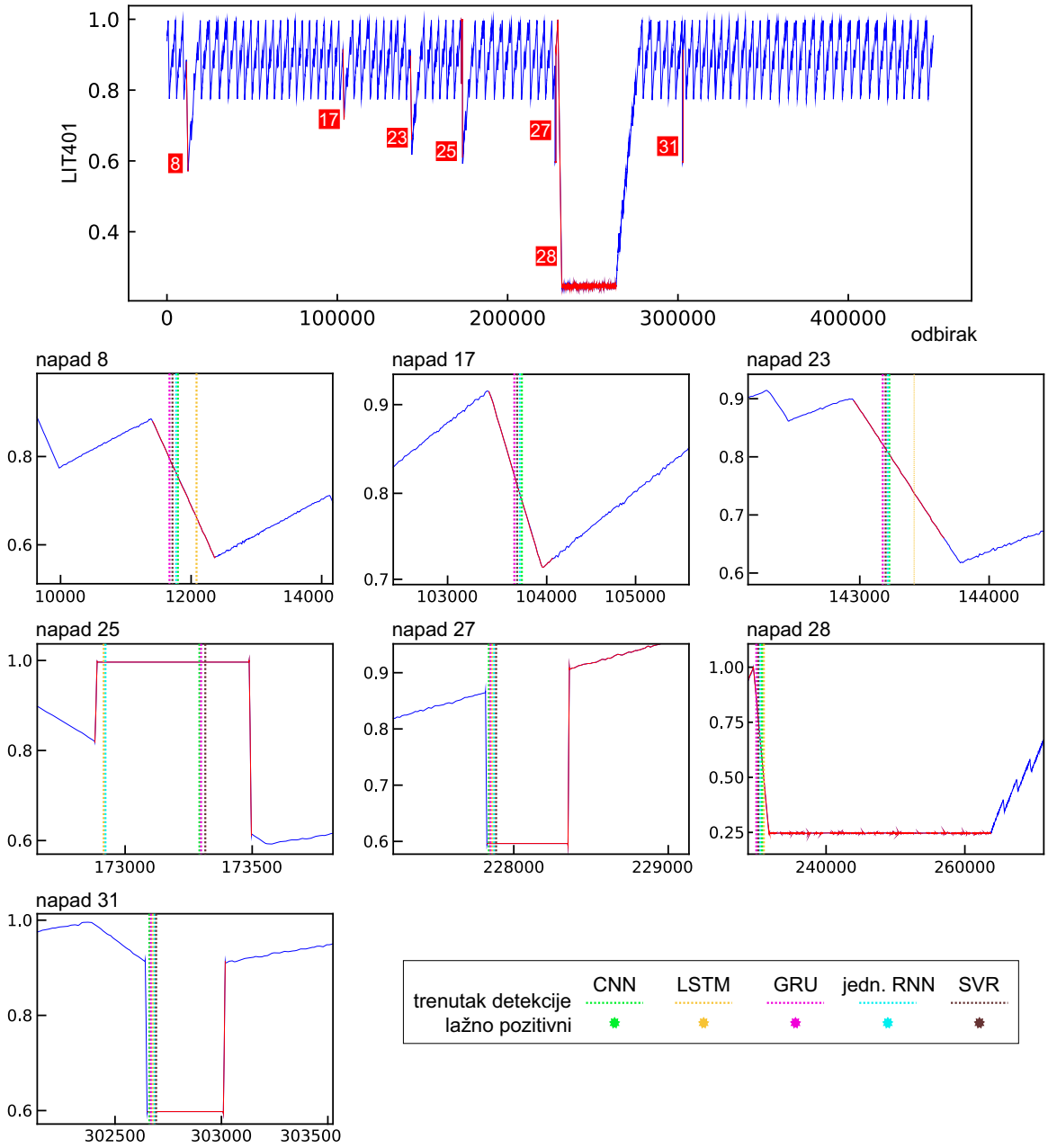
U primeru senzora FIT201 ne postoje napadi koji su direktno bili usmereni na ovaj uređaj. Ipak, korišćenjem DNN zasnovanih tehnika detektovani su napadi 2, 26 i 30, koji su ostvarili indirektno dejstvo (slika 33). Ovim napadima direktno su ugroženi signali sa sledećih uređaja: P102 (napad 2), P101 i LIT301 (napad 26), P101, LIT101 i MV201 (napad 30). Primenom SVR modela u ovom slučaju nije detektovan nijedan napad. Trenuci detekcije napada korišćenjem različitih DNN zasnovanih tehnika bili su približno isti. Primena modela jednostavne RNN prouzrokovala je dva lažno pozitivna rezultata.



Slika 33: Detekcija napada na signalu sa senzora FIT201

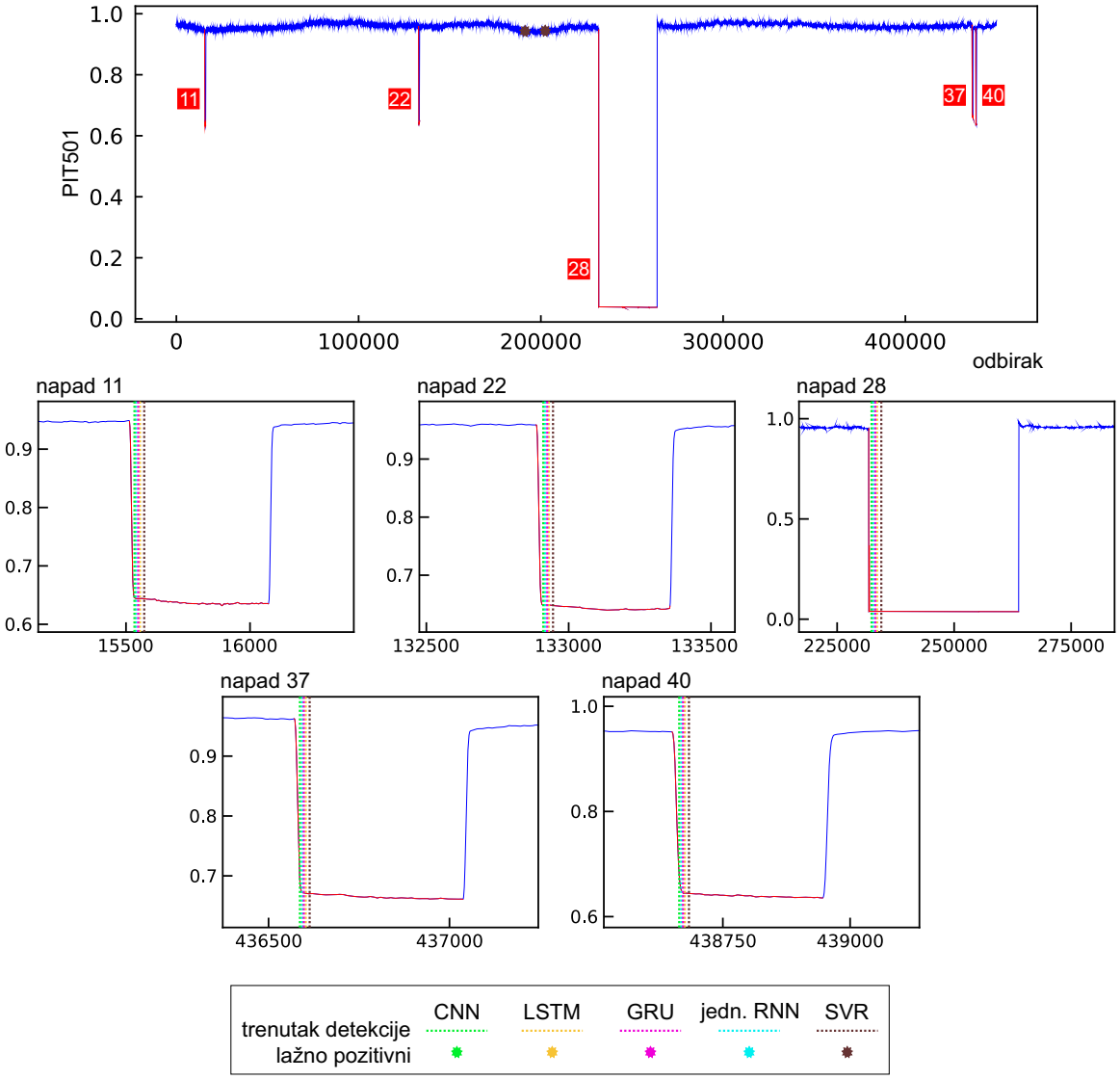
Primena predložene metodologije na LIT401 rezultirala je detekcijom napada 25, 27 i 31 koji su direktno delovali na taj senzor, gde je napad 25 delovao i na aktuator P401, dok je dejstvo napada 27 bilo usmereno i na P302 (slika 34). Pored toga, detektovani su: napad 8 na DPIT301, napad 17 na aktuator MV303, napad 23 na uređaje u sektorima 3 i 6 (senzor DPIT301 i aktuatore MV302 i P602) i napad 28 na aktuator P302.

Kod LSTM modela izostala je detekcija napada 17, dok su ostali modeli uspešno detektovali svih sedam navedenih napada. U slučaju napada 8 i 23, LSTM modelu je bilo potrebno najviše vremena kako bi detektovao napad. S druge strane, značajna razlika u trenutku detekcije ostvarena je i na primeru napada 25, gde su se najbolje pokazali modeli jednostavne RNN i LSTM. U svim ostalim napadima, trenuci detekcije su približno isti.



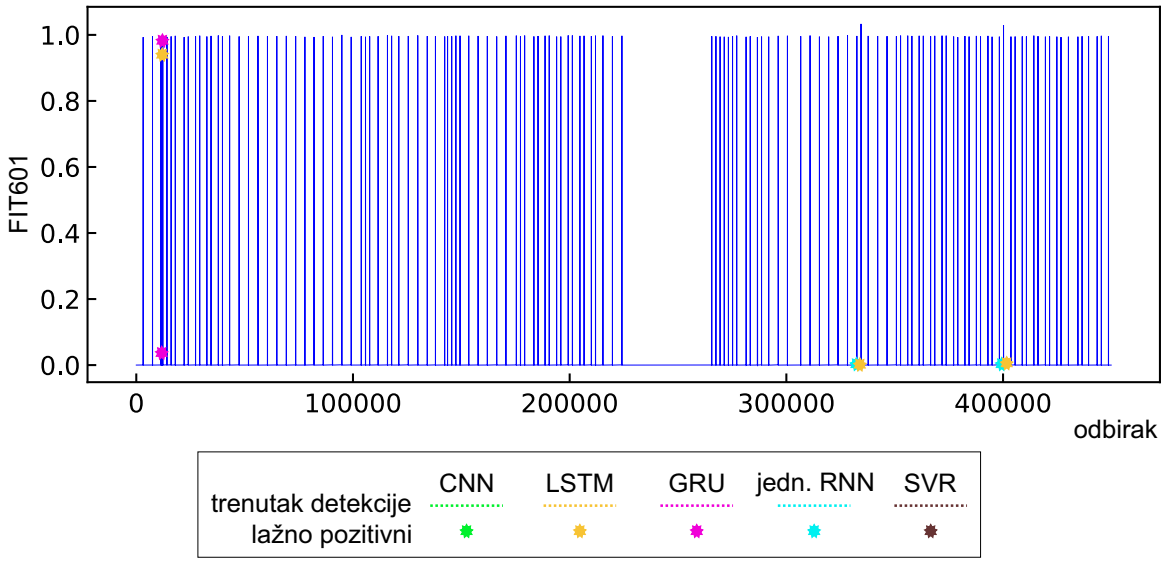
Slika 34: Detekcija napada na signalu sa senzora LIT401

Kada se posmatra signal prikupljen sa senzora PIT501 (slika 35), svi kreirani modeli detektovali su pet napada koji su imali indirektno dejstvo: napad 11 (na FIT401), napad 22 (na UV401, AIT502 i P501), napad 28 (na P302), napad 37 (na P501 i FIT502) i napad 40 (na FIT 401). Dva lažna rezultata dobijena su primenom SVR modela. Kada se porede momenti detekcije različitim pristupima, nisu primetne značajne razlike.



Slika 35: Detekcija napada na signalu sa senzora PIT501

Signal sa senzora FIT601 koji nije bio predmet nijednog napada, uključen je u analizu kako bi se ispitala mogućnost razmatranih tehnika da modeliraju signal sa dinamikom koju karakterišu nagli i učestali prelazi između dve vrednosti (slika 36). U ovom slučaju nije bilo detektovanih napada, ali se ispostavilo da je kod svih RNN pristupa bilo minimum 2 lažno pozitivna rezultata (ukupno 7 lažno pozitivnih rezultata).



Slika 36: Detekcija napada na signalu sa senzora FIT601

Sprovedenom analizom na 5 senzora detektovano je ukupno 20 napada[6], od čega je 8 napada imalo za ciljnu tačku neki od razmatranih senzora, dok je ostalih 14 detekcija predstavljalo uticaj napada koji su kao metu imali neki od susednih uređaja. Kao metrika za merenje performansi predloženih pristupa, korišćen je F_1 skor po događaju. U tabeli 8 za svaki ML model (ukupno 25) prikazane su vrednosti parametara (tp, lp i ln) koji se koriste za izračunavanje F_1 skora.

Tabela 8: Komparativna analiza detekcije napada razmatranim tehnikama

Signal / Tehnika	LIT101			FIT201			LIT401			PIT501			FIT601			Ukupno		
	tp	lp	ln	tp	lp	ln	tp	lp	ln	tp	lp	ln	tp	lp	ln	tp	lp	ln
SVR	6	2	1	0	0	0	7	0	0	5	2	0	0	0	0	18	4	4
jednostavne RNN	7	0	0	3	2	0	7	0	0	5	0	0	0	2	0	22	4	0
LSTM	7	0	0	3	0	0	6	0	1	5	0	0	0	3	0	21	3	1
GRU	7	1	0	3	0	0	7	0	0	5	0	0	0	2	0	22	3	0
CNN	7	0	0	3	0	0	7	0	0	5	0	0	0	0	0	22	0	0

Tabela 9 prikazuje *preciznost*, *odziv* i F_1 skor za sistem u celini za svaku od razmatranih tehnika mašinskog učenja.

Tabela 9: Poređenje rezultata primene razmatranih tehnika za detekciju napada korišćenjem F_1 skora po napadu

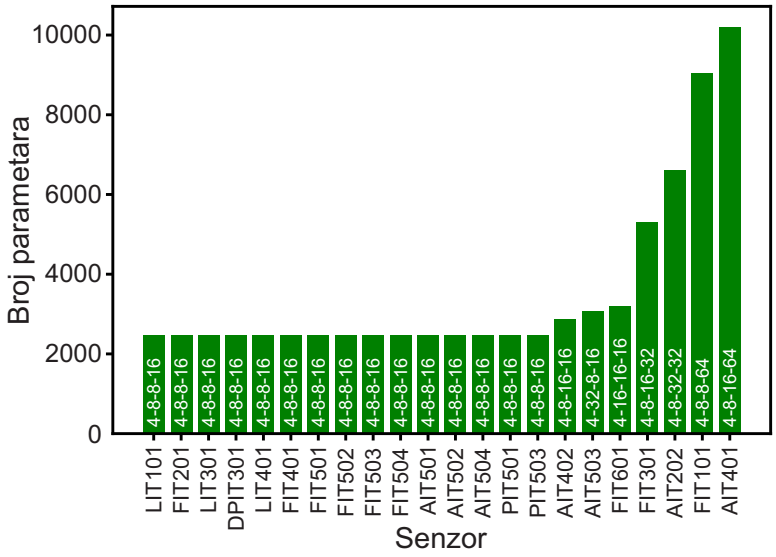
Metod	**prec.**	**odziv**	**F_1**
SVR	0,818	0,818	0,818
jednostavne RNN	0,846	1	0,917
LSTM	0,875	0,954	0,913
GRU	0,880	1	0,936
CNN	1	1	1

Kako je najniža ostvarena vrednost F_1 skora 0,818 u slučaju SVR pristupa, može se reći da su sve tehnike dale zadovoljavajuće rezultate. Odlični rezultati dobijeni su primenom RNN tehnika, gde je korišćenjem jednostavne RNN, LSTM i GRU arhitektura postignuta vrednost F_1 preko 0,9. Ipak, na datom skupu signala, najbolji učinak ostvaren je CNN modelima: maksimalne

[6]Na nekim senzorima pojedini napadi se ponavljaju.

vrednosti za *preciznost* (bez lažno pozitivnih rezultata) i *odziv* (22/22 detektovanih napada), samim tim i maksimalna vrednost F_1 skora. Ovi rezultati pokazuju da je korišćenjem ove tehnike moguće modelirati ponašanje sistema sa visokom tačnošću uključujući dobro svojstvo generalizacije, što se kasnije uspešno koristi u procesu detekcije napada. Iz tog razloga, ova tehnika je korišćena u nastavku za generisanje modela svih odabranih senzorskih signala iz SWaT skupa podataka, kao i u detaljnoj analizi i poređenju sa postojećim pristupima.

Prethodnom analizom utvrđeno je da su najbolje performanse detekcije ostvarene modelima na bazi CNN tehnike pa će u nastavku ova tehnika biti primenjena za modeliranje signala sa svih senzora. Postupak pretprocesiranja obuhvatio je primenu istog niskopropusnog FIR filtera koji je razvijen i korišćen u prethodnoj analizi za pet signala iz SWaT skupa podataka. Predloženom metodologijom za svaki od razmatranih senzora (ukupno 22) kreiran je CNN model. Arhitekture svih modela sastoje se iz dva konvoluciona bloka (c=2) sa veličinom filtera fs=2 u svim konvolucionim slojevima, dužina bafera je v=16, dok prvi potpuno povezani sloj sadrži 30 neurona. S druge strane, razliku između arhitektura čini broj filtera u konvolucionim slojevima 1-4 čije su vrednosti prikazane na slici 37 u formatu f_1-f_2-f_3-f_4.



Slika 37: Broj parametara modela različitih senzora. Broj filtera u CNN slojevima predstavljen je u formatu f_1-f_2-f_3-f_4

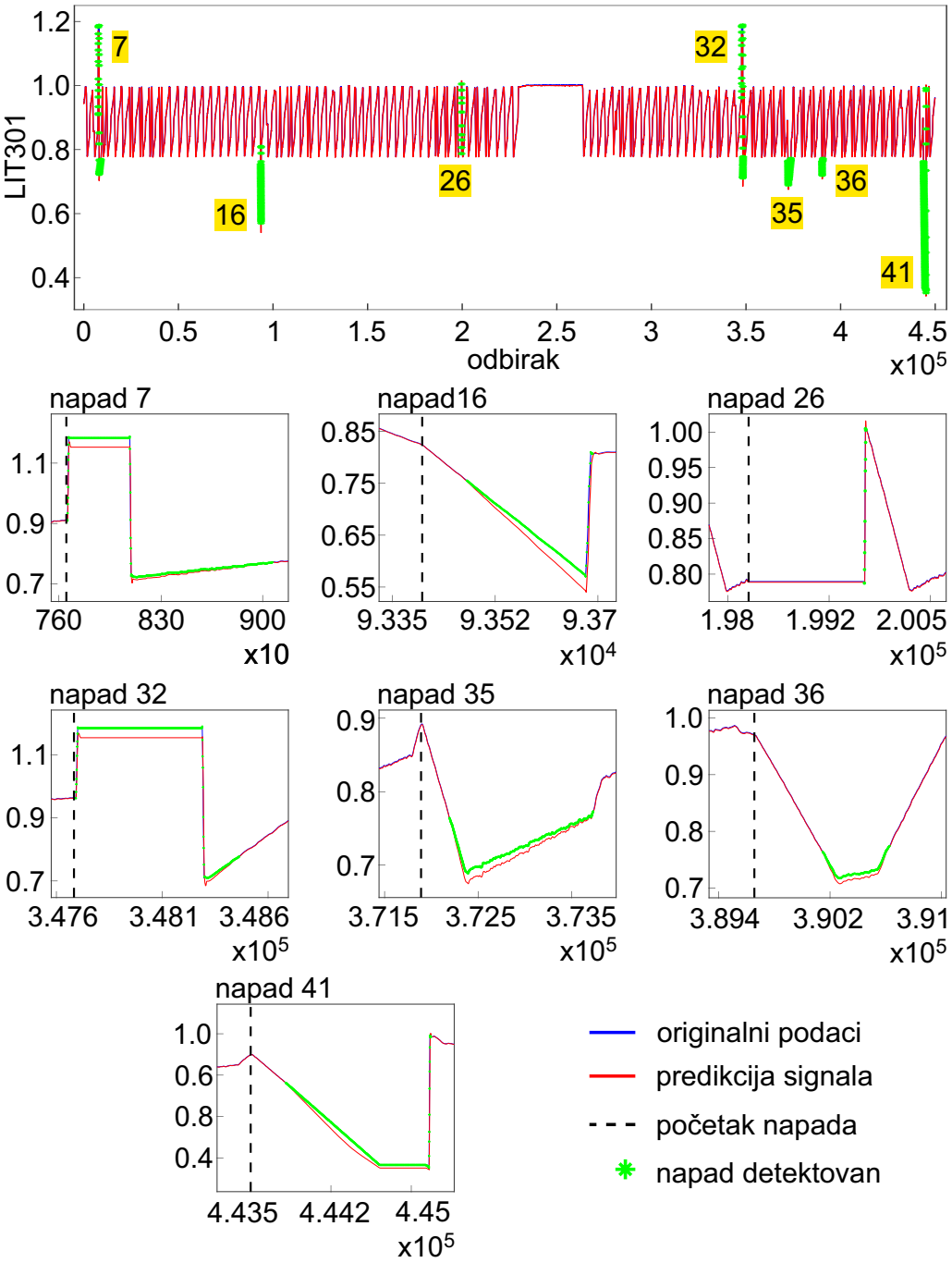
Pored broja filtera, na slici 37 prikazan je i broj parametara CNN modela razvijenih za različite senzore. Broj parametara modela u konkretnom slučaju definisan je brojem filtera u konvolucionim slojevima, veličinom matrica težinskih koeficijenata (koje su određene veličinom filtera), kao i brojem neurona u potpuno povezanom sloju. Modeli najmanje kompleksnosti (2.473 parametara) generisani su za 15 senzora (LIT101, FIT201, LIT301 itd.), dok je model sa najvećim brojem parametara (10.209) kreiran za senzor AIT401.

Algoritmima za detekciju napada koji su dobijeni predloženom metodologijom detektovano je ukupno 30 napada. U tabeli 10 navedeni su svi detektovani napadi sa pojedinačnim opisom, uključujući i spisak senzora na kojima je izvršena detekcija. Senzori na čijim signalima su detektovani napadi sa direktnim dejstvom (na te senzore) označeni su podebljanim slovima (tabela 10). Kao što je i očekivano, nisu detektovani napadi koji su prilikom kreiranja SWaT skupa podataka [51] neuspešno izvedeni (napadi 13, 14 i 29) ili napadi čije dejstvo nije prouzrokovalo nikakav uticaj na sistem (napadi 4 i 34). Pored toga, napad 24 koji nije detektovan imao je zanemarljivo mali uticaj na proces [51]. Napad 24 usmeren je na pumpe za doziranje NaOCl i NaCl (P203 i P205) i njegov uticaj (kada je prisutan) može se posmatrati samo na signalu sa AIT203 senzora koji meri nivo NaOCl, a koji je isključen iz razmatranja kao irelevantan zbog nemogućnosti da se stabilizuje nakon dejstva napada (slika 29c).

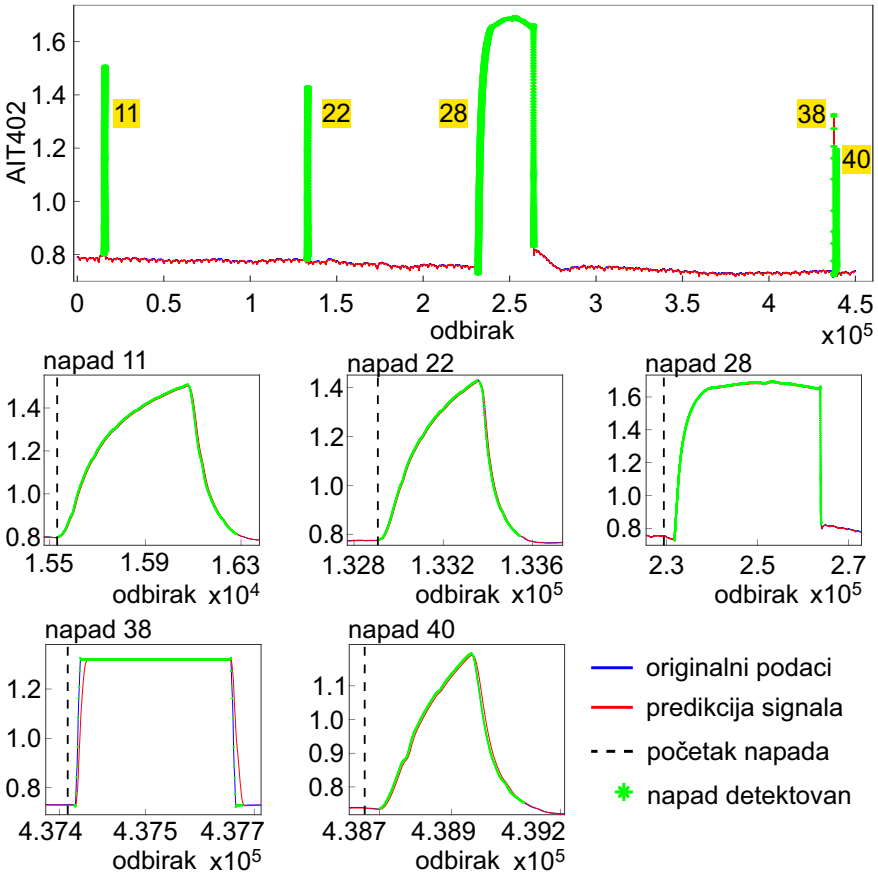
Tabela 10: Pregled svih detektovanih napada iz SWaT skupa podataka

Napad	**Opis** [51]	**Sektor**	**Senzor na kojem je detektovan napad**
1	Otvoriti MV101	P1	LIT101
2	Uključiti P102	P1	FIT201
3	Povećati nivo vode za 1 mm/s	P1	**LIT101**
6	Postaviti vrednost AIT202 na 6	P2	**AIT202**
7	Nivo vode povećan iznad HH	P3	**LIT301**
8	Postaviti vrednost DPIT na $>$ 40 kPa	P3	**DPIT301**, LIT401
10	Postaviti vrednost FIT401 na $<$ 0.7	P4	**FIT401**
11	Postaviti vrednost FIT401 na 0	P4	**FIT401**, AIT402, FIT501, FIT502, FIT503, FIT504, AIT501, AIT502, AIT504, PIT501, PIT503
16	Smanjiti nivo vode za 1 mm/s	P3	**LIT301**
17	Neprekidno držati MV303 zatvoren	P3	LIT401
19	Postaviti vrednost AIT504 na 16 uS/cm	P5	**AIT504**
20	Postaviti vrednost AIT504 na 255 uS/cm	P5	**AIT504**
21	Neprekidno držati MV101 otvoren; Postaviti vrednost LIT101 na 700 mm	P1	**LIT101**
22	Zaustaviti UV401; Postaviti vrednost AIT502 na 150; Zadržati P501 da ostane uključen	P4, P5	**AIT502**, FIT401, AIT402, FIT501, FIT502, FIT503, FIT504, AIT501, AIT504, PIT501, PIT503
23	Postaviti vrednost DPIT301 na $>$ 0.4 bar; Zadržati MV302 otvoren i P602 zatvoren;	P3, P6	**DPIT301**, LIT401
25	Postaviti vrednost LIT401 na 1000; držati P402 uključen	P4	**LIT401**
26	Neprekidno držati P101 uključen;	P1, P3	**LIT301**, LIT101, FIT201
27	Neprekidno držati P302 uključen; Postaviti vrednost LIT401 na 600 mm Postaviti vrednost LIT301 na 801 mm	P3, P4	**LIT401**
28	Zatvoriti P302	P3	DPIT301, FIT401, AIT402, FIT501, FIT502, FIT503, FIT504, AIT501, AIT502, AIT504, PIT501, PIT503
30	Neprekidno držati P101 i MV101 uključene; Postaviti vrednost LIT101 na 700 mm; P102 se pokreće kada je nivo LIT301 nizak	P1, P2	**LIT101**, FIT201
31	Postaviti vrednost LIT401 ispod L	P4	**LIT401**
32	Postaviti vrednost LIT301 iznad HH	P3	**LIT301**
33	Postaviti vrednost LIT101 iznad H	P1	**LIT101**
35	Isključiti P101; Držati P102 isključen	P1	LIT301
36	Postaviti LIT101 ispod LL	P1	**LIT101**, LIT301
37	Zatvoriti P-501; Postaviti vrednost FIT502 na 1.29	P5	**FIT502**, FIT401, FIT501, FIT503, FIT504, AIT502, AIT504, PIT501, PIT503
38	Postaviti vrednost AIT402 na 260; Postaviti vrednost AIT502 na 260	P4, P5	**AIT402, AIT502**
39	Postaviti vrednost FIT401 na 0.5; Postaviti vrednost AIT502 na 140 mV	P4, P5	**FIT401**
40	Postaviti vrednost FIT401 na 0	P4	**FIT401**, AIT402, FIT501, FIT502, FIT503, FIT504, AIT501, AIT502, AIT504, PIT501, PIT503
41	Smanjiti vrednost LIT301 za 0.5 mm/s	P3	**LIT301**

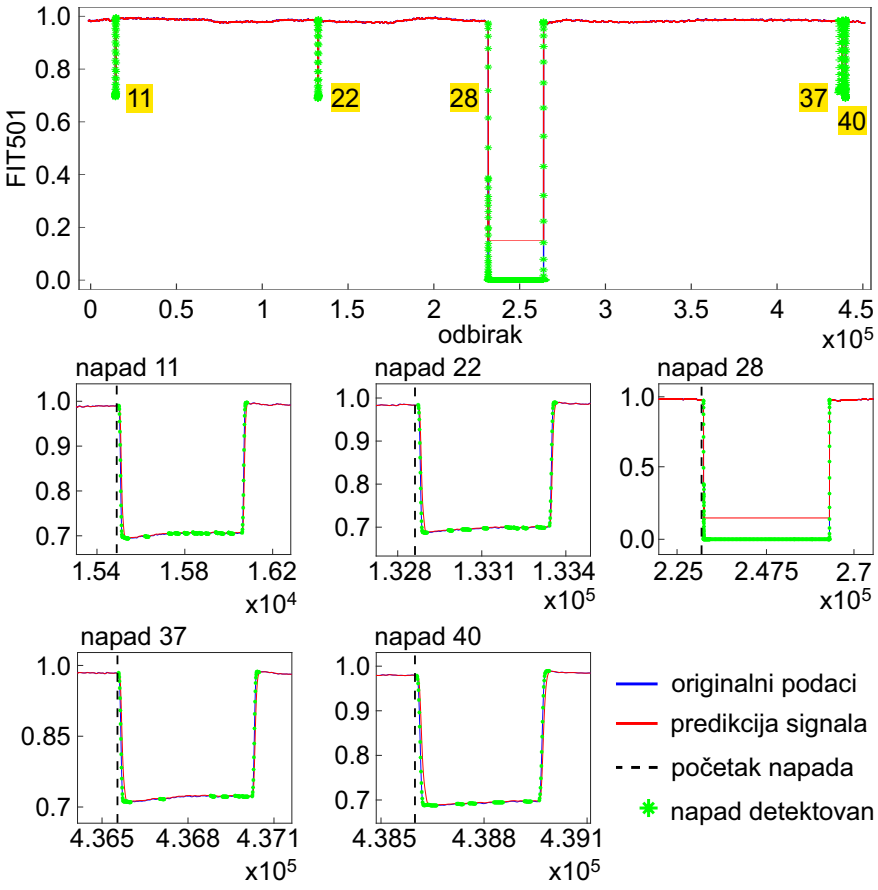
Na slikama 38, 39 i 40 prikazane su performanse onlajn detekcije na senzorima iz tri različite klase LIT, AIT i FIT konkretno za: senzor nivoa LIT301, ORP (engl. *Oxidation Reduction Potential* – ORP) senzor AIT402 i senzor protoka FIT501. Na ovim slikama originalni signal tokom napada prikazan je plavom, vrednosti njegove predikcije crvenom linijom, početak napada vertikalnom crnom isprekidanom linijom, dok su momenti detekcije napada označeni zelenim markerom. Koristeći signale ova tri senzora, detektovano je ukupno 12 napada (neki napadi se ponavljaju). Kada se posmatra signal sa senzora LIT301, pored napada 7, 16, 26, 32 i 41 koji direktno ugrožavaju LIT301, detektovani su i napad 36 na LIT101, odnosno napad 35 na P101 i P102. Primena predložene metode na AIT402, rezultirala je detekcijom napada 11 i 40 koji su direktno delovali na FIT401, napada 28 na P302, napada 38 na senzore u sektorima 4 i 5 (AIT402 i AIT502) i napada 22 sa direktnim dejstvom na tri uređaja (UV401, AIT502 i P501) u dva različita sektora. Iako FIT501 nije bio direktna meta nijednog napada, na signalu sa ovog senzora detektovana su četiri ista napada kao i u slučaju senzora AIT402: napadi 11, 22, 28 i 40. Pored toga, na signalu sa senzora FIT501 detektovan je i napad 37 koji je bio direktno usmeren na P501 i FIT502. Prikazani rezultati jasno pokazuju da razvijeni algoritmi uspešno detektuju napade na aktuatore koristeći samo signale dobijene sa senzora.



Slika 38: Detektovani napadi na signalu sa senzora nivoa LIT301



Slika 39: Detektovani napadi na signalu sa ORP senzora AIT402



Slika 40: Detektovani napadi na signalu sa senzora protoka FIT501

U cilju sveobuhvatne analize i poređenja sa postojećim pristupima, u nastavku će se koristiti sve tri razmatrane metrike: F_1 skor, *tačnost* i *FPR* definisane izrazima (37), (40) i (41), tim redom. Treba napomenuti da se u industrijskim sistemima prilikom detekcije napada, već nakon prvog odbirka aktiviraju određene akcije u cilju sprečavanja daljih posledica koje napad može prouzrokovati. U tom kontekstu, veoma je važno svesti na minimum broj lažno pozitivnih detekcija napada, odnosno izbeći učestale prekide normalnog funkcionisanja sistema. Iz navedenih razloga, F_1 skor po događaju uzima se kao relevantniji pristup procene performansi. Međutim, *tačnost* i *FPR* se ne mogu računati po događaju, jer ne postoji način da se izračuna broj *tn* događaja (ispravno okarakterisanih ponašanja sistema u normalnom radu).

U tabeli 11 prikazan je F_1 skor po događaju za sve sektore, gde su redni brojevi uspešno detektovanih napada podvučeni. Neki napadi (npr. napadi 22 i 23) koji su kao direktan cilj imali uređaje u više sektora, navedeni su u više odgovarajućih vrsta. Primenom predloženog metoda za detekciju napada, samo dva događaja su prepoznata kao lažno pozitivni, jedan na senzoru diferencijalnog pritiska DPIT301, a drugi na ORP senzoru AIT502. Ukupni F_1 skor po događaju je 0,882, gde je ostvarena *preciznost* od 0,938 i *odziv* od 0,833. Kako F_1 skor u tabeli 11 uključuje u razmatranje i napade koji su neuspešni ili su prouzrokovali zanemarljivo mali uticaj na sistem, da bi se stekao bolji uvid u performanse sistema tabela 12 prikazuje F_1 skor izračunat na bazi događaja ne uzimajući u obzir napade 4, 13, 14, 29 i 34. Prema tabeli 12, ukupni F_1 skor je 0,953, *preciznost* je 0,938, dok je postignut *odziv* od 0,968.

Tabela 11: F_1 skor po događaju uključujući i neuspele napade

Sektor	**Napadi**	**lp**	**prec.**	**odziv**	**F_1**
P1	1, 2, 3, 21, 26, 30, 33, 34, 35, 36	0	1	0,9	0,947
P2	6, 24, 29, 30	0	1	0,5	0,667
P3	7, 8, 13, 14, 16, 17, 23, 26, 27, 28, 32, 41	1	0,909	0,833	0,869
P4	10, 11, 22, 25, 27, 31, 38, 39, 40	0	1	1	1
P5	4, 19, 20, 22, 37, 38, 39	1	0,857	0,857	0,857
P6	23	0	1	1	1
Ukupno (P1-P6)		2	**0,938**	**0,833**	**0,882**

Tabela 12: F_1 skor po događaju (neuspeli napadi nisu razmatrani)

Sektor	**lp**	**tp**	**ln**	**prec.**	**odziv**	**F_1**
P1	0	9	0	1	1	1
P2	0	2	1	1	0,667	0,8
P3	1	10	0	0,909	1	0,952
P4	0	9	0	1	1	1
P5	1	6	0	0,857	1	0,923
P6	0	1	0	1	1	1
Ukupno	2	30	1	**0,938**	**0,968**	**0,953**

5.1.2. Uporedna analiza razvijenog IDS-a sa postojećim pristupima

Performanse dobijene upotrebom predložene metodologije upoređene su sa prethodno razvijenim pristupima baziranim na samonadgledanom učenju, a koji su razmatrali SWaT skup podataka. Pojedini pristupi nisu uključeni u ovu komparativnu analizu zbog nedovoljnog broja signala na kojima je izvršena evaluacija, kao npr. [4] gde su razmatrana samo dva signala ili

[37] i [100] gde su obuhvaćeni samo signali iz prvog sektora SWaT postrojenja. U tabeli 13 je prikazan *odziv* i broj detektovanih napada kod različitih pristupa. U većini radova koji opisuju metode detekcije napada na SWaT nije prikazan *lp* po događaju pa nije moguće izračunati F_1 po događaju i uporediti pristup iz ove doktorske disertacije sa drugim pristupima koristeći ovaj (najrelevantniji) kriterijum. Iz tabele 13 može se uočiti da je predloženi IDS po kriterijumima broja detektovanih napada i *odziva* po događaju dao bolje rezultate od ostalih pristupa, izuzimajući [59] gde je detektovan 31 napad. U [28] navodi se da je detektovano 30 napada, međutim u taj skup spadaju i napadi 14 i 29 čije izvršavanje tokom kreiranja skupa podataka nije uspelo, stoga nisu mogli ostaviti nikakav uticaj na sistem.

Iako je izračunavanje F_1 skora po odbirku donekle irelevantno zbog predubeđenja definisanog dužinom napada (uspešna detekcija dugih napada može potpuno maskirati neuspešnu detekciju kraćih napada) i nije u skladu sa funkcionisanjem IDS-a u realnim aplikacijama, u tabeli 14 prikazani su rezultati poređenja metoda za detekciju korišćenjem ovog kriterijuma. Naime, ostali autori su u svojim radovima prikazivali samo metriku po odbirku i samo ova metrika je na raspolaganju za poređenje. U tabeli 14, vrednosti za *preciznost*, *odziv* i F_1 skor preuzete su iz navedenih literaturnih izvora. Međutim, kako su na raspolaganju *preciznost*, *odziv* i ukupni broj odbiraka u signalima sa napadima, moguće je izračunati *tačnost* i *FPR* za navedene metode. Naime, dužina signala sa napadima koji su korišćeni za proveru performansi u svim razmatranim pristupima je 449.919 odbiraka, što predstavlja sumu svih kategorija ($tp+tn+lp+ln$). S druge strane, poznato je da je ukupno trajanje svih 36 napada u okviru ovih signala 53.850 odbiraka što odgovara sumi tačno pozitivnih i lažno negativnih ($tp+ln$). Na osnovu tih vrednosti, kao i *preciznosti* i *odziva* datih u tabeli 14 i izraza definisanih u (38) i (39), moguće je izračunati tp, tn, lp i ln, a shodno tome i *tačnost* i *FPR* (definisani izrazima (40) i (41)). Vrednosti za *tačnost* i *FPR* takođe su prikazane u tabeli 14. Iz ove tabele može se primetiti da je metodologija predložena u ovoj doktorskoj disertaciji nadmašila ostale pristupe sa F_1 skorom po odbirku od 0,902 pri čemu treba napomenuti da je F_1 skor računat za svih 36 napada. Takođe, i u slučaju ostale dve metrike ovde predloženi pristup dao je najbolje rezultate ostvarivši *tačnost* od 97,846% i *FPR* od 0,135%. Posebno treba naglasiti da su u ostalim pristupima za razvoj IDS-a korišćeni signali sa senzora i aktuatora, dok su u predloženom pristupu korišćeni samo signali sa senzora (ukupno 22).

Tabela 13: Poređenje rezultata (po događaju) mehanizama za detekciju napada

Tehnika	**odziv**	**Broj detekt. napada**
NN-SVM [7]	0,806	29
1D CNN-IF [27]	0,722	26
DIF [28]	0,833	30
OC-SVM [46]	0,556	20
LSTM RNN [46]	0,361	13
1D CNN [59]	**0,861**	**31**
TABOR [68]	0,667	24
1D CNN [98]	0,639	23
MLP [104]	0,694	25
Nepotpuni autoenkoderi [124]	0,722	26
Predloženi pristup – 1D CNN	0,833	30

Tabela 14: Poređenje rezultata mehanizama za detekciju napada korišćenjem F_1 skora po odbirku, *tačnosti* i *FPR*

Tehnika	**prec.**	**odziv**	**F_1**	**tačn.(%)**	**FPR(%)**
NN-SVM [7]	0,940	0,820	0,876	97,219	0,712
DIF [28]	0,935	0,835	0,882	97,375	0,738
OC-SVM [46]	0,925	0,699	0,796	95,770	0,713
LSTM RNN [46]	0,983	0,678	0,803	96,008	0,157
1D CNN [59]	0,968	0,791	0,871	97,195	0,344
UAE Frequency [60]	0,911	0,860	0,885	97,408	1,041
TABOR [68]	0,862	0,788	0,823	96,161	1,479
LSTM RNN [98]	0,984	0,750	0,851	94,634	0,166
MLP [104]	0,967	0,696	0,812	96,087	0,312
Nepotpuni autoenkoderi [124]	0,856	0,885	0,870	96,840	2,024
NN [132]	0,967	0,725	0,828	96,422	0,325
Predloženi pristup – 1D CNN	0,988	0,830	**0,902**	**97,846**	**0,135**

Iz prikazane analize može se primetiti da su pristupi bazirani na CNN – sistem za detekciju napada predložen u ovoj doktorskoj disertaciji i [59] dali najbolje rezultate kada je broj detektovanih napada u pitanju. U slučaju F_1 skora po odbirku, ovde predloženi IDS nadmašio je IDS iz [59] sa 0,902 u poređenju sa 0,871. Pored toga, tačnost predložene metodologije – 97,846% veća je nego u [59] gde je ostvaren rezultat od 97,195%. Konačno, *FPR* od 0,135% dobijen primenom predložene metodologije značajno je niži od vrednosti 0,344% postignute u [59] što je izuzetno važno za ICS kod kojih prekid rada sistema usled lažno detektovanog napada može dovesti do nepotrebnog zaustavljanja procesa i nepotrebnih gubitaka.

S druge strane, u pristupu iz [59] detektovan je 31 napad, dok je predloženim pristupom detektovano 30, gde su u oba pristupa detektovani isti napadi sa dva izuzetka. Naime, primenom pristupa zasnovanog na CNN-u [59] detektovan je napad 24 koji je imao zanemarljiv uticaj na sistem, dok je pristupom predloženim u okviru ove doktorske disertacije taj napad bilo nemoguće otkriti zbog izostavljanja (razlozi su prethodno objašnjeni) senzora AIT203. Drugu razliku između ova dva pristupa čine napad 34 (prema [51] nije imao uticaja na sistem) koji je detektovan metodom iz [59], dok napad 35 koji je prouzrokovao očekivani uticaj na sistem pristupom iz [59] nije detekovan. Suprotno tome, metodologijom koja je predložena u ovoj doktorskoj disertaciji napad 34 nije detektovan, dok je napad 35 uspešno detektovan. Stoga, ukoliko se isključe iz razmatranja napadi koji nisu izazvali nikakav uticaj na sistem (između ostalog i napad 34) i uzmu u obzir samo napadi koji su prouzrokovali određene promene, može se zaključiti da su oba IDS-a detektovala isti broj napada.

Potrebno je naglasiti da mehanizam za detekciju napada iz [59] ima bitne nedostatke u odnosu na predloženu metodologiju. Naime, u [59] korišćen je F_1 skor da optimizuje IDS hiperparametre, konkretno dužinu bafera v kao i vrednost praga detekcije T. Kako proračun F_1 skora zahteva upotrebu signala sa napadima, ne može se smatrati da metod iz [59] spada u kategoriju samonadgledanog učenja. U ovom slučaju, CNN model je kreiran na osnovu vrednosti signala dobijenih tokom normalnog rada sistema, što odgovara samonadgledanom učenju, ali su vrednosti hiperparametara optimizovane korišćenjem signala sa napadima kao u nadgledanim pristupima. S druge strane, u pristupu predloženom u ovoj doktorskoj disertaciji svi hiperparametri dobijeni su samo na osnovu signala prikupljenih tokom normalnog rada sistema, odakle se može zaključiti da kreirani IDS nema predubeđenja koja potiču od napada prisutnih u odgovarajućim signalima.

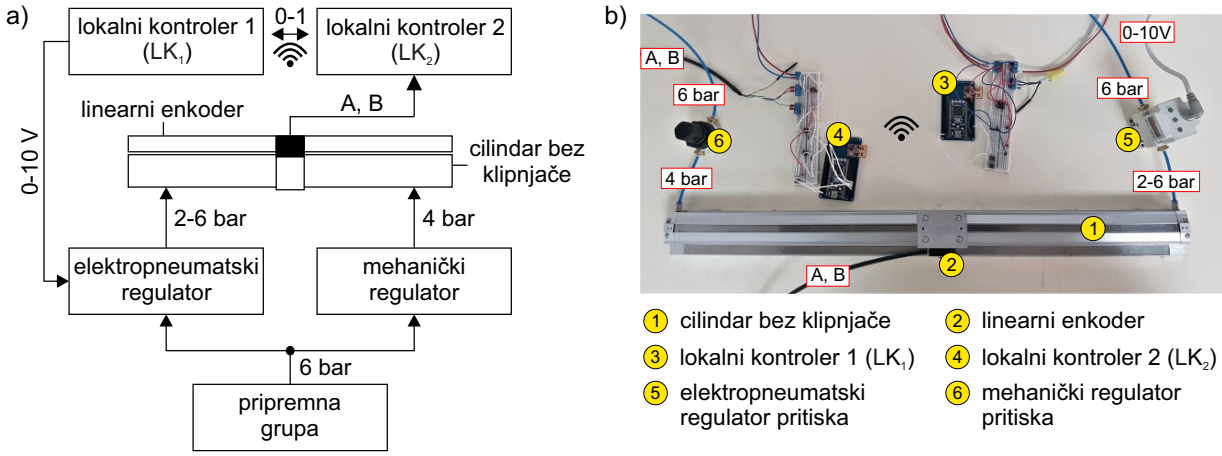
Još jedna prednost algoritama predloženih ovom doktorskom disertacijom u poređenju sa pristupom iz [59] ogleda se u niskoj računskoj složenosti i malom kašnjenju koje je pritom prouzrokovano. Kako je krajnji cilj razvoja IDS-a implementacija u ICS, njihova primenljivost

u realnom vremenu i niska računska složenost kao neophodan uslov za primenu na energetski ograničenim uređajima (koji su prisutni u ICS) su od ključnog značaja. U ovom pristupu, za razvoj svih CNN modela korišćena je veličina ulaznog bafera v od 16 odbiraka, dužina FIR filtera od 11 odbiraka, što ukupno dovodi do kašnjenja od 27 odbiraka. U pristupu [59] korišćena dužina bafera bila je u opsegu od 50 do 300 odbiraka, što dalje prouzrokuje znatno veće kašnjenje. Pored veličine bafera, računska složenost koja je direktno proporcionalna kompleksnosti ML modela, može uvesti dodatna kašnjenja u zavisnosti od proračunskih kapaciteta uređaja na koji se algoritam implementira, kao i od frekvencije odabiranja. CNN modeli kreirani u [59] sadrže 8 konvolucionih slojeva sa po 32 filtera, gde svaki filter sadrži po 2 koeficijenta što je rezultiralo veličinom modela od 1.585 KB. S druge strane, veličina modela kreirana u ovde predloženom pristupu figuriše u rasponu od CNN arhitekture sa 4 konvoluciona sloja sa 4, 8, 8 i 16 filtera sa po 2 koeficijenta, gde je ukupna veličina modela 89 KB do CNN arhitekture sa 4 sloja sa 4, 8, 16 i 64 filtera, takođe sa po 2 koeficijenta i veličinom modela od 177 KB; srednja veličina modela je 102,7 KB. Kada se uporede rezultati, jasno je da je računska složenost ovde kreiranih modela višestruko manja nego u slučaju modela generisanih u [59].

Iz prikazane analize dolazi se do zanimljivog zaključka da su dve metode zasnovane upravo na CNN arhitekturi (pristup predložen u ovoj doktorskoj disertaciji i [59]) dale najbolje rezultate u pogledu broja detektovanih napada. Jedan od razloga je u tome što CNN kao klasa DNN metoda ima sposobnost da inkrementalno uči obeležja iz podataka. Pored toga, kada se uporedi sa ostalim DNN pristupima kao što je RNN, CNN ima značajne prednosti za razmatrane studije slučaja. Ako se na slici 29 posmatraju signali sa senzora prikupljeni tokom normalnog rada sistema (plava), moguće je primetiti da raspodela podataka nije ista duž vremenske ose. Stoga, ukoliko se izabere uzastopno npr. 80% podataka za obučavanje/validaciju i 20% za izbor modela, moguće je da raspodela pomenutih podskupova podataka neće odgovarati raspodeli celog signala, što može prouzrokovati lošije performanse. Kao što je prethodno rečeno, suprotno pristupima koji koriste RNN, CNN ima mogućnost mešanja podataka, čime se postiže približno ista raspodela u delu za obučavanje/validaciju i izbor modela i rešava prethodno navedeni problem.

5.2. Studija slučaja 2 - Elektropneumatski sistem za pozicioniranje

Upotrebljivost razvijene metodologije u realnim uslovima testirana je na Elektropneumatskom sistemu za pozicioniranje (EpSP) koji je razvijen u Laboratoriji za automatizaciju proizvodnje na Mašinskom fakultetu. Ovaj sistem je baziran na interakciji između pametnog aktuatora i pametnog senzora. Šematski prikaz i fotografija eksperimentalne postavke EpSP prikazani su na slici 41. Pametni aktuator sastoji se od linearnog cilindra bez klipnjače *SMC MY3B16-600* koji je napajan mehanički upravljanim regulatorom pritiska *AZ Pneumatica MREG 2-08*, sa jedne, i elektropneumatskim regulatorom pritiska *SMC ITV2050-33F2N3* sa druge strane. Na oba razvodnika se preko pripremne grupe dovodi vazduh pod pritiskom od 6 bara. Elektropneumatski regulator propušta pritisak u opsegu 2-6 bara sa intenzitetom koji je proporcionalan analognom signalu u opsegu od 0-10 V koji dobija na ulazu. S druge strane cilindra mehanički regulator propušta konstantan pritisak od 4 bara. Razlika pritisaka između dve strane cilindra prouzrokuje kretanje klipa. Pored elektropneumatskih komponenti, pametni aktuator sadrži lokalni kontroler 1 (LK_1) baziran na *ARM Cortex-M3* procesoru sa radnim taktom od 96 MHz [92] kome je dodeljen transiver *Microchip MRF24J40MA* [73] za bežičnu komunikaciju zasnovanu na standardu *IEEE 802.15.4*.



Slika 41: Elektropneumatski sistem za pozicioniranje: a) šematski prikaz; b) eksperimentalna postavka

Linearni inkrementalni magnetni enkoder *Balluff BML-S1B0-Q53G-M400-L0-KA05* pozicioniran duž cilindra korišćen je za merenje pozicije klipa, čime se zatvara upravljačka petlja u EpSP. Enkoder je opremljen lokalnim kontrolerom 2 (LK_2) koji je zasnovan na istim uređajima kao LK_1, takođe uključujući i mogućnost bežične komunikacije. Ukupan hod cilindra je 600 mm što odgovara 60.000 impulsa (1 mm=100 impulsa). Inicijalna pozicija je krajnja tačka cilindra na strani na kojoj se nalazi mehanički regulator pritiska. Upravljački zadatak sistema distribuiran je na LK_1 i LK_2. Koristeći A i B faze enkodera, LK_2 određuje trenutnu poziciju klipa. Željena pozicija definisana je na LK_2, gde je implementiran PID regulator čiji su parametri određeni eksperimentalnim putem. Poredeći trenutnu i željenu poziciju klipa dobija se vrednost u opsegu [0, 1] koja odgovara narednoj akciji aktuatora, a koja se prenosi bežičnim putem na LK_1. Po prihvatanju te vrednosti, LK_1 je pretvara u analogni napon u opsegu 0-10 V što je proporcionalno pritisku vazduha koji je neophodan za ostvarivanje željenog kretanja klipa. Algoritam za detekciju napada kreiran je korišćenjem dva skupa podataka koji su prikupljeni u okviru ove disertacije:

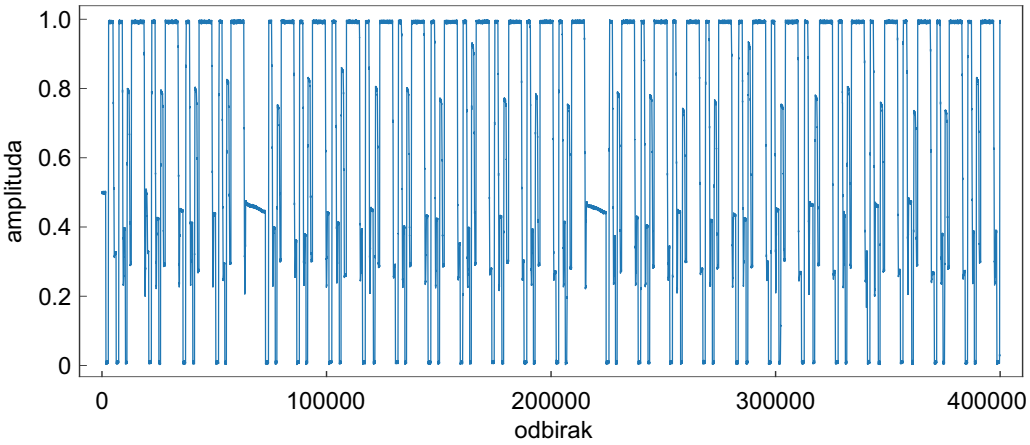
1. Skup podataka dobijen akvizicijom signala između LK_1 i elektropneumatskog regulatora pritiska [80] koji je korišćen za inicijalna istraživanja;

2. Skup podataka koji sadrži podatke koji su komunicirani između LK_2 i LK_1 [81].

Korišćenjem podataka dobijenih akvizicijom signala između LK_1 i elektropneumatskog regulatora pritiska ispituje se mogućnost tehnika mašinskog i dubokog učenja da na osnovu sirovog signala prikupljenog sa realne eksperimentalne instalacije modeliraju ponašanje sistema. Za potrebe snimanja podataka korišćen je *National Instruments* sistem za akviziciju (engl. *Data Acquisition* – DAQ). Frekvencija odabiranja prilikom snimanja podataka bila je 100 Hz. Nakon inicijalizacije kojom se klip dovodi u početnu poziciju, klip je ciklično ponavljao trajektoriju sačinjenu od pet pozicija (50-400-250-400-100 mm) što je rezultiralo sa ukupno 400.000 odbiraka.

Kako je dinamika signala određena naglim promenama koje mogu negativno uticati na performanse modela, neophodno je bilo kreirati odgovarajući FIR filter što predstavlja prvi korak u procesu pretprocesiranja signala. Izabrani filter definisan je propusnim opsegom [0; 0,11π], nepropusnim opsegom [0,35π; π] i prelaznim regionom između. Parks-Meklelanovim algoritmom dobijene su vrednosti 11 koeficijenata: $h(n)$=[0,0202; 0,0230; 0,0541; 0,1039; 0,1255; 0,1234; 0,1255; 0,1039; 0,0541; 0,0230; 0,0202]. Varijacijom vrednosti parametara prikazanih u tabelama 3, 4 i 5 kreirani su odgovarajući modeli za svaku od razmatranih tehnika (tabela 15).

Iz tabele 15 može se primetiti da je SVR kao najkompleksniji model opisan sa 3.964 nosećih vektora, širinom margine razdvajanja ε=0,01, RBF kernelom i parametrom regularizacije



Slika 42: Signal snimljen sa EpSP – napon između LK_1 i elektropneumatskog regulatora pritiska; sekvenca klipa 50-400-250-400-100 mm

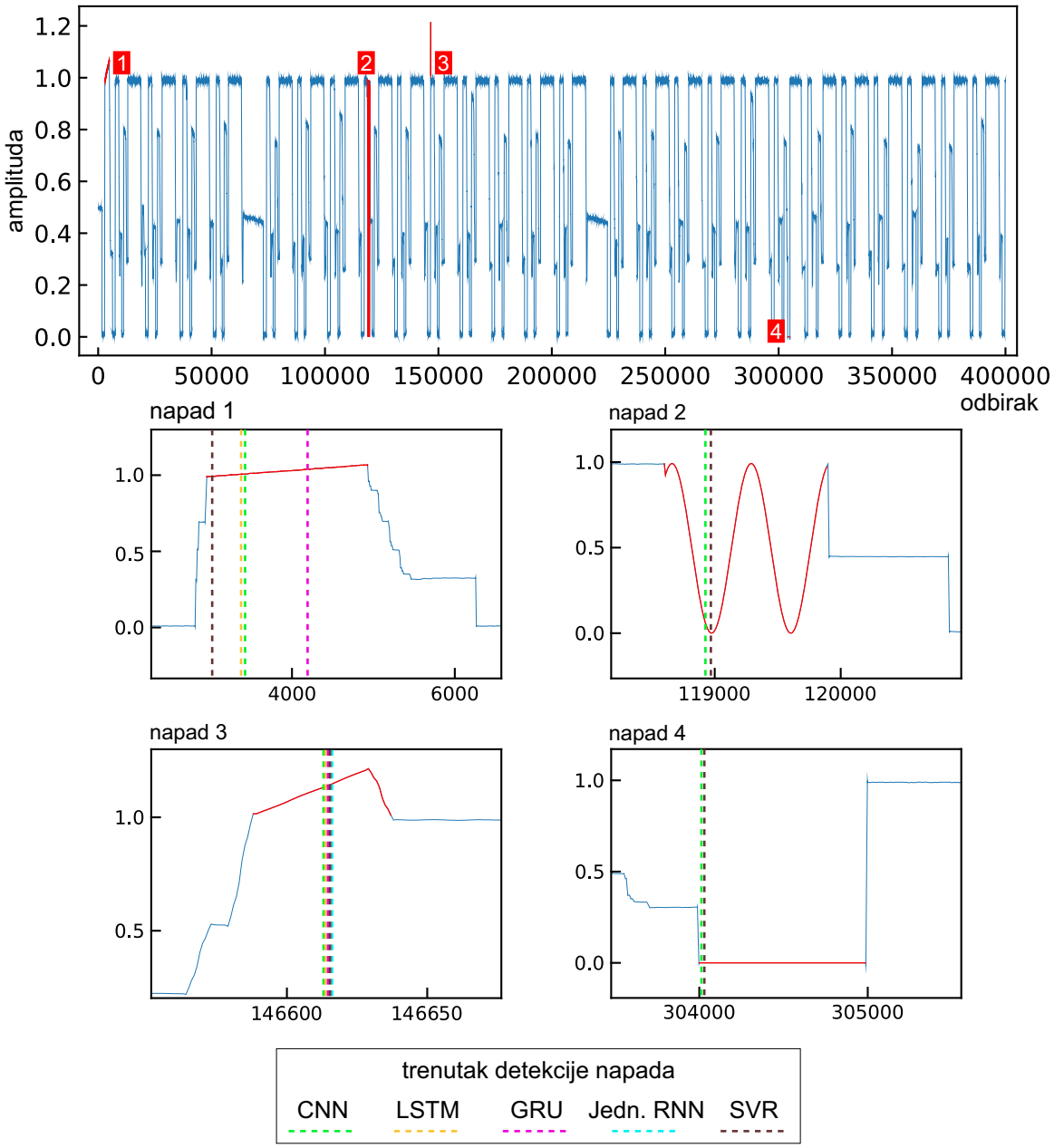
Tabela 15: Arhitekture kreiranih modela za signal sa EpSP (napon između LK_1 i elektropneumatskog regulatora pritiska)

Tehnika \ Signal		EpSP
SVR	v, ε, kr, C	v=2, $\varepsilon = 0,01$ kr=RBF, C=0,501
	br. nos. vekt.	3.964
jednostavne RNN	u_1-u_2-u_3-u_4, v, dr	8-8, v=2, dr=0,05
	br. param.	225
LSTM	u_1-u_2-u_3-u_4, v, dr	8-8, v=2, dr=0
	br. param.	873
GRU	u_1-u_2-u_3-u_4, v, dr	8-8, v=2, dr=0,05
	br. param.	705
CNN	f_1-f_2-f_3-f_4, v	4-8-16-16, v=16
	br. param.	2.865

C=0,501. S druge strane, RNN modeli su opisani sa po dva rekurentna sloja (c=1), dok je CNN model sastavljen od dva bloka, odnosno četiri konvoluciona sloja. CNN model koristio je najveću dužinu bafera v=16 u poređenju sa ostalim pristupima (v=2). Broj neurona u prvom potpuno povezanom sloju u okviru CNN arhitekture bio je d_1=30. Modeli jednostavne RNN i GRU definisani su stopom izostavljanja dr=0,05, dok je u slučaju LSTM modela vrednost ovog hiperparametra bila 0. Treba napomenuti da su ostali parametri/karakteristike poput vrste aktivacione funkcije, funkcije cilja, tipa optimizatora, broja epoha prilikom obučavanja itd. isti kao i u primerima vezanim za SWaT skup podataka.

Verifikacija kreiranih sistema za detekciju sprovedena je na 4 različita napada (slika 43). Napadi 1 i 3 linearno povećavaju vrednost za 0,00003 i 0,005 po odbirku, gde napad 1 uključuje i nasumično generisani šum. Napadom 2 vrednost signala menja se na osnovu sinusne funkcije, dok napad 4 postavlja vrednost signala na 0 određeni vremenski period. Uslovi detekcije napada definisani su podalgoritmom sa slike 25, kao i u prethodnim slučajevima. Na slici 43 originalni signal prikazan je plavom, period dejstva napada crvenom bojom, dok su trenuci detekcije napada označeni vertikalnim linijama (svakom modelu dodeljena je odgovarajuća boja).

Pristupima zasnovanim na SVR i CNN detektovana su sva četiri napada. S druge strane, LSTM i GRU modelima detektovani su napadi 1 i 3, dok je model jednostavne RNN detektovao samo napad 1. Sa aspekta momenta detekcije, na napadu 1 je uočljivo da je SVR pristup dao



Slika 43: Detektovani napadi na signalu sa EpSP

najbolje rezultate, dok je GRU modelu bilo potrebno najviše vremena za ostvarivanje detekcije. Na primerima ostalih napada nema značajnijih razlika u tom pogledu.

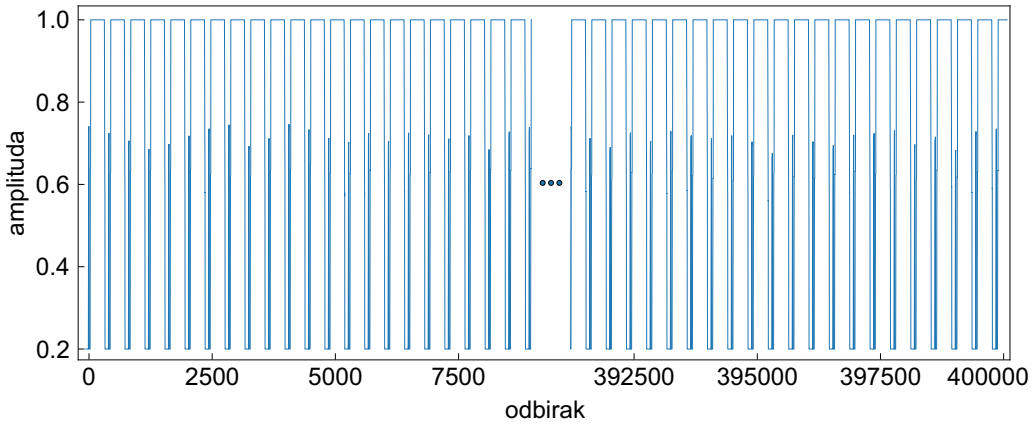
Iako su se modeli generisani na osnovu signala napona između LK_1 i elektropneumatskog regulatora pritiska iz skupa podataka [80] u prethodnim istraživanjima [88, 89] pokazali kao odgovarajući za implementaciju na lokalnom kontroleru, kreiran je drugi skup podataka [81] koji sadrži originalne vrednosti komunicirane između LK_1 i LK_2. Drugi skup podataka ne sadrži šum generisan prilikom akvizicije napona između mikrokontrolera i regulatora, već podatke koji se u stvarnosti razmenjuju između kontrolera. Ovaj skup podataka obuhvata signale koji su definisani različitim trajektorijama klipa, gde je najkraća trajektorija sastavljena od dve, a najduža od pet zadatih pozicija klipa. Svaki signal snimljen je tokom 200 minuta kretanja klipa duž definisane putanje, pri čemu je dužina svakog signala 400.000 odbiraka. Snimanje podataka izvršeno je korišćenjem softvera *Tera Term* [94] instaliranog na računaru koji je serijskom komunikacijom (USB protokolom) povezan sa lokalnim kontrolerima. Frekvencija odabiranja prilikom snimanja podataka bila je 33,3 Hz. Podaci koji će biti u nastavku korišćeni

predstavljaju vrednosti koje je LK_1 primio od LK_2 tokom rada sistema u normalnim uslovima. Iz navedenog skupa podataka izabrana su četiri signala za koje će biti kreirani jedinstveni ML modeli korišćenjem svih razmatranih tehnika. Oznake signala i trajektorije koje je klip ostvarivao prilikom njihovog snimanja navedeni su u tabeli 16.

Tabela 16: Oznake razmatranih signala i trajektorije klipa

Oznaka	**Trajektorija [mm]**
tr_1	150-350-450
tr_2	50-400-250-400-100
tr_3	100-350
tr_4	400-150

Kako su signali tr_3 i tr_4 definisani najmanjim, a signal tr_2 najvećim brojem pozicija klipa (dve pozicije i pet pozicija), kao referentni signal koji će u nastavku biti detaljnije prikazan izabran je tr_1 (definisan sa tri pozicije klipa). Deo signala tr_1 prikazan je na slici 44 gde se može primetiti cikličnost navedene sekvence klipa.



Slika 44: Deo signala tr_1 snimljenog na EpSP – komunicirani podaci između LK_1 i LK_2

Sa slike 44 mogu se primetiti određene sličnosti sa signalom koji je prikazan na slici 42 u vidu dinamike karakterisane naglim promenama. Stoga, za potrebe pretprocesiranja svih signala prikazanih u tabeli 16 izabran je isti FIR filter (sa propusnim opsegom $[0;\ 0{,}11\pi]$, nepropusnim opsegom $[0{,}35\pi;\ \pi]$ i prelaznim regionom između) koji sadrži ukupno 11 koeficijenata. Takođe, shodno primeru prvog razmatranog signala iz EpSP (slika 42) izabrane su iste vrednosti dodatnih parametara modela, kao i karakteristike koje definišu proces obučavanja modela. Arhitekture koje su se prema definisanim kriterijumima za odabir modela pokazale kao odgovarajuće prikazane su u tabeli 17.

Tabela 17: Arhitekture kreiranih modela za signale sa EpSP (komunicirani podaci između LK_1 i LK_2)

Tehnika \ Signal		tr_1	tr_2	tr_3	tr_4
SVR	v, ε, kr, C	v=4, ε=0,1 kr=RBF C=0,360	v=2, ε=0,1 kr=*linearni* C=1	v=4, ε=1 kr=*linearni* C=1	v=2, ε=0,1 kr=*linearni* C=1
	br. nos. vekt.	4.060	2.953	3.044	1.845
jedn. RNN	u_1-u_2-u_3-u_4 v, dr	8-8 v=2, dr=0,05	8-8 v=2, dr=0,05	8-8 v=2, dr=0	8-8 v=2, dr=0,05
	br. param.	225	225	225	225
LSTM	u_1-u_2-u_3-u_4 v, dr	8-8 v=2, dr=0,05	8-8 v=2, dr=0,05	8-8 v=2, dr=0	8-8 v=2, dr=0
	br. param.	873	873	873	873
GRU	u_1-u_2-u_3-u_4 v, dr	8-8 v=2, dr=0,05	8-8 v=2, dr=0	8-8 v=2, dr=0	8-8 v=2, dr=0,05
	br. param.	705	705	705	705
CNN	f_1-f_2-f_3-f_4 v	8-16-16-16 v=16	16-8-8-16 v=16	4-8-32-16 v=16	4-8-8-16 v=16
	br. param.	3.333	2.701	3.649	2.473

Interesantno je da su u poređenju sa modelima iz tabele 15 arhitekture za sve RNN modele identične (dva RNN sloja sa po 8 jedinica), uključujući i dužinu bafera v, a samim tim i broj obučavajućih parametara (razlikuje se u zavisnosti od primenjene RNN tehnike). Za vrednost stope izostavljanja kod RNN modela ne može se uvideti određena zavisnost, sem da je za sve RNN modele vezane za signal tr_1 iznosila dr=0,05, dok je za sve RNN modele vezane za signal tr_3 bila dr=0. S druge strane, stopa izostavljanja za signale tr_2 i tr_4 varira u odnosu na primenjenu RNN tehniku.

Glavna razlika u odnosu na tabelu 15 ogleda se u SVR i CNN modelima. Naime, kod SVR modela u zavisnosti od signala korišćeno je od 1.845 do 4.060 nosećih vektora sa dužinom bafera v=2 ili v=4. Za tri od četiri signala izabran je *linearni* kernel i vrednost parametra regularizacije C=1. Širina margine razdvajanja u većini slučajeva bila je ε=0,1, sem u SVR modelu kreiranom za signal tr_3 gde je iznosila ε=1. Iako su svi CNN modeli opisani drugačijim arhitekturama (broj obučavajućih parametara varira u opsegu od 2.473 do 3.649), dužina bafera za sve modele bila je v=16. Takođe, broj neurona u prvom potpuno povezanom sloju kod svih CNN modela bio je d_1=30.

Kreirani modeli biće korišćeni u nastavku za implementaciju na eksperimentalnoj instalaciji EpSP. Kada se posmatraju svi modeli iz tabele 17 vidi se da najkompleksniji model ima 4.060 obučavajućih parametara (SVR model za signal tr_1) što se može smatrati prihvatljivim brojem parametara za implementaciju algoritma na lokalni kontroler, a što će u nastavku biti i utvrđeno.

5.3. Implementacija ML algoritama na kontrolere pametnih uređaja

U ovom poglavlju biće prikazana procedura za implementaciju svih razmatranih algoritama za detekciju napada na kontrolere pametnih uređaja u okviru EpSP. Naime, mehanizmi za detekciju napada biće primenjeni na lokalnom kontroleru (LK_1) pametnog aktuatora koji je prijemnik (potencijalno kompromitovanog) signala u okviru EpSP. Kao što je ranije navedeno, procedura kreiranja RNN i CNN modela izvodi se u *Python v3.8.5* programskom jeziku u okviru koga je integrisana platforma *TensorFlow v2.3.0* namenjena za razvoj modela baziranih na mašinskom učenju. U slučaju SVR modela korišćen je programski paket *Matlab 2022b*. Upravljački

zadaci oba lokalna kontrolera u ovom slučaju izvode se u *C++* programskom jeziku koristeći *Keil uVision5* okruženje [5]. Stoga, da bi se primenili ML algoritmi na LK_1, svaki sloj u okviru arhitekture mora biti programiran samo sa elementarnim funkcijama programskog jezika *C++*. Pre nego što bude prikazana procedura transformacije iz *TensorFlow* platforme u okruženje lokalnog kontrolera, ukratko će biti prikazana i struktura bazičnih slojeva (jednostavne RNN, LSTM, GRU i CNN sloj) za svaki razmatrani tip neuronskih mreža. Implementacija ostalih slojeva (izostavljajući sloj, sloj sažimanja, ispravljajući sloj i potpuno povezan sloj) je prilično jednostavna pa se iz tog razloga neće detaljnije opisivati.

Izlaz iz *TensorFlow*-a je obučeni model koji se sastoji od konačnog broja obučavajućih parametara struktuiranih u prethodno definisan format. S druge strane, izlaz iz *Matlab* okruženja predstavlja SVR model gde su vrednosti nosećih vektora sadržani u jednoj matrici. U poređenju sa RNN i CNN tehnikama, struktura SVR algoritma je jednostavnija pa je samim tim i njegova implementacija manje komplikovana.

5.3.1. Implementacija SVR algoritma

Implementacija SVR algoritma na kontroler je jednostavna i svodi se na izračunavanje vrednosti jednačine (12). SVR model definisan je matricom nosećih vektora $\mathbf{S}$ i vektorom $\mathbf{a}$:

$$\mathbf{S} = \begin{bmatrix} s_{11} & s_{12} & \cdots & s_{1v} \\ s_{21} & s_{22} & \cdots & s_{2v} \\ \vdots & \vdots & \ddots & \vdots \\ s_{n1} & s_{n2} & \cdots & s_{nv} \end{bmatrix} \tag{42}$$

$$\mathbf{a} = [\alpha_1, \alpha_2, ...\alpha_n] \tag{43}$$

gde n i v predstavljaju broj nosećih vektora i dužinu bafera, tim redom. Pored vrednosti nosećih vektora i $\alpha_i, i \in \{1, \dots n\}$ koeficijenata, model sadrži i *bias* vrednost.

Izabrani kernel *kr* u programu kontrolera treba zapisati pomoću osnovnih matematičkih operacija. U primeru prikazanom u algoritmu 1 korišćen je kernel koji koristi radijalnu funkciju (definisan u (13)).

Algoritam 1: Implementacija SVR modela

for $i = 1$ **to** n **do**

$y_{fit} = y_{fit} + \mathbf{a}[i] * exp((-1/(g * g)) * ((\mathbf{x}[1] - \mathbf{S}[1][i]) * (\mathbf{x}[1] - \mathbf{S}[1][i]) + ... + (\mathbf{x}[v] - \mathbf{S}[v][i]) * (\mathbf{x}[v] - \mathbf{S}[v][i])))$

end for

$y_{fit} \mathrel{+}= b$

U tabeli 18 navedene su oznake korišćene u algoritmu 1. Treba napomenuti da se u postupku kreiranja SVR modela, vrednost parametra radijalnog kernela γ najčešće dobija automatski korišćenjem metoda na bazi heuristike, iako je vrednost ovog parametra moguće i ručno definisati pre procesa obučavanja.

5.3.2. Implementacija RNN algoritama

Kod RNN modela, parametri RNN slojeva čine strukturu $\mathbf{S}$ sa tri osnovna člana koji se za sloj t mogu zapisati na sledeći način: $\mathbf{W}_{xh}$ i $\mathbf{W}_{hh}$ – ulazna matrica i matrica skrivenog stanja, $\mathbf{b}_h$ – *bias* vektor. U zavisnosti od tipa neuronske mreže, dimenzije matrica $\mathbf{W}_{xh}$ i $\mathbf{W}_{hh}$ i vektora $\mathbf{b}_h$ se razlikuju. Struktura sloja jednostavne RNN opisana je na sledeći način:

Tabela 18: Notacija korišćena u algoritmu za implementaciju SVR

Oznaka	Opis	Oznaka	Opis
$\mathbf{x}$	vektor ulaza sa v vrednosti	b	*bias* vrednost
y_{fit}	izlazna vrednost regresije	n	broj nosećih vektora
$\mathbf{S}$	matrica vrednosti nosećih vektora	v	dužina bafera
$\mathbf{a}$	vektor α koeficijenata sa n vrednosti	γ	parametar radijalnog kernela

$$\mathbf{S} = \begin{bmatrix} \mathbf{W}_{xh}[i_t][u_t] \\ \mathbf{W}_{hh}[u_t][u_t] \\ \mathbf{b}_h[u_t] \end{bmatrix} \tag{44}$$

gde i_t predstavlja broj ulaznih vektora, dok je u_t broj jedinica u sloju t. Implementacija sloja jednostavne RNN opisana je u algoritmu 2, dok je korišćena notacija navedena u tabeli 19.

Algoritam 2: Implementacija sloja jednostavne RNN

for $i = 1$ **to** v_t **do** // za svaku vrednost iz bafera
  // proizvod $\mathbf{x}_t[i] \cdot \mathbf{W}_{xh}$
  for $j = 1$ **to** u_t **do**
    for $k = 1$ **to** i_t **do**
      $\mathbf{m}_1[k] = \mathbf{x}_t[i] * \mathbf{W}_{xh}[k][j]$
      $\mathbf{m}_2[j]$ += $\mathbf{m}_1[k]$ // sumiranje po kolonama
    end for
  end for
  // proizvod $\mathbf{h}_{t-1} \cdot \mathbf{W}_{hh}$
  for $p = 1$ **to** u_t **do**
    for $q = 1$ **to** u_t **do**
      $\mathbf{n}_1[q] = \mathbf{h}_{t-1}[q] * \mathbf{W}_{hh}[q][p]$
      $\mathbf{n}_2[p]$ += $\mathbf{n}_1[q]$ // sumiranje po kolonama
    end for
  end for
  for $r = 1$ **to** u_t **do**
    $\mathbf{h}_t[r] = \mathbf{m}_2[r] + \mathbf{n}_2[r] + \mathbf{b}_h[r]$ // skriveno stanje
    $\mathbf{h}_t[r] = \mathbf{h}_t[r] * (\mathbf{h}_t[r] > 0)$ // ReLU aktivaciona funkcija
  end for
end for
za sloj t+1 važi $\mathbf{x}_{t+1} = \mathbf{h}_t$

Tabela 19: Notacija korišćena u algoritmu za implementaciju sloja jednostavne RNN

Oznaka	**Opis**	**Oznaka**	**Opis**
v_t	dužina bafera za sloj t	i_t	broj ulaznih vektora u sloju t
$\mathbf{x}_t$	ulazni vektor za sloj t	$\mathbf{b}_h$	*bias* vektor
$\mathbf{W}_{xh}$	ulazna matrica	$\mathbf{h}_t$	vektor skrivenog stanja za sloj t
$\mathbf{W}_{hh}$	matrica skrivenog sloja	$\mathbf{h}_{t-1}$	vektor skrivenog stanja za sloj t-1
u_t	broj jedinica u sloju t		

Zbog četiri kapije koje sadrži unutar ćelije, LSTM sloj ima kompleksniju strukturu u odnosu na jednostavne RNN:

$$\mathbf{S} = \begin{bmatrix} \mathbf{W}_{xh}[i_t][4 * u_t] \\ \mathbf{W}_{hh}[u_t][4 * u_t] \\ \mathbf{b}_h[4 * u_t] \end{bmatrix} \tag{45}$$

Notacija za u_t i i_t ista je kao u slučaju sloja jednostavne RNN. Algoritam 3 prikazuje proceduru primene LSTM sloja u okruženje lokalnog kontrolera, dok je notacija vezana za ovaj algoritam data u tabeli 20.

Tabela 20: Notacija korišćena u algoritmu za implementaciju LSTM sloja

Oznaka	**Opis**	**Oznaka**	**Opis**
v_t	dužina bafera za sloj t	$\mathbf{b}_h$	*bias* vektor
$\mathbf{x}_t$	ulazni vektor za sloj t	$\mathbf{c}_t$	vektor unutrašnjeg stanja ćelije za sloj t
$\mathbf{W}_{xh}$	ulazna matrica	$\mathbf{c}_{t-1}$	vektor unutrašnjeg stanja ćelije za sloj t-1
$\mathbf{W}_{hh}$	matrica skrivenog sloja	$\mathbf{i}$	ulazna kapija
u_t	broj jedinica u sloju t	$\mathbf{f}$	kapija zaboravljanja
i_t	broj ulaznih vektora u sloju t	$\mathbf{g}$	kapija stanja kandidata
$\mathbf{h}_t$	vektor skrivenog stanja za sloj t	$\mathbf{o}$	izlazna kapija
$\mathbf{h}_{t-1}$	vektor skrivenog stanja za sloj t-1		

U slučaju GRU sloja, matrice težinskih koeficijenata ($\mathbf{W}_{xh}$ i $\mathbf{W}_{hh}$) i *bias* matrica $\mathbf{B}_h$ imaju sledeći oblik:

$$\mathbf{S} = \begin{bmatrix} \mathbf{W}_{xh}[i_t][3 * u_t] \\ \mathbf{W}_{hh}[u_t][3 * u_t] \\ \mathbf{B}_h[2][3 * u_t] \end{bmatrix} \tag{46}$$

Implementacija GRU sloja pomoću osnovnih funkcija programskog jezika *C++* prikazana je u algoritmu 4, dok je notacija vezana za ovaj algoritam prikazana u tabeli 21.

Algoritam 3: Implementacija LSTM sloja

for $i = 1$ **to** v_t **do** // za svaku vrednost iz bafera
  // proizvod $\mathbf{x}_t[i] \cdot \mathbf{W}_{xh}$
  for $j = 1$ **to** $4 * u_t$ **do**
    for $k = 1$ **to** i_t **do**
      $\mathbf{m}_1[k] = \mathbf{x}_t[i] * \mathbf{W}_{xh}[k][j]$
      $\mathbf{m}_2[j]$ += $\mathbf{m}_1[k]$ // sumiranje po kolonama
    end for
  end for
  // proizvod $\mathbf{h}_{t-1} \cdot \mathbf{W}_{hh}$
  for $p = 1$ **to** $4 * u_t$ **do**
    for $q = 1$ **to** u_t **do**
      $\mathbf{n}_1[q] = \mathbf{h}_{t-1}[q] * \mathbf{W}_{hh}[q][p]$
      $\mathbf{n}_2[p]$ += $\mathbf{n}_1[q]$ // sumiranje po kolonama
    end for
  end for
  for $r = 1$ **to** $3 * u_t$ **do**
    $\mathbf{s}[r] = \mathbf{m}_2[r] + \mathbf{n}_2[r] + \mathbf{b}_t[1][r] + \mathbf{b}_t[2][r]$
  end for
  for $g = 1$ **to** u_t **do**
    $\mathbf{i}[g] = (1/(1 + \exp(-\mathbf{s}[g])))$ // ulazna kapija (sa sigmoidnom aktivacionom funkcijom)
    $\mathbf{f}[g] = (1/(1 + \exp(-\mathbf{s}[g + u_t])))$ // kapija zaboravljanja (sa sigmoidnom aktivacionom funkcijom)
    $\mathbf{g}[g] = \mathbf{s}[g + 2 * u_t] * (\mathbf{s}[g + 2 * u_t] > 0)$ // kapija stanja kandidata (sa ReLU aktivacionom funkcijom)
    $\mathbf{o}[g] = (1/(1 + \exp(-\mathbf{s}[g + 3 * u_t])))$ // izlazna kapija (sa ReLU aktivacionom funkcijom)
    $\mathbf{c}_t[g] = \mathbf{i}[g] * \mathbf{g}[g] + \mathbf{f}[g] * \mathbf{c}_{t-1}[g]$ // unutrašnje stanje ćelije
    $\mathbf{h}_t[g] = \mathbf{o}[g] * (\mathbf{c}_t[g] * (\mathbf{c}_t[g] > 0))$ // skriveno stanje (sa ReLU aktivacionom funkcijom)
  end for
za sloj $t + 1$ važi $\mathbf{x}_{t+1} = \mathbf{h}_t$

Tabela 21: Notacija korišćena u algoritmu za implementaciju GRU sloja

Oznaka	**Opis**	**Oznaka**	**Opis**
v_t	dužina bafera za sloj t	$\mathbf{b}_h$	*bias* vektor
$\mathbf{x}_t$	ulazni vektor za sloj t	$\mathbf{h}_t$	vektor skrivenog stanja za sloj t
$\mathbf{W}_{xh}$	ulazna matrica	$\mathbf{h}_{t-1}$	vektor skrivenog stanja za sloj t-1
$\mathbf{W}_{hh}$	matrica skrivenog sloja	$\mathbf{u}$	kapija za ažuriranje
u_t	broj jedinica u sloju t	$\mathbf{r}$	kapija za resetovanje
i_t	broj ulaznih vektora u sloju t	$\mathbf{g}$	kapija stanja kandidata

Algoritam 4: Implementacija GRU sloja

for $i = 1$ **to** v_t **do** // za svaku vrednost iz bafera
  // proizvod $\mathbf{x}_t[i] \cdot \mathbf{W}_{xh}$
  for $j = 1$ **to** $3 * u_t$ **do**
    for $k = 1$ **to** i_t **do**
      $\mathbf{m}_1[k] = \mathbf{x}_t[i] * \mathbf{W}_{xh}[k][j]$
      $\mathbf{m}_2[j] \mathrel{+}= \mathbf{m}_1[k]$ // sumiranje po kolonama
    end for
  end for
  // proizvod $\mathbf{h}_t \cdot \mathbf{W}_{hh}$
  for $p = 1$ **to** $3 * u_t$ **do**
    for $q = 1$ **to** u_t **do**
      $\mathbf{n}_1[q] = \mathbf{h}_{t-1}[q] * \mathbf{W}_{hh}[q][p]$
      $\mathbf{n}_2[p] \mathrel{+}= \mathbf{n}_1[q]$ // sumiranje po kolonama
    end for
  end for
  for $r = 1$ **to** $3 * u_t$ **do**
    $\mathbf{s}[r] = \mathbf{m}_2[r] + \mathbf{n}_2[r] + \mathbf{B}_h[1][r] + \mathbf{B}_h[2][r]$
  end for
  for $e = 1$ **to** u_t **do**
    $\mathbf{u}[e] = (1/(1 + \exp(-\mathbf{s}[e])))$ // kapija za ažuriranje (sa sigmoidnom aktivacionom funkcijom)
    $\mathbf{r}[e] = (1/(1 + \exp(-\mathbf{s}[e + u_t])))$ // kapija za resetovanje (sa sigmoidnom aktivacionom funkcijom)
  end for
  // proizvod $\mathbf{x}_t[i] \cdot \mathbf{W}_{xh}$
  for $f = 1$ **to** u_t **do**
    for $g = 1$ **to** i_t **do**
      $\mathbf{m}_3[g] = \mathbf{x}_t[i] * \mathbf{W}_{xh}[g][f + 2 * u_t]$
      $\mathbf{m}_4[f] \mathrel{+}= \mathbf{m}_3[g]$ // sumiranje po kolonama
    end for
  end for
  // proizvod $\mathbf{x}_t \cdot \mathbf{W}_{hh}$
  for $d = 1$ **to** u_t **do**
    for $e = 1$ **to** u_t **do**
      $\mathbf{n}_3[e] = \mathbf{h}_t[e] * \mathbf{W}_{hh}[e][d + 2 * u_t]$
      $\mathbf{n}_4[d] \mathrel{+}= \mathbf{n}_3[e]$ // sumiranje po kolonama
    end for
  end for
  for $w = 1$ **to** u_t **do**
    $\mathbf{g}[w] = \mathbf{m}_4[w] + \mathbf{B}_h[1][w + 2 * u_t] + \mathbf{r}[w] * (\mathbf{n}_4[w] + \mathbf{B}_h[2][w + 2 * u_t])$ // kapija stanja kandidata
    $\mathbf{g}[w] = \mathbf{g}[w] * (\mathbf{g}[w] > 0)$ // ReLU aktivaciona funkcija
    $\mathbf{h}_t[w] = (1 - \mathbf{u}[w]) * \mathbf{g}[w] + \mathbf{u}[w] * \mathbf{h}_t[w]$ // skriveno stanje
    $\mathbf{h}_t[w] = \mathbf{h}_t[w] * (\mathbf{h}_t[w] > 0)$ // ReLU aktivaciona funkcija
  end for
za sloj $t + 1$ važi $\mathbf{x}_{t+1} = \mathbf{h}_t$

5.3.3. Implementacija CNN algoritma

U slučaju CNN sloja, struktura **S** opisana je sa dva člana: trodimenzionalna matrica težinskih koeficijenata – **W** i *bias* vektor – **b** [83], odnosno kao:

$$\mathbf{S} = \begin{bmatrix} \mathbf{W}[m_t][i_t][f_t] \\ \mathbf{b}[f_t] \end{bmatrix} \tag{47}$$

gde m_t označava veličinu filtera, i_t predstavlja broj ulaznih vektora, dok je f_t broj filtera u sloju t. Postupak implementacije CNN sloja prikazan je u algoritmu 5. Notacija korišćena u algoritmu 5 navedena je u tabeli 22.

Algoritam 5: Implementacija CNN sloja

for $i = 1$ **to** f_t **do** // za sve filtere
  for $j = 1$ **to** len($\mathbf{O}_{t-1}[1]$) **do** // za sve ulazne vektore
    $\mathbf{w} = \{\mathbf{W}[m_t][j][i] : \mathbf{W}[1][j][i]\}$ // trenutni filter
    // konvolucija – početak
    for $v = 1$ **to** v_t **do**
      $\mathbf{c}[v-1] = \mathbf{w}[m_t] * \mathbf{x}_t[j][v-m_t] + ... + \mathbf{w}[1] * \mathbf{x}_t[j][v]$
    end for
    $\mathbf{c}$=$\mathbf{c}[m_t : \text{end}]$ // izostavljanje prvih $m_t - 1$ vrednosti
    // konvolucija – kraj
    // postavljanje konvolucije ulaznog vektora u matricu $\mathbf{I}_t$
    for $k = 1$ **to** len($\mathbf{O}_{t-1}[2]$) **do**
      $\mathbf{I}_t[j][k] = \mathbf{c}[k]$
    end for
  end for
  // sumiranje po kolonama ulaza filtriranog sa **w**
  for $q = 1$ **to** len($\mathbf{O}_{t-1}[2]$) **do**
    for $h = 1$ **to** len($\mathbf{O}_{t-1}[1]$) **do**
      $\mathbf{p}[q]$ += $\mathbf{I}_t[h][q]$
    end for
    $\mathbf{p}[q]$ += $\mathbf{b}[i]$ // dodavanje vrednosti *bias*-a
    $\mathbf{p}[q] = \mathbf{p}[q] * (\mathbf{p}[q] > 0)$ // ReLU aktivaciona funkcija
    $\mathbf{O}_t[i][q] = \mathbf{p}[q]$
  end for // dobijen je jedan izlazni vektor
end for

Tabela 22: Notacija korišćena u algoritmu za implementaciju CNN sloja

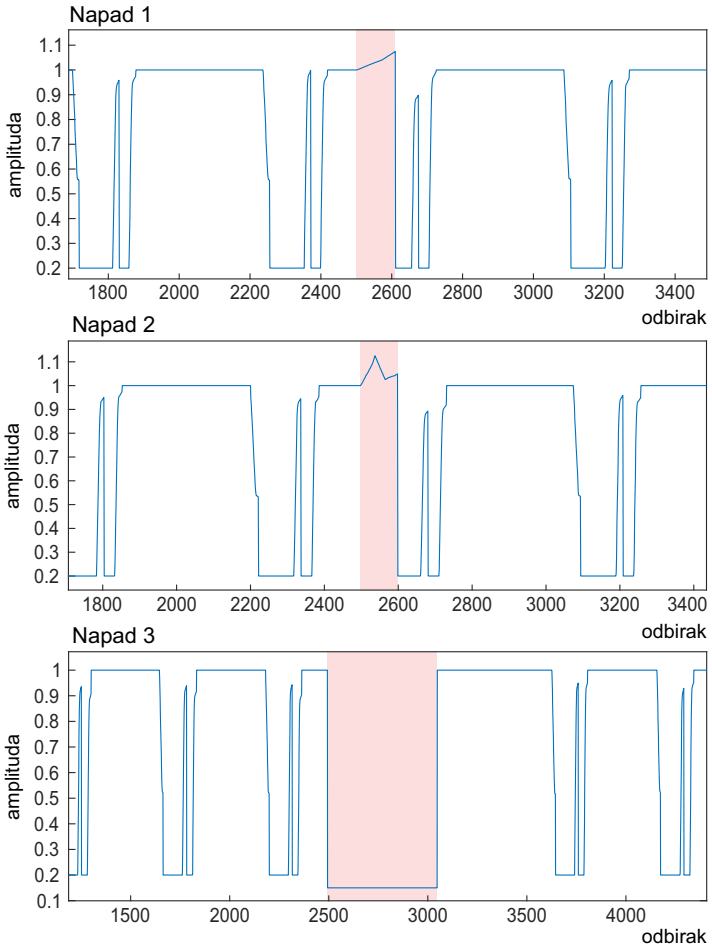
Oznaka	Opis	Oznaka	Opis
v_t	dužina bafera za sloj t	$\mathbf{b}_h$	*bias* vektor
$\mathbf{x}_t$	ulazni vektor za sloj t	$\mathbf{O}_t$	matrica izlaza u sloju t
$\mathbf{W}$	matrica težinskih koeficijenata	$\mathbf{m}_t$	veličina filtera u sloju t
f_t	broj filtera u sloju t	$\mathbf{w}$	trenutni filter
i_t	broj ulaznih vektora u sloju t	$\mathbf{c}$	vektor konvolucije

5.3.4. Validacija algoritama na eksperimentalnoj instalaciji

Na primeru modela kreiranih korišćenjem signala tr_1 iz tabele 17 biće prikazani rezultati implementacije koja je izvršena prema predstavljenoj proceduri na LK_1 kao delu pametnog aktu-

atora u okviru EpSP. Na ovaj način, na primeru industrijske aplikacije testirane su performanse algoritama u realnom vremenu. Odabir signala na kojem će biti prikazana implementacija bio je uslovljen najkompleksnijim modelom (SVR model za tr_1 sa 4.060 nosećih vektora). Naime, ukoliko implementacija IDS-a koji koristi ovaj model ne izazove kašnjenja ili neke slične poremećaje koji bi poremetili rad sistema, može se očekivati da će i svi ostali modeli iz tabele 17 biti uspešno implementirani.

U procesu testiranja generisano je više napada, od čega su ovde prikazana ukupno tri (slika 45). Napad 1 linearno povećava vrednost signala sa korakom od 0,0005 po odbirku. Trajanje napada 1 ograničeno je na 110 odbiraka kada se dostiže maksimalna vrednost od 1,055. Napad 2 linearno uvećava vrednost signala za 0,0025 po odbirku u trajanju od 35 odbiraka kada se dostiže maksimalna vrednost od 1,0875. Po dostizanju ove vrednosti, tokom narednih 20 odbiraka vrši se smanjivanje vrednosti signala za 0,002 po odbirku. Poslednja sekvenca napada 2 jeste uvećavanje vrednosti signala za 0,0005 po odbirku narednih 35 odbiraka. Stoga, ukupno trajanje napada 2 je 100 odbiraka. Napad 3 predstavlja najduži napad sa ukupnim trajanjem od 550 odbiraka gde se vrednost signala već u prvom odbirku postavlja na 0,15 i na toj vrednosti ostaje do kraja dejstva.

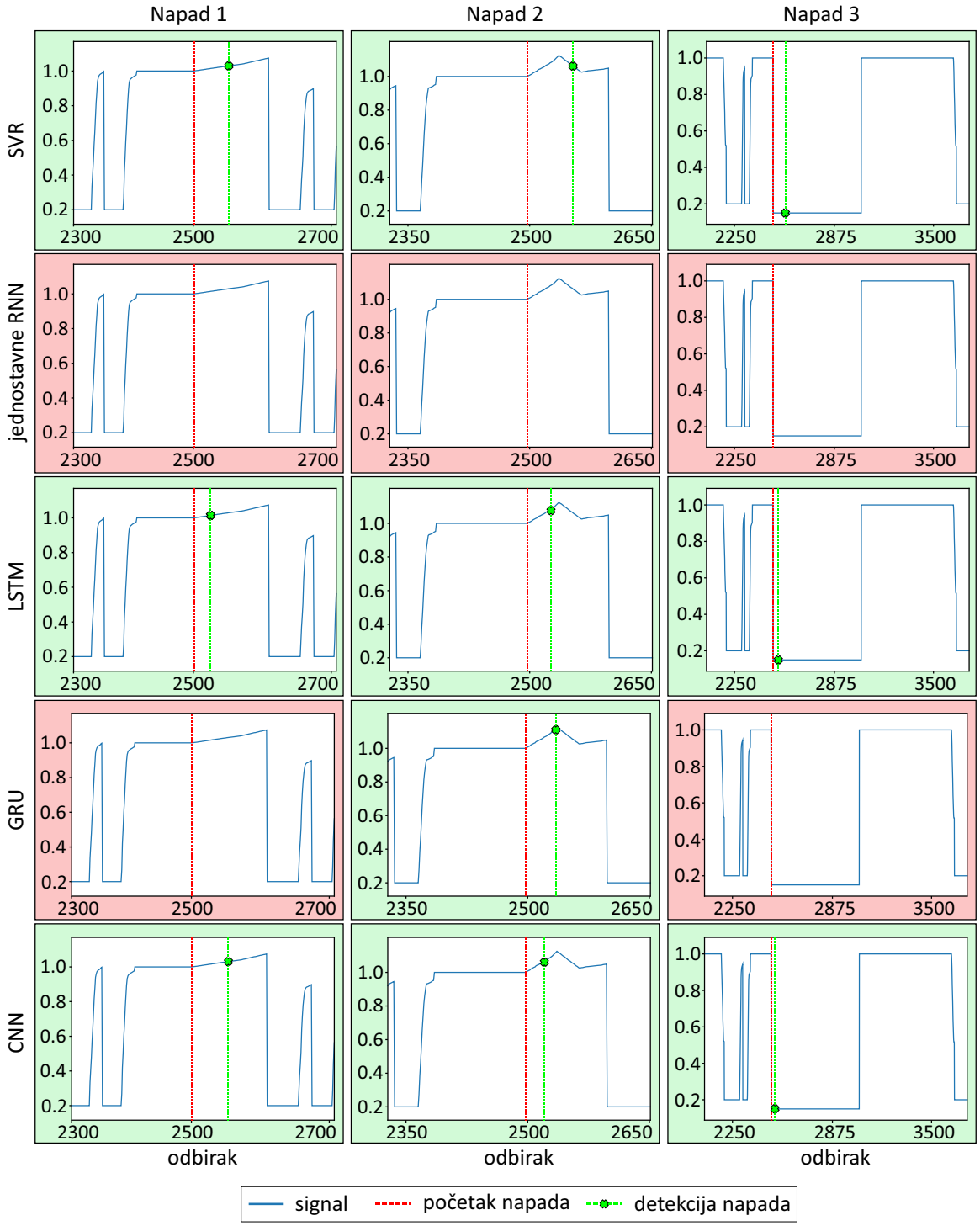


Slika 45: Napadi korišćeni za validaciju razvijenih algoritama za detekciju napada

SVR, LSTM i CNN algoritmi uspešno su detektovali sva tri napada (slika 46). Kada se porede ove tri tehnike kojima su uspešno detektovani svi napadi, može se primetiti da je SVR algoritmima bilo potrebno najviše odbiraka pre trenutaka detekcije napada. S druge strane, pomoću GRU algoritma detektovan je samo napad 2, dok algoritam jednostavne RNN u ovom slučaju nije detektovao nijedan napad. Početak napada je na slici 46 označen crvenom isprekidanom linijom, dok je trenutak detekcije označen zelenom linijom. Takođe, boja u pozadini

dijagrama označava da li je napad detektovan (zelena) ili ne (crvena). Lažno pozitivnih rezultata nije bilo ni u jednom slučaju primene razvijenih algoritama. Pored toga, treba naglasiti da implementacija algoritama za detekciju napada nije prouzrokovala kašnjenja koja bi na bilo koji način ugrozila funkcionalnost sistema i izvršavanje definisanog zadatka upravljanja.

U tabeli 23 prikazane su vrednosti kašnjenja svake tehnike prilikom detekcije razmatranih napada. Kašnjenje je izraženo brojem odbiraka kao i vremenom u sekundama koje je dobijeno deljenjem broja odbiraka sa frekvencijom odabiranja od 33,3 Hz. Iz tabele 23 može se primetiti da je najmanje kašnjenje ostvareno prilikom detekcije napada 1 algoritmom baziranim na LSTM arhitekturi. S druge strane, detekcija napada 3 algoritmom koji koristi SVR model zahtevela je najveći broj odbiraka (75 odbiraka = 2,25 s).



Slika 46: Performanse algoritama za detekciju napada na EpSP u realnom vremenu

Razmatrajući ove, kao i prethodne rezultate primene algoritama, dolazi se do zaključka da je CNN pokazao najbolje performanse u zadacima detekcije napada. Ostvarenim rezultatima detekcije napada na eksperimentalnoj instalaciji gde se proces odvija u realnom vremenu, stiče se utisak da se razvijeni algoritmi mogu uspešno koristiti i u sličnim industrijskim aplikacijama.

Stoga, dokazana je tvrdnja da je primenom algoritama za detekciju napada zasnovanih na mašinskom i dubokom učenju, moguće u realnom vremenu u okviru proračunski i energetski ograničenih kibernetsko-fizičkih sistema, postići visok nivo detekcije napada ne izazivajući pritom kašnjenja koja će ugroziti funkcionalnost sistema.

Tabela 23: Kašnjenja pri detekciji napada na EpSP

Tehnika	**Kašnjenje**	**Napad 1**	**Napad 2**	**Napad 3**
SVR	broj odbiraka	52	56	75
	vreme [s]	1,56	1,68	2,25
jednostavne RNN	broj odbiraka	/	/	/
	vreme [s]	/	/	/
LSTM	broj odbiraka	25	30	33
	vreme [s]	0,75	0,90	1,00
GRU	broj odbiraka	/	38	/
	vreme [s]	/	1,14	/
CNN	broj odbiraka	54	28	27
	vreme [s]	1,62	0,84	0,81

6. Proširivanje skupa podataka

Tehnike za kreiranje IDS-a koje su zasnovane na podacima zahtevaju veliku količinu podataka prikupljenih iz realnih instalacija u okviru ICS. Međutim, proces prikupljanja podataka iz ICS-a često je praćen određenim ograničenjima prvenstveno vezanim za činjenicu da prikupljanje dovoljne količine podataka zahteva rad sistema u izolovanim uslovima određen vremenski period koji može biti značajan, a naročito u slučaju česte rekonfiguracije sistema u okviru Industrije 4.0. Pored toga, mnogi ICS koriste vlasničke protokole i interfejse, što može otežati povezivanje sa sistemima i akviziciju podataka. Prikupljanje podataka iz izolovanog ICS-a može da poremeti normalan rad sistema i da izazove neželjene posledice kroz prekid usluge tokom određenog vremenskog perioda.

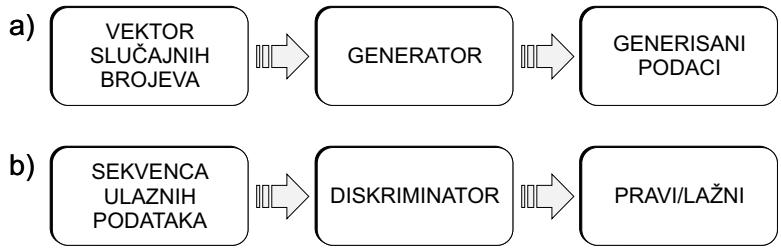
Jedan od načina za smanjenje efekta navedenog ograničenja i obezbeđivanje dovoljne količine podataka jeste primena generativnih tehnika koje se mogu iskoristiti za proširenje ograničenog skupa podataka prikupljenog iz realnog sistema. U ovom poglavlju ispitana je mogućnost proširivanja podataka dobijenih iz realnog sveta korišćenjem Generativnih suparničkih mreža (engl. *Generative Adversarial Networks* – GAN). Skup podataka proširen upotrebom GAN-a korišćen je u nastavku za kreiranje IDS-a.

6.1. Generativne suparničke mreže

GAN predstavljaju metod za kreiranje generativnih modela na osnovu podataka za obučavanje korišćenjem suparničkog procesa [39] između dva igrača:

1. Generatora G čiji je cilj da generiše podatke sa raspodelom sličnom raspodeli podataka za obučavanje (slika 47a) i

2. Diskriminatora D koji treba da prepozna da li podaci dolaze iz originalnog skupa za obučavanje ili ih je kreirao generator. Prepoznavanje se u ovom slučaju svodi na klasifikaciju ulaza kao realnih ili generisanih (lažnih) podataka (slika 47b).



Slika 47: Generativne suparničke mreže: a) Generator; b) Diskriminator

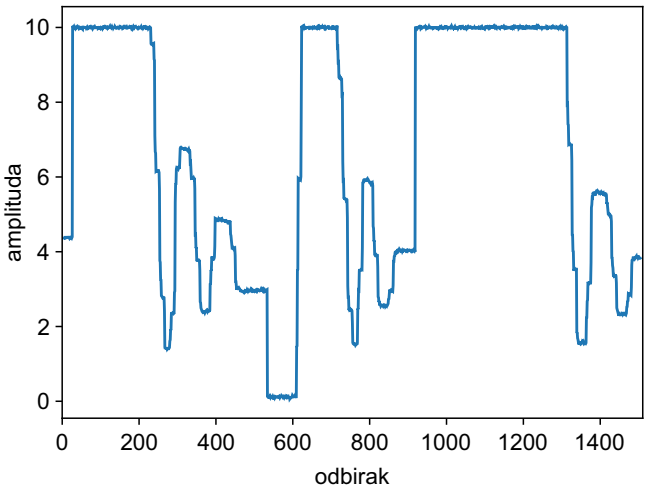
Suparnički proces između dva igrača odvija se tako što generator kreira podatke, prosleđuje ih diskriminatoru i pokušava da ga ubedi da napravi grešku i prepozna generisane podatke kao realne [9, 15]. Po pravilu, ulaz u generator predstavlja vektor nasumičnih vrednosti (vektor latentnih promenljivih), dok se na izlazu iz generatora dobija višedimenzionalni vektor koji predstavlja generisane podatke (slika 47a). Generisani podaci su ujedno i ulaz u diskriminator, dok je na izlazu iz diskriminatora skalar koji predstavlja verovatnoću da su ulazni podaci realni (slika 47b). Generator i diskriminator mogu biti zasnovani na proizvoljno odabranim tehnikama mašinskog učenja, uključujući i DNN.

Obučavanje generatora i diskriminatora izvodi se istovremeno gde generator kreira sekvencu lažnih odbiraka koji se sa sekvencom odbiraka iz skupa za obučavanje dovode diskriminatoru

radi klasifikacije. U zavisnosti od kvaliteta klasifikacije diskriminatora, generator i diskriminator se ažuriraju u cilju generisanja "boljih" lažnih podataka, odnosno ostvarivanja bolje klasifikacije, tim redom. Ovaj proces se ponavlja kroz prethodno definisan broj iteracija.

6.2. Primena GAN-a u proširivanju skupa podataka za kreiranje IDS-a u okviru ICS

U okviru ovog odeljka će biti prikazan postupak generisanja podataka primenom GAN-a korišćenjem relativno malog broja odbiraka signala snimljenog u EpSP (slika 41) [85]. Kao što je ranije navedeno, ceo signal sadrži 400.000 odbiraka prikupljenih korišćenjem sistema za akviziciju podataka i predstavlja vrednost napona između LK_1 i elektropneumatskog regulatora pritiska (slika 42). Frekvencija odabiranja prilikom akvizicije podataka bila je 100 Hz. Na slici 48 prikazan je isečak ovog signala od 1.500 odbiraka.

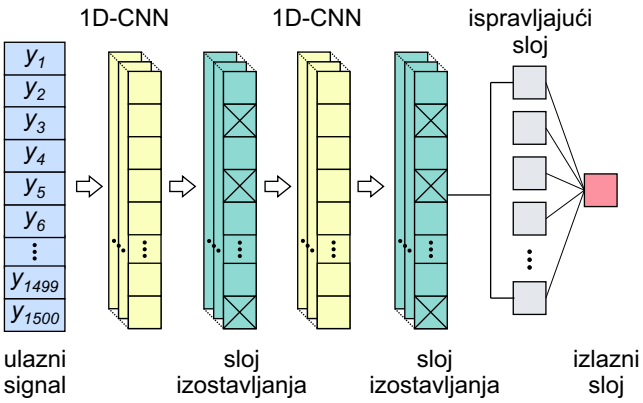


Slika 48: Deo signala prikupljenog iz EpSP

Kao što je pomenuto, arhitekture generatora i diskriminatora mogu biti zasnovane na različitim ML arhitekturama. Za kreiranje generativnog modela signala prikupljenog iz EpSP, diskriminator je definisan CNN arhitekturom, dok je model generatora baziran na potpuno povezanoj neuronskoj mreži – višeslojnom perceptronu. Arhitektura diskriminatora (slika 49) ima:

1. Dva bloka koji sadrže: (i) konvolucioni sloj sa 30 filtera (svaki sa po 10 neurona i ReLU aktivacionom funkcijom) praćen (ii) izostavljajućim slojem (stopa izostavljanja je 0,2);
2. Ispravljajući sloj;
3. Izlazni sloj sa 1 neuronom.

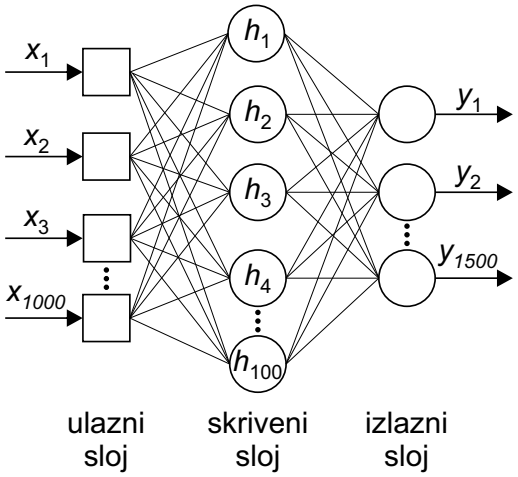
Na ulaz diskriminatora dovodi se generisan/pravi signal dužine 1.500 odbiraka, dok se na izlazu iz mreže dobija estimacija koja govori da li je signal generisan ili pravi (0/1).



Slika 49: Arhitektura diskriminatora

S druge strane, arhitektura generatora (slika 50) sadrži sledeće slojeve:

1. Ulazni latentni vektor sastavljen od 1.000 nasumično generisanih vrednosti;
2. Skriveni sloj sa 100 neurona i sigmoidnom aktivacionom funkcijom;
3. Potpuno povezani izlazni sloj sa 1.500 neurona.



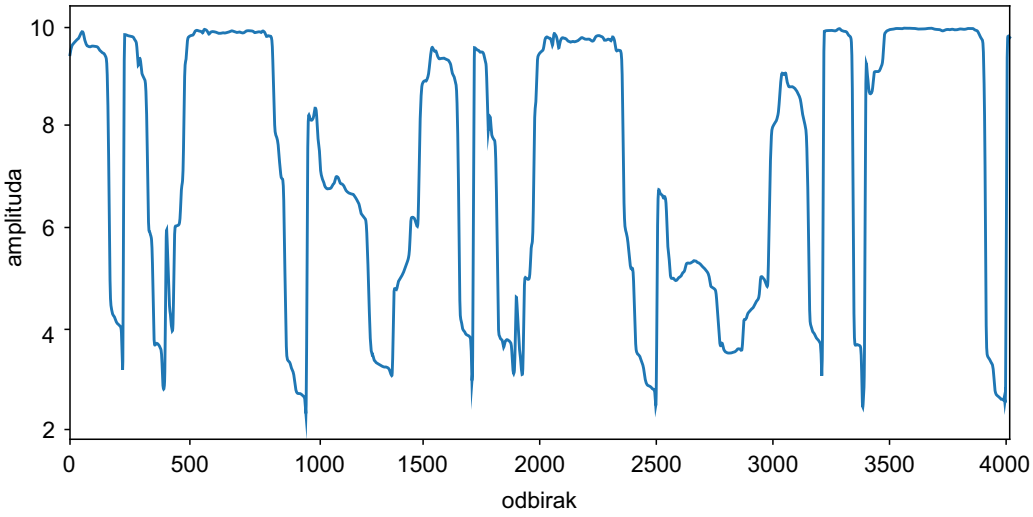
Slika 50: Arhitektura generatora

Dakle, u datoj postavci generator generiše 1.500 odbiraka na izlazu koristeći 1.000 nasumično dobijenih vrednosti na ulazu. Za potrebe obučavanja neuronske mreže korišćen je skup od 2.000 signala $\mathbf{s}_i$, $i \in [0, 1999]$, svaki dužine od 1.500 odbiraka izvučenih iz originalnog skupa podataka na sledeći način:

$$\mathbf{s}_i = [x_{50i+1}, x_{50i+2}, ...x_{50i+1500}] \tag{48}$$

čime je iskorišćeno ukupno 101.450 odbiraka prikupljenih iz EpSP. Proces obučavanja neuronske mreže izveden je kroz 500 epoha.

Primer signala koji je generator generisao nakon obučavanja prikazan je na slici 51. Može se primetiti sličnost između obrazaca koji se ponavljaju u signalu sa slike 51 i sekvence prikazane na delu signala dobijenog iz stvarnog procesa (slika 48).



Slika 51: Primer signala dobijenog korišćenjem generatora

Korišćenjem podataka generisanih pomoću GAN-a, prema proceduri koja je prikazana u poglavlju 4 kreiran je IDS. U fazi pretprocesiranja signala izabran je FIR filter sa propusnim opsegom $[0; 0{,}11\pi]$, nepropusnim opsegom $[0{,}35\pi; \pi]$ i prelaznim regionom između. Kreirani filter sadržao je ukupno 11 koeficijenata čije su vrednosti dobijene korišćenjem Parks-Meklelanovog algoritma. Ulazni podaci dobijeni korišćenjem razvijenog generativnog modela sadrže 266 signala sa po 1.500 odbiraka što je ukupno 391.818 uređenih parova. Generisani podaci podeljeni su na delove za obučavanje, validaciju i izbor modela sa po 70/10/20%, tim redom. Obučavanje neuronske mreže sprovedeno je kroz 5 epoha korišćenjem Adam optimizatora (parametar učenja α=0,001) i funkcije cilja srednje kvadratne greške.

Model koji je zadovoljio definisane kriterijume za odabir modela (predstavljene u poglavlju 4) i izabran je kao optimalni, definisan je sledećom arhitekturom:

- CNN (4 filtera, veličina kernela=2),
- CNN (8 filtera, veličina kernela=2),
- Sloj sažimanja (stepen sažimanja=2),
- CNN (16 filtera, veličina kernela=2),
- CNN (16 filtera, veličina kernela=2),
- Sloj sažimanja (stepen sažimanja=2),
- Ispravljajući sloj,
- Potpuno povezan sloj (30 neurona),
- Potpuno povezan sloj (1 neuron)

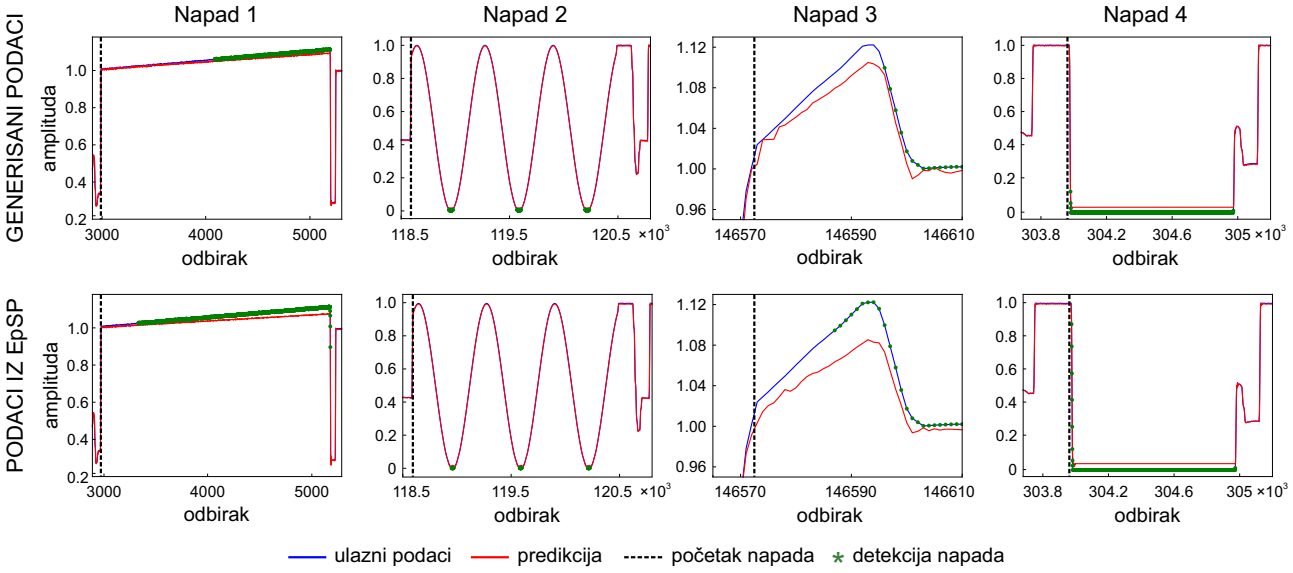
sa ukupno 2.865 obučavajućih parametara. Dužina bafera u ovom slučaju bila je 16.

Ovde je potrebno naglasiti da CNN zasnovan model ponašanja sistema koji je kreiran na osnovu realnih podataka i iskorišćen za detekciju napada na EpSP u okviru poglavlja 5 (tabela 15) ima istu arhitekturu kao i model kreiran na osnovu generisanih podataka. Međutim, za kreiranje modela na osnovu samo realnih podataka (poglavlje 5) upotrebljeno je 399.000 odbiraka, čime je dobijeno 398.973 uređenih parova, dok je upotrebom GAN-a i generisanih podataka broj realnih podataka značajno smanjen na 101.450 odbiraka.

Prilikom kreiranja IDS-a na osnovu generisanih podataka, vrednosti praga detekcije izračunate su korišćenjem izraza (36) odakle je dobijeno da je T=0,00941. S druge strane, za IDS baziran na realnim podacima ovaj prag iznosi T=0,00956 (očigledno je da se radi o izuzetno bliskim vrednostima). Verifikacija kreiranog sistema za detekciju sprovedena je korišćenjem 4 napada koji su prethodno opisani u poglavlju 5 (slika 43). Uslovi za detekciju napada definisani su kao i u prethodnim slučajevima podalgoritmom detekcije (slika 25).

Rezultati detekcije napada IDS-a kreiranog na osnovu generisanih podataka prikazani su na slici 52. Ulazni podaci i ostvarena predikcija prikazani su plavom i crvenom bojom, dok su početak napada i trenutak detekcije napada prikazani crnom isprekidanom linijom i zelenim markerom, tim redom.

Radi poređenja IDS-a kreiranog na osnovu generisanih sa IDS-om kreiranim na osnovu realnih podataka, na slici 52 su paralelno sa rezultatima detekcije napada, korišćenjem IDS-a kreiranog u ovom poglavlju, prikazani i rezultati detekcije korišćenjem IDS-a kreiranog u odeljku 5.2. Oba IDS-a pokazala su odlične performanse i detektovali su sve napade bez lažno pozitivnih rezultata.



Slika 52: Poređenje performansi IDS-a kreiranog na osnovu generisanih podataka i IDS-a kreiranog na osnovu originalnih podataka

Sa slike 52 može se primetiti da je IDS koji koristi model zasnovan na realnim podacima detektovao napad 1 ranije nego IDS kod koga su prilikom obučavanja modela korišćeni generisani podaci. U slučaju napada 2, 3 i 4, razlika između trenutaka detekcije je zanemarljiva. Vrednosti kašnjenja prilikom detekcije napada (izražene brojem odbiraka i vremenom u sekundama) prikazane su u tabeli 24.

Tabela 24: Kašnjenja pri detekciji napada (pristupi bazirani na generisanim podacima i na podacima iz EpSP)

Tehnika	**Kašnjenje**	**Napad 1**	**Napad 2**	**Napad 3**	**Napad 4**
CNN (generisani podaci)	broj odbiraka	1081	358	24	21
	vreme [s]	10,81	3,58	0,24	0,21
CNN (podaci iz EpSP)	broj odbiraka	376	368	16	20
	vreme [s]	3,76	3,68	0,16	0,20

Prikazani rezultati pokazuju da je, na osnovu relativno male količine podataka, primenom GAN-a moguće generisati podatke na kojima će kasnije biti baziran model ponašanja. IDS koji koristi model generisan na taj način pruža gotovo iste rezultate kao i IDS kreiran na bazi podataka prikupljenih sa realne instalacije.

7. Detekcija kibernetskih napada na sekvence dvodimenzionalnih signala

U prethodnim poglavljima ove doktorske disertacije fokus je bio usmeren na jednodimenzionalne signale, odnosno na detekciju kibernetskih napada na vremenske serije. Međutim, u određenim industrijskim aplikacijama informacije koje se prenose putem slika (dvodimenzionalnih – 2D signala) često imaju značajnu ulogu. Pre svega, misli se na sisteme veštačkog gledanja (engl. *vision systems*) u kojima se slika dobijena sa senzora (kamere) koristi kao glavni nosilac informacija. Ovi sistemi uključujući obradu slike predstavljaju oblast koja se brzo razvija sa već brojnim primenama u različitim sferama industrije. Savremene kamere po pravilu predstavljaju pametne uređaje koji sekvence prikupljenih slika šalju ostalim uređajima u okviru upravljačkog sistema različitim komunikacionim protokolima. Samim tim i sekvence slika kao i signali u formi klasičnih vremenskih serija predstavljaju pogodno polje za dejstvo različitih kibernetskih napada. Imajući u vidu specifičnosti ove vrste signala, potrebno je kreirati namenske sisteme zaštite za njih.

Prateći rezultate dobijene za klasične vremenske serije u prethodnim poglavljima, u okviru ovog poglavlja će biti ispitane mogućnosti primene predikcije naredne slike u sekvenci slika za kreiranje IDS-a za ovakve vrste signala. Ovde treba naglasiti da se poslednjih godina intenziviraju istraživanja u oblasti predikcije naredne slike u sekvenci slika, čime se može postići predviđanje kretanja objekata u okruženju, sigurna i efikasna navigacija [43] kao i sprečavanje potencijalnih odstupanja ili nepravilnosti [111] i slično. S obzirom na složenost slika dobijenih korišćenjem sistema veštačkog gledanja, za predikciju naredne slike se po pravilu koriste modeli autoregresije – sledeća slika se estimira na osnovu niza prethodnih slika.

Za kreiranje modela autoregresije sekvence slika u postojećim istraživanjima korišćeni su različiti pristupi zasnovani na mašinskom i dubokom učenju. CNN su našle široku primenu u različitim oblicima kao što su autoenkoderi [41] ili klasične CNN [95]. Autoenkoderi zasnovani na LSTM arhitekturi takođe su korišćeni [108] da bi se izvršila predikcija sledeće slike. Pošto u nekim slučajevima LSTM nije u mogućnosti da dovoljno precizno modelira prostornu strukturu podataka, u [105, 119] uvedene su konvolucione LSTM gde su unutrašnje transformacije LSTM ćelije zamenjene konvolucijom. Na ovaj način, model je sposoban da prepozna prostorne zavisnosti korišćenjem operacije konvolucije, dok je i dalje u mogućnosti da prepozna vremenske zavisnosti pomoću LSTM-a [31].

U ovom poglavlju biće predstavljena procedura kreiranja i evaluacije algoritama namenjenih za detekciju kibernetskih napada na sekvence slika. Celokupna metodologija predložena u ovoj doktorskoj disertaciji (poglavlje 4) prilagođena je i primenjena na 2D problem. Evaluacija performansi razvijenih algoritama za detekciju napada sprovedena je korišćenjem skupa podataka koji je prikupljen u okviru ove doktorske disertacije.

7.1. Oflajn generisanje i odabir autoregresionog modela

Oflajn faza kreiranja IDS-a koja je namenjena za generisanje i odabir modela za slučaj 2D signala u potpunosti prati predloženu metodologiju u kojoj prvi korak podrazumeva pretprocesiranje podataka. U slučaju 2D signala pretprocesiranje obuhvata redukciju veličine slike, normalizaciju signala, kreiranje uređenih parova i mešanje podataka. Redukcija veličine slike sprovodi se u cilju smanjenja računske složenosti, odnosno vremena potrebnog za obučavanje modela. Redukcijom veličine slike ne postiže se manji broj obučavajućih parametara koje model sadrži, ali ovaj korak pretprocesiranja može uticati na ukupan broj računskih operacija koje se izvršavaju tokom primene IDS-a što je u direktnoj korelaciji sa ostvarenim kašnjenjem. Normalizacija signala se kao i u slučaju jednodimenzionalnog signala sprovodi njegovom maksimalnom

apsolutnom vrednošću čime se dobijaju vrednosti u opsegu [-1,1].

Prilikom kreiranja obučavajućih parova postojeći način primene autoregresije neophodno je samo prilagoditi sa 1D na 2D problem. Za razliku od prethodnog problema gde su razmatrani 1D signali i gde je vršena predikcija jedne vrednosti na osnovu v prethodnih vrednosti, u slučaju 2D signala vrši se predikcija jedne slike (dimenzija $n_x \times n_y$ piksela) korišćenjem prethodnih v slika. Stoga, ulazni oblik podataka u ML algoritam za obučavanje koji je za 1D signale definisan izrazom (4) sada se takođe može definisati preko obučavajućih parova na sledeći način [86]:

$$(\mathbf{X}_i, Y_i) \in [([X_1, ..., X_v], X_{v+1}), ([X_2, ..., X_{v+1}], X_{v+2}), ..., ([X_{i-v}, ..., X_{i-1}], X_i), ..., ([X_{n-v}, ..., X_{n-1}], X_n)] \tag{49}$$

Ulazna sekvenca $\mathbf{X}_i$ i odgovarajući odziv Y_i koji figurišu u izrazu (49) definisani su sa:

$$\mathbf{X}_i = [X_{i-v}, ..., X_{i-1}] = \left[\begin{bmatrix} x_{11}^{i-v} & x_{12}^{i-v} & \cdots & x_{1n_y}^{i-v} \\ x_{21}^{i-v} & x_{22}^{i-v} & \cdots & x_{2n_y}^{i-v} \\ \vdots & \vdots & \ddots & \vdots \\ x_{n_x1}^{i-v} & x_{n_x2}^{i-v} & \cdots & x_{n_xn_y}^{i-v} \end{bmatrix}, ..., \begin{bmatrix} x_{11}^{i-1} & x_{12}^{i-1} & \cdots & x_{1n_y}^{i-1} \\ x_{21}^{i-1} & x_{22}^{i-1} & \cdots & x_{2n_y}^{i-1} \\ \vdots & \vdots & \ddots & \vdots \\ x_{n_x1}^{i-1} & x_{n_x2}^{i-1} & \cdots & x_{n_xn_y}^{i-1} \end{bmatrix} \right] \tag{50}$$

$$Y_i = X_i = \begin{bmatrix} x_{11}^{i} & x_{12}^{i} & \cdots & x_{1n_y}^{i} \\ x_{21}^{i} & x_{22}^{i} & \cdots & x_{2n_y}^{i} \\ \vdots & \vdots & \ddots & \vdots \\ x_{n_x1}^{i} & x_{n_x2}^{i} & \cdots & x_{n_xn_y}^{i} \end{bmatrix} \tag{51}$$

gde $x_{rq}^i, r \in [1, n_x], q \in [1, n_y]$ predstavlja intenzitet osvetljenosti piksela na poziciji (r,q) u okviru i-te slike. Izrazi (50) i (51) predstavljaju primer za sliku sa jednim slojem osvetljenja (npr. monohromatska ili crno bela slika). Međutim, predložena metodologija nije ograničena na ovaj slučaj i može se lako primeniti i na slikama sa više komponenata (na primer slike u boji poput RGB (engl. *Red Green Blue*) slike koja je definisana sa tri 2D matrice intenziteta crvene, zelene i plave boje).

Mešanje podataka kao poslednji korak pretprocesiranja sprovodi se isto kao i kod 1D signala gde se raspored obučavajućih parova nasumično meša čime se postiže da delovi namenjeni za obučavanje/validaciju/izbor modela pripadaju istoj raspodeli.

7.1.1. Tehnike korišćene za razvoj autoregresionog modela za sekvence 2D signala

Shodno obliku podataka i predviđenoj nameni, za kreiranje autoregresionog modela ispitana je mogućnost korišćenja dve različite tehnike uključujući pritom i njihovu kombinaciju:

1. dvodimenzionalne CNN (2D-CNN),
2. dvodimenzionalne konvolucione LSTM (2D-ConvLSTM),
3. kombinacija 2D-CNN i 2D-ConvLSTM.

Kako su struktura i princip funkcionisanja 1D-CNN i RNN prethodno objašnjeni u poglavlju 4, u nastavku će biti prikazane samo jednačine koje opisuju 2D-CNN i 2D-ConvLSTM tehnike.

2D-CNN

U slučaju 2D-CNN diskretna konvolucija definisana je na sledeći način:

$$(f * g)(i, j) = \sum_{j=1}^{n} \sum_{i=1}^{m} f(i, j) g(m - i + 1, n - j + 1) \tag{52}$$

gde (m,n) predstavljaju dimenzije filtera koji je po pravilu višestruko manji od veličine slike. Izlaz iz konvolucionog sloja definisan je sa [38]:

$$\mathbf{h}_d^l = a_l \left(\sum_k \mathbf{W}_{dk}^l * \mathbf{y}_k^{l-1} + b_d^l \right) \tag{53}$$

gde $\mathbf{W}_{dk}^l$ i b_d^l predstavljaju matricu težinskih koeficijenata i *bias* za sloj l. Broj filtera u trenutnom (l) i prethodnom (l-1) sloju označeni su sa d i k, tim redom. Izlaz iz prethodnog sloja (ulaz u trenutni sloj) označen je sa $\mathbf{y}_k^{l-1}$, dok a_l označava aktivacionu funkciju u sloju l.

2D-ConvLSTM

Za razliku od osnovnog oblika i strukture LSTM-a (prikazane u poglavlju 4) gde se matrično množenje koristi za transformacije ulaza i rekurentne transformacije izlaza [38] – relacije (17)-(22), u 2D-ConvLSTM se ulazne i rekurentne transformacije vrše primenom konvolucije [105]. Shodno tome, vektori odgovarajućih kapija ($\mathbf{i}(t)$, $\mathbf{f}(t)$ i $\mathbf{o}(t)$), vektor unutrašnjeg stanja ćelije $\mathbf{c}(t)$ i vektor skrivenog stanja $\mathbf{h}(t)$ 2D-ConvLSTM sloja definisani su na sledeći način [105]:

$$\mathbf{i}(t) = \sigma \left(\mathbf{W}_{xi} * \mathbf{x}(t) + \mathbf{W}_{hi} * \mathbf{h}(t-1) + \mathbf{W}_{ci} \odot \mathbf{c}(t-1) + \mathbf{b}_i \right) \tag{54}$$

$$\mathbf{f}(t) = \sigma \left(\mathbf{W}_{xf} * \mathbf{x}(t) + \mathbf{W}_{hf} * \mathbf{h}(t-1) + \mathbf{W}_{cf} \odot \mathbf{c}(t-1) + \mathbf{b}_f \right) \tag{55}$$

$$\mathbf{c}(t) = \mathbf{f}(t) \odot \mathbf{c}(t-1) + \mathbf{i}(t) \odot tanh \left(\mathbf{W}_{xc} * \mathbf{x}(t) + \mathbf{W}_{hc} * \mathbf{h}(t-1) + \mathbf{b}_c \right) \tag{56}$$

$$\mathbf{o}(t) = \sigma \left(\mathbf{W}_{xo} * \mathbf{x}(t) + \mathbf{W}_{ho} * \mathbf{h}(t-1) + \mathbf{W}_{co} \odot \mathbf{c}(t) + \mathbf{b}_o \right) \tag{57}$$

$$\mathbf{h}(t) = \mathbf{o}(t) \odot tanh \left(\mathbf{c}(t) \right) \tag{58}$$

gde $\mathbf{x}$(t) označava ulazni vektor, dok $\mathbf{W}_{\llcorner}$ i $\mathbf{b}_{\llcorner}$ predstavljaju odgovarajuće matrice težinskih koeficijenata i *bias*-e, tim redom. Na ovaj način 2D-ConvLSTM kombinuje prednosti CNN-a u modeliranju prostornih karakteristika i prednosti LSTM-a u modeliranju vremenskih karakteristika podataka, što dovodi to toga da 2D-ConvLSTM predstavlja veoma koristan alat za modeliranje prostorno-vremenskih zavisnosti 2D signala.

7.1.2. Opšte arhitekture autoregresionih modela

U cilju redukcije prostora za pretragu odgovarajućeg modela i smanjenja vremena neophodnog za njegov razvoj, a shodno postupku kreiranja autoregresionog modela za 1D signale (poglavlje 4), predložene su opšte arhitekture za svaku od korišćenih tehnika (slika 53). Sve predložene arhitekture sadrže ulazni sloj sa ulaznim formatom podataka (x,y,v) i izlazni sloj u kojem je oblik podataka (x,y,1) određen brojem izlaznih parametara. Razliku između arhitektura definišu slojevi koji se nalaze između ulaznog i izlaznog sloja.

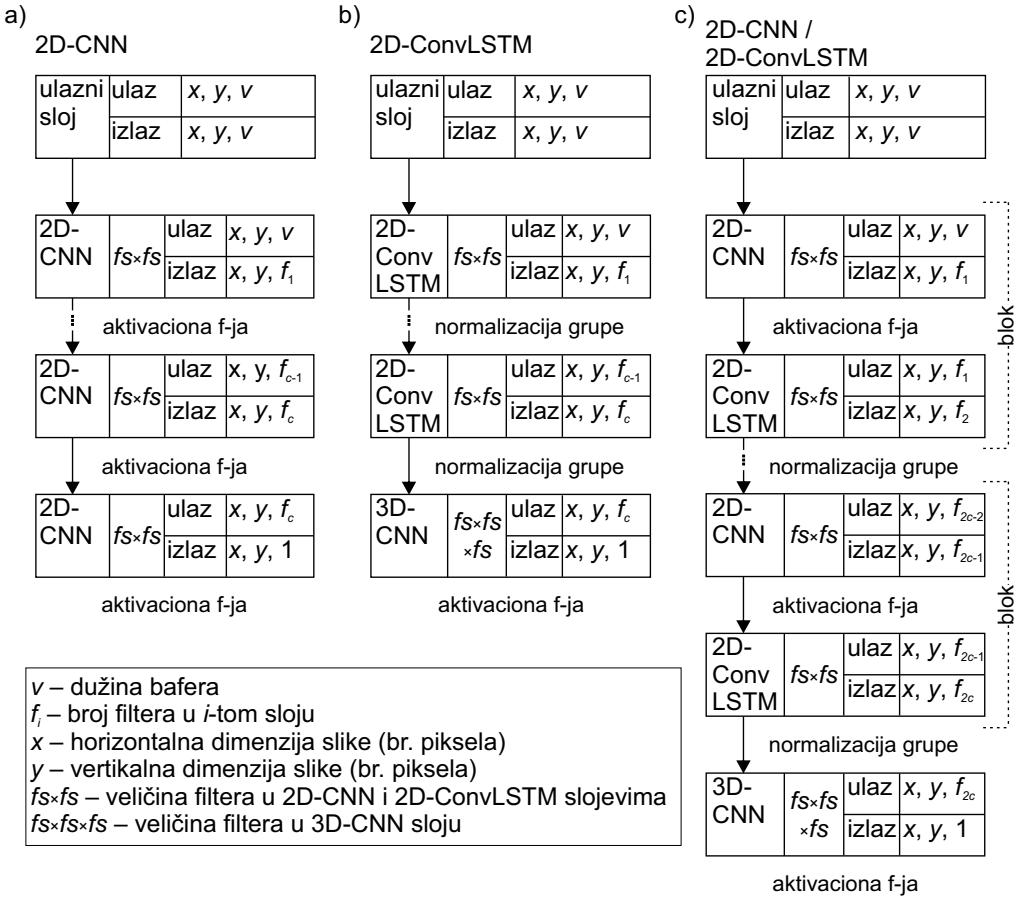
U slučaju 2D-CNN arhitekture (slika 53a), nakon ulaznog sloja sledi ukupno c 2D-CNN slojeva koji su definisani brojem filtera $f_i, i \in \{1, ...c\}$ i veličinom filtera $fs \times fs$ (usvojene su iste veličine filtera u svim slojevima). Nakon svakog 2D-CNN sloja primenjena je aktivaciona funkcija. Stoga, za kreiranje 2D-CNN modela na bazi opisane arhitekture biće varirane vrednosti sledećih parametara:

- c – broj 2D-CNN slojeva;

- f_i – broj filtera u 2D-CNN sloju i;
- $fs \times fs$ – veličina filtera u 2D-CNN slojevima.

Predložena 2D-ConvLSTM arhitektura (slika 53b) u kojoj nakon ulaznog sloja slede 2D-ConvLSTM slojevi po strukturi je slična 2D-CNN arhitekturi pa se broj slojeva, broj filtera u svakom 2D-ConvLSTM sloju i veličina filtera označavaju isto kao i kod 2D-CNN arhitekture. S druge strane, postoje dve značajne razlike između ovih arhitektura. Prva razlika ogleda se u primeni tehnike normalizacije grupe (engl. *batch normalization*) nakon svakog 2D-ConvLSTM sloja. Ova tehnika podrazumeva normalizaciju parametara neuronske mreže sa ciljem da se proces obučavanja učini stabilnijim i vremenski manje zahtevnim [50]. Drugu razliku predstavlja izlazni sloj u formi trodimenzionalnog CNN-a (3D-CNN) nakon kojeg se primenjuje aktivaciona funkcija. Parametri čije će vrednosti biti varirane u okviru ove arhitekture su:

- c – broj 2D-ConvLSTM slojeva;
- f_i – broj filtera u 2D-ConvLSTM sloju i;
- $fs \times fs$ – veličina filtera u 2D-ConvLSTM slojevima.



Slika 53: Opšte arhitekture modela za: a) 2D-CNN; b) 2D-ConvLSTM; c) 2D-CNN/2D-ConvLSTM

2D-CNN/2D-ConvLSTM arhitektura (slika 53c) definisana je sa ukupno c sekvencijalno poređanih blokova gde se jedan blok sastoji od 2D-CNN i 2D-ConvLSTM slojeva. Ukupan broj filtera i veličina filtera u 2D-CNN ili 2D-ConvLSTM sloju i označeni su sa $f_i, i \in \{1, ...2c\}$ i $fs \times fs$, tim redom. Nakon svakog 2D-CNN sloja primenjuje se aktivaciona funkcija, dok se nakon 2D-ConvLSTM slojeva primenjuje normalizacija grupe. Kao i u primeru 2D-ConvLSTM

arhitekture, na izlazu se koristi 3D-CNN sloj koji prati aktivaciona funkcija. Generisanje jedinstvenih 2D-CNN/2D-ConvLSTM modela biće izvršeno varijacijom vrednosti sledećih parametara:

- c – broj 2D-CNN/2D-ConvLSTM blokova;
- f_i – broj filtera u 2D-CNN ili 2D-ConvLSTM sloju i;
- $fs \times fs$ – veličina filtera u 2D-CNN ili 2D-ConvLSTM slojevima.

Nezavisno od izabrane arhitekture, prilikom kreiranja autoregresionog modela varirana je vrednost dužine bafera v, tj. broj prethodnih slika koje se koriste za predikciju naredne slike. Kao i u slučaju predloženih arhitektura za modeliranje 1D signala, arhitekture predložene za 2D signale (slika 53) pokazale su se kao najbolje, ali razvijena metodologija za detekciju kibernetskih napada nije ograničena na korišćenje isključivo ovih arhitektura.

7.1.3. Izbor odgovarajućeg autoregresionog modela

Izbor odgovarajućeg modela za sekvence 2D signala sproveden je po istim kriterijumima koji su definisani u predloženoj metodologiji za 1D signale: prvi kriterijum je definisan izrazima (34) i (35) i odnosi se na razliku između prikupljenih i estimiranih podataka, dok se drugi kriterijum odnosi na proveru da li će neki deo podataka snimljenih u toku normalnog rada biti okarakterisan kao napad. Kako su u slučaju 1D signala u izrazima (32) i (33) srednja vrednost i standardna devijacija izračunate na osnovu razlike ostvarene y_i i estimirane vrednosti $\hat{y}_i$ koji predstavljaju skalare, te izraze je potrebno prilagoditi 2D problemu gde su ostvareni izlaz Y_i i predikcija $\hat{Y}_i$ zapisani u obliku matrica:

$$\mu_k = \frac{\frac{1}{n_k - v} \sum_{i=v+1}^{n_k} \sum_{r=1}^{n_x} \sum_{q=1}^{n_y} \left| Y_i^k(r,q) - \hat{Y}_i^k(r,q) \right|}{n_x \cdot n_y}, \; k = \{obučavanje, izbor\} \tag{59}$$

$$\sigma_k = \frac{\sqrt{\frac{1}{n_k - v} \sum_{i=v+1}^{n_k} \sum_{r=1}^{n_x} \sum_{q=1}^{n_y} \left(\left| Y_i^k(r,q) - \hat{Y}_i^k(r,q) \right| - \mu_k \right)^2}}{n_x \cdot n_y}, \; k = \{obučavanje, izbor\} \tag{60}$$

gde n_k predstavlja broj slika u skupovima za obučavanje/izbor modela, dok $Y_i^k(r,q)$ i $\hat{Y}_i^k(r,q)$ označavaju intenzitet osvetljenosti piksela na poziciji (r,q) i-te ostvarene i estimirane slike, tim redom. Dimenzije slike (izražene brojem piksela) u horizontalnom i vertikalnom pravcu označene su sa n_x i n_y. Kako će se detekcija napada vršiti na nivou piksela, a ne na nivou cele slike, vrednosti μ_k i σ_k koje se računaju u izrazima (59) i (60) podeljene su sa ukupnim brojem piksela na slici $(n_x \cdot n_y)$.

Vrednost praga detekcije T izračunava se na osnovu prethodno definisanog izraza (36). Za potrebe detekcije napada na 2D signalima kreira se prozor detekcije dimenzija p_x i p_y koji na svakoj poziciji ima vrednost T. S obzirom na razlike između 1D i 2D signala, u podalgoritmu detekcije napada (slika 25) će se umesto prekoračenja z uzastopnih vrednosti sada proveravati da li vrednosti razlike intenziteta piksela u prozoru ostvarene i estimirane slike prekoračuju prag detekcije T, što se može zapisati na sledeći način:

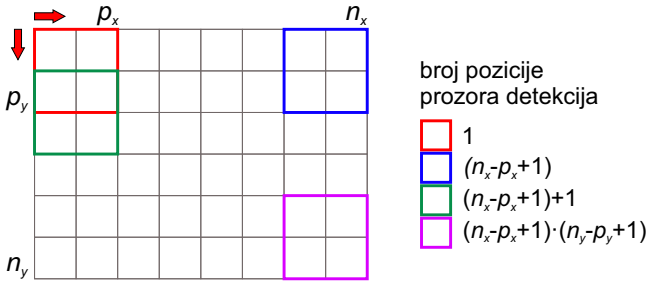
$$r_{det} = \left\| \begin{bmatrix} x_{11}^{ij} & x_{12}^{ij} & \cdots & x_{1p_y}^{ij} \\ x_{21}^{ij} & x_{22}^{ij} & \cdots & x_{2p_y}^{ij} \\ \vdots & \vdots & \ddots & \vdots \\ x_{p_x1}^{ij} & x_{p_x2}^{ij} & \cdots & x_{p_xp_y}^{ij} \end{bmatrix} - \begin{bmatrix} \hat{x}_{11}^{ij} & \hat{x}_{12}^{ij} & \cdots & \hat{x}_{1p_y}^{ij} \\ \hat{x}_{21}^{ij} & \hat{x}_{22}^{ij} & \cdots & \hat{x}_{2p_y}^{ij} \\ \vdots & \vdots & \ddots & \vdots \\ \hat{x}_{p_x1}^{ij} & \hat{x}_{p_x2}^{ij} & \cdots & \hat{x}_{p_xp_y}^{ij} \end{bmatrix} \right\| > \begin{bmatrix} T & T & \cdots & T \\ T & T & \cdots & T \\ \vdots & \vdots & \ddots & \vdots \\ T & T & \cdots & T \end{bmatrix} \tag{61}$$

gde x_{rq}^{ij} i $\hat{x}_{rq}^{ij}$ predstavljaju intenzitet osvetljenosti piksela na poziciji (r,q) j-tog prozora na i-toj realnoj i estimiranoj slici, tim redom. Ukoliko apsolutna razlika između realne i estimirane slike na svim pozicijama unutar jednog prozora detekcije prekorači prag detekcije, uslov iz jednačine (61) biće zadovoljen čime promenljiva r_{det} dobija vrednost 1. Vraćanje promenljive r_{det} koja ima vrednost 1 u glavni algoritam onlajn detekcije (slika 25) znači da je napad detektovan.

Kako su dimenzije prozora detekcije obično višestruko manje od dimenzija slike, celokupna slika uključuje se u razmatranje pomeranjem prozora detekcije u horizontalnom i vertikalnom pravcu. Ukupan broj pozicija prozora detekcije (bp) na slici može se izračunati na sledeći način:

$$bp = \left(floor\left(\frac{n_x - p_x}{s}\right) + 1\right) \cdot \left(floor\left(\frac{n_y - p_y}{s}\right) + 1\right) \tag{62}$$

gde s označava korak (izražen u pikselima) sa kojim se prozor detekcije pomera kada menja poziciju, dok *floor* predstavlja funkciju koja uzima celobrojnu vrednost broja. Iz izraza (62) može se primetiti da broj pozicija prozora detekcije zavisi isključivo od njegovih dimenzija ($p_x \times p_y$), dimenzija slike ($n_x \times n_y$) i koraka s. Na slici 54 prikazane su četiri različite pozicije prozora detekcije. U datom primeru radi jednostavnosti korišćen je korak s=1.



Slika 54: Pomeranje prozora detekcije

Na primeru koji je prikazan na slici 54 vidi se da prozor detekcije prvo izvršava kretanje u horizontalnom pravcu sve do poslednjeg piksela n_x, nakon čega se u vertikalnom pravcu pomera za s piksela i opet iterativno ponavlja kretanje u horizontalnom pravcu do piksela n_x.

Nakon što se generiše predikcija slike $\hat{Y}_i$ na osnovu prethodnih v slika, izračunava se apsolutna razlika između intenziteta osvetljenosti piksela te slike i piksela odgovarajuće realne slike Y_i. Dobijena 2D matrica razlike ima iste dimenzije kao i slike Y_i i $\hat{Y}_i$ ($n_x \times n_y$ piksela). Onlajn detekcija napada vrši se pomeranjem prozora detekcije (na način koji je prikazan na slici 54) po 2D matrici razlike. Svaki put kada prozor detekcije zauzme novu poziciju proverava se da li vrednosti 2D matrice razlike na trenutnoj poziciji prozora detekcije prekoračuju vrednost praga detekcije T. Ukoliko sve vrednosti 2D matrice razlike koje preklapa trenutna pozicija prozora detekcije ($p_x \cdot p_y$ vrednosti) prekoračuju vrednost T smatra se da je napad detektovan.

7.2. Primena razvijene metodologije za kreiranje IDS-a za detekciju napada na sekvence 2D signala

U okviru ovog odeljka biće opisani rezultati primene IDS razvijenih korišćenjem predložene metodologije u detekciji napada na odabrane sekvence 2D signala. Vrednosti parametara koje će biti varirane tokom kreiranja autoregresionih modela za sve tri predložene arhitekture navedene su u tabelama 25, 26 i 27. U okviru ovih tabela mogu se primetiti određene sličnosti poput vrednosti parametara dužine bafera i veličine filtera u odgovarajućim slojevima. Naime, predložene dužine bafera variraju u opsegu od 2 (što predstavlja najmanju moguću vrednost na osnovu koje se mogu očekivati odgovarajući rezultati predikcije naredne slike) do 10 što je u ovom slučaju predstavljalo maksimalnu vrednost. Smatra se da bi veće vrednosti dužine bafera

izazvale preveliko kašnjenje u primeni modela u realnom vremenu. S druge strane, varirane vrednosti veličine filtera iste su za sva tri tipa arhitektura.

Tabela 25: Varijacija vrednosti hiperparametara – 2D-CNN

Parametar	**Oznaka**	**Vrednost**
dužina bafera	v	2, 5, 10
broj 2D-CNN slojeva	c	2, 3, 4, 5
veličina filtera u 2D-CNN slojevima	$fs \times fs$	2×2, 3×3, 4×4
broj filtera u 1. 2D-CNN sloju	f_1	4, 8, 16, 32
broj filtera u 2. 2D-CNN sloju	f_2	4, 8, 16, 32
broj filtera u 3. 2D-CNN sloju	f_3	8, 16, 32, 64
broj filtera u 4. 2D-CNN sloju	f_4	8, 16, 32, 64
broj filtera u 5. 2D-CNN sloju	f_5	8, 16, 32, 64

Tabela 26: Varijacija vrednosti hiperparametara – 2D-ConvLSTM

Parametar	**Oznaka**	**Vrednost**
dužina bafera	v	2, 5, 10
broj 2D-ConvLSTM slojeva	c	2, 3, 4
veličina filtera u 2D-ConvLSTM slojevima	$fs \times fs$	2×2, 3×3, 4×4
broj filtera u 1. 2D-ConvLSTM sloju	f_1	4, 8, 16, 32
broj filtera u 2. 2D-ConvLSTM sloju	f_2	4, 8, 16, 32
broj filtera u 3. 2D-ConvLSTM sloju	f_3	8, 16, 32, 64
broj filtera u 4. 2D-ConvLSTM sloju	f_4	8, 16, 32, 64

Tabela 27: Varijacija vrednosti hiperparametara – 2D-CNN/2D-ConvLSTM

Parametar	**Oznaka**	**Vrednost**
dužina bafera	v	2, 5, 10
broj 2D-CNN/2D-ConvLSTM blokova	c	1, 2, 3
veličina filtera u 2D-CNN i 2D-ConvLSTM slojevima	$fs \times fs$	2×2, 3×3, 4×4
broj filtera u 1. 2D-CNN sloju	f_1	4, 8, 16, 32
broj filtera u 1. 2D-ConvLSTM sloju	f_2	4, 8, 16, 32
broj filtera u 2. 2D-CNN sloju	f_3	8, 16, 32, 64
broj filtera u 2. 2D-ConvLSTM sloju	f_4	8, 16, 32, 64
broj filtera u 3. 2D-CNN sloju	f_5	8, 16, 32, 64
broj filtera u 3. 2D-ConvLSTM sloju	f_6	8, 16, 32, 64

Broj 2D-CNN slojeva u okviru 2D-CNN arhitekture variran je u opsegu od 2 do 5. Razmotrene su i 2D-CNN arhitekture koje obuhvataju samo jedan 2D-CNN, ali se ti slučajevi prema predloženim kriterijumima nisu pokazali kao odgovarajući. Broj 2D-CNN slojeva veći od 5 u ovom slučaju prouzrokovao je ML modele sa velikim brojem obučavajućih parametara. Slično tome, u slučaju 2D-ConvLSTM arhitekture broj 2D-ConvLSTM slojeva variran je u opsegu od 2 do 4, gde su manje i veće vrednosti izostavljene iz istih razloga kao u 2D-CNN arhitekturi. Kada je 2D-CNN/2D-ConvLSTM arhitektura u pitanju, broj blokova je variran u opsegu od 1 do 3 što je podrazumevalo ukupno 2, 4 ili 6 2D-CNN i 2D-ConvLSTM slojeva.

Skupovi vrednosti koji su predloženi za broj filtera u 2D-CNN i 2D-ConvLSTM slojevima za svaku od tri arhitekture određeni su empirijskim putem gde su manje vrednosti f_i dovodile do nedovoljno dobrih modela koji nisu zadovoljavali definisane kriterijume za izbor modela, dok

je njihovo dalje uvećavanje prouzrokovalo veliki broj obučavajućih parametara. U slučaju 2D-ConvLSTM i 2D-CNN/2D-ConvLSTM arhitektura, veličina filtera u izlaznom 3D-CNN sloju odgovarala je veličini filtera u 2D-CNN ili 2D-ConvLSTM slojevima. Na primer, ukoliko je veličina filtera u 2D-CNN ili 2D-ConvLSTM sloju bila 2×2 u okviru te arhitekture veličina filtera u 3D-CNN sloju bila je 2×2×2. Shodno tome, vrednosti ovog parametra nisu navedene u tabelama 26 i 27.

Izuzimajući izlazne slojeve (2D-CNN i 3D-CNN) u kojima je primenjena sigmoidna aktivaciona funkcija, u svim 2D-CNN slojevima korišćena je ReLU aktivaciona funkcija. Prilikom obučavanja 2D-ConvLSTM i 2D-CNN/2D-ConvLSTM modela korišćena je funkcija cilja srednje kvadratne greške, dok je u slučaju 2D-CNN izabrana binarna unakrsna entropija. *Adam* optimizator sa parametrom učenja α=0,001 izabran je kod svih arhitektura kao pogodna tehnika za zadatak optimizacije. Skup podataka podeljen je na podskupove za obučavanje/validaciju/izbor modela sa udelima 70%/10%/20%.

U okviru dela za pretprocesiranje podataka izvršena je normalizacija intenziteta osvetljenosti piksela, čime je opseg vrednosti 0-255 sveden na opseg 0-1. Pored toga, dimenzije slike $n_x \times n_y$ redukovane na 160×120 piksela. Kako su gubici prilikom obučavanja i validacije ostali gotovo nepromenjeni nakon 50 epoha, taj broj epoha izabran je kao optimalan. Da bi se izbegla eventualna predubeđenja koja mogu nastati kao uticaj stohastike, obučavanje neuronskih mreža svake jedinstvene arhitekture ponovljeno je tri puta.

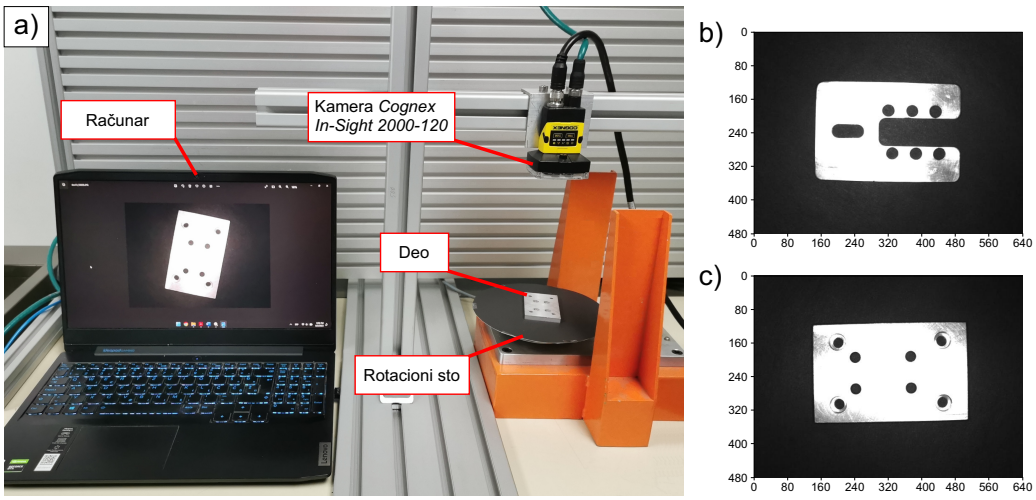
Za slike dimenzija 160×120 piksela koje su korišćene u postupku oflajn obučavanja modela i onlajn detekcije napada, kao optimalna veličina prozora detekcije usvojena je $p_x \times p_y$=10×10 piksela. Treba napomenuti da izbor dimenzije prozora detekcije predstavlja kompromis između osetljivosti primenjene tehnike na napade i broja operacija potrebnih za primenu algoritma za detekciju. Stoga, smatra se da bi prozor detekcije manjih dimenzija povećao računsku složenost i vreme potrebno da se izvrši provera cele slike, dok bi korišćenje većih dimenzija prozora detekcije za posledicu moglo imati nemogućnost detekcije manjih delova slike koji su izmenjeni usled dejstva napada. Usvojeni korak pomeranja prozora detekcije bio je s=1 piksel.

Kreiranje svih ML modela sprovedeno je korišćenjem *Python* programskog jezika na bazi *Tensorflow v2.3.0* platforme. U poređenju sa 1D signalima, neophodni proračunski kapaciteti višestruko su veći kada je u pitanju rad sa 2D signalima pa je korišćena radna stanica koja se sastoji od dva procesora *Intel Xeon Silver 4208*, tri grafičke karte *Nvidia Quadro RTX6000* i 192GB DDR4 RAM.

7.2.1. Eksperimentalna instalacija za prikupljanje podataka za obučavanje

Za potrebe razvoja ML modela u okviru ove doktorske disertacije u Laboratoriji za automatizaciju proizvodnje na Univerzitetu u Beogradu – Mašinskom fakultetu kreirana je nova eksperimentalna postavka (slika 55a) namenjena za prikupljanje slika iz realnog sistema. Ova postavka sastoji se od kamere *Cognex In-Sight 2000-120*, rotacionog stola na kojem se postavljaju delovi i računara koji služi za akviziciju slika. *Cognex In-Sight 2000-120* je monohromatska kamera sa rezolucijom 640×480 piksela koja ima mogućnost generisanja maksimalno 75 slika (frejmova) u sekundi (engl. *frames per second* – fps) [17]. Akvizicija slika izvršena je korišćenjem softvera *In-Sight Explorer* [18] instaliranog na računaru koji je sa kamerom bio povezan putem Eternet protokola.

Tokom akvizicije podataka rotacioni sto na kojem je u jednom trenutku bio postavljen uvek samo jedan deo, vršio je obrtno kretanje tako da su delovi slikani u različitim orijentacijama. Kamera je bila postavljena upravno na rotacioni sto, ali se njena udaljenost od rotacionog stola razlikovala u zavisnosti od sekvence slika koja je snimana. Takođe, snimljene sekvence slika razlikuju se i po poziciji dela u odnosu na centar rotacionog stola. Ukupno 16 različitih delova korišćeno je prilikom snimanja 82 sekvence što je rezultiralo sa preko 640.000 slika. Najkraća i najduža snimljena sekvenca sadrže 293 i 10.000 slika, tim redom.



Slika 55: Eksperimentalna postavka: a) fotografija postavke; b) primer slike iz sekvence *deo1*; c) primer slike iz sekvence *deo2*

Za evaluaciju performansi razvijene metodologije izabrane su dve sekvence slika nazvane *deo1* i *deo2* čiji su primeri slika prikazani na slici 55b i slici 55c, tim redom; obe sekvence sadrže po 10.000 slika. Iako nije striktno definisano, za obe sekvence slika prilikom jedne pune rotacije stola zabeleženo je 110 slika. Delovi koji su korišćeni za kreiranje sekvenci slika izrađeni su od aluminijuma pa je zbog refleksije svetlosti koja se javlja prilikom slikanja bilo pogodno koristiti crnu podlogu. Sivi segmenti koji su jasno primetni na delovima posledica su određenih neravnina na snimanim površinama.

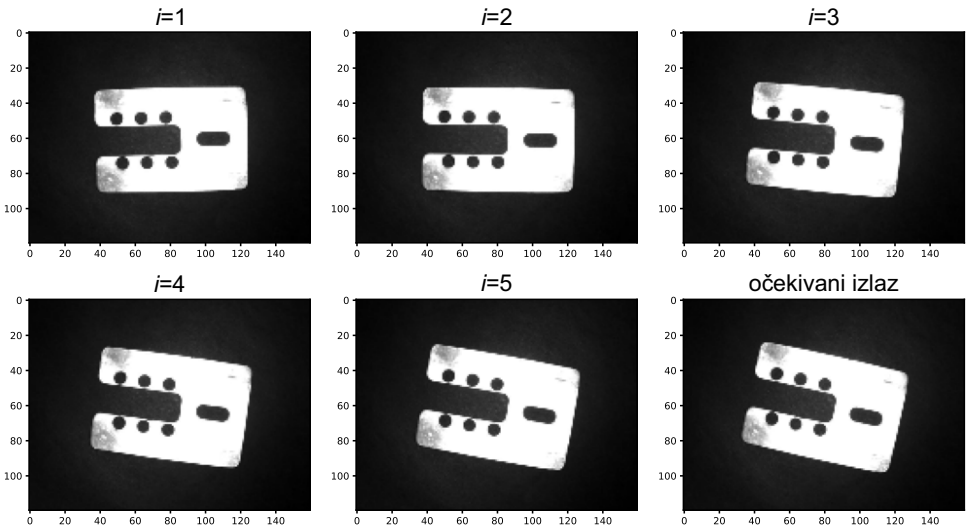
7.2.2. Generisanje IDS-a za izabrane sekvence 2D signala

Prema predloženoj metodologiji za razvoj ML modela, varijacijom vrednosti parametara iz tabela 25, 26 i 27, za obe sekvence slika kreirani su odgovarajući modeli čije su arhitekture prikazane u tabeli 28. Pored arhitektura, u tabeli 28 prikazan je i broj obučavajućih parametara za svaki ML model. Broj parametara ML modela varirao je od 993 u slučaju 2D-CNN modela za *deo1* do 25.313 za 2D-CNN/2D-ConvLSTM model za istu sekvencu slika. Izabrana veličina filtera koju su koristili prikazani ML modeli varirala je u opsegu od 2×2 do 3×3. Interesantno je da su svi odabrani modeli koristili dužinu bafera v=5.

Tabela 28: Arhitekture kreiranih modela za sekvence slika *deo1* i *deo2*

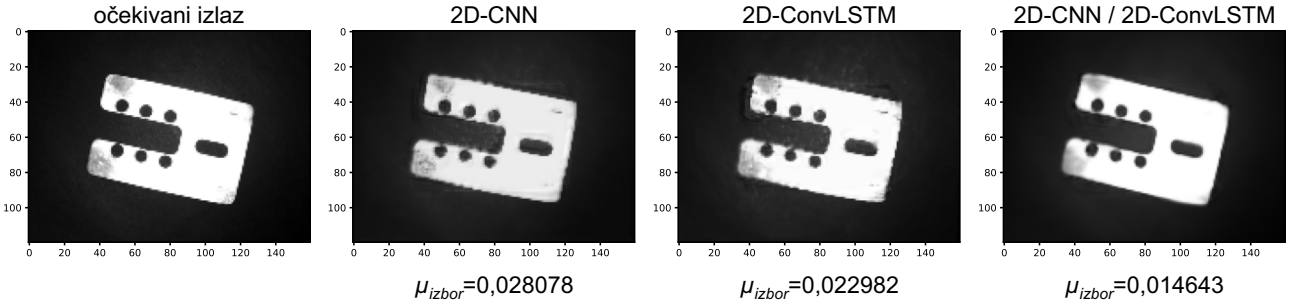
Tehnika \ Sekvenca slika		*deo1*	*deo2*
2D-CNN	f_1-...-$f_c, c \in \{4,5\}$, fs, v	8-8-8-8, fs=2×2, v=5	16-16-8-8-16, fs=2×2, v=5
	broj parametara	993	2.753
2D-ConvLSTM	f_1-f_2, fs, v	8-8, fs=2×2, v=5	8-8, fs=3×3, v=5
	broj parametara	3.905	8.697
2D-CNN/2D-ConvLSTM	f_1-f_2-f_3-f_4, fs, v	16-16-8-8, fs=3×3, v=5	8-8-8-8, fs=2×2, v=5
	broj parametara	25.313	4.689

Kako su svi ML modeli iz tabele 28 koristili dužinu bafera v=5, u cilju grafičke ilustracije mogućnosti kreiranih modela da generišu odgovarajuću predikciju odabran je bafer od 5 sukcesivno snimljenih slika iz sekvence *deo1* i prikazan je očekivani izlaz (slika 56). Na ovoj slici može se primetiti promena orijentacije dela u sceni koja je uslovljena rotacijom stola prilikom snimanja sekvence.



Slika 56: Bafer od 5 slika sa očekivanim izlazom

Na slici 57 prikazani su rezultati primene ML modela iz tabele 28 (modeli za *deo1*) na primeru sekvence sa slike 56. Na osnovu srednje vrednosti razlike između ostvarenog izlaza i predikcije koja je izračunata na podskupu za izbor modela korišćenjem izraza (59), kao i na osnovu vizuelnog prikaza generisanih slika, može se primetiti da je u ovom slučaju najbolja predikcija postignuta primenom 2D-CNN/2D-ConvLSTM modela.



Slika 57: Grafička reprezentacija rezultata predikcije naredne slike

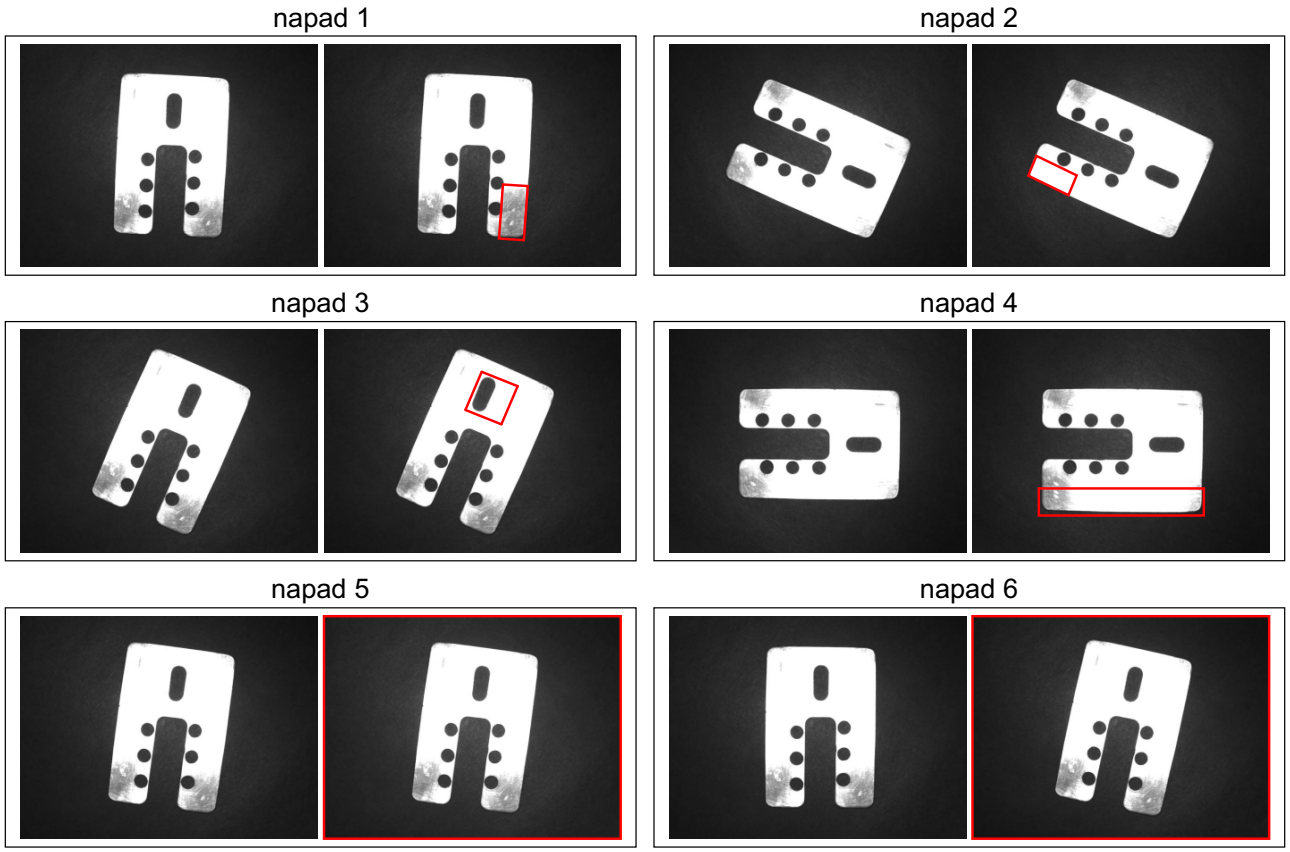
7.2.3. Rezultati primene IDS-a u detekciji napada

Performanse detekcije IDS zasnovanih na ML modelima iz tabele 28 ispitane su na po 6 napada za svaku od sekvenci slika. Napadi na sekvence slika *deo1* i *deo2* (slike 58 i 59) podrazumevali su:

- napad 1 – proširivanje regiona na kojem se nalazi površina sive boje;
- napad 2 – uklanjanje sivog segmenta sa jednog kraja dela i njegova zamena belim segmentom;
- napad 3 – izmeštanje određenog segmenta na drugu poziciju unutar granica dela;
- napad 4 – proširivanje jedne strane dela;
- napad 5 – promena intenziteta osvetljenosti cele slike za 5%;
- napad 6 – izbacivanje 4 uzastopne slike iz sekvence slika.

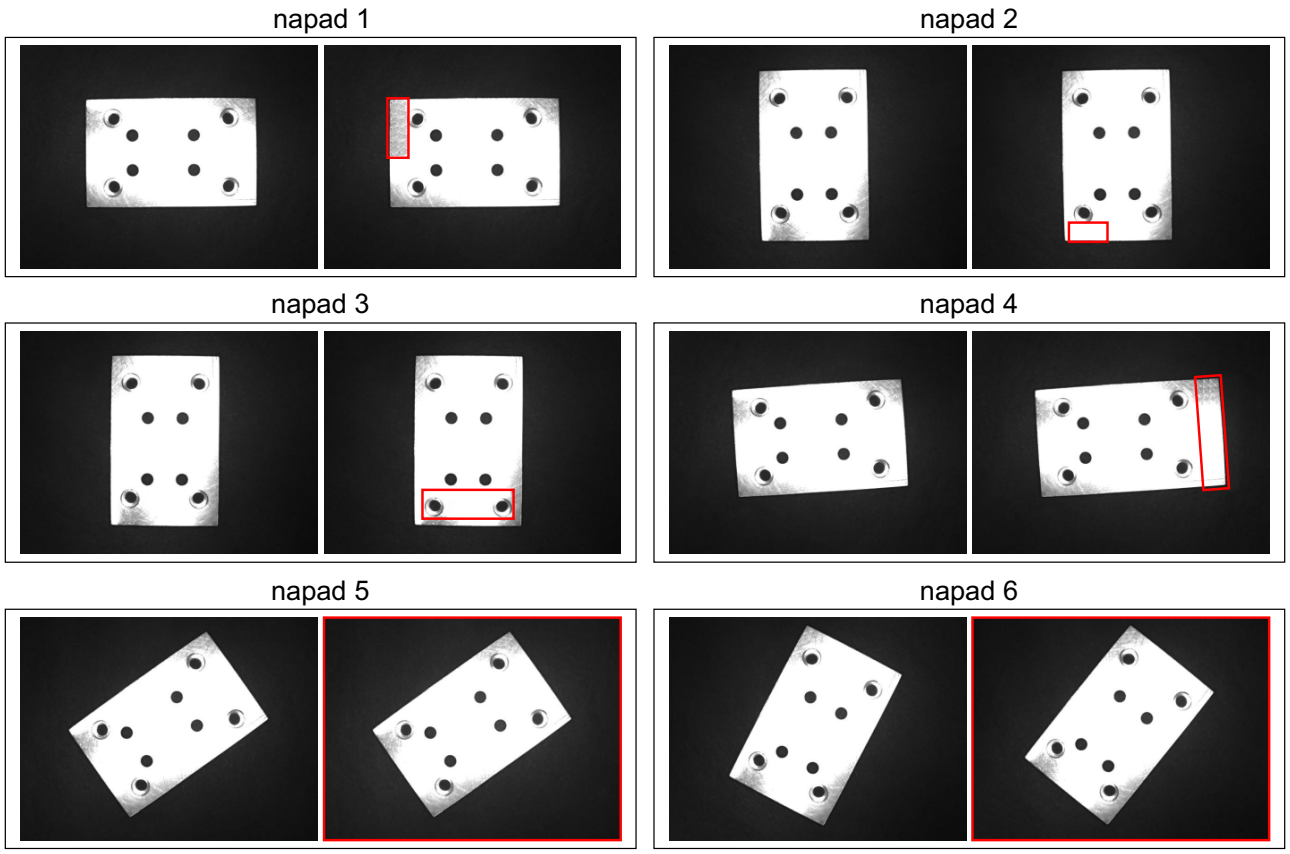
Na slikama 58 i 59 crvenim pravougaonicima označeni su segmenti slike koji su na određeni način izmenjeni dejstvom napada. Može se primetiti da su u slučaju napada 5 i 6 crvenim pravougaonicima obuhvaćene cele slike. Naime, napad 5 se odnosi na promenu intenziteta osvetljenosti svih piksela na slici, dok se kod napada 6 izostavljanjem 4 slike daje lažna informacija o sledećem stanju. Treba napomenuti da su svi napadi sem napada 6 bili usmereni samo na po jednu sliku.

Ukoliko bi se prema taksonomiji koja je predložena u ovoj doktorskoj disertaciji (slika 5) napadi svrstavali u određenu klasu, prvih pet napada bili bi svrstani u napade obmanom, dok se napad 6 može smatrati napadom uskraćivanja pristupa servisu – uklanjanje poruke. Razmatrani napadi pokazuju na koje sve načine napadač može uticati na promenu dimenzija dela, strukturu dela, poziciju i orijentaciju dela unutar scene, dostupnost slika itd, čime se ističe neophodnost i značaj razvoja mehanizama za detekciju napada koji bi sprečili posledice navedenih izmena sekvenci slika.



Slika 58: Napadi na slike iz sekvence *deo1* (u slučaju svih napada leva slika označava očekivani izlaz, dok je sa desne strane prikazana izmenjena slika kao posledica dejstva napada)

Rezultati primene IDS koji su zasnovani na odabranim ML modelima prikazani su u tabeli 29. Predstavljeni rezultati pokazuju da je ostvaren visok nivo detekcije (u proseku 10/12 napada je detektovano) primenom svih odabranih modela. IDS zasnovani na 2D-CNN/2D-ConvLSTM modelima pokazali su se kao najbolji i detektovali su svih 12 napada. S druge strane, modeli bazirani na 2D-ConvLSTM nisu uspeli da detektuju napad 3 ni u jednoj sekvenci slika, kao ni napad 4 u sekvenci slika *deo1*. Napad 3 na sekvencu slika *deo2* ostao je jedini nedetektovan prilikom primene modela zasnovanih na 2D-CNN.

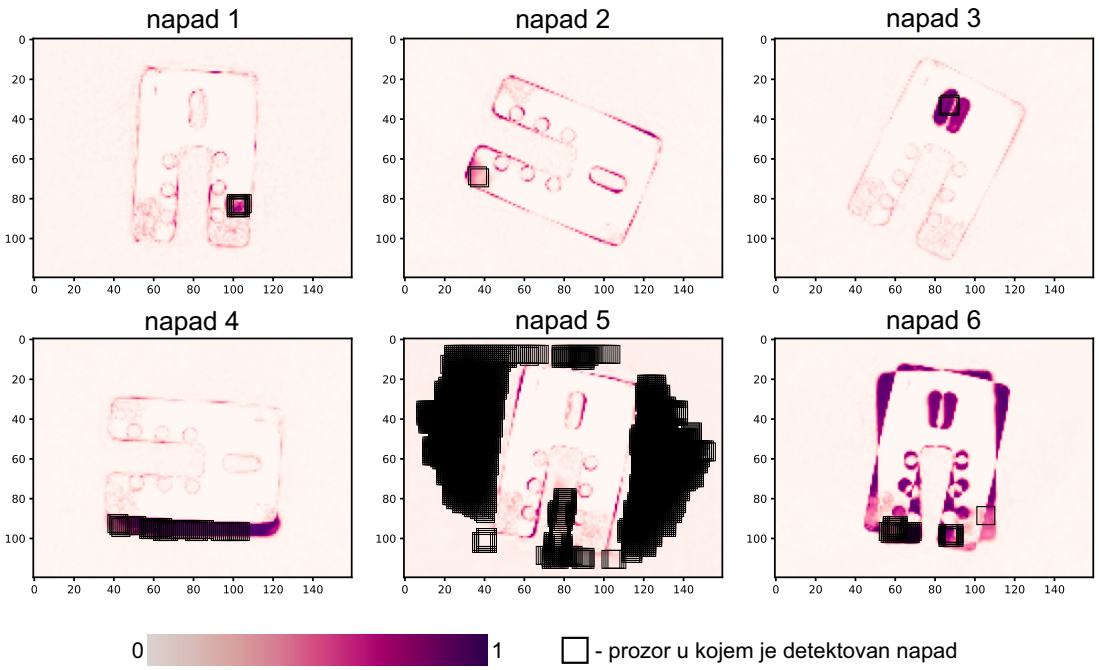


Slika 59: Napadi na slike iz sekvence *deo2* (u slučaju svih napada leva slika označava očekivani izlaz, dok je sa desne strane prikazana izmenjena slika kao posledica dejstva napada)

Tabela 29: Rezultati detekcije napada na sekvence slika *deo1* i *deo2*

Sekvenca slika / Tehnika	*deo1*–napadi						*deo2*–napadi						Ukupno
	1	2	3	4	5	6	1	2	3	4	5	6	
2D-CNN	✓	✓	✓	✓	✓	✓	✓	✓	✗	✓	✓	✓	11/12
2D-ConvLSTM	✓	✓	✗	✗	✓	✓	✓	✓	✗	✓	✓	✓	9/12
2D-CNN/2D-ConvLSTM	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	12/12

Za svih 6 napada na slike iz sekvence *deo1* na slici 60 prikazana je razlika između realne i slike estimirane primenom 2D-CNN/2D-ConvLSTM modela. Razlika koja može biti u opsegu od [0,1] prikazana je različitim nijansama ljubičaste boje gde svetlija nijansa označava manju razliku. Za većinu napada u regionu u kojem su delovali jasno su uočljive razlike između ostvarene i slike dobijene predikcijom. Pored toga, na slici 60 crnim kvadratima su označene pozicije prozora detekcije u trenutku detekcije napada.



Slika 60: Detektovani napadi na sekvenci *deo1* primenom IDS-a koji je zasnovan na 2D-CNN/ConvLSTM modelu

8. Zaključak

U fokusu ove doktorske disertacije su kibernetski napadi na proizvodne resurse i industrijske sisteme upravljanja. Naime, savremeni zahtevi tržišta u pogledu masovne kastomizacije podrazumevaju brzo prilagođavanje proizvodnje različitim vrstama proizvoda, što dovodi do primene novih pristupa i tehnologija u upravljanju proizvodnim sistemima. Odgovor na ove zahteve daje se kroz integraciju kibernetsko-fizičkih sistema i Industrijskog interneta stvari u proizvodne pogone i prelazak sa centralizovanih na potpuno distribuirane sisteme upravljanja gde lokalni kontroleri razmenjuju relevantne informacije kako bi postigli željeno ponašanje sistema u celini.

Međutim, komunikacija između umreženih uređaja u okviru ICS-a (posebno putem bežičnih veza), kao i sve veća povezanost ICS-a sa spoljnim mrežama (kao što je internet) otvaraju nova pitanja sajber bezbednosti. ICS više nisu izolovana ostrva, već su izloženi različitim kibernetskim pretnjama kao što su kibernetski napadi. Dejstvo kibernetskih napada može prouzrokovati ozbiljne posledice u vidu disfunkcije industrijskih sistema, ugrožavanja životne sredine pa čak i ljudskih života. Stoga, razvoj i implementacija bezbednosnih sistema, pre svega sistema za detekciju kibernetskih napada predstavljaju ključni pravac za sprečavanje posledica koje napadi mogu prouzrokovati.

Dosadašnja istraživanja koja su bila usmerena na razvoj sistema za detekciju kibernetskih napada dovela su do značajnih rezultata. Međutim, metode razvijene u prethodnim istraživanjima poseduju ograničenja različite prirode. U slučaju metoda koje su zasnovane na nadgledanom učenju, pored toga što su razvijene korišćenjem malog broja scenarija napada i ne pokrivaju široki spektar mogućih napada, one su testirane na istim ili vrlo sličnim napadima koji su korišćeni u procesu generisanja modela, čime se stvaraju predubeđenja koja kasnije često imaju posledice u pogledu loših svojstava generalizacije. Kod razvijenih metoda zasnovanih na samonadgledanom učenju glavni nedostaci ogledaju se u ručnom postavljanju vrednosti određenih hiperparametara poput praga detekcije. Vrednosti ovih hiperparametara često su određivane po principu pokušaja i greške uključujući u taj postupak i napade koji se nalaze u signalima. Na taj način dovode se u pitanje performanse detekcije napada koji prethodno nisu bili prisutni u razmatranom skupu podataka.

Kako bi se rešili navedeni nedostaci u postojećim pristupima, predmet istraživanja ove disertacije bili su sistemi za detekciju kibernetskih napada u okviru ICS, sa fokusom na razvoj metodologije za kreiranje IDS-a za sisteme sa kontinualnim upravljanjem. U skladu sa navedenim, osnovni naučni cilj disertacije odnosio se na razvoj metodologije za kreiranje algoritama koji predstavljaju osnove sistema za detekciju kibernetskih napada u okviru industrijskih sistema upravljanja, a on je i ostvaren kroz postizanje odgovarajućih specifičnih ciljeva. Naime, u ovoj doktorskoj disertaciji razvijena je originalna metodologija za generisanje IDS-a zasnovana na principima samonadgledanog učenja koja se može koristiti kako za sisteme iz kojih je moguće prikupiti dovoljnu količinu podataka tako i za sisteme iz kojih je količina podataka koja se može prikupiti ograničena. Metodologija se sastoji od dve faze: 1) oflajn generisanje modela na bazi tehnika mašinskog učenja i 2) onlajn detekcija kibernetskih napada. Prva faza se odnosi na generisanje modela podataka koji se razmenjuju između uređaja u normalnim uslovima rada (bez napada) i na najvišem nivou podrazumeva pretprocesiranje signala, razvoj različitih ML modela i izbor odgovarajućeg modela za dalje kreiranje IDS-a.

Postupak pretprocesiranja signala kao bitan aspekat za pripremu podataka i ostvarivanje dobrih performansi modela, obuhvata filtriranje, normalizaciju, kreiranje uređenih parova i mešanje podataka. Odabir odgovarajuće tehnike za filtriranje signala zasnovan je na statističkim metrikama (MSE, PSNR i μ) uzimajući u obzir i kašnjenja koja primena određene tehnike prouzrokuje. Prema tako postavljenim kriterijumima, FIR filter izabran je kao odgovarajuća tehnika za filtriranje signala i u nastavku je korišćen u te svrhe prilikom kreiranja svih modela.

Za razvoj ML modela istraživane su tri klase tehnika mašinskog učenja: 1) regresija nose-

ćim vektorima, 2) konvolucione neuronske mreže i 3) rekurentne neuronske mreže. Cilj primene navedenih tehnika bio je da modeliraju transmitovane podatke, odnosno da pronađu karakteristike i zavisnosti koje se u njima nalaze. Nezavisno od izabrane ML tehnike, modeliranje je sprovedeno principom autoregresije gde se trenutne vrednosti podataka estimiraju na osnovu njihovih prethodnih vrednosti. Pritom, korišćena je univarijatna autoregresija što znači da je svaka razmatrana promenljiva (signal) bila posebno modelirana. Samim tim izvršeno je potpuno rasprezanje signala koji se prenose između različitih uređaja i obezbeđena je mogućnost implementacije sistema za detekciju napada u skladu sa arhitekturom sistema upravljanja – uvek na prijemnom uređaju. Na ovaj način napravljen je veliki iskorak u odnosu na postojeće tehnike zasnovane na multivarijatnoj regresiji za koje često nije bilo jasno na kom bi uređaju bilo moguće implementirati sistem za detekciju napada.

U okviru doktorske disertacije razvijena je i originalna metoda za automatski izbor autoregresionog modela zasnovanog na tehnikama mašinskog učenja (uključujući i sve njegove parametre) koji će se efikasno koristiti u procesu detekcije napada u realnom, proračunski i energetski ograničenom sistemu. Ova metoda za izbor modela koristi dva kriterijuma, od kojih prvi kriterijum korišćenjem statističkih parametara utvrđuje da model nije sklon preobučavanju ili nedovoljnom obučavanju, dok drugi kriterijum razmatra robusnost modela na poremećaje koji nastaju kao posledica različitih uticaja radnog okruženja, mehaničkih svojstava itd. Kako je svaka od razmatranih ML tehnika okarakterisana relativno velikim brojem različitih arhitektura i hiperparametara, u okviru disertacije je sprovedeno istraživanje koje je rezultovalo generisanjem preporučenih opštih arhitektura sa delimično određenim pozicijama i brojem slojeva u okviru modela.

ML model koji je zadovoljio postavljene kriterijume i odabran je u oflajn fazi generisanja modela kao odgovarajući, predstavljao je jedan od ulaza u onlajn algoritam za detekciju napada. U onlajn algoritmu vršena je provera da li apsolutno odstupanje između primljene vrednosti i predikcije premašuje prag detekcije za određeni broj uzastopno primljenih vrednosti. Ukoliko bi taj uslov bio ispunjen, smatralo bi se da je na sistem izvršen napad. Naveden pristup se po pravilu koristi u svim metodama za detekciju napada zasnovanim na samonadgledanom učenju pri čemu se vrednost praga određuje metodom pokušaja i greške i to na osnovu signala sa napadima postavljajući pitanje generalizacije tako generisanog praga za slučajeve napada koji nisu razmatrani prilikom njegovog određivanja.

U okviru ove doktorske disertacije predložena je metoda za automatsko određivanje praga detekcije na osnovu statističkih karakteristika odstupanja između stvarnih i estimiranih vrednosti koji je isključivo funkcija podataka, tj. ne predstavlja jedan od parametara koji je potrebno na bilo koji način podešavati. Shodno navedenom, za svaki razmatrani signal izračunata je posebna vrednost praga detekcije, što nije slučaj u istraživanjima koja su koristila multivarijatni pristup i u kojima je često korišćena jedna vrednost praga detekcije za sve signale, što često nije odgovarajuće rešenje s obzirom da su različiti signali karakterisani i različitom dinamikom. Takođe, originalan način određivanja vrednosti praga detekcije u ovoj doktorskoj disertaciji prevazilazi probleme loših svojstava generalizacije i stvaranja predubeđenja koji su prisutni u velikom broju postojećih istraživanja, a koji su posledica određivanja ovog hiperparametra po principu pokušaja i greške.

Performanse algoritama za detekciju napada koji su kreirani korišćenjem predložene metodologije verifikovane su na javno dostupnom skupu podataka i skupovima podataka dobijenih sa eksperimentalne instalacije kroz dve studije slučaja koje se odnose na jednodimenzionalne signale i dve studije slučaja koje se odnose na sekvence slika (dvodimenzionalne signale). U prvoj studiji slučaja koja je podrazumevala SWaT skup podataka, sprovedena je uvodna analiza koja je obuhvatala primenu svih razmatranih ML tehnika na 5 signala i kojom je utvrđeno da su algoritmi za detekciju napada bazirani na CNN modelima dali najbolje rezultate. Iz tog razloga, CNN modeli korišćeni su u sveobuhvatnoj analizi koja je uključivala signale sa 22

senzora iz SWaT skupa podataka. Performanse detekcije upoređene su sa postojećim pristupima koji su predloženi od strane drugih autora i utvrđeno je da su algoritmi predloženi u ovoj doktorskoj disertaciji dali najbolje rezultate u pogledu tri razmatrane metrike: F_1 skor po odbirku (ostvareno 0,902), *tačnost* (ostvareno 97,846%) i *FPR* (ostvareno 0,135%).

Poređenje performansi algoritama za detekciju napada na SWaT izvršeno je i po kriterijumu broja detektovanih napada. Kada se iz razmatranja isključe napadi koji nisu izazvali nikakav uticaj na sistem i uzmu u obzir samo napadi koji su prouzrokovali određene promene, može se zaključiti da su u odnosu na sve ostale pristupe IDS predložen u ovoj doktorskoj disertaciji i IDS predložen u [59] detektovali najveći broj napada (ukupno 30). Dalje poređenje ova dva pristupa dovelo je do zaključka da mehanizam za detekciju napada iz [59] ima bitne nedostatke u odnosu na predloženu metodologiju. Navedeni nedostaci ogledaju se u: 1) načinu određivanja vrednosti praga detekcije (u pristupu iz [59] za ovu svrhu korišćeni su i signali sa napadima što odgovara nadgledanom učenju), 2) kašnjenju prilikom detekcije napada u realnom vremenu (korišćena dužina bafera u [59] bila je u opsegu od 50 do 300 odbiraka što je višestruko veće od kašnjenja prouzrokovanog primenom IDS-a razvijenog u ovoj doktorskoj disertaciji koje je iznosilo 27 odbiraka), 3) kompleksnosti ML modela (veličina modela od 1.585 KB u [59] je višestruko veća od 102,7 KB što predstavlja srednju veličinu CNN modela korišćenih za SWaT skup podataka u ovoj doktorskoj disertaciji) i 4) mogućnosti implementacije (nije jasno na kom uređaju bi IDS razvijen u [59] koji koristi model kreiran multivarijatnim pristupom mogao biti implementiran, s obzirom da u sistemu ne postoji uređaj koji ima uvid u sve senzorske i aktuatorske signale pod dejstvom napada; prilikom projektovanja metodologije u okviru ove doktorske disertacije vođeno je računa o arhitekturi upravljačkog sistema odnosno korišćene su samo informacije (senzorski signali) dostupne uređajima na koje se IDS implementira).

Rezultati dobijeni sprovedenom analizom potvrđuju **prvu i drugu hipotezu** postavljenu u okviru ove doktorske disertacije: 1) da autoregresioni modeli podataka čija se komunikacija vrši između elemenata sistema upravljanja distribuiranih na pametne uređaje, a koji su kreirani korišćenjem tehnika mašinskog i dubokog učenja, mogu predstavljati osnovu za kreiranje sistema za detekciju napada na komunikacione veze u okviru sistema upravljanja proizvodnim resursima i 2) da je iz familije autoregresionih modela moguće automatski odabrati model kao i sve ostale parametre algoritma za detekciju napada zasnovanog na tom modelu koji će se efikasno koristiti u procesu detekcije napada u realnom sistemu uzimajući u obzir arhitekturu sistema upravljanja.

Druga studija slučaja podrazumevala je proveru performansi algoritama za detekciju napada na dva skupa podataka koji su u okviru ove doktorske disertacije prikupljeni sa, u te svrhe razvijenog, elektropneumatskog sistema za pozicioniranje (EpSP). Prvi skup podataka korišćen je za inicijalna istraživanja gde su pristupi zasnovani na SVR i CNN tehnikama dali najbolje rezultate (detektovali su sva 4 generisana napada). Na drugom skupu podataka sprovedena je detaljna analiza i prema opisanoj proceduri kreirani su modeli za 4 različita signala korišćenjem svih razmatranih tehnika (ukupno 20 ML modela). Svi razvijeni algoritmi za detekciju napada su implementirani i testirani na realnoj instalaciji, a u okviru disertacije prikazana je procedura za njihovu implementaciju na kontrolerima pametnih uređaja u okviru EpSP. Rezultati primene algoritama za detekciju napada na realnoj instalaciji prikazani su na primeru jednog signala pri čemu je bilo neophodno detektovati 3 napada. SVR, LSTM i CNN algoritmi pokazali su se kao najuspešniji i detektovali su sva tri napada bez lažno pozitivnih rezultata. Implementacija nijednog algoritma nije prouzrokovala kašnjenja koja bi na bilo koji način i u bilo kojoj meri ugrozila funkcionalnost sistema i izvršavanje definisanog zadatka upravljanja. Vrednosti kašnjenja prilikom detekcije napada varirale su u opsegu od 25 odbiraka (0,75 sekundi) do 75 odbiraka (2,25 sekundi) u zavisnosti od primenjene tehnike i napada. Na ovaj način dokazana je **treća hipoteza** da je primenom algoritama za detekciju napada zasnovanih na mašinskom i dubokom učenju moguće u realnom vremenu u okviru proračunski i energetski ograničenih kibernetsko-

fizičkih sistema postići visok nivo detekcije napada ne izazivajući pritom kašnjenja koja će ugroziti funkcionalnost sistema.

Prilikom kreiranja IDS-a jedan od preduslova za uspešnu primenu tehnika zasnovanih na podacima (u koje spadaju SVR, RNN i CNN) jeste velika količina podataka prikupljenih iz realnih instalacija u okviru ICS. Međutim, zbog različitih ograničenja neretko nije moguće prikupiti dovoljne količine podataka iz realnog sistema. Stoga, mogućnost proširivanja podataka razmatrana je u okviru ove doktorske disertacije kroz primenu generativnih suparničkih mreža. Za drugu studiju slučaja, na osnovu podataka koji su generisani primenom GAN-a, kreiran je IDS čije su performanse upoređene sa IDS-om kreiranim na osnovu originalnih podataka. Primena oba IDS-a obezbedila je detekciju svih razmatranih napada bez lažno pozitivnih rezultata. Može se zaključiti da je na osnovu relativno male količine podataka moguće generisati podatke na kojima će biti zasnovan IDS čijom se primenom postižu gotovo isti rezultati detekcije napada kao i primenom IDS-a koji je kreiran na bazi podataka prikupljenih sa realne instalacije.

Pored prikazanih rezultata primene predložene metodologije na 1D signale, u poglavlju 7 predstavljena je procedura kreiranja i evaluacije algoritama namenjenih za detekciju kibernetskih napada na sekvence 2D signala. Za razvoj ML modela ispitana je mogućnost primene arhitektura zasnovanih na 2D-CNN slojevima, 2D-ConvLSTM slojevima, kao i na njihovoj kombinaciji. Performanse razvijenih algoritama za detekciju napada testirane su na dve sekvence 2D signala koje su kreirane u okviru ove doktorske disertacije. Primenom IDS baziranih na odabranim ML modelima ostvaren je visok nivo detekcije kibernetskih napada. Kao najbolji pokazali su se IDS zasnovani na modelima sa 2D-CNN/2D-ConvLSTM (kombinovanom) arhitekturom koji su detektovali sve razmatrane napade. Ovim rezultatima pokazano je da primena metodologije predložene u ovoj doktorskoj disertaciji nije ograničena samo na 1D signale, već se uspešno može koristiti i za detekciju napada na signale 2D oblika.

Buduća istraživanja treba da doprinesu daljem napretku u domenu sajber bezbednosti u okviru ICS-a što može biti od ključnog značaja kako bi se obezbedila njihova sigurna i pouzdana funkcionalnost u eri Industrije 4.0. Pravci daljeg istraživanja biće usmereni na:

- Razvoj IDS-a koji ima mogućnost da prilikom detekcije napada odredi i njegovu klasu u skladu sa taksonomijom koja je predložena u ovoj doktorskoj disertaciji;
- Integraciju ML modela u IDS namenjen za sisteme sa diskretnim događajima koji uključuju elemente sistema sa kontinualnom prirodom;
- Razvoj opštih modela za kreiranje IDS-a za specifične zadatke upravljanja i vrste napada poput upravljanja kretanjem i desinhronizacije taktova kontrolera;
- Analizu potencijalnih posledica kibernetskih napada na ICS, koja može uključivati simulacije različitih scenarija napada kako bi se bolje razumeli potencijalni rizici i razvili efikasni sistemi za detekciju napada.

Literatura

[1] Abokifa, A. A., Haddad, K., Lo, C. and Biswas, P. [2019], 'Real-time identification of cyber-physical attacks on water distribution systems via machine learning–based anomaly detection techniques', *Journal of Water Resources Planning and Management* **145**(1), 04018089.

[2] Ahmed, C. M., Palleti, V. R. and Mathur, A. P. [2017], Wadi: a water distribution testbed for research in the design of secure cyber physical systems, *in* 'Proceedings of the 3rd international workshop on cyber-physical systems for smart water networks', pp. 25–28.

[3] Al-Abassi, A., Karimipour, H., Dehghantanha, A. and Parizi, R. M. [2020], 'An ensemble deep learning-based cyber-attack detection in industrial control system', *IEEE Access* **8**, 83965–83973.

[4] Aoudi, W., Iturbe, M. and Almgren, M. [2018], Truth will out: Departure-based process-level detection of stealthy attacks on control systems, *in* 'Proceedings of the ACM Conference on Computer and Communications Security', pp. 817–831.

[5] Arm Keil [2019], 'MDK-ARM - Keil uVision', https://www.keil.com/download. datum pristupa: 22.08.2023.

[6] Asghar, M., Hu, Q. and Zeadally, S. [2019], 'Cybersecurity in industrial control systems: Issues, technologies, and challenges', *Computer Networks* **165**.

[7] Boateng, E. A., Bruce, J. and Talbert, D. A. [2022], 'Anomaly detection for a water treatment system based on one-class neural network', *IEEE Access* **10**, 115179–115191.

[8] Branco, P., Torgo, L. and Ribeiro, R. P. [2016], 'A survey of predictive modeling on imbalanced domains', *ACM Computing Surveys (CSUR)* **49**(2), 1–50.

[9] Brownlee, J. [2019], *Generative adversarial networks with python: deep learning generative models for image synthesis and image translation*, Machine Learning Mastery.

[10] Carvalho, L. K., Wu, Y.-C., Kwong, R. and Lafortune, S. [2018], 'Detection and mitigation of classes of attacks in supervisory control systems', *Automatica* **97**, 121–133.

[11] Chandola, V., Banerjee, A. and Kumar, V. [2009], 'Anomaly detection: A survey', *ACM computing surveys (CSUR)* **41**(3), 1–58.

[12] Chaparro, L. and Akan, A. [2018], *Signals and Systems using MATLAB*, Academic Press.

[13] Cherepanov, A. and Lipovsky, R. [2016], 'Blackenergy–what we really know about the notorious cyber attacks', *Virus Bull. October* .

[14] Cho, K., Van Merriënboer, B., Gulcehre, C., Bahdanau, D., Bougares, F., Schwenk, H. and Bengio, Y. [2014], 'Learning phrase representations using rnn encoder-decoder for statistical machine translation', *arXiv preprint arXiv:1406.1078* .

[15] Chollet, F. [2021], *Deep learning with Python*, Simon and Schuster.

[16] Chung, J., Gulcehre, C., Cho, K. and Bengio, Y. [2014], 'Empirical evaluation of gated recurrent neural networks on sequence modeling', *arXiv preprint arXiv:1412.3555* .

[17] Cognex [2023*a*], 'In-Sight 2000 Vision Sensors', https://www.cognex.com/products/machine-vision/vision-sensors/in-sight-2000-vision-sensors. datum pristupa: 22.08.2023.

[18] Cognex [2023*b*], 'In-Sight Explorer Software', https://support.cognex.com/en/downloads/in-sight/software-firmware. datum pristupa: 22.08.2023.

[19] Conti, M., Donadel, D. and Turrin, F. [2021], 'A survey on industrial control system testbeds and datasets for security research', *IEEE Communications Surveys & Tutorials* **23**(4), 2248–2294.

[20] Das, T., Adepu, S. and Zhou, J. [2020], 'Anomaly detection in industrial control systems using logical analysis of data', *Computers and Security* **96**.

[21] de Sa, A. O., Carmo, L. F. d. C. and Machado, R. C. [2020], 'Bio-inspired active system identification: a cyber-physical intelligence attack in networked control systems', *Mobile Networks and Applications* **25**(5), 1944–1957.

[22] de Sá, A. O., da Costa Carmo, L. F. R. and Machado, R. C. [2017], 'Covert attacks in cyber-physical control systems', *IEEE Transactions on Industrial Informatics* **13**(4), 1641–1651.

[23] Ding, D., Han, Q.-L., Xiang, Y., Ge, X. and Zhang, X.-M. [2018], 'A survey on security control and attack detection for industrial cyber-physical systems', *Neurocomputing* **275**, 1674–1683.

[24] Ding, J., Tarokh, V. and Yang, Y. [2018], 'Model selection techniques: An overview', *IEEE Signal Processing Magazine* **35**(6), 16–34.

[25] Elbez, G., Keller, H. B. and Hagenmeyer, V. [2018], A new classification of attacks against the cyber-physical security of smart grids, *in* 'Proceedings of the 13th International Conference on Availability, Reliability and Security', pp. 1–6.

[26] Elman, J. L. [1990], 'Finding structure in time', *Cognitive science* **14**(2), 179–211.

[27] Elnour, M., Meskin, N. and Khan, K. [2020], Hybrid attack detection framework for industrial control systems using 1d-convolutional neural network and isolation forest, *in* 'CCTA 2020 - 4th IEEE Conference on Control Technology and Applications', pp. 877–884.

[28] Elnour, M., Meskin, N., Khan, K. and Jain, R. [2020], 'A dual-isolation-forests-based attack detection framework for industrial control systems', *IEEE Access* **8**, 36639–36651.

[29] Fernandes, G., Rodrigues, J. J., Carvalho, L. F., Al-Muhtadi, J. F. and Proença, M. L. [2019], 'A comprehensive survey on network anomaly detection', *Telecommunication Systems* **70**, 447–489.

[30] Filonov, P., Kitashov, F. and Lavrentyev, A. [2017], 'Rnn-based early cyber-attack detection for the tennessee eastman process', *arXiv preprint arXiv:1709.02232* .

[31] Finn, C., Goodfellow, I. and Levine, S. [2016], 'Unsupervised learning for physical interaction through video prediction', *Advances in neural information processing systems* **29**.

[32] Fritz, R., Schwarz, P. and Zhang, P. [2019], Modeling of cyber attacks and a time guard detection for ics based on discrete event systems, *in* '2019 18th European Control Conference (ECC)', IEEE, pp. 4368–4373.

[33] Fritz, R. and Zhang, P. [2018], 'Modeling and detection of cyber attacks on discrete event systems', *IFAC-PapersOnLine* **51**(7), 285–290.

[34] Gao, W. and Morris, T. H. [2014], 'On cyber attacks and signature based intrusion detection for modbus based industrial control systems', *Journal of Digital Forensics, Security and Law* **9**(1), 3.

[35] Góes, R. M., Kang, E., Kwong, R. and Lafortune, S. [2017], Stealthy deception attacks for cyber-physical systems, *in* '2017 IEEE 56th Annual Conference on Decision and Control (CDC)', IEEE, pp. 4224–4230.

[36] Goh, J., Adepu, S., Junejo, K. N. and Mathur, A. [2016], A dataset to support research in the design of secure water treatment systems, *in* 'International conference on critical information infrastructures security', Springer, pp. 88–99.

[37] Goh, J., Adepu, S., Tan, M. and Lee, Z. [2017], Anomaly detection in cyber physical systems using recurrent neural networks, *in* 'Proceedings of IEEE International Symposium on High Assurance Systems Engineering', pp. 140–145.

[38] Goodfellow, I., Bengio, Y. and Courville, A. [2016], *Deep learning*, MIT press.

[39] Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A. and Bengio, Y. [2014], 'Generative adversarial nets', *Advances in neural information processing systems* **27**.

[40] Hao, X., Zhou, F. and Chen, X. [2016], Analysis on security standards for industrial control system and enlightenment on relevant chinese standards, *in* '2016 IEEE 11th Conference on Industrial Electronics and Applications (ICIEA)', IEEE, pp. 1967–1971.

[41] Hasan, M., Choi, J., Neumann, J., Roy-Chowdhury, A. K. and Davis, L. S. [2016], Learning temporal regularity in video sequences, *in* 'Proceedings of the IEEE conference on computer vision and pattern recognition', pp. 733–742.

[42] Hochreiter, S. and Schmidhuber, J. [1997], 'Long short-term memory', *Neural computation* **9**(8), 1735–1780.

[43] Hu, A., Cotter, F., Mohan, N., Gurau, C. and Kendall, A. [2020], Probabilistic future prediction for video scene understanding, *in* 'Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XVI 16', Springer, pp. 767–785.

[44] Humayed, A., Lin, J., Li, F. and Luo, B. [2017], 'Cyber-physical systems security—a survey', *IEEE Internet of Things Journal* **4**(6), 1802–1831.

[45] Industrial Control Systems Cyber Emergency Response Team [2016], 'Recommended Practice: Improving Industrial Control System Cybersecurity with Defense-in-Depth Strategies)', https://www.cisa.gov. datum pristupa: 22.08.2023.

[46] Inoue, J., Yamagata, Y., Chen, Y., Poskitt, C. M. and Sun, J. [2017], Anomaly detection for a water treatment system using unsupervised machine learning, *in* '2017 IEEE International Conference on Data Mining Workshops (ICDMW)', IEEE, pp. 1058–1065.

[47] International Electrotechnical Commission [2013], 'IEC 62264-1:2013 Enterprise-control system integration — Part 1: Models and terminology', https://www.iso.org/standard/57308.html. datum pristupa: 22.08.2023.

[48] International Electrotechnical Commission (IEC) [2019], 'Industrial Process Measurement, Control and Automation – Network and System Information Security', https://www.iecee.org/certification/iec-standards/iec-62443-4-22019-34421. datum pristupa: 22.08.2023.

[49] International Organization for Standardization (ISO)/International Electrotechnical Commission (IEC) [2022], 'nformation security, cybersecurity and privacy protection — Information security management systems — Requirements', https://www.iso.org/standard/82875.html. datum pristupa: 22.08.2023.

[50] Ioffe, S. and Szegedy, C. [2015], Batch normalization: Accelerating deep network training by reducing internal covariate shift, *in* 'International conference on machine learning', pmlr, pp. 448–456.

[51] iTrust – SUTD [2015], 'Secure Water Treatment (SWaT)', https://itrust.sutd.edu.sg/itrust-labs_datasets/dataset_info/. datum pristupa: 22.08.2023.

[52] Jakovljevic, Z., Lesi, V. and Pajic, M. [2021], 'Attacks on distributed sequential control in manufacturing automation', *IEEE Transactions on Industrial Informatics* **17**(2), 775–786.

[53] Jakovljevic, Z., Majstorovic, V., Stojadinovic, S., Zivkovic, S., Gligorijevic, N. and Pajic, M. [2017], Cyber-physical manufacturing systems (cpms), *in* 'Proceedings of 5th International Conference on Advanced Manufacturing Engineering and Technologies: NEWTECH 2017 5', Springer, pp. 199–214.

[54] Jakovljevic, Z. and Nedeljkovic, D. [2021], Distribution of control tasks to smart devices in industrial control systems: a case study, *in* 'In 8th International Conference on Electrical, Electronics and Computing Engineering (IcETRAN 2021)', pp. 585–590.

[55] Jakovljevic, Z. and Nedeljkovic, D. [2023], Cybersecurity issues in motion control – an overview of challenges, *in* '2023 10th International Conference on Electrical, Electronic and Computing Engineering (IcETRAN)', pp. 1–6.

[56] Joseph, V. R. [2022], 'Optimal ratio for data splitting', *Statistical Analysis and Data Mining: The ASA Data Science Journal* **15**(4), 531–538.

[57] Kagermann, H., Helbig, J., Hellinger, A. and Wahlster, W. [2013], *Recommendations for implementing the strategic initiative INDUSTRIE 4.0*, Forschungsunion.

[58] Khan, R., Maynard, P., McLaughlin, K., Laverty, D. and Sezer, S. [2016], Threat analysis of blackenergy malware for synchrophasor based real-time control and monitoring in smart grid, *in* '4th International Symposium for ICS & SCADA Cyber Security Research 2016 4', pp. 53–63.

[59] Kravchik, M. and Shabtai, A. [2018], Detecting cyber attacks in industrial control systems using convolutional neural networks, *in* 'Proceedings of the 2018 Workshop on Cyber-Physical Systems Security and PrivaCy', CPS-SPC '18, ACM, New York, NY, USA, pp. 72–83.

[60] Kravchik, M. and Shabtai, A. [2021], 'Efficient cyber attack detection in industrial control systems using lightweight neural networks and pca', *IEEE Transactions on Dependable and Secure Computing* pp. 1–1.

[61] Krithivasan, K., Pravinraj, S., VS, S. S. et al. [2020], 'Detection of cyberattacks in industrial control systems using enhanced principal component analysis and hypergraph-based convolution neural network (epca-hg-cnn)', *IEEE Transactions on Industry Applications* **56**(4), 4394–4404.

[62] Krizhevsky, A., Sutskever, I. and Hinton, G. E. [2012], 'Imagenet classification with deep convolutional neural networks', *Advances in neural information processing systems* **25**.

[63] Lee, E. A. and Seshia, S. A. [2016], *Introduction to embedded systems: A cyber-physical systems approach*, Mit Press.

[64] Li, D., Chen, D., Jin, B., Shi, L., Goh, J. and Ng, S.-K. [2019], Mad-gan: Multivariate anomaly detection for time series data with generative adversarial networks, *in* 'Artificial Neural Networks and Machine Learning–ICANN 2019: Text and Time Series: 28th International Conference on Artificial Neural Networks, Munich, Germany, September 17–19, 2019, Proceedings, Part IV', Springer, pp. 703–716.

[65] Lima, P. M., Alves, M. V., Carvalho, L. K. and Moreira, M. V. [2017], 'Security against network attacks in supervisory control systems', *IFAC-PapersOnLine* **50**(1), 12333–12338.

[66] Lima, P. M., Alves, M. V. S., Carvalho, L. K. and Moreira, M. V. [2019], 'Security against communication network attacks of cyber-physical systems', *Journal of Control, Automation and Electrical Systems* **30**, 125–135.

[67] Lima, P. M., Carvalho, L. K. and Moreira, M. V. [2018], 'Detectable and undetectable network attack security of cyber-physical systems', *IFAC-PapersOnLine* **51**(7), 179–185.

[68] Lin, Q., Adepu, S., Verwer, S. and Mathur, A. [2018], Tabor: A graphical model-based approach for anomaly detection in industrial control systems, *in* 'Proceedings of the 2018 on asia conference on computer and communications security', pp. 525–536.

[69] Manca, G. [2020], '"Tennessee-Eastman-Process" Alarm Management Dataset', https://dx.doi.org/10.21227/326k-qr90. datum pristupa: 22.08.2023.

[70] McAvoy, T. and Ye, N. [1994], 'Base control for the tennessee eastman problem', *Computers & Chemical Engineering* **18**(5), 383–413.

[71] Meira-Góes, R., Lafortune, S. and Marchand, H. [2021], 'Synthesis of supervisors robust against sensor deception attacks', *IEEE Transactions on Automatic Control* **66**(10), 4990–4997.

[72] Mensah, F. N. and Helps, R. G. [2013], Security analysis of cps: Understanding current concerns as a foundation for future design, *in* '2013 ASEE Annual Conference & Exposition', pp. 23–1057.

[73] Microchip Technology Inc. [2008], 'MRF24J40MA 2.4 GHz IEEE Std. 802.15.4T MRF Transceiver Module', http://ww1.microchip.com/downloads/en/DeviceDoc/70329b.pdf. datum pristupa: 22.08.2023.

[74] Mihalič, F., Truntič, M. and Hren, A. [2022], 'Hardware-in-the-loop simulations: A historical overview of engineering challenges', *Electronics* **11**(15), 2462.

[75] Mishra, P., Varadharajan, V., Tupakula, U. and Pilli, E. [2019], 'A detailed investigation and analysis of using machine learning techniques for intrusion detection', *IEEE Communications Surveys and Tutorials* **21**(1), 686–728.

[76] Monmasson, E. and Cirstea, M. N. [2007], 'Fpga design methodology for industrial control systems—a review', *IEEE transactions on industrial electronics* **54**(4), 1824–1842.

[77] National Institute of Standards and Technology [2013], 'Security and Privacy Controls for Federal Information Systems and Organizations', https://nvlpubs.nist.gov/nistpubs/SpecialPublications/NIST.SP.800-53r4.pdf. datum pristupa: 22.08.2023.

[78] National Institute of Standards and Technology [2015], 'Guide to Industrial Control Systems (ICS) Security', https://nvlpubs.nist.gov/nistpubs/SpecialPublications/NIST.SP.800-82r2.pdf. datum pristupa: 22.08.2023.

[79] National Institute of Standards and Technology [2020], 'Security and Privacy Controls for Information Systems and Organizations', https://nvlpubs.nist.gov/nistpubs/SpecialPublications/NIST.SP.800-53r5.pdf. datum pristupa: 22.08.2023.

[80] Nedeljkovic, D. and Jakovljevic, Z. [2021*a*], 'New datasets obtained from experimental installations with centralized control - v1.0', https://zenodo.org/record/4556924. datum pristupa: 22.08.2023.

[81] Nedeljkovic, D. and Jakovljevic, Z. [2021*b*], 'New datasets obtained from experimental installations with centralized control - v2.0', https://zenodo.org/record/5514351. datum pristupa: 22.08.2023.

[82] Nedeljković, D. and Jakovljević, Ž. [2020], Cyber-attack detection method based on rnn, *in* '7th International Conference on Electrical, Electronic and Computing Engineering (IcETRAN 2020), Proceedings, Belgrade, Čačak, Niš, Novi Sad, September 2020.', ETRAN Society, Belgrade, Academic Mind, Belgrade, pp. 726–731.

[83] Nedeljković, D. and Jakovljević, Ž. [2021], Implementation of cnn based algorithm for cyber-attacks detection on a real-world control system, *in* 'Proceedings of the 14th International Scientific Conference MMA 2021-Flexible Technologies, Novi Sad, september 2021', Faculty of Technical Sciences, Department of Production Engineering, Novi Sad, pp. 119–122.

[84] Nedeljkovic, D. and Jakovljevic, Z. [2022*a*], 'Cnn based method for the development of cyber-attacks detection algorithms in industrial control systems', *Computers & Security* **114**, 102585.

[85] Nedeljković, D. and Jakovljević, Ž. [2022*b*], Gan-based data augmentation in the design of cyber-attack detection methods, *in* '9th International Conference on Electrical, Electronic and Computing Engineering (IcETRAN 2022), Proceedings, Novi Pazar, June 2022, ROI1. 4', ETRAN Society, Belgrade, Academic Mind, Belgrade, pp. ROI1–4.

[86] Nedeljkovic, D. and Jakovljevic, Z. [2023], Deep learning prediction models for the detection of cyber-attacks on image sequences, *in* 'International Conference on Robotics in Alpe-Adria Danube Region', Springer, pp. 62–70.

[87] Nedeljković, D., Jakovljević, Ž. and Miljković, Z. [2020], 'The detection of sensor signal attacks in industrial control systems', *FME Transactions, New Series* **48**(1), 7–12.

[88] Nedeljkovic, D., Jakovljevic, Z., Miljkovic, Z. and Pajic, M. [2019], Detection of cyber-attacks in electro-pneumatic positioning system with distributed control, *in* '2019 27th Telecommunications Forum (TELFOR)', IEEE, pp. 1–4.

[89] Nedeljković, D., Jakovljević, Ž., Miljković, Z. and Pajić, M. [2020], 'Detection of cyber-attacks in systems with distributed control based on support vector regression', *Telfor Journal* **12**(2), 104–109.

[90] Neshenko, N., Bou-Harb, E. and Furht, B. [2021], 'A behavioral-based forensic investigation approach for analyzing attacks on water plants using gans', *Forensic Science International: Digital Investigation* **37**, 301198.

[91] Nourian, A. and Madnick, S. [2015], 'A systems theoretic approach to the security threats in cyber physical systems applied to stuxnet', *IEEE Transactions on Dependable and Secure Computing* **15**(1), 2–13.

[92] NXP Semiconductors [2009], 'Arm Mbed LPC1768 Board', https://www.nxp.com/products/processors-and-microcontrollers/arm-microcontrollers/general-purpose-mcus/lpc1700-arm-cortex-m3/arm-mbed-lpc1768-board:OM11043. datum pristupa: 22.08.2023.

[93] OPC foundation, OPC UA Online Reference [2018], 'OPC 10000-2: OPC Unified Architecture, Part 2: Security Model, Release 1.04,', https://reference.opcfoundation.org. datum pristupa: 22.08.2023.

[94] OSDN [2021], 'Tera Term software', https://ttssh2.osdn.jp/index.html.en. datum pristupa: 22.08.2023.

[95] Paraskevoudis, K., Karayannis, P. and Koumoulos, E. P. [2020], 'Real-time 3d printing remote defect detection (stringing) with computer vision and artificial intelligence', *Processes* **8**(11), 1464.

[96] Parks, T. and McClellan, J. [1972], 'Chebyshev approximation for nonrecursive digital filters with linear phase', *IEEE Transactions on circuit theory* **19**(2), 189–194.

[97] Pasqualetti, F., Dörfler, F. and Bullo, F. [2013], 'Attack detection and identification in cyber-physical systems', *IEEE Transactions on Automatic Control* **58**(11), 2715–2729.

[98] Perales Gómez, Á. L., Fernández Maimó, L., Huertas Celdrán, A. and García Clemente, F. J. [2020], 'Madics: A methodology for anomaly detection in industrial control systems', *Symmetry* **12**(10), 1583.

[99] Priyanga, S., Krithivasan, K., Pravinraj, S. and Shankar Sriram, V.S., S. [2020], 'Detection of cyberattacks in industrial control systems using enhanced principal component analysis and hypergraph-based convolution neural network (EPCA-HG-CNN)', *IEEE Transactions on Industry Applications* **56**(4), 4394–4404.

[100] Raman MR, G., Somu, N. and Mathur, A. [2020], 'A multilayer perceptron model for anomaly detection in water treatment plants', *International Journal of Critical Infrastructure Protection* **31**.

[101] Rubio, J. E., Alcaraz, C., Roman, R. and Lopez, J. [2019], 'Current cyber-defense trends in industrial control systems', *Computers & Security* **87**, 101561.

[102] Sapkota, S., Mehdy, A. N., Reese, S. and Mehrpouyan, H. [2020], 'Falcon: Framework for anomaly detection in industrial control systems', *Electronics* **9**(8), 1192.

[103] Scholkopf, B., Burges, C. and Smola, A. [1998], 'Advances in kernel methods: Support vector machines'.

[104] Shalyga, D., Filonov, P. and Lavrentyev, A. [2018], 'Anomaly detection for water treatment system based on neural network with automatic architecture optimization'.

[105] Shi, X., Chen, Z., Wang, H., Yeung, D.-Y., Wong, W.-K. and Woo, W.-c. [2015], 'Convolutional lstm network: A machine learning approach for precipitation nowcasting', *Advances in neural information processing systems* **28**.

[106] Smola, A. J. and Schölkopf, B. [2004], 'A tutorial on support vector regression', *Statistics and computing* **14**(3), 199–222.

[107] Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I. and Salakhutdinov, R. [2014], 'Dropout: a simple way to prevent neural networks from overfitting', *The journal of machine learning research* **15**(1), 1929–1958.

[108] Srivastava, N., Mansimov, E. and Salakhudinov, R. [2015], Unsupervised learning of video representations using lstms, *in* 'International conference on machine learning', PMLR, pp. 843–852.

[109] Stouffer, K., Falco, J., Scarfone, K. et al. [2011], 'Guide to industrial control systems (ics) security', *NIST special publication* **800**(82), 16–16.

[110] Su, R. [2018], 'Supervisor synthesis to thwart cyber attack with bounded sensor reading alterations', *Automatica* **94**, 35–44.

[111] Sultani, W., Chen, C. and Shah, M. [2018], Real-world anomaly detection in surveillance videos, *in* 'Proceedings of the IEEE conference on computer vision and pattern recognition', pp. 6479–6488.

[112] Sundar, A., Pahwa, V., Das, C., Deshmukh, M. and Robinson, N. [2016], 'A comprehensive assessment of the performance of modern algorithms for enhancement of digital volume pulse signals', *International Journal of Pharma Medicine and Biological Sciences* **5**(1), 91.

[113] Taormina, R., Galelli, S., Tippenhauer, N. O., Salomons, E., Ostfeld, A., Eliades, D. G. et al. [2018], 'Battle of the attack detection algorithms: Disclosing cyber attacks on water distribution networks', *Journal of Water Resources Planning and Management* **144**(8), 04018048.

[114] Teixeira, A., Perez, D., Sandberg, H. and Johansson, K. H. [2012], Attack models and scenarios for networked control systems, *in* 'Proceedings of the 1st international conference on High Confidence Networked Systems', pp. 55–64.

[115] Teixeira, A., Shames, I., Sandberg, H. and Johansson, K. [2015], 'A secure control framework for resource-limited adversaries', *Automatica* **51**, 135–148.

[116] Thistle, J. G. [1996], 'Supervisory control of discrete event systems', *Mathematical and Computer Modelling* **23**(11-12), 25–53.

[117] Tidy, J. [2021*a*], 'Colonial hack: How did cyber-attackers shut off pipeline?', https://www.bbc.com/news/technology-57063636. datum pristupa: 22.08.2023.

[118] Tidy, J. [2021*b*], 'Hacker tries to poison water supply of florida city', https://www.bbc.com/news/world-us-canada-55989843. datum pristupa: 22.08.2023.

[119] Tsironi, E., Barros, P., Weber, C. and Wermter, S. [2017], 'An analysis of convolutional long short-term memory recurrent neural networks for gesture recognition', *Neurocomputing* **268**, 76–86.

[120] Umer, M. A., Mathur, A., Junejo, K. N. and Adepu, S. [2020], 'Generating invariants using design and data-centric approaches for distributed attack detection', *International Journal of Critical Infrastructure Protection* **28**, 100341.

[121] Upadhyay, D. and Sampalli, S. [2020], 'SCADA (Supervisory Control and Data Acquisition) systems: Vulnerability assessment and security recommendations', *Computers and Security* **89**.

[122] Vapnik, V. [1999], *The nature of statistical learning theory*, Springer science & business media.

[123] Wakaiki, M., Tabuada, P. and Hespanha, J. P. [2019], 'Supervisory control of discrete-event systems under attacks', *Dynamic Games and Applications* **9**, 965–983.

[124] Wang, C., Wang, B., Liu, H. and Qu, H. [2020], 'Anomaly detection for industrial control system based on autoencoder neural network', *Wireless Communications and Mobile Computing* **2020**, 1–10.

[125] Wang, X., Mizuno, M., Neilsen, M., Ou, X., Rajagopalan, S. R., Boldwin, W. G. and Phillips, B. [2015], Secure rtos architecture for building automation, *in* 'Proceedings of the First ACM Workshop on Cyber-Physical Systems-Security and/or PrivaCy', pp. 79–90.

[126] Wang, Y., Bozkurt, A. K. and Pajic, M. [2019], 'Attack-resilient supervisory control of discrete-event systems', *arXiv preprint arXiv:1904.03264* .

[127] Wang, Y., Li, Y., Yu, Z., Wu, N. and Li, Z. [2021], 'Supervisory control of discrete-event systems under external attacks', *Information Sciences* **562**, 398–413.

[128] Wang, Z., Xie, W., Wang, B., Tao, J. and Wang, E. [2021], 'A survey on recent advanced research of cps security', *Applied Sciences* **11**(9), 3751.

[129] Wun, A., Cheung, A. and Jacobsen, H.-A. [2007], A taxonomy for denial of service attacks in content-based publish/subscribe systems, *in* 'Proceedings of the 2007 inaugural international conference on Distributed event-based systems', pp. 116–127.

[130] Xie, Y., Wang, W., Wang, F. and Chang, R. [2018], Vtet: A virtual industrial control system testbed for cyber security research, *in* '2018 Third International Conference on Security of Smart Cities, Industrial Control System and Communications (SSIC)', IEEE, pp. 1–7.

[131] Xu, Y., Yang, Y., Li, T., Ju, J. and Wang, Q. [2017], Review on cyber vulnerabilities of communication protocols in industrial control systems, *in* '2017 IEEE Conference on Energy Internet and Energy System Integration, EI2 2017 - Proceedings', pp. 1–6.

[132] Xue, F., Yan, W., Wang, T., Huang, H. and Feng, B. [2020], Deep anomaly detection for industrial systems: A case study, *in* 'Proceedings of the Annual Conference of the Prognostics and Health Management Society, PHM', Vol. 12, pp. 1–8.

[133] Yaacoub, J.-P. A., Salman, O., Noura, H. N., Kaaniche, N., Chehab, A. and Malli, M. [2020], 'Cyber-physical systems security: Limitations, issues and future trends', *Microprocessors and microsystems* **77**, 103201.

[134] You, D., Wang, S. and Seatzu, C. [2021], 'A liveness-enforcing supervisor tolerant to sensor-reading modification attacks', *IEEE Transactions on Systems, Man, and Cybernetics: Systems* **52**(4), 2398–2411.

[135] Zhang, Q., Li, Z., Seatzu, C. and Giua, A. [2018], Stealthy attacks for partially-observed discrete event systems, *in* '2018 IEEE 23rd International Conference on Emerging Technologies and Factory Automation (ETFA)', Vol. 1, IEEE, pp. 1161–1164.

[136] Zvei - electrifying ideas [2015], 'Reference Architecture Model Industrie 4.0 (RAMI4.0)', http://www.zvei.org. datum pristupa: 22.08.2023.

Biografija autora

Dušan (Momir) Nedeljković rođen je 03. decembra 1992. godine u Aranđelovcu, Republika Srbija. Osnovnu školu "Svetolik Ranković" i srednju tehničku školu "Mileta Nikolić" završio je u Aranđelovcu kao odličan đak.

Na Mašinski fakultet Univerziteta u Beogradu upisao se školske 2011/2012. godine. Osnovne akademske studije završio je 2014. godine sa prosečnom ocenom 8,48 (osam i 48/100). Školske 2014/2015. godine upisao je Master akademske studije na Katedri za proizvodno mašinstvo, a iste završio 2016. godine sa prosečnom ocenom 9,75 (devet i 75/100) odbranivši master rad iz predmeta Automatizacija proizvodnje pod mentorstvom prof. dr Živane Jakovljević. Nakon diplomiranja započinje radni odnos u kompaniji "Servoteh d.o.o." kao mašinski inženjer–projektant. Školske 2017/2018. godine upisao je Doktorske akademske studije na Mašinskom fakultetu Univerziteta u Beogradu gde je položio sve ispite sa prosečnom ocenom 9,86 (devet i 86/100). Od 19. januara 2018. godine zaposlen je na Univerzitetu u Beogradu – Mašinskom fakultetu u zvanju asistenta na Katedri za proizvodno mašinstvo. Od prolećnog semestra školske 2017/2018. godine aktivno učestvuje u realizaciji laboratorijskih vežbi, pregledu samostalnih i projektnih zadataka na ukupno 7 nastavnih predmeta na OAS i MAS. Član je Laboratorije za automatizaciju proizvodnje čiji je rukovodilac prof. dr Živana Jakovljević.

Od 1. februara 2018. godine angažovan je na projektu Tehnološkog razvoja (ev. br. TR-35004) koji finansira Ministarstvo prosvete, nauke i tehnološkog razvoja Republike Srbije. Pored toga, od 2020-2022. godine bio je angažovan na projektu (akronim–MISSION4.0, ev. br. 6523109) koji je finansiran od strane Fonda za nauku Republike Srbije u okviru poziva "Program za razvoj projekata iz oblasti veštačke inteligencije". Do sada je bio autor ili koautor 19 radova i jednog tehničkog rešenja. Od 15. decembra 2021. god. boravio je na Djuk Univerzitetu iz Durhama, SAD u okviru tromesečnog *Erasmus+* projekta razmene studenata gde je bio uključen u aktivnosti Laboratorije za kibernetsko-fizičke sisteme.

Izjava o autorstvu

Ime i prezime autora Dušan Nedeljković

Broj indeksa D13/2017

Izjavljujem

da je doktorska disertacija pod naslovom:

Detekcija kibernetskih napada na sisteme za upravljanje proizvodnim resursima

- rezultat sopstvenog istraživačkog rada;
- da disertacija u celini ni u delovima nije bila predložena za sticanje druge diplome prema studijskim programima drugih visokoškolskih ustanova;
- da su rezultati korektno navedeni i
- da nisam kršio/la autorska prava i koristio/la intelektualnu svojinu drugih lica.

Potpis autora

U Beogradu, _____

Dušan Nedeljković

Izjava o istovetnosti štampane i elektronske verzije doktorskog rada

Ime i prezime autora Dušan Nedeljković

Broj indeksa D13/2017

Studijski program Doktorske akademske studije – Mašinsko inženjerstvo

Naslov rada **Detekcija kibernetskih napada na sisteme za upravljanje proizvodnim resursima**

Mentor dr Živana Jakovljević, redovni profesor

Izjavljujem da je štampana verzija mog doktorskog rada istovetna elektronskoj verziji koju sam predao/la radi pohranjivanja u **Digitalnom repozitorijumu Univerziteta u Beogradu**.

Dozvoljavam da se objave moji lični podaci vezani za dobijanje akademskog naziva doktora nauka, kao što su ime i prezime, godina i mesto rođenja i datum odbrane rada.

Ovi lični podaci mogu se objaviti na mrežnim stranicama digitalne biblioteke, u elektronskom katalogu i u publikacijama Univerziteta u Beogradu.

Potpis autora

U Beogradu, _____

Izjava o korišćenju

Ovlašćujem Univerzitetsku biblioteku „Svetozar Marković“ da u Digitalni repozitorijum Univerziteta u Beogradu unese moju doktorsku disertaciju pod naslovom:

Detekcija kibernetskih napada na sisteme za upravljanje proizvodnim resursima

koja je moje autorsko delo.

Disertaciju sa svim prilozima predao/la sam u elektronskom formatu pogodnom za trajno arhiviranje.

Moju doktorsku disertaciju pohranjenu u Digitalnom repozitorijumu Univerziteta u Beogradu i dostupnu u otvorenom pristupu mogu da koriste svi koji poštuju odredbe sadržane u odabranom tipu licence Kreativne zajednice (Creative Commons) za koju sam se odlučio/la.

1. Autorstvo (CC BY)
2. Autorstvo – nekomercijalno (CC BY-NC)
3. Autorstvo – nekomercijalno – bez prerada (CC BY-NC-ND)
4. Autorstvo – nekomercijalno – deliti pod istim uslovima (CC BY-NC-SA)
5. Autorstvo – bez prerada (CC BY-ND)
6. Autorstvo – deliti pod istim uslovima (CC BY-SA)

(Molimo da zaokružite samo jednu od šest ponuđenih licenci.
Kratak opis licenci je sastavni deo ove izjave).

Potpis autora

U Beogradu, _____

Душан Недељковић

1. Autorstvo. Dozvoljavate umnožavanje, distribuciju i javno saopštavanje dela, i prerade, ako se navede ime autora na način određen od strane autora ili davaoca licence, čak i u komercijalne svrhe. Ovo je najslobodnija od svih licenci.

2. **Autorstvo** – **nekomercijalno**. Dozvoljavate umnožavanje, distribuciju i javno saopštavanje dela, i prerade, ako se navede ime autora na način određen od strane autora ili davaoca licence. Ova licenca ne dozvoljava komercijalnu upotrebu dela.

3. **Autorstvo** – **nekomercijalno** – **bez prerada**. Dozvoljavate umnožavanje, distribuciju i javno saopštavanje dela, bez promena, preoblikovanja ili upotrebe dela u svom delu, ako se navede ime autora na način određen od strane autora ili davaoca licence. Ova licenca ne dozvoljava komercijalnu upotrebu dela. U odnosu na sve ostale licence, ovom licencom se ograničava najveći obim prava korišćenja dela.

4. **Autorstvo** – **nekomercijalno** – **deliti pod istim uslovima**. Dozvoljavate umnožavanje, distribuciju i javno saopštavanje dela, i prerade, ako se navede ime autora na način određen od strane autora ili davaoca licence i ako se prerada distribuira pod istom ili sličnom licencom. Ova licenca ne dozvoljava komercijalnu upotrebu dela i prerada.

5. **Autorstvo** – **bez prerada**. Dozvoljavate umnožavanje, distribuciju i javno saopštavanje dela, bez promena, preoblikovanja ili upotrebe dela u svom delu, ako se navede ime autora na način određen od strane autora ili davaoca licence. Ova licenca dozvoljava komercijalnu upotrebu dela.

6. **Autorstvo** – **deliti pod istim uslovima**. Dozvoljavate umnožavanje, distribuciju i javno saopštavanje dela, i prerade, ako se navede ime autora na način određen od strane autora ili davaoca licence i ako se prerada distribuira pod istom ili sličnom licencom. Ova licenca dozvoljava komercijalnu upotrebu dela i prerada. Slična je softverskim licencama, odnosno licencama otvorenog koda.