

УНИВЕРЗИТЕТ „УНИОН - НИКОЛА ТЕСЛА“
ФАКУЛТЕТ ЗА ИНФОРМАЦИОНЕ ТЕХНОЛОГИЈЕ И
ИНЖЕЊЕРСТВО

МИЉАН П. МИЛЕТИЋ

ПРИМЕНА ЦЕЛОБРОЈНИХ SEC-DAES
КОДОВА И ВЕШТАЧКЕ
ИНТЕЛИГЕНЦИЈЕ У СИСТЕМИМА ЗА
ПРЕРАДУ ОТПАДНИХ ВОДА

ДОКТОРСКА ДИСЕРТАЦИЈА

БЕОГРАД, 2024.

UNIVERSITY "UNION - NIKOLA TESLA"
FACULTY OF INFORMATION TECHNOLOGIES AND
ENGINEERING

MILJAN P. MILETIC

APPLICATION OF INTEGER SEC-DAEC
CODES AND ARTIFICIAL INTELLIGENCE
IN WASTE WATER TREATMENT
SYSTEMS

DOCTORAL DISSERTATION

BELGRADE, 2024.

Подаци о члановима комисије:

1. проф. др Радоје Цвејић, редовни професор у пензији, Факултет за информационе технологије и инжењерство, Универзитет „Унион - Никола Тесла“ Београд - председник,
2. проф. др Акександар Радоњић, ванредни професор, Факултет за информационе технологије и инжењерство, Универзитет "Унион - Никола Тесла"у Београду - ментор,
3. проф. др Платон Совиљ, редовни професор, Факултет техничких наука, Универзитет у Новом Саду - члан,
4. проф. др Слађана Вујичић, ванредни професор, Факултет за информационе технологије и инжењерство, Универзитет "Унион - Никола Тесла"у Београду - члан.

ПРИМЕНА ЦЕЛОБРОЈНИХ SEC-DAEC КОДОВА И ВЕШТАЧКЕ ИНТЕЛИГЕНЦИЈЕ У СИСТЕМИМА ЗА ПРЕРАДУ ОТПАДНИХ ВОДА

Сажетак: Предмет истраживања докторске дисертације јесте унапређење рада система за прераду отпадних вода. Пошто је поменути систем сам по себи врло сложен, у дисертацији су дати предлози како се он може унапредити у два сегмента. Први сегмент тиче се даљинског видео надзора када у којим се налази отпадна вода. Како би се спречило изливање отпадних вода, а самим тим и плављење постројења, изнад када су инсталиране видео камере које путем Интернета омогућавају даљинско праћење нивоа воде. У досадашњој пракси, подаци са камера су били заштићени Интернет чексумом, која нема способност корекције грешака које се могу десити током преноса. Због тога је сваки оштећени пакет бивао одбачен, што је повремено доводило до непотпуног увида у тренутно стање система. У дисертацији ће бити показано да се употребом целобројних SEC-DAEC кодова (уместо Интернет чексуме) може значајно редуковати количина изгубљених података, а самим тим и обезбедити поузданији рад система. Други сегмент истраживања односи се на развијање и оптимизацију алгоритама вештачке интелигенције ради ефикаснијег управљања постројењем за пречишћавање и прераду отпадних вода. У ову сврху, коришћене су две дубоке неуронске мреже: CNN (енг. *Convolution Neural Network*) и MLP (енг. *Multi Layer Perceptron*). Прва мрежа је служила за обучавање алгоритама за детекцију исправних и неисправних индустријских уређаја, док је друга коришћена за регулацију рН вредности воде. Колико је аутору познато, у нашој земљи и у окружењу до сада није забележена примена вештачке интелигенције у контроли и одлучивању у процесу прераде отпадних вода. Овај рад се, стога, може сматрати пионирским истраживањем у нашем региону.

Кључне речи: целобројни SEC-DAEC кодови, вештачка интелигенција, отпадне воде, информационе технологије.

APPLICATION OF INTEGER SEC-DAEC CODES AND ARTIFICIAL INTELLIGENCE IN WASTE WATER TREATMENT SYSTEMS

Abstract: The subject of the research of the doctoral dissertation is the improvement of the process of the wastewater treatment system. Since the mentioned system is very complex itself, the dissertation offers suggestions on how it can be improved in two segments. The first segment concerns the remote video surveillance of the tank in which there is waste water. In order to prevent the spillage of waste water, and thus the flooding of the plant, video cameras have been installed above that enable remote monitoring of the water level via the Internet. In current practice, camera data is protected by Internet checksum, which does not have the ability to correct errors that may occur during transmission. Because of this, every damaged packet was dropped, which occasionally led to an incomplete view of the current state of the system. In the dissertation, it will be shown that the use of integer SEC-DAEC codes (instead of the Internet checksum) can significantly reduce the amount of lost data, and thus ensure more reliable operation of the system. The second segment of the research is related to the development and optimization of artificial intelligence algorithms for more efficient management of the wastewater treatment and processing plant. For this purpose, two deep neural networks were used: CNN (Convolution Neural Network) and MLP (Multi Layer Perceptron). The first network was used to train algorithms for the detection of correct and incorrect industrial devices, while the second was used to regulate the pH value of water. As far as the author is aware, the application of artificial intelligence in control and decision-making in the wastewater treatment process has not been recorded in our country and in the surrounding area. This work can therefore be considered a pioneering research in our region.

Keywords: integer SEC-DAEC codes, artificial intelligence, wastewater, information technologies.

САДРЖАЈ

1. Увод	1
2. Досадашња истраживања у предметној области.....	3
2.1. Граничне вредности емисије отпадних вода.....	3
2.2. Резултати мерења пре модернизације постројења.....	5
3. Систем за прераду отпадних вода (СПОВ)	9
3.1. Третман отпадне воде процесом неутрализације	9
3.2. Процес прераде отпадне воде таложењем	11
3.3. Принцип функционисања постројења.....	13
3.3.1. Прерада базних отпадних вода.....	14
3.3.2. Прерада киселих отпадних вода.....	16
3.3.3. Прерада тешких метала у отпадној води	18
3.3.4. Испуштање прерађених отпадних вода	19
3.4. Увођење мерно-информационог система	19
3.4.1. Инсталирање мерних и надзорних уређаја	20
3.4.2. Инсталирање командно-управљачких уређаја	24
3.4.3. Уписивање мерних података у MySQL базу	27
3.5. Проблеми са видео надзором постројења	28
4. Пренос података заснован на Интернет протоколу	33
4.1. Хијерархије протокола	33
4.2. Референтни модели	35
4.2.1. OSI референтни модел.....	35
4.2.2. TCP/IP референтни модел.....	37
4.2.3. Поређење OSI и TCP/IP референтних модела.....	38
4.3. Контрола преноса података на транспортном слоју	39
4.3.1. Принцип рада TCP протокола.....	39
4.3.2. Принцип рада UDP протокола.....	40
4.4. Пренос мултимедијалних података	41
4.4.1. Систем видео комуникације.....	41
4.4.2. Битске грешке у преносу података: проблеми и постојећа решења	41
5. Целобројни заштитни кодови	43
5.1. Заједничке особине свих целобројних заштитних кодова	43
5.1.1. Целобројни SEC-B _{l/b} ED кодови	46
5.1.2. Целобројни SEC-SB _{l/b} ED-DB _{n/b} ED кодови	47

5.1.3. Целобројни $SB_{h/b}EC-SB_{l/b}ED$ кодови	48
5.1.4. Целобројни $SB_{h/b}EC-SB_{l/b}ED-DB_{d/b}ED$ кодови	49
5.1.5. Целобројни $SB_{l/b}EC-DED$ кодови	50
5.1.6. Целобројни $R_{t/b}EC$ кодови	51
5.1.7. Целобројни $D_{1/b}EC$ кодови	52
5.1.8. Целобројни $DEC-(TAEC)_b$ кодови.....	53
5.1.9. Целобројни $SB_{l/b}EC-DB_{h/b}EC$ кодови	54
5.1.10. Целобројни $SB_{l/b}EC-D_{1/b}EC$ кодови	55
5.1.11. Целобројни $DAEC$ кодови.....	56
5.1.12. Целобројни $SB_{l/b}AEC-R_{t/b}AEC$ кодови	58
5.1.13. Целобројни $SB_{l/b}AEC-DAEC$ кодови	59
5.2. Целобројни $SEC-DAEC$ кодови	61
5.2.1. Мотивација.....	61
5.2.2. Конструкција	62
5.2.3. Процедура корекције грешака	65
5.2.4. Поређење са Интернет чексумом	70
6. Примена вештачке интелигенције у СПОВ-у.....	73
6.1. Употреба CNN-а.....	73
6.1.1. Прикупљање података за обуку CNN-а.....	74
6.2. Детекција висине звука на бази цепструма	75
6.2.1. Алгоритми за детекцију висине тона.....	76
6.2.2. Примењена метода	77
6.2.3. Анализа детективног кола	80
6.3. Методе анализе звука применом мере сличности слике	84
6.3.1. Уводна разматрања.....	84
6.3.2. Мере за процену сличности слика.....	88
6.3.3. Структурна сличност	89
6.4. Тестирање	91
6.4.1. Скуп података	91
6.4.2. Прорачун и анализа мера сличности слика	92
6.4.3. Резултати	93
6.4.4. Анализа на нивоу мера сличности слике усредњених по узорку	94
6.5. Класификација звука мотора пумпи помоћу DNN-а.....	97
6.5.1. Уводна разматрања.....	97

6.5.2. DNN и CNN	97
6.5.3. Мапирање аудио сигнала у слике.....	98
6.5.4. Развијена CNN	100
6.5.5. Резултати класификације оштећења мотора пумпе	102
6.5.6. Резултати класификације оштећења семеринга	103
6.5.7. Резултати класификације оштећења лежајева	104
6.5.8. Закључна разматрања.....	106
6.6. Унапређење процеса прераде отпадних вода помоћу MLP-а	106
6.6.1. Уводна разматрања.....	106
6.6.2. Архитектура MLP-а.....	107
6.6.3. Резултати обуке и тестирања MLP мреже	110
6.6.4. Закључна разматрања.....	112
7. Стратегија за унапређење рада СПОВ-а	114
7.1. Видео надзор када	114
7.2. Предвиђање кvara индустријских уређаја	114
7.3. Регулација рН вредности отпадне воде	116
7.4. Вредности параметара отпадне воде у унапређеном СПОВ-у.....	119
8. Закључак.....	120
9. Литература	121
10. Прилози	129

1. Увод

Вода представља основну потребу и главни елемент за живот човека. Само здрава исправна вода може се употребити за пиће. Наша основна улога је да је чувамо и штитимо за будућа покољења.

Дисертација је настала као сублимација рада на заштити исправности квалитета вода којом се аутор бави од 2016. године (првенствено са стране имплементације и примене аутоматизације и информационих технологија у процесу рада фабрика за пречишћавање хемијских загађивача у индустрији). Приликом рада на неколико фабрика примећено је да, услед застарелости опреме и утицаја човека на систем управљања, постоји доста простора за осавремењавање процеса пречишћавања. Фабрике у којима се вршила реконструкција постројења изграђене су 1980-их и њихова употребна вредност и ефикасност је одавно превазишла данашње стандарде. Реконструкција се углавном односила на процесе аутоматизације постројења, контролу вредности параметара у кадама (која се прати 24 часа) и контролу улазних и излазних вредности воде која се испушта у реципијент. Досадашња пракса показала је да се вредности параметара проверавају једном месечно од стране Завода за јавну заштиту, што доводи у сумњу вредност добијених резултата (испуштање отпадне воде се може десити пре доласка контроле или током ноћних сати). Због ових разлога нису коришћене вредности које се могу пронаћи на Интернету или у другим истраживањима. У систему, који је описан и предложен у дисертацији, вредности параметара воде (количина испуштене воде у реципијент, рН вредност, Redox потенцијал, температура, боја, итд.) се мере на сваких пет минута, а потом се уписују у базу података која се налази на серверу. Проблеми који су се јављали приликом имплементације система (употреба разних сензора, пловака за мерење нивоа воде у кадама, итд.) доводили су до изливања воде у постројењу. Ово се дешавало као последица отказа различитих мерно-контролних уређаја (како због прљавштине и агресивности отпадне воде, тако и због високе влажности ваздуха и околне температуре). Као решење за ове проблеме наметнула се употреба посебних индустријских камера које ће мерити промене нивоа воде и вршити контролу исправности осталих сензора. Иако је ово решење дало добре резултате, повремено се дешавало, да се због проблема у преносу података (губитак пакета), читава акција извршавала са одређеном несигурношћу. Како би се овај проблем превазишао, у дисертацији је предложен нови начин заштите видео података заснован на целобројним SEC-DAEC кодовима. Поред тога, у дисертацији је показано како је употребом неуронских мрежа MLP (енг. *Multi Layer Perceptron*) и CNN (енг. *Convolution Neural Network*) развијен алгоритам за аутоматизовано управљање постројењем. Наравно, због законских регулатива, требало

је обезбедити људски надзор и управљање. Аутоматизација је извршена на основу велике количине података у бази и вештачке интелигенције која сама одлучује колика количина базе или киселине треба да се дода у одређену каду пре него што се вода испусти у реципијент. Дакле, вештачка интелигенција не дозвољава испуштање недовољно прерађене воде у реципијент. Улога људи у описаном систему јесте да прате рад постројења, да врше допуну потребних сировина за пречишћавање и да одвозе продукте рада постројења након филтрације. Свакодневним складиштењем података долази се до скоро савреног учења вештачке интелигенције, чији проценат учења прелази 87%.

2. Досадашња истраживања у предметној области

Пре прегледа досадашњих истраживања везаних за ниво испуштања штетних материја у реципијенте биће приказани законски оквири везани за ову индустрију. Тачније, биће приказан извод из „Службеног гласника Републике Србије“ о граничним вредностима штетних материјала који могу да се испусте у реципијент за већи спектар операција које се изводе на разним материјалима и процесима у индустрији. Постројења у којима су вршена истраживања налазе се у наменској индустрији Републике Србије, и стога неће бити помињана њихова имена. Проблеми до којих се дошло приликом одбраде и прикупљања информација проситичу из чињенице да у готово свим постројењима није постојао 24-часовни мониторинг и контрола, односно узорковање материјала који се испушта у реципијент. Мерене вредности су добијене током узорковања и контроле постројења надлежних биохемијских лабораторија и Завода за јавно здравље.

2.1. Граничне вредности емисије отпадних вода

Граничне вредности емисије наведене у наставку односе на отпадну воду чије загађење првенствено потиче од једног од наведених објеката или постројења, укључујући одговарајући предтретман, међутретман и накнадни третман. Прерада и фина обрада метала обухвата следеће процесе (слике 2.1 и 2.2) [1]:

- 1 – галванизација
- 2 – декапирање
- 3 – анодизација
- 4 – брунирање
- 5 – топло превлачење цинка, топло калајисање
- 6 – калење
- 7 – производњу штампаних кола
- 8 – производња батерија
- 9 – емајлирање
- 10 – радионице за обраду метала
- 11 – брушење
- 12 – фарбање

Оптерећење загађујућим материјама се може одржавати на следећи начин:

- а) третманом процесних купки, као што су јонска измена, мембранска филтрација, електролиза и термални процес, итд.
- б) задржавањем састојака купки, као што су смањен улаз сировина, оптимизовани састав купке, итд.
- в) повраћајем погодних састојака из купки за испирање у процесне купке.

г) вишеструком употребом воде за испирање, као што је каскадно испирање и рециркулациона технологија коришћењем јонске измене.

д) “ЕДТА (етилен, диамин, тетрасирћетна киселина)” и њихове соли могу се ефикасно уклонити из хемијских раствора за бакар короз процес повраћаја након чега следи одговарајућ раствор за испирање [1].

Граничне вредности емисије пре мешања са осталим отпадним водама Службени гласник Републике Србије br. 67/2011,48/2012 i 1/2016													
Параметри	Јединица мере	Број процеса											
		1	2	3	4	5	6	7	8	9	10	11	12
АОХ	mg/l	1	1	1	1	1	1	1	1	1	1	1	1
Арсен	mg/l	0,1	-	-	-	-	-	0,1	0,1	-	-	-	-
Баријум	mg/l	-	-	-	-	-	2	-	-	-	-	-	-
Олово	mg/l	0,5	-	-	-	0,5	-	0,5	0,5	0,5	0,5	-	0,5
Кадмијум	mg/l	0,2				0,1			0,2	0,2	0,1		0,2
	kg/t	0,3							1,5				
Слободни хлор	mg/l	0,5	0,5		0,5		0,5				0,5		
Укупни хром	mg/l	0,5	0,5	0,5	0,5			0,5		0,5	0,5	0,5	0,5
Хром VI	mg/l	0,1	0,1	0,1	0,1			0,1		0,1	0,1		0,1
Кобалт	mg/l			1						1			
Цијаниди	mg/l	0,2					1	0,2			0,2		
Бакар	mg/l	0,5	0,5					0,5	0,5	0,5	0,5	0,5	0,5
Никл	mg/l	0,5	0,5		0,5			0,5	0,5	0,5	0,5	0,5	0,5
Жива	mg/l								0,05				
	kg/t								0,03				
Селен	mg/l									1			
Сребро	mg/l	0,1						0,1	0,1				
Сулфиди	mg/l	1	1		1			1	1	1			
Калај	mg/l	2		2		2		2					
Цинк	mg/l	2	2	2		2			2	2	2	2	2

Слика 2.1. Граничне вредности емисије пре мешања са осталим отпадним водама на нивоу погона (вредности се односе на двочасовни узорак).

Граничне вредности емисије пре мешања са осталим отпадним водама Службени гласник Републике Србије br. 67/2011,48/2012 i 1/2016													
Параметри	Јединица мере	Број Процеса											
		1	2	3	4	5	6	7	8	9	10	11	12
Алуминијум	mg/l	3	3	3	-	-	-	-	-	2	3	3	3
Азот из амонијака	mg/l	100	30	-	30	30	50	50	50	20	30	-	-
ХПК	mgO ₂ /l	400	100	100	200	200	400	600	200	100	400	400	300
Гвожђе	mg/l	3	3	-	3	3	-	3	3	3	3	3	3
Флуориди	mg/l	50	20	50	-	50	-	50	-	50	30	-	-
Азот и нитрати	mg/l	-	5	5	5	-	5	-	-	5	5	-	-
Угљоводоници ⁺	mg/l	10	10	10	10	10	10	10	10	10	10	10	10
Фосфор	mg/l	2	2	2	2	2	2	2	2	2	2	2	2
Токсичност за		6	4	2	6	6	6	6	6	4	6	6	6

Слика 2.2. Граничне вредности емисије на месту испуштања у површинске воде (вредности се односе на двочасовни узорак).

У [1] су наведени и додатни захтеви за достизање граничне вредности емисије на месту настанка отпадне воде:

1. Отпадна вода мора да садржи само оне халогене раствараче одобрене за употребу на основу студије утицаја. Осим тога, насумични узорак мора да успостави границу од 0,1 mg/l за испарљиве халогенизоване угљоводонике.

2. Када се ради са отпадним водама које су контаминирание живом, неопходно је придржавати се захтева одржавања концентрације живе не веће од 0,05 mg/l било у случајном узорку или двочасовном композитном узорку.

3. Отпадна вода добијена из када за одмашћивање, демејализованих када и спољашњих никлованих када не сме садржати никакве трагове ЕДТА.

4. Када се ради са отпадном водом из када које садрже кадмијум, укључујући испирање, неопходно је осигурати да ниво у случајном узорку или двочасовном композитном узорку не прелази 0,2 mg/l.

5. Отпадне воде потичу из испуста посторјења за предтретман, што је одређено специфичним параметром који се мери.

2.2. Резултати мерења пре модернизације постројења

У наставку су приказани резултати добијени пре модернизације постројења за прераду и пречишћавање отпадних вода. Анализиране су фабрике које налазе се у Републици Србији, и које у својим процесима производње имају обраду металних производа, поцинкавање, хромирање и галванизацију. Подаци (слике 2.3 - 2.8) су добијени у фабрикама на основу средње вредности мерних података у периоду од једне године (мерења на месечном нивоу обавиле су овлашћене лабораторије [2]).

Резултати мерења акредитоване лабораторије													
Параметри	Јединица мере	Број процеса											
		1	2	3	4	5	6	7	8	9	10	11	12
Алуминијум	mg/l	-	-	-	-	-	-	-	-	-	3.8	3.6	3.4
Азот из амонијака	mg/l	-	-	-	-	-	65	-	-	-	40	-	-
НРК	mgO ₂ /l	-	-	-	-	-	485	-	-	-	480	420	360
Гвожђе	mg/l	-	-	-	-	-	-	-	-	-	4.8	4.2	3.2
Флуориди	mg/l	-	-	-	-	-	-	-	-	-	40	-	-
Азот и нитрати	mg/l	-	-	-	-	-	6	-	-	-	7	-	-
Угљоводоници(III)	mg/l	-	-	-	-	-	12	-	-	-	12	11	10.8
Фосфор	mg/l	-	-	-	-	-	2.6	-	-	-	2.4	2.7	2.1
Токсичност за рибе		-	-	-	-	-	6.9	-	-	-	7	7.7	8

Слика 2.3. Вредности отпадних вода при испуштању у реципијент у случају фабрике која се бави производњом инокс галантерије и алата.

Резултати мерења акредитоване лабораторије													
Параметри	Јединица мере	Број процеса											
		1	2	3	4	5	6	7	8	9	10	11	12
АОХ	mg/l	-	-	-	-	-	1.61	-	-	-	2.66	2.66	1.6
Арсен	mg/l	-	-	-	-	-	-	-	-	-	-	-	-
Баријум	mg/l	-	-	-	-	-	2.33	-	-	-	-	-	-
Олово	mg/l	-	-	-	-	-	-	-	-	-	1.33	-	1.03
Кадмијум	mg/l	-	-	-	-	-	-	-	-	-	0.15	-	0.47
	kg/t	-	-	-	-	-	-	-	-	-	-	-	-
Слободни хлор	mg/l	-	-	-	-	-	0.92	-	-	-	0.83	-	-
Укупни хром	mg/l	-	-	-	-	-	-	-	-	-	1.13	1.25	0.92
Хром VI	mg/l	-	-	-	-	-	-	-	-	-	0.35	0.38	0.28
Кобалт	mg/l	-	-	-	-	-	-	-	-	1.66	-	-	-
Цијанид	mg/l	-	-	-	-	-	1.06	-	-	-	1.14	-	-
Бакар	mg/l	-	-	-	-	-	-	-	-	-	0.93	0.95	0.91
Никл	mg/l	-	-	-	-	-	-	-	-	-	0.84	0.95	0.62
Жива	mg/l	-	-	-	-	-	-	-	-	-	-	-	-
	kg/t	-	-	-	-	-	-	-	-	-	-	-	-
Селен	mg/l	-	-	-	-	-	-	-	-	-	-	-	-
Сребро	mg/l	-	-	-	-	-	-	-	-	-	-	-	-
Сулфиди	mg/l	-	-	-	-	-	-	-	-	-	-	-	-
Калај	mg/l	-	-	-	-	-	-	-	-	-	-	-	-
Цинк	mg/l	-	-	-	-	-	-	-	-	-	3.33	3.52	2.344

Слика 2.4. Вредности отпадних вода при мешању са осталим отпадним водама у случају фабрике која се бави производњом инокс галантерије и алата.

Резултати мерења акредитоване лабораторије													
Параметри	Јединица мере	Број Процеса											
		1	2	3	4	5	6	7	8	9	10	11	12
Алуминијум	mg/l	3.8	4	4.2	-	-	-	-	-	-	4.8	3.78	3.52
Азот из амонијака	mg/l	110	42	-	38	35	55	-	-	-	38	-	-
НРК	mgO ₂ /l	440	120	118	222	224	524	-	-	-	430	424	320
Гвожђе	mg/l	3.8	3.25	-	3.54	3.48	-	-	-	-	3.46	3.45	3.87
Флуориди	mg/l	54	25	54	-	58	-	-	-	-	39.5	-	-
Азот и нитрати	mg/l	-	6.4	5.8	6.1	-	5.4	-	-	-	5.4	-	-
Угљоводоници	mg/l	10.8	10.7	10.87	10.6	10.8	10.9	-	-	-	10.8	10.7	10.9
Фосфор	mg/l	2.21	2.3	2.7	2.5	2.8	2.7	-	-	-	2.1	2.3	2.4
Токсичност за рибе		6.5	4.3	2.4	6.5	6.8	6.5	-	-	-	6.4	6.4	6.5

Слика 2.5. Вредности отпадних вода при испуштању у реципијент у случају фабрике која се бави производњом заштитних средстава и алата.

Резултати мерења акредитоване лабораторије													
Параметри	Јединица мере	Број процеса											
		1	2	3	4	5	6	7	8	9	10	11	12
АОХ	mg/l	1.67	1.69	1.62	1.8	1.59	1.61	-	-	-	1.66	2.06	1.6
Арсен	mg/l	0.1	-	-	-	-	-	-	-	-	-	-	-
Баријум	mg/l	-	-	-	-	-	2.33	-	-	-	-	-	-
Олово	mg/l	0.93	-	-	-	0.82	-	-	-	-	0.93	-	0.83
Кадмијум	mg/l	0.33	-	-	-	0.27	-	-	-	-	0.5	-	0.27
	kg/t	0.45	-	-	-	-	-	-	-	-	-	-	-
Слободни хлор	mg/l	0.83	0.78	-	0.8	-	0.82	-	-	-	0.93	-	-
Укупни хром	mg/l	0.83	0.97	0.93	1.1	-	-	-	-	-	1.33	1.05	1.32
Хром VI	mg/l	0.66	0.56	0.26	0.4	-	-	-	-	-	0.25	-	0.28
Кобалт	mg/l	-	-	1.65	-	-	-	-	-	1.66	-	-	-
Цијанид	mg/l	0.3	-	-	-	-	1.06	-	-	-	0.34	-	-
Бакар	mg/l	0.83	0.97	-	-	-	-	-	-	-	0.93	1.05	1.31
Никл	mg/l	0.93	0.8	-	0.9	-	-	-	-	-	1.34	1.5	1.32
Жива	mg/l	-	-	-	-	-	-	-	-	-	-	-	-
	kg/t	-	-	-	-	-	-	-	-	-	-	-	-
Селен	mg/l	-	-	-	-	-	-	-	-	-	-	-	-
Сребро	mg/l	0.6	-	-	-	-	-	-	-	-	-	-	-
Сулфиди	mg/l	1.67	1.87	-	1.7	-	-	-	-	-	-	-	-
Калај	mg/l	2.73	-	2.38	-	2.91	-	-	-	-	-	-	-
Цинк	mg/l	2.93	2.8	2.86	-	2.97	-	-	-	-	2.93	2.92	2.94

Слика 2.6. Вредности отпадних вода при мешању са осталим отпадним водама у случају фабрике која се бави производњом заштитних средстава и алата.

Резултати мерења акредитоване лабораторије													
Параметри	Јединица мере	Број процеса											
		1	2	3	4	5	6	7	8	9	10	11	12
Алуминијум	mg/l	3.98	4.4	4.5	-	-	-	-	-	-	4.8	4.78	3.22
Азот из амонијака	mg/l	180	48	-	43	37	55	-	-	-	39	-	-
НРК	mgO ₂ /l	498	159	138	254	275	598	-	-	-	478	457	378
Гвожђе	mg/l	4.2	4.25	-	4.54	4.48	-	-	-	-	4.46	4.45	4.87
Флуориди	mg/l	59	34	61	-	59	-	-	-	-	41.5	-	-
Азот и нитрати	mg/l	-	6.9	6.8	6.5	-	5.9	-	-	-	6.14	-	-
Угљоводоници	mg/l	11.6	11.5	11.7	11.6	11.8	11.9	-	-	-	10.8	11.2	10.7
Фосфор	mg/l	2.91	2.83	2.79	2.85	2.87	2.79	-	-	-	2.5	2.4	2.3
Токсичност за рибе		6.9	4.85	3.2	6.9	7.1	6.9	-	-	-	6.9	7.2	6.9

Слика 2.7. Вредности отпадних вода при испуштању у реципијент у случају фабрике која се бави производњом алата и машинских делова.

Резултати мерења акредитоване лабораторије													
Параметри	Јединица мере	Број процеса											
		1	2	3	4	5	6	7	8	9	10	11	12
АОХ	mg/l	2.07	2.09	2.02	1.9	1.9	1.91	-	-	-	2.06	2.06	1.86
Арсен	mg/l	0.12	-	-	-	-	-	-	-	-	-	-	-
Баријум	mg/l	-	-	-	-	-	2.93	-	-	-	-	-	-
Олово	mg/l	1.03	-	-	-	0.92	-	-	-	-	1.03	-	0.93
Кадмијум	mg/l	0.73	-	-	-	0.27	-	-	-	-	0.6	-	0.27
	kg/t	0.65	-	-	-	-	-	-	-	-	-	-	-
Слободни хлор	mg/l	1.03	1.08	-	1.1	-	1.02	-	-	-	1.03	-	-
Укупни хлор	mg/l	1.03	1.07	1.13	1.4	-	-	-	-	-	1.63	1.15	1.39
Хром VI	mg/l	0.86	0.76	0.46	0.5	-	-	-	-	-	0.35	-	0.38
Кобалт	mg/l	-	-	1.95	-	-	-	-	-	1.86	-	-	-
Цијанид	mg/l	0.64	-	-	-	-	1.26	-	-	-	0.44	-	-
Бакар	mg/l	1.03	1.07	-	-	-	-	-	-	-	1.03	1.25	1.39
Никл	mg/l	1.13	0.98	-	1.5	-	-	-	-	-	1.94	1.8	1.67
Жива	mg/l	-	-	-	-	-	-	-	-	-	-	-	-
	kg/t	-	-	-	-	-	-	-	-	-	-	-	-
Селен	mg/l	-	-	-	-	-	-	-	-	-	-	-	-
Сребро	mg/l	0.8	-	-	-	-	-	-	-	-	-	-	-
Сулфиди	mg/l	1.77	1.97	-	1.9	-	-	-	-	-	-	-	-
Калај	mg/l	2.93	-	2.89	-	2.99	-	-	-	-	-	-	-
Цинк	mg/l	3.03	2.9	2.96	-	3.07	-	-	-	-	3.03	3.12	3.04

Слика 2.8. Вредности отпадних вода при мешању са осталим отпадним водама у случају фабрике која се бави производњом алата и машинских делова.

3. Систем за прераду отпадних вода (СПОВ)

Систем за прераду отпадних вода (СПОВ), који ће бити описан у овом поглављу, користи се у металско-хемијској индустрији која се бави наменском производњом. Истраживања и резултати до којих се дошло приликом тестирања и пуштања у рад проверена су од стране сертификованих лабораторија и Завода за јавно здравље Републике Србије.

3.1. Третман отпадне воде процесом неутрализације

Неутрализација отпадних вода је веома значајна, а у многим случајевима и неопходна фаза обраде воде пре њеног испуштања у природни реципијент. Ако је отпадна вода загађена искључиво високом концентрацијом H^+ или OH^- јона, неутрализација је једина фаза обраде пре испуштања у природне реципијенте. Ово је нужно, јер су законски одређене границе рН вредности за отпадне воде које се испуштају у природне реципијенте. Ако отпадна вода садржи и друге загађиваче који се могу уклонити физичко-хемијском обрадом, неутрализација је неопходна као претходна фаза обраде.

Без обзира због којих се потреба врши, неутрализација отпадних вода захтева утрошак киселина или база, значи захтева улагања. Неутрализацију треба вршити на најјефтинији и најефикаснији начин, а да би се то постигло, потребно је познавати основне законитости и начине њеног извођења.

Пре него што се приступи процесу неутрализације мора се одредити параметри, како они који се односе на извођење процеса, тако и они који су везани за инвестициона улагања и оперативне трошкове. Низ параметара се може добити једино након лабораторијских испитивања карактеристика отпадних вода. Мање количине отпадних вода се обично неутралишу шаржним поступком, а веће континуалним. У извесним случајевима отпадну воду је погодније претходно сакупљати у прихватним резервоарима, па повремено, зависно од динамике настајања, неутралисати. Приликом избора начина неутрализације треба имати на уму да су континуални процеси рационалнији, да се могу лакше контролисати и аутоматизовати, али да нису увек и најрационалнији и могу се оправдати једино код постројења већег капацитета. Посебно је значајан избор неутрализационог агенса. Неутрализација јаким киселинама и базама је најефикаснија и најбржа, али у принципу најскупља. Због тога се користе јефтиније материје као што су угљен-диоксид и креч од киселих, односно кречно млеко од алкалних.

Кад год је могуће, а при неутрализацији индустријских отпадних вода то је чест случај, треба употребљавати она средства која ће потпомоћи или још боље, изазвати таложње и других загађивача присутних у отпадној води (слика 3.1).

Овим ће се олакшати и појефтинити следећа фаза обраде. Наравно, када се вода након неутрализације испушта у природне реципијенте, морају се задовољити законске одредбе у погледу рН вредности ефлуета.

Неутрализација воденим раствором креча тече знатно брже од неутрализације кречом. Водени раствор креча, који се користи за неутрализацију млечних масти, потребно је претходно припремити. Припрема се врши у посебном суду који се интензивно меша, а у који се доводи вода и креч у праху. Пошто је растворљивост $\text{Ca}(\text{OH})_2$ у води мала, суд се мора интензивно мешати. И поред интензивног мешања за постизање zasiћеног раствора потребно је дуже време. Неутрализација кречним млеком је суштински истоветна неутрализацији кречом. Разлика је у томе што се код овог поступка претходно припрема zasiћен раствор $\text{Ca}(\text{OH})_2$ који садржи одређену количину у води нераствореног, суспендованог, CaO .

Када су у питању да креч и кречно млеко треба истаћи и да они имају велики значај за уклањање колоидно растворених органских материја из отпадних вода, као што је случај са водама из млекара. Овај начин „неутрализације“ има честу примену, јер обезбеђује задовољавајући степен неутрализације који се може и подешавати по потреби, а обезбеђује и бистрење отпадне воде, јер нерастворене калцијумове соли (калцијев сапун) формирају талог [3]-[5].



Слика 3.1. Када за неутрализацију.

3.2. Процес прераде отпадне воде таложењем

Таложење припада групи седиментационих процеса, односно процеса сепарације чврсте и течне фазе под утицајем гравитације, при чему су код таложења чврсте честице теже од течности и издвајају се наниже. Примена таложења је могућа под условом да су суспендоване честице нестабилне у односу на суспензију због своје величине или густине, односно да се претходним третманом (флокулацијом или коагулацијом) могу превести у нестабилне агломерате. На такве честице, односно агломерате у хидраулички мирној средини, примаран је утицај гравитационе силе. Операцију таложења карактерише један улазни ток и два излазна тока. Термин „суспензија“ обично се користи за улазни ток хемијских и интерних система са ниском концентрацијом чврсте материје, док се термин „ретки муљ“ примењује на исте системе са високом концентрацијом чврсте материје. Излазни ток течности назива се „бистри флуент“, док „муљ“ представља излазни ток исталоженог чврстог материјала.

Главни циљ примене операције таложења јесте добијање бистре пречишћене воде (ефлуента) и што згуснутијег муља. Уређаји у којима се одиграва операција таложења називају се таложници. Често се, обзиром на примарну намену, сусреће и назив „избистривач“. Иако постоје различите изведбе таложника, основне зоне (делови) сваког таложника су *улазна зона* кроз коју се суспензија уводи у уређај, *зона бистрења* (таложења) у којој се при хидраулички мирним условима одбија сепарација чврсто-течно, *зона муља* у којој се издвојени чврсти материјал прикупља, сабија и затим изводи из уређаја као муљ, и *излазна зона* кроз коју се бистри ефлуент изводи из уређаја.

Честице се таложе из суспензије у зависности од концентрације чврстог материјала у суспензији и карактеристика самих честица. На основу ових особина описана су четири различита начина класа таложења:

1. *Слободно таложење*, које представља таложење дискретних честица из разблажене суспензије, при чему честице немају никакву или имају врло слабу тенденцију ка флокулацији. Овакав начин таложења је типичан за разблажене суспензије чврстог зрнастог инертног материјала (нпр. песак).
2. *Степињено таложење*, које представља таложење честица које имају тенденцију флокулације из разблажене суспензије. Честице се током таложења спајају, повећавајући тако масу појединих агрегата и брзина таложења.
3. *Зонално таложење*, настаје када се честице из суспензија умерених концентрација таложе у непосредној близини једна другој. Међучестичне силе у овој ситуацији су довољно јаке да осигурају да честице остану у константним

позицијама једна у односу на другу. Као резултат, честице се таложе као јединствена маса, формирајући међупростор између таложених честица и течности.

4. *Сабијање (компресија)*, које представља таложење које је карактеристично за суспензије флокулентних честица високе концентрације. Ове честице стварају одређену физичку структуру, па до даљњег таложења може да дође једино сабијањем настале структуре. Сабијање се одиграва под дејством тежине честица које се и даље таложе из течности на створену структуру и улазе у њен састав. Овај начин таложења одвија се обично у доњим слојевима дебелих муљних маса.

У основи таложник мора да испуни ефикасно уклањање чврстих материја из суспензије у циљу добијања чистог филтрата. Поред тога, таложник има и низ подређених намена, чије остваривање може бити значајно (нпр. може се радити о потреби измене згртача муља због појаве гасних мехурова услед делимичне биолошке разградње у сталоженом муљу). Сви таложници који данас постоје деле се на вертикалне, хоризонталне и кружне. За протоке отпадних вода мање од $100 \text{ m}^3/\text{h}$ користе се вертикални таложници (слика 3.2). Они се конструишу као округли базени са коничним дном чији је нагиб износи најмање 45° (како би се обезбедило клизање муља). Улазни део је у облику цилиндра са дифузионим конусом који се налази у центру таложника. Централни цилиндар се пројектује за брзину суспензије $0,03 \text{ m/s}$, док његова висина одговара висини цилиндричног дела таложника. Дубина цилиндричног дела таложника је обично једнака његовом пречнику, док се запремина и висина коничног дела димензионишу на основу количине муља дводневне акумулације [6]-[14].



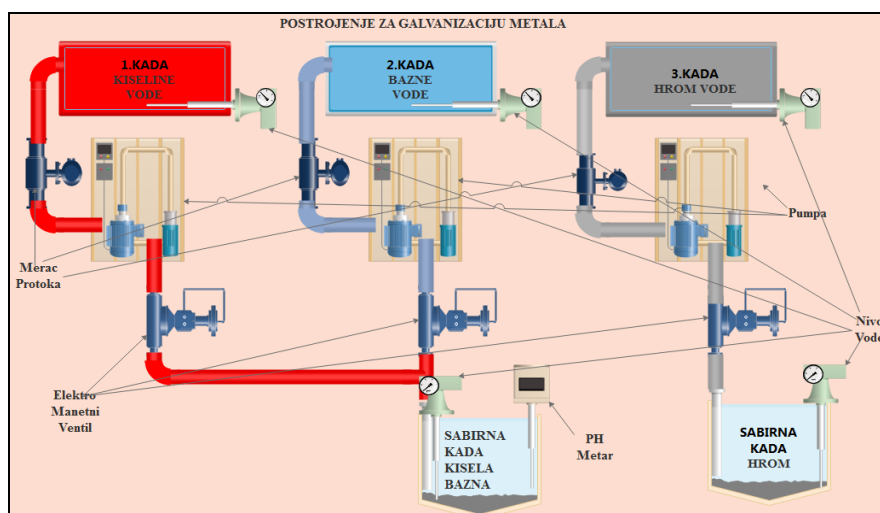
Слика 3.2. Ламеларни таложник.

3.3. Принцип функционисања постројења

У овом потпоглављу биће описан принцип рада постројења за галванизацију и обраду метала и транспорт у постројење за прераду и пречишћавање отпадних вода. У постројењу за галванизацију и обраду метала врши се одвајање три врсте отпадних вода:

- киселе отпадне воде,
- базне отпадне воде,
- хромиране отпадне воде и отпадне воде које садрже тешке метале.

У свим наведеним случајевима, процес пречишћавања започиње испуштањем отпадне воде која се гравитационим путем доводи у каду. Ту се врши одвајање страних елемената (стакло, дрво, пластика, метал, итд.) који се у току транспорта кроз цевоводе могу наћи у води. Након одвајања страних тела, вода се пумпом транспортује у одговарајуће сабирне каде (слика 3.3). Свака од њих је опремљена мерачом нивоа воде и мерачом протока воде, као и пумпом за пребацивање воде у сабирну каду. У сабирној кади за базне и киселе воде налази се рН-метар који мери рН вредност отпадних вода [15].



Слика 3.3. Процеси прераде отпадних вода.

На слици 3.4. приказан је доводни канал за базну и киселу отпадну воду. У овом каналу врши се мерење протока, односно количине воде која се доводи у једну од када за неутрализацију. Канал је тачно дефинисаних димензија и на њему су уграђени претварачи притиска који су повезани са PLC контролером који прерачунава количину пристигле воде. Изнад канала постављају се магнети за одвајање металних делова коју су се нашли у цевоводу након испуштања воде из

индустрије. Поред ових магнета у овом постројењу постоје и специјално дизајнирани филтери и решетке које служе за одвајање немагнетних материјала (гума, дрво, стакло и пластика), односно свих материјала које магнети не могу привући. За довод отпадне воде са садржајем тешких метала користи се цевовод на коме се налази ултразвучни мерач протока воде. На улазу у каду за обраду воде са тешким металима налазе се филтери за одстрањивање отпадних материјала пластике, стакла, металних делова и дрвета како би се заштитила опрема у кади (пумпе, мешачи, мерни инструменти, итд.).

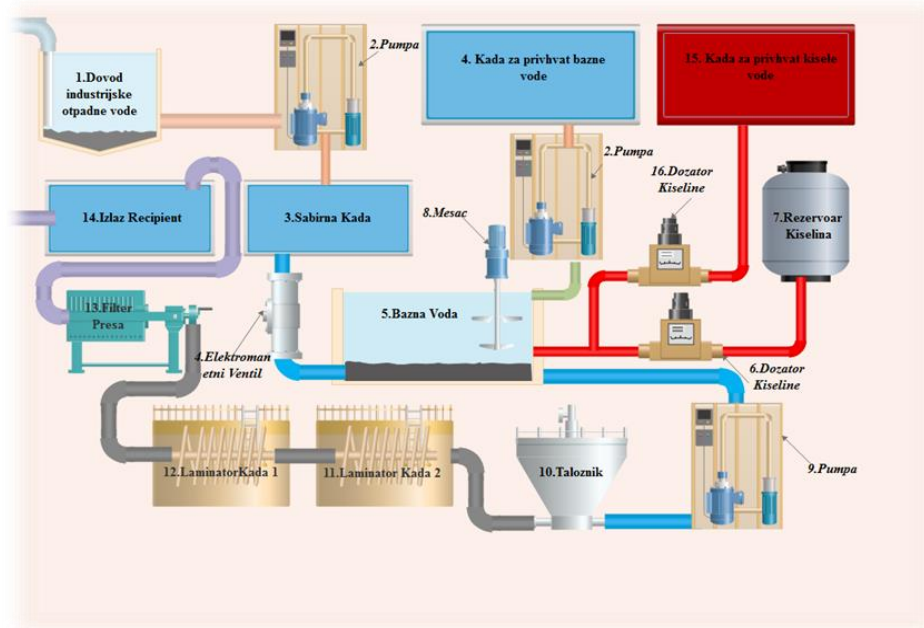


Слика 3.4. Доводни канал за базну и киселу отпадну воду.

3.3.1. Прерада базних отпадних вода

Након што се пумпама (блок 2) из пријемне каде (блок 1) отпадна вода доведе у сабирну (блок 3), врши мерење њене рН вредности (слика 3.5). Када се установи да су рН вредности отпадне воде изнад 7, она се транспортује у резервоар за базну воду (блок 5). У овом резервоару се налази мешач (блок 8) који спречава таложење нечистоћа. Након извесног времена, у резервоару се врши поновно мерење рН вредности воде. Ако се испостави да је измерена вредност већа од 8, коришћењем дозатора (блок 6) додаје се одређена количина киселине из резервоара киселина (блок 7) након чега врши се мешање воде уз упумпавање кисеоника. Ако после ове процедуре рН вредност воде не буде у границама између 6 до 8, врши се ново додавање концентроване киселине. Та процедура се понавља све док се не добије жељена рН вредност воде. Када се то деси, вода се пумпом (блок 9) транспортује

до таложника (блок 10), где се, након извесног времена, врши поновна провера њене рН вредности. У следећој фази вода се одводи у ламинирану каду (блокови 11 и 12), где се врши додатно таложње и пречишћавање. Након тога, вода долази у "излазни резервоар" (блок 14), где се врши последње мерење рН вредности. У зависности од добијених вредности, вода ће се испустити у реципијент или ће се вратити на поновну обраду [16].



Слика 3.5. Процес прераде базних отпадних вода.

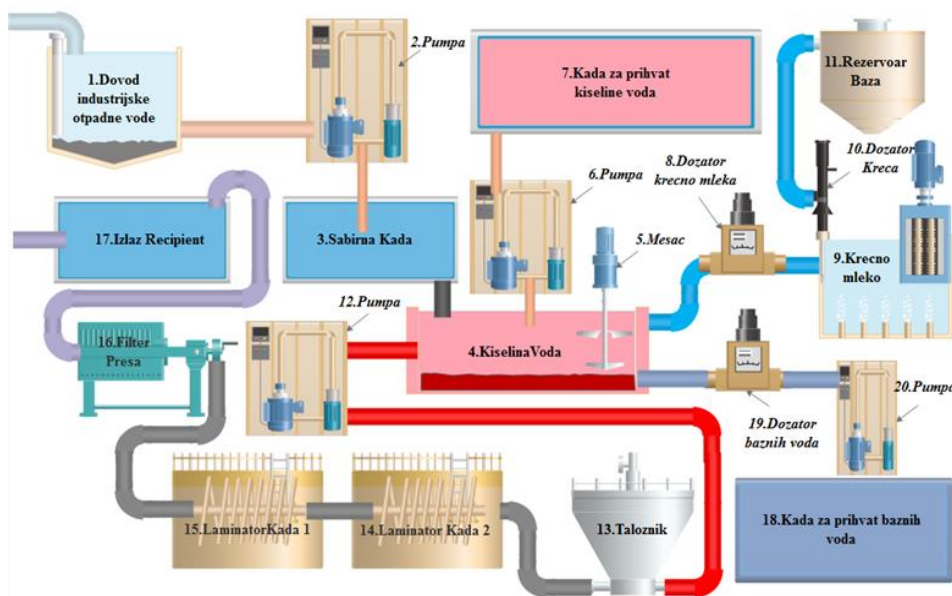
На слици 3.6. приказан је резервоар за прихват и припрему киселине. Направљена концентрација киселине се цевоводом прослеђује до дозатора где се врши одговарајуће дозирање.



Слика 3.6. Резервоар за припрему киселине.

3.3.2. Прерада киселих отпадних вода

Као и у претходном случају, отпадна вода се прво доводи у пријемну каду (блок 1), да би се потом пумпама (блок 2) "пребацила" у сабирну каду (блок 3) где се врши мерење њене киселости (слика 3.7). Након тога, вода се транспортује у резервоар за киселу воду (блок 4), у којем се, коришћењем мешача (блок 5), њена концентрација одржава стабилном. У резервоару за киселу воду врши се поновна провера рН вредности воде, и ако је она мања од 5, помоћу дозатора (блок 8) врши се додавање кречног млека (блок 9). Након тога, врши се мешање уз упумпавање кисеоника. Ако нису испуњени услови за исправне параметре воде (од 6 до 8), врши се поновно дозирање кречног млека. Вода остаје у овом резервоару све док њена вредност не буде у границама од 6 до 8. Кад овај услов буде испуњен, вода се пумпом (блок 12) транспортује до таложника (блок 13), где извесно време "мирује" (ради талочења). Након тога, вода се пребацује у ламинирану каду (блокови 14 и 15) у којој врши додатно талочење и пречишћавање. У последњем кораку, вода долази у резервоар „излаз реципијент“ (блок 17), где се врши мерење њене рН вредности. У зависности од добијених резултата, вода се или испушта у реципијент или враћа поново на обраду [16].



Слика 3.7. Процес прераде киселих отпадних вода.

На слици 3.8 приказан је резервоар за смештање базе, која се транспортује пужним транспортером до каде за прављење кречног млека. У овој индустрији због великог капацитета производње инсталиран је силос за прихват до 40000 kg креча. Допремање креча у праху врши се камион-цистернама које својим пумпама убацују цевоводом креч у резервоар. У резервоар су уграђени ултразвучни сензо-

ри нивоа који врше мерење нивоа креча у резервоару. Податке шаљу микроконтролеру који врши приказ на командној табли, односно врши унос промена количине креча у резервоару у базу података. На основу ових података могуће је благовремено вршити набавку сировине. У остале две индустрије резервоари за складиштење креча су знатно мањих капацитета до 500 kg.



Слика 3.8. Резервоар за смештање базе.

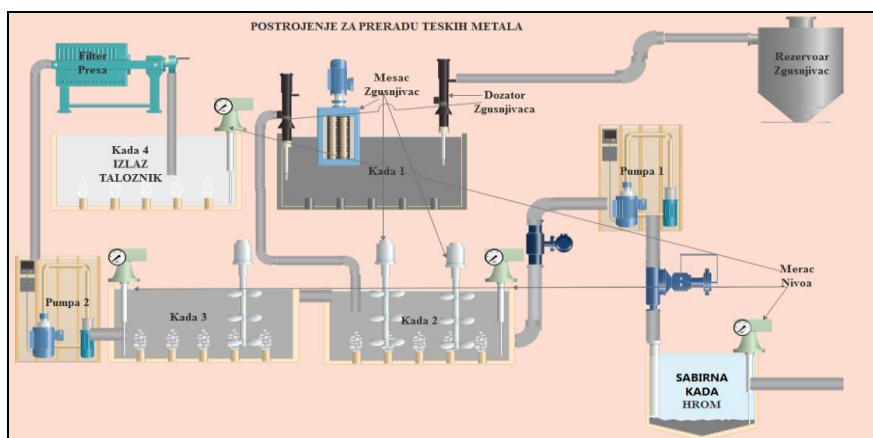
Када се пужним транспортом изврши дозирање базе, односно креч допреми до каде за производњу кречног млека, врши се додавање чисте воде, након чега се укључује мешач. Цео поступак је у потпуности аутоматизован: рачунар прати ниво креча у резервоару, врши одговарајуће дозирање, отвара вентил за чисту воду и започиње мешање. Овај поступак је временски дефинисан и након истека потребног времена могуће је вршити дозирање кречног млека. На слици 3.9 приказана је када за производњу кречног млека.



Слика 3.9. Када за производњу кречног млека.

3.3.3. Прерада тешких метала у отпадној води

Када отпадна вода садржи тешке метале, она се прво пребацује у каду 1 (слика 3.10), односно у каду за припрему полиелектролита (слика 3.11). Ту се врши њено мешање, мерење нивоа, и убацивање кисеоника и згушњивача. Таква смеша се, помоћу другог дозатора, пребацује у каду 2, где се врши њено вишечасовно мешање. Након тога, вода, преко таложника, из каде 2 прелази у каду 3, где се наставља њено мешање и убацивање кисеоника. У претпоследњој фази, вода се, помоћу пумпе 2, транспортује у филтер пресу (слика 3.12). Улога ове пресе је да под великим притиском филтер платна у преси (притисак износи и до 100 бара) исцеди чисту воду, која се поново враћа на проверу квалитета и након тога се или поново третира или испушта у реципијент (када 4) [17]. Треба навести и да се на филтер платнима сакупљају честице муља, које се након истресања са платна пакују у пластичне џакове и одвозе се на уништавање. Тиме се уједно и завршава процес обраде воде са примесама тешких метала.



Слика 3.10. Процес одвајања тешких метала из отпадне воде.



Слика 3.11. Када за припрему полиелектролита.



Слика 3.12. Филтер преса.

3.3.4. Испуштање прерађених отпадних вода

Испуштање воде након прераде је последња акција у постројењу. Након провере рН вредности воде и Redox потенцијала, вода се цевоводом води до канала за испуст у реципијент (слика 3.13).



Слика 3.13. Испуштање прерађене воде у реципијент.

3.4. Увођење мерно-информационог система

Због застарелости опреме, која је уграђена пре 40 година, било је немогуће ефикасно управљати постројењем. Поменута опрема није омогућавала 24-часовно праћење (мониторинг) процеса рада, што је једна од главних карактеристика свих модерних постројења. Осим тога, донете су и нове законске норме које прописују обавезу бележења података о пријему количине отпадних вода у постројење и ефикасности њене прераде. Ове законске обавезе није било могуће испоштовати употребом старе опреме.

3.4.1. Инсталирање мерних и надзорних уређаја

За мерне уређаје се често каже да представљају „очи и уши“ сваког индустријског система. Они мере различите физичке величине, исказују их у електричном облику, и потом такву информацију прослеђују микроконтролеру који врши обраду сигнала. Први мерни уређаји који су били инсталирани у модернизовани систем били су сензори температуре Pt-100 (слика 3.14). Њихов задатак је да мере температуру отпадне воде приликом њеног уласка у постројење. Ови сензори су направљени од материјала који има позитиван термички коефицијент, тако да са повећањем температуре расте вредност отпора коју сензор генерише.



Слика 3.14. Pt-100 сензор који је инсталиран у постројењу.

После инсталације Pt-100 сензора, инсталиране су рН и Redox електроде које мере Redox потенцијал (слика 3.15) и рН вредност отпадних вода (слика 3.16). За ове електроде је битно да не остану без течности, јер их у супротном није могуће користити за даља мерења [18].



Слика 3.15. рН електрода која је инсталирана у постројењу.



Слика 3.16. Redox електрода која је инсталирана у постројењу.

На неколико места инсталирани су и дигитални рН-метри, који могу истовремено да мере Redox потенцијал и рН вредност отпадних вода (слика 3.17). Овим уређајима се управља помоћу тастера и LCD екрана на којем се приказују резултати мерења. Овај уређај поседује и комуникациони модул захваљујући којем се може повезати са PLC уређајем (енг. *Programmable Logic Controller*).



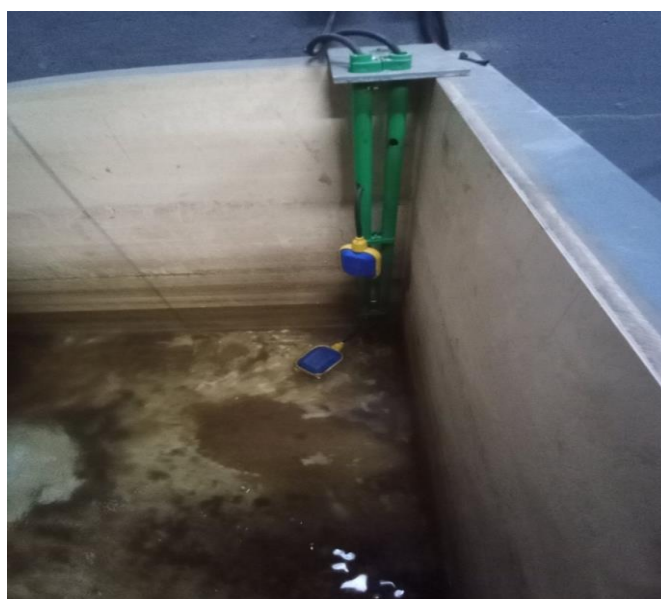
Слика 3.17. Дигитални рН-метар који је инсталиран у постројењу.

Након дигиталних рН-метара, инсталирани су мерачи протока воде на улазу и на излазу из постројења (слика 3.18). Избор је пао на електромагнетне мераче због њиховог једноставног принципа рада (лопатице мерача, приликом проласка воде, окрећу магнет који индукује струју, а онда се та струја, помоћу трансмитера, шаље до PLC-а који уноси њене вредности у базу података).



Слика 3.18. Мерач протока воде који је инсталиран у постројењу.

За потребе мерења нивоа воде коришћени су сензори са пловком (слика 3.19). Принцип рада ових сензора је да се постављају изнад жељеног нивоа воде. У случају да ниво воде пређе жељену вредност, сензори се издижу у води, почињу да пливају и у том тренутку жива која се налази унутар ових сензора спаја контакте и врши се акција укључења аларма или укључења пумпи за регулацију нивоа воде.



Слика 3.19. Мерач нивоа воде који је инсталиран у постројењу.

У модернизовани систем уграђени су и нови електромагнетни вентили (слика 3.20) за пропуштање воде [19]. Ови уређаји раде на принципу отварања шпулне приликом пуштања напона кроз намотаје соленоида. У постројењу су примењене различите величине ових уређаја у зависности од димензија цеви [20], [21].



Слика 3.20. Електромагнетни вентил који је инсталиран у постројењу.

Последњи уређаји који су били инсталирани у модернизовани систем биле су Интернет камере (слике 3.21 и 3.22). Оне су биле монтиране на неколико места у постројењу, да би потом, путем Интернет везе, слале слике удаљеном PLC-у.



Слика 3.21. Камера за мерење нивоа крчног млека.



Слика 3.22. Камера за мерење нивоа неутрализације.

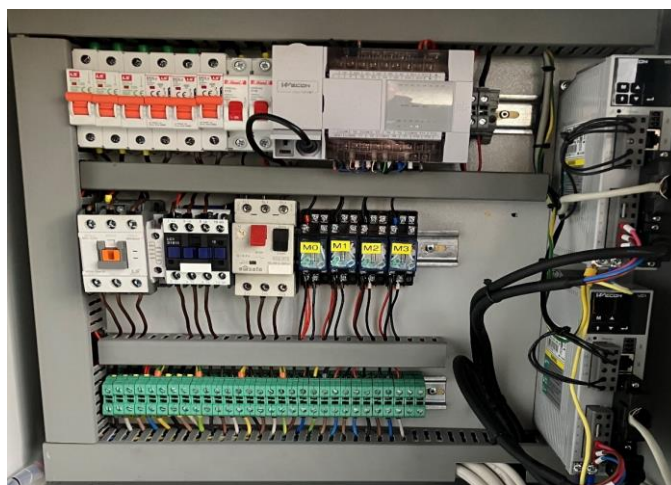
3.4.2. Инсталирање командно-управљачких уређаја

Релеји и контактори су једни од главних уређаја за управљање актуаторима и моторима (служе за укључивање/искључивање струјног кола) (слика 3.23). Израђују се у више варијанти у зависности од радне струје, командног напона и броја парова контакта. Постоје нормално отворени и нормално затворени контакти.



Слика 3.23. Контактори и релеји који су инсталирани у постројењу.

Да би се регулисао проток и контролисало отварање и затварање вентила, коришћени су драјвери. Ова електронска кола су одговорна за регулисање и напајање корачних и серво мотора. Примарна разлика између њих лежи у њиховим капацитетима снаге, конкретно у количини струје коју троши серво мотор [20]. На слици 3.24 је приказан изглед ормана са драјверима серво мотора, док је на слици 3.25 приказан фреквентни регулатор. Оба се користе за регулисање брзине мешача, регулисање брзине мотора пумпе, итд.



Слика 3.24. Драјвер серво мотора који је инсталиран у постројењу.



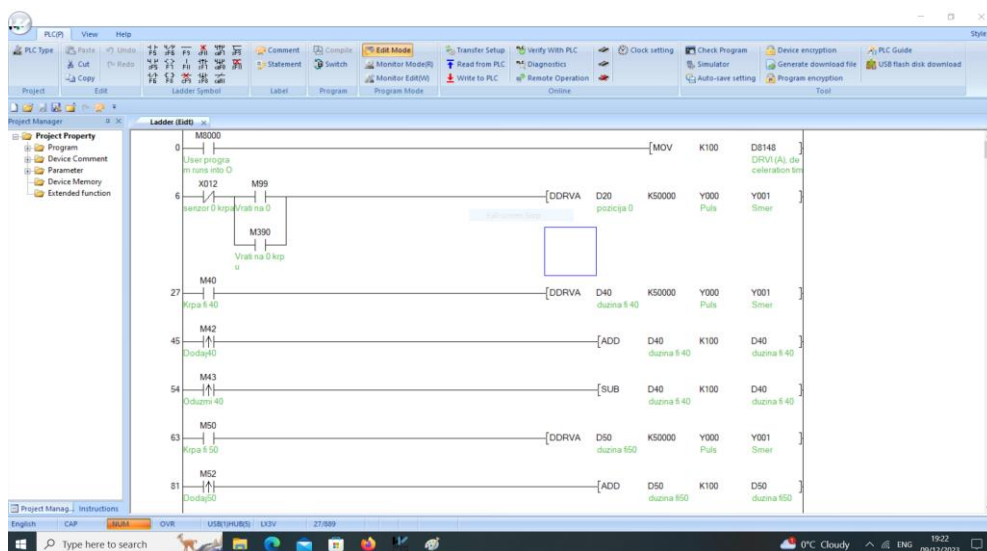
Слика 3.25. Фреквентни регулатор мотора који је инсталиран у постројењу.

Ниједан од наведених уређаја не би имао своју улогу у систему да није повезан са PLC-ом (слика 3.26). Његова улога јесте да прикупља податке са различитих мерних и надзорних уређаја, а затим анализира и покреће излазе (команде) према унапред одређеним параметрима. Са могућношћу посматрања и документовања података који се односе на трајање рада, PLC може самостално да започне и прекине процесе између осталих функција. У модернизованом систему, он се користи за прикупљање информација са сензора, за управљање актуаторима, за потребе отварања и затварања електромагнетних вентила, за потребе укључивања и искључивања мешача, итд.



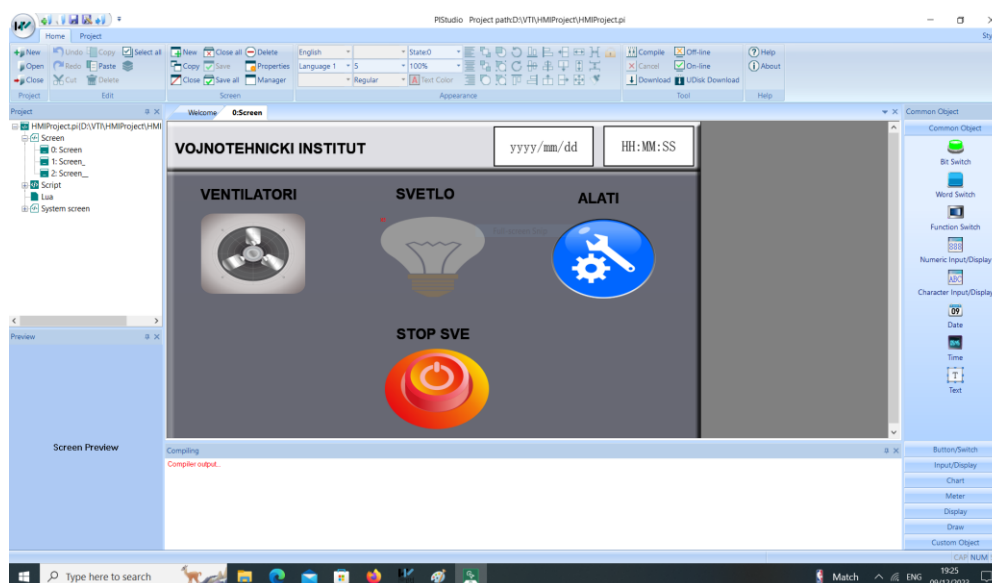
Слика 3.26. PLC који је инсталиран у постројењу.

Наравно, да би све ове функције могао да обавља, PLC мора бити програмиран. Најчешће коришћен програмски језик за ту сврху јесте ледер (енг. *ladder*) логика. Ради се о једном од најинтуитивнијих језика који користи графички интерфејс са електричним дијаграмима (слика 3.27).



Слика 3.27. Ледер дијаграм који је коришћен током модернизације постројења.

Како би оператори могли да остваре интеракцију са PLC-ом, било је потребно инсталирати НМИ уређаје (енг. *Human Machine Interface*). Ови оператерски интерфејси омогућавају корисницима да прегледају и уносе одговарајуће параметре и информације у PLC у реалном времену (слике 3.28 и 3.29) [22].



Слика 3.28. Једна од фаза пројектовања интерфејса НМИ уређаја.



Слика 3.29. Оператерски интерфејс HMI уређаја који је инсталиран у постројењу.

За потребе даљинског мониторинга постројења, отклањања евентуалних грешака и преноса података са PLC и HMI уређаја у виртуелну приватну мрежу коришћени су V-BOX уређаји (слика 3.30) [23].

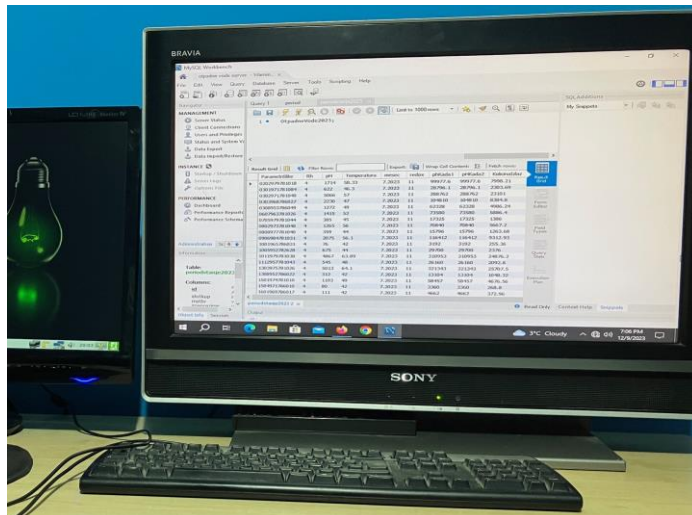


Слика 3.30. V-BOX уређај који је инсталиран у постројењу.

3.4.3. Уписивање мерних података у MySQL базу

Да би постројење могло да функционише на унапређен начин, било је неопходно је да се сви мерни подаци чувају у једној централизованом бази података. За ту потребу коришћена је MySQL база, која је инсталирана на серверу којег покреће

OpenSUSE Linux оперативни систем (слика 3.31). Овај оперативни систем је изабран како због високе поузданост и једноставног интерфејса, тако и због чињенице да за његову инсталацију није потребна лиценца (пошто је отвореног кода).



Слика 3.31. Уписивање мерних података у MySQL базу.

За потребе смештања мерних података у MySQL базу било је потребно написати неколико програмских кодова у Python-у. Један од њих, приказан у Прилогу 1, служи за упис података који су исчитани са улазних портова једног PLC-а. У питању су три вредности температуре, три вредности рН параметара и две вредности Redox потенцијала. Други програмски код, приказан у Прилогу 2, служи за упис измерених вредности температура и рН параметара са улазних портова другог PLC-а.

3.5. Проблеми са видео надзором постројења

Агресивне киселине, које су саставни део отпадних вода, често су механички оштећивале сензоре за мерење нивоа воде. Као резултат тога, у постројењу је долазило до изливања отпадних вода, а самим тим и до оштећења друге инсталиране опреме (пумпе за избацивање воде, мешачи, рН-метри, итд.) (слика 3.32). Како би се у будућности избегли овакви сценарији, дошло се на идеју да се, паралелно са осталим врстама сензора, поставе и камере које ће, путем Интернет везе, пратити нивое воде у кадама (слике 3.21 и 3.22).

Након инсталације камера, у постројењу није долазило до изливања отпадне воде. Међутим, повремено се јављао један други проблем: подаци са постављених камера нису увек били уписивани у MySQL базу (табела 3.1). Овај проблем први

пут је уочен у фебруару 2017. године, односно непуних месец дана након што су камере биле инсталиране.



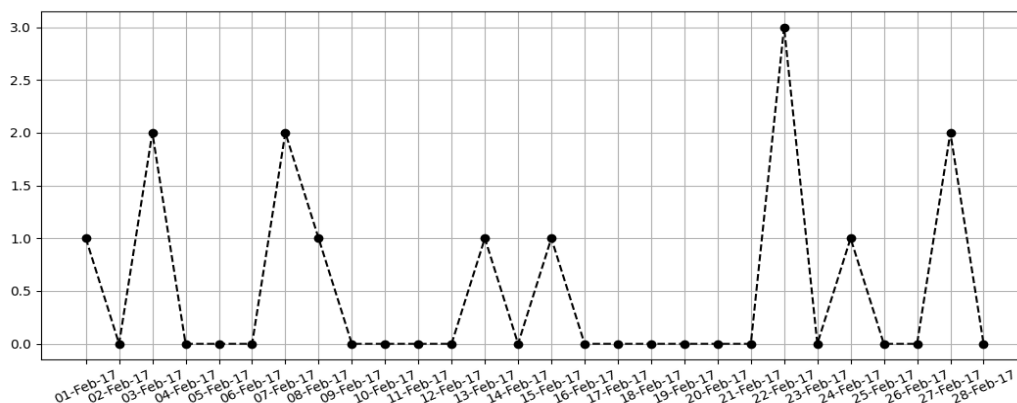
Слика 3.32. Приказ оштећене опреме након плављења постројења.

У наредних неколико месеци вршило се испитивање узрока ових проблема са акцентом на упоређивање тренутака "отказивања камера" (слике 3.33 - 3.39) са са-тницама дневних активности постројења. Оваква стратегија, међутим, није довела

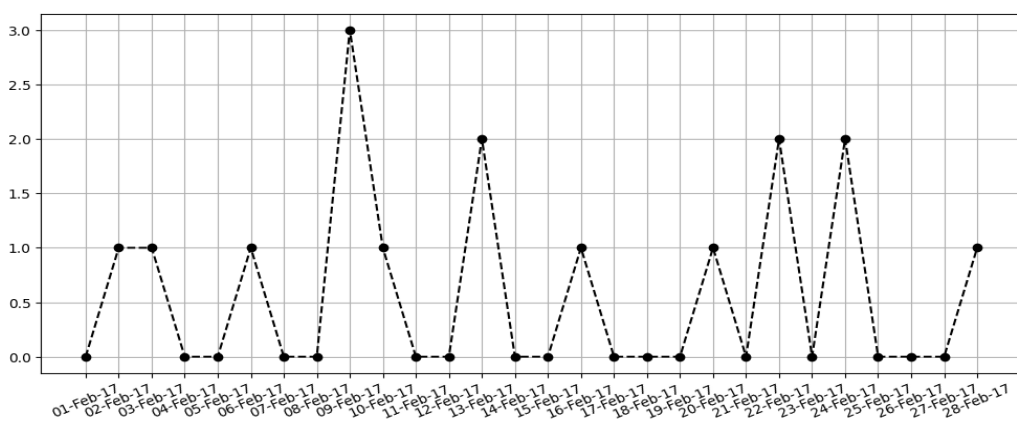
Табела 3.1. Проблеми са уписом података које су слале Интернет камере.

Камера за каду 1	Камера за каду 2	Камера за каду 3	Датум/Време
413	715	413	25.2.2017 12:00
421	714	421	25.2.2017 12:02
413	703	413	25.2.2017 12:04
434	723	434	25.2.2017 12:06
422	713	422	25.2.2017 12:08
468	759	468	25.2.2017 12:10
454	773	454	25.2.2017 12:12
Null	823	Null	25.2.2017 12:14
455	746	455	25.2.2017 12:16
460	Null	460	25.2.2017 12:18
491	779	491	25.2.2017 12:20
498	782	498	25.2.2017 12:22
524	828	524	25.2.2017 12:24
534	880	534	25.2.2017 12:26
541	896	541	25.2.2017 12:28
585	893	585	25.2.2017 12:30
602	927	602	25.2.2017 12:32
572	895	572	25.2.2017 12:34
549	905	549	25.2.2017 12:36
521	989	521	25.2.2017 12:38
482	942	482	25.2.2017 12:40
455	874	455	25.2.2017 12:42
450	859	450	25.2.2017 12:44
454	860	454	25.2.2017 12:46
446	859	446	25.2.2017 12:48
448	862	448	25.2.2017 12:50

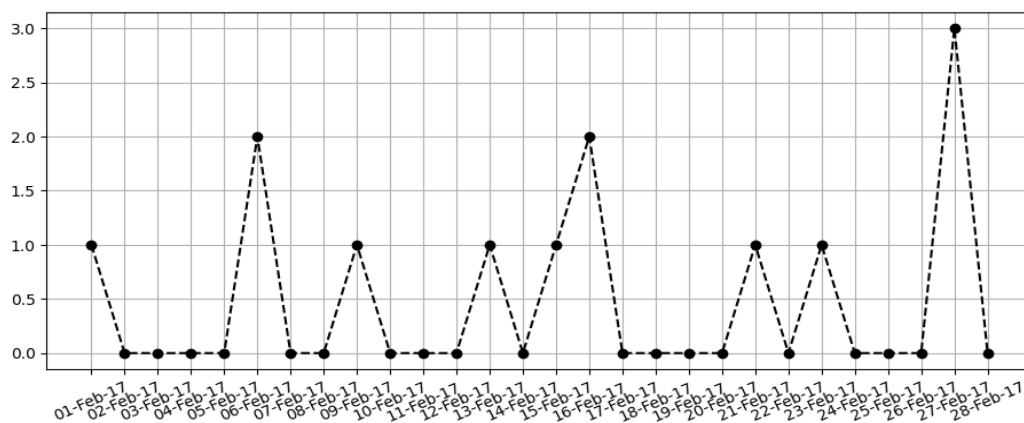
до одговарајућег закључка. Стога се претпоставило да главни проблем ипак лежи у несавршености UTP кабла, преко којег су Интернет камере биле повезане са PLC-овима. То је, уједно, и мотивисало аутора ове дисертације да се посвети конструкцији заштитних кодова који би овај проблем могли да реше.



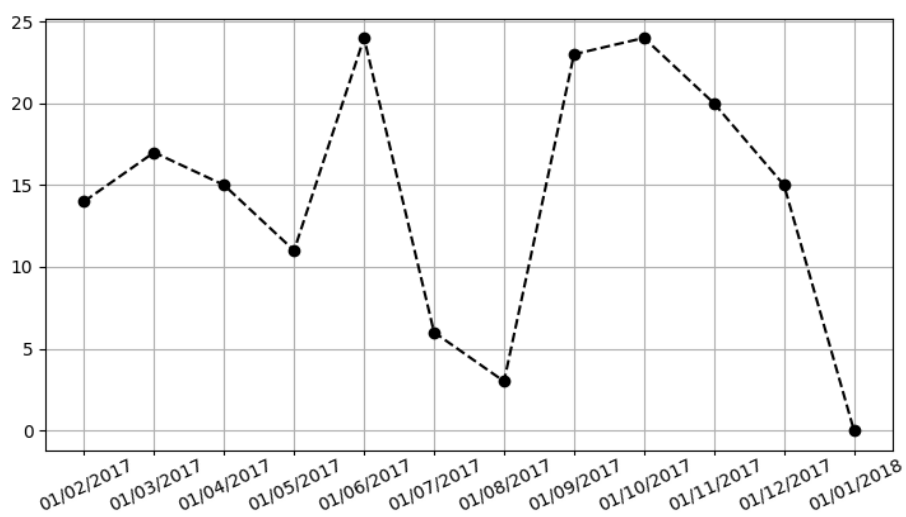
Слика 3.33. Број "отказа" камере за каду 1 током фебруара 2017. године.



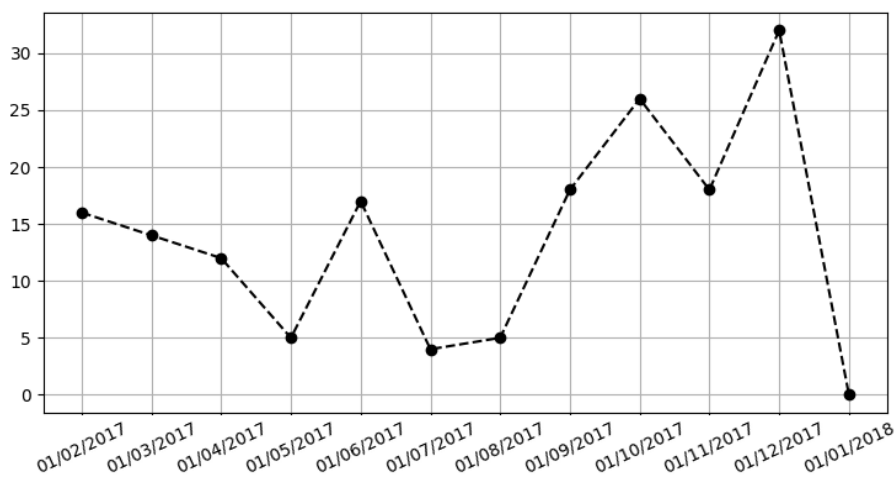
Слика 3.34. Број "отказа" камере за каду 2 током фебруара 2017. године.



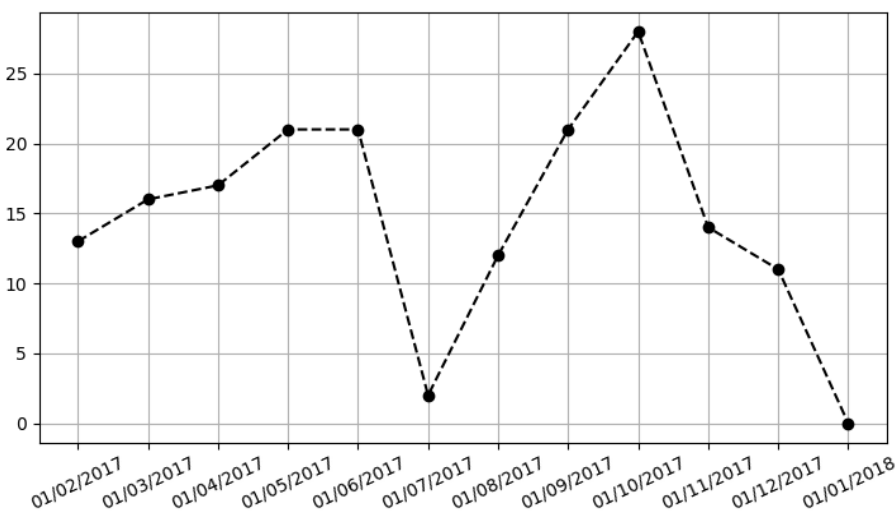
Слика 3.35. Број "отказа" камере за каду 3 током фебруара 2017. године.



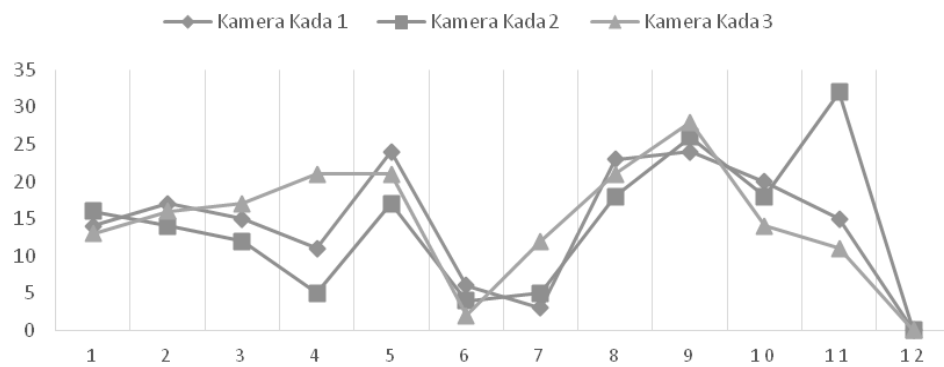
Слика 3.36. Број "отказа" камере за каду 1 током 12 месеци.



Слика 3.37. Број "отказа" камере за каду 2 током 12 месеци.



Слика 3.38. Број "отказа" камере за каду 3 током 12 месеци.



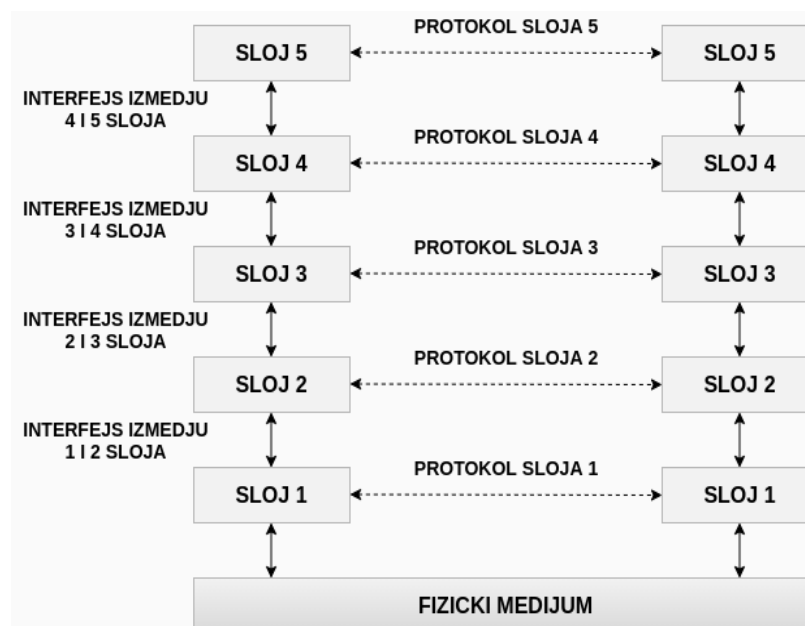
Слика 3.39. Број "отказа" све три камере током 12 месеци.

4. Пренос података заснован на Интернет протоколу

У првим пројектима рачунарских мрежа хардвер је имао главну улогу. Оваква стратегија, међутим, данас више није заступљена. У одељцима који следе биће појашњене основе Интернет комуникација које су засноване на софтверском структурирању података.

4.1. Хијерархије протокола

Данашње комуникационе мреже су организоване као скуп слојева, односно нивоа. Број слојева, њихови називи, садржај и функција зависе од мреже. Сваки од слојева је заправо виртуелна машина која пружа неку услугу слоју који се налази изнад њега. Овакав концепт је заступљен у рачунарским наукама, али има различите називе: скривање података, енкапсулирање података, итд. [24]. Основа је у томе да конкретна софтверска/хардверска компонента обезбеди корисницима услугу, а да при том од корисника "сакрије" примењене алгоритме.

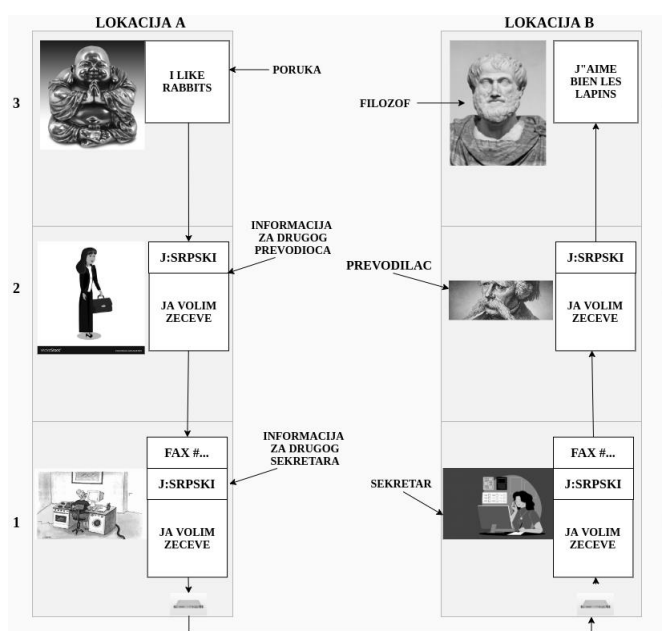


Слика 4.1. Слојеви, протоколи и интерфејси у рачунарским комуникацијама.

У реалним ситуацијама подаци се никада не преносе директно од n -слоја на једном рачунару ка n -слоју другог рачунара. Комуникација је олакшана преношењем података и контролних информација са једног слоја на слој директно испод њега, све док не дође до најнижег слоја. Стварна размена комуникације се одвија преко физичког медијума који се налази испод слоја 1, као што је илустровано на слици 4.1 (виртуелна комуникација је означена тачкастим, а стварна комуникација

пуним линијама). Интерфејс између суседних слојева дефинише функционалности и услуге које доњи слој пружа горњем слоју [25].

Скуп свих слојева и протокола назван је једним именом архитектура мреже. Неопходно је да спецификација архитектуре садржи довољан број информација како би пројектант за сваки слој могао да напише програм или пројектује хардвер који ће поштовати правила одговарајућег протокола. Детаљи реализације, као и спецификација интерфејса нису део архитектуре („сакривени“ су у рачунарима). Није обавезно ни да интерфејси на свим умреженим рачунарима буду исти, уколико сваки од рачунара правилно користи све протоколе.



Слика 4.2. Особа-преводиалац-секретар архитектура.

Концепт комуникације између слојева може се боље разумети кроз пример разговора две особе (слика 4.2), од којих једна говори урду и енглески језик, а друга кинески и француски језик (процеси 3. слоја). Да би ове особе могле да комуницирају, оне морају да ангажују одговарајуће преводиоце (процеси 2. слоја), а сваки преводилац мора да ангажује секретара (процеси 1. слоја). Прва особа, на пример, жели да каже шта мисли и осећа према зечевима одређене врсте, па из тог разлога шаље поруку преводиоцу поруку на енглеском, путем интерфејса 2/3. Преводиоци су се међусобно договорили да користе језик који обоје познају и говоре (на пример српски), па ће поруку „*I like rabbits*“ превести на „*Ja volim zecuve*“. Преводилац даје инструкције секретару да проследи поруку протоколу првог слоја путем факса, а након стижања поруке до циља, врши се превод на француски језик, па путем интерфејса 2/3 бива прослеђена другом преводиоцу. Пре-

водиоци могу прећи са српског језика на фински, ако се претходно око тога обоје сложе, а при том тиме неће изменити своје интерфејсе према 1. и 3. слоју. На сличан начин, секретари поруку могу разменити путем телефона или електронске поште чиме не ремете друге слојеве (не морају их чак ни обавестити). Сваки процес може прикључити поруци додатне податке који су намењени искључиво „колеги“ (не прослеђују се слоју изнад).

4.2. Референтни модели

У наредним одељцима биће обрађени референтни модели OSI и TCP/IP, који се користе у рачунарским комуникацијама. Протоколи који директно користе OSI модел су данас врло ретки, али је сам модел још увек важећи. TCP/IP модел се, с друге стране, не користи много, али протоколи засновани на њему имају широку примену на Интернету.

4.2.1. OSI референтни модел

Међународна организација за стандардизацију (ISO) је замислила OSI модел као средство за покретање стандардизације протокола применљивих на различите слојеве, чиме се отвара пут међународној униформности [24]. Овај модел служи и за повезивање система који су отворени за комуникацију са другим системима.

SLOJEVI OSI MODELA		SLOJEVI TCP/IP MODELA	
APLIKACIJA	7	APLIKACIJA	
PRESENTACIJA	6		
SESIJA	5		
TRANSPORTNI	4	TRANSPORTNI	
MREZNI	3	MEDJUMREZNI	
VEZA PODATAKA	2	ZA POVEZIVANJE RACUNARA S MREZOM	
FIZICKI	1		

Слика 4.3. Референтни модели: OSI (лево) и TCP/IP (десно).

Као што се може видети (слика 4.3), OSI модел има седам слојева. Њихове функције ће укратко бити описане у наставку.

1. Физички слој има улогу преноса битова података путем комуникационог канала. Један од проблема његовог пројектовања је да се обезбеди да онда када једна страна пошаље бит 1 да и друга страна такође прими бит 1, не бит 0. Обично треба размислити о томе колики ће напон представљати јединицу, а колики нулу,

колико наносекунди може да траје бит, да ли ће пренос моћи да се обавља у оба смера истовремено, итд. Главна одговорност дизајнера у овом слоју је да се позабаве механичким и електричним аспектима система. Они такође играју кључну улогу у синхронизацији интерфејса и дизајнирању физичког медијума који се користи за пренос, а који се налази испод физичког слоја.

2. Слој везе података је углавном задужен за структурирање битова података у фрејмове (пакете података). Један фрејм може да садржи од неколико стотина до неколико хиљада битова података. Пренос оквира се одвија у секвенцијалном редоследу, а у случају поуздане услуге прималац потврђује пријем сваког оквира слањем оквира потврде пошиљаицу. Један од проблема који се може јавити у слоју везе података је неусаглашеност брзина слања и примања података. Други чести проблем јесте провера приступа заједничком каналу. Унутар слоја везе података постоји јединствени подслој познат као средњи подслој контроле приступа, који је посебно дизајниран да се „ухвати у коштац“ са овим посебним изазовом.

3. Мрежни слој управља радом подмреже и игра кључну улогу у дизајну мреже адресирајући рутирање пакета од њиховог извора до циљаног одредишта. Ово подразумева разматрање два приступа: коришћење статичких табела које су интегрисане у мрежну инфраструктуру или примену аутоматских ажурирања за динамичко прилагођавање рута. Ако је превише пакета присутно у подмрежи, они се могу међусобно ометати, стварајући уска грла. Руковање загушењима је такође одговорност мрежног слоја. Када пакет мора да путује кроз више мрежа могу настати проблеми везани за адресирање. Примарни циљ мрежног слоја је превазилажење горе поменуте проблематике и могућност беспрекорног међусобног повезивања различитих мрежа.

4. Транспортни слој на предаји дели податке са виших слојева (на мање делове) и прослеђује их мрежном слоју. На пријемној страни, транспортни слој обезбеђује да се сви делови података секвенцијално испоручују вишим слојевима. Транспортни слој преузима одговорност одлучивања о специфичној услузи која треба да се пружи слоју сесије и, на крају, корисницима мреже. Најраспрострањенији облик транспортне везе је канал од тачке до тачке без грешака који гарантује испоруку података у тачном редоследу којим су првобитно послати. Међутим, на располагању су и друге потенцијалне врсте транспортних услуга, као што је пренос изолованих порука без икакве гаранције за њихову испоруку.

5. Слој сесије омогућава корисницима за рачунарима да успоставе међусобне сесије. Сесије нуде различите услуге, као што су синхронизација (провера преноса података како би се омогућио наставак преноса од тачке прекида у случају пада система и накнадног опоравка) и контролу дијалога.

6. Слој презентације се, за разлику од нижих слојева, бави семантиком и синтаксом пренетих података. За ефикасну комуникацију између рачунара, неопходно је успоставити стандардизован и апстрактан оквир за размену структура података. Ово укључује дефинисање и кодирање структура података у универзално прихваћеном формату.

7. Слој апликације обухвата низ протокола који задовољавају свакодневне захтеве корисника. Међу њима, HTTP (енг. *HyperText Transfer Protocol*) се појављује као широко прихваћен протокол апликације, који служи као основни оквир за прегледање веб страница. Поред тога, други протоколи, укључујући FTP (енг. *File Transfer Protocol*) и IMAP (енг. *Internet Message Access Protocol*) се користе за ефикасан пренос датотека и комуникацију путем електронске поште.

4.2.2. TCP/IP референтни модел

Референтни модел који ће надаље бити описан потиче од ARPANET-а: истраживачке рачунарске мреже, која је крајем 1960-их повезивала неколико америчких универзитета и владиних институција. Услед забринутости америчке владе да би ARPANET могао бити уништен у потенцијалном нападу Совјетског Савеза, наложено је да његов дизајн буде "отпоран" на било какав напад. То значи да је његова архитектура морала бити флексибилна, и да је морала омогућити пренос једноставних датотека и говора у реалном времену. У деценијама које су уследиле ARPANET архитектура је постала позната као TCP/IP референтни модел. Име је дато на основу два протокола која се најчешће користе: TCP-а (енг. *Transmission Control Protocol*) [25] и IP-а (енг. *Internet Protocol*) [26]. За разлику од традиционалног OSI-а, TCP/IP референтни модел има четири слоја. Њихове функције ће укратко бити описане у наставку.

1. Слој везе описује које везе се морају направити да би биле задовољене потребе за пренос података. У суштини, овај слој се може сматрати интерфејсом пре него традиционалним слојем, јер олакшава везу између рачунара и преносне везе.

2. Интернет слој служи као основни стуб који повезује целокупну архитектуру. Његов примарни циљ је да олакша пренос пакета до њихових назначених крајњих тачака коришћењем IP адреса. Важно је имати на уму да пакети могу повремено стићи на своја одредишта другачијим редоследом него што су првобитно послати. У таквим случајевима, постаје дужност виших слојева да преуреде ове пакете у одговарајући низ.

3. Транспортни слој је дизајниран да омогући конверзацију између процеса на изворном и одредишном рачунару (као OSI транспортни слој). Данас се, на транспортном слоју најчешће користе два протокола: TCP, који омогућава поуздан и

секвенцијалан пренос података, и UDP (енг. *User Datagram Protocol*), који је непоуздан и који се користи за потребе преноса аудио и видео података.

4. Слој апликације се налази на врху TCP/IP референтних модела, па самим тим садржи различите протоколе вишег нивоа, као што су FTP, HTTP, SMTP (енг. *Simple Mail Transfer Protocol*), итд. Наведени протоколи се користе приликом преноса веб страница, разних фајлова, аудио/видео података, када је потребно даљински приступити неком рачунару, итд.

4.2.3. Поређење OSI и TCP/IP референтних модела

OSI и TCP/IP референтни модели показују бројне заједничке карактеристике. Оба су изграђена на идеји серије самосталних протокола, где су функционалности слојева релативно сличне. На пример, у оба модела, очекује се да сваки слој понуди независну транспортну услугу од краја до краја која повезује крајње тачке и задовољава захтеве комуникације. Даљи слојеви изнад транспортног су апликације које су корисници услуге транспорта.

Упркос овим сличностима, ова два модела имају и много разлика. Очигледна разлика је број слојева: први модел (OSI) има седам слојева, а други (TCP/IP) четири. Друга разлика је у области комуникације са успостављањем везе или без ње. На мрежном слоју, OSI модел прихвата оба типа комуникације, док искључиво подржава комуникацију оријентисану на везу ка транспортном слоју [24]. TCP/IP референтни модел, с друге стране, подржава само један режим на мрежном слоју (без успостављања директне везе), али подржава оба режима на транспортном слоју. Иако се 1980-их чинило да ће OSI модел и његови протоколи преузети свет, то се, ипак, није десило. Разлози за то су бројни. Један од њих је лош тајминг, јер су TCP/IP протоколи већ били у широкој употреби. Други разлог јесте избор седам слојева, од којих су два скоро празни (слојеви сесије и презентације), док су два претрпани (слој везе за пренос података и мрежни слој). Трећи, и можда најважнији, разлог јесте мишљење да је OSI створен од стране Европске заједнице, а касније и Владе САД. Овај став је био само делимично тачан, али је сама идеја да иза OSI-а стоји гомила бирократа била одбојна код истраживача који развијају рачунарске мреже. Међутим, ни TCP/IP модел, који се данас користи на Интернету, није савршен. Прво, њему недостаје могућност разликовања између услуга, интерфејса и протокола. Сходно томе, није погодан за развој нових мрежа које укључују нове технологије. Друго, TCP/IP није погодан за описивање било ког стек протокола осим TCP/IP. Описати Bluetooth, на пример, је потпуно немогуће уколико се користи TCP/IP референтни модел. Треће, важно је напоменути да слој везе, упркос томе што се назива слојем, није у складу са конвенционалном дефиницијом слоја, пошто служи као интерфејс који повезује сло-

јеве мреже и везе података. Осим наведених недостатака, TCP/IP модел не прави јасну разлику између физичког слоја и слоја везе података, упркос њиховим инхерентним разликама. Физички слој се бави транспортним особинама медијума, док слој везе података разграничава почетак и крај оквира података и гарантује поузданост у преносу података.

4.3. Контрола преноса података на транспортном слоју

У претходним поглављима поменуто је да се на транспортном слоју углавном користе TCP и UDP протоколи. Њихова улога јесте да од апликације на предајној страни преузму, и да апликацији на пријемној испоруче одговарајуће податке. За ту потребу они користе два механизма: проверу грешке и адресирање апликација преко бројева портова. Иако су ове особине заједничке за оба протокола, њихови принципи рада се разликују. Ове разлике биће кратко објашњене у наставку.

4.3.1. Принцип рада TCP протокола

Једна од главних особина TCP протокола јесте да се пренос података обавља тек када се успостави веза између два рачунара. Та веза је увек од тачке до тачке, односно између два удаљена рачунара. Оног тренутка када се веза успостави, TCP преузима апликативне податке, дели их на сегменте, и сваком сегменту додаје заглавље које садржи неколико поља: број изворног порта (који идентификује апликацију на предаји), број одредишног порта (који идентификује апликацију на пријему), редни број сегмента, број потврде (редни број примљеног сегмента), величина заглавља (изражена у бајтовима), резервисана поља, контролни бити (URG, ACK, PSH, RST, SYN, FIN), величина прозора (број бајтова који се могу примити), Интернет контролни збир (за потребе детекције грешака), показивач на ургентне податке, опциони бити (за потребе подешавања величине прозора) и подаци (који су променљиве дужине) (слика 4.4).

Bajt	bit	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
0	0	Izvorišni port																Određišni port															
4	32	Redni broj																															
8	64	Broj potvrde																															
12	96	Veličina zaglavlja				Rezervisano 0 0 0			N S	C W R E	E C E G	U R C K	A S S H	P S S T	R Y N	S I N	F	Veličina prozora															
16	128	Internet kontrolni zbir (čeksuma)																Pokazivač na urgentne podatke															
20	160	Opcije + podaci																															
...	...																																

Слика 4.4. Изглед TCP заглавља.

Сваки пут када пошање неки сегмент, предајник покреће тајмер дефинишући тако време током којег ће чекати на потврду од пријемника да је послати сегмент успешно примљен (АСК порука). Ако за предвиђено време не стигне позитивна потврда од пријемника, предајник ће поново послати исти сегмент. Идентична процедура ће се десити и уколико предајник од пријемника добије поруку да је примљени сегмент садржи битске грешке (NACK порука). Провера интегритета примљеног сегмента остварује се коришћењем Интернет чексуме, чије ће особине бити објашњене у следећем поглављу. Овде је битно истаћи само то да Интернет контролне бите генеришу и предајник и пријемник, и да њиховим упоређивањем пријемник закључује да ли је примљени сегмент оштећен или не. С обзиром да се не може предвидети динамика пријема АСК и NACK порука, дешава се да TCP сегменти пристижу у редоследу другачије од послатог. Због тога се, приликом коришћења овог протокола, мора резервисати одређени меморијски простор који ће омогућити пријем свих сегмената како би се они могли сортирати пре испоруке апликацији. По завршетку преноса података, веза између рачунара се раскида.

4.3.2. Принцип рада UDP протокола

За разлику од TCP-а, UDP омогућава да удаљени рачунари комуницирају без успостављања везе. Пренос података одвија се по принципу од једног рачунара ка више рачунара, што UDP чини изузетно погодним за масовне мултимедијалне комуникације. Да би се направила додатна разлика у односу на TCP, сегменти података који се размењују називају се датаграмима, и њихова достава и редослед пријема нису загарантовани. UDP, као и у случају TCP-а, преузима апликативне податке, дели их на сегменте, и сваком сегменту додаје мало заглавље које садржи четири поља: број изворног порта (који идентификује апликацију на предаји), број одредишног порта (који идентификује апликацију на пријему), дужина (дефинише величину датаграма) и Интернет контролни збир (за потребе детекције грешака) (слика 4.5) [27].

Бajt	bit	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
0	0	Izvorišni port																Određišni port															
4	32	Dužina																Internet kontrolni zbir (čeksuma)															

Слика 4.5. Изглед UDP заглавља.

Када пријемник, коришћењем Интернет чексуме, утврди да је UDP датаграм оштећен, он ће га одбацити без слања NACK поруке. Исто тако, ако је датаграм примљен без грешке, пријемник ће га прихватити без слања АСК поруке. Дакле, ни у једном од наведених случајева предајник нема информацију шта се десило са

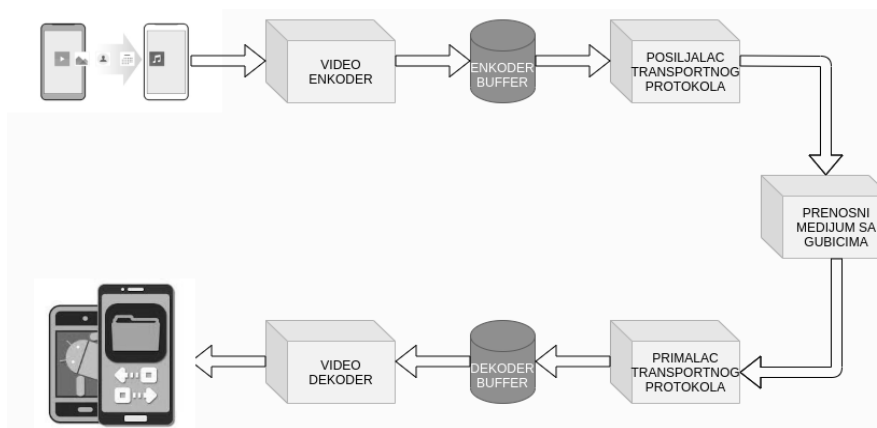
послатим подацима. Због тога се често каже да се UDP преноси податке по тзв. "best effort" принципу, односно да не води рачуна о поузданој достави.

4.4. Пренос мултимедијалних података

Упркос напретку у погледу побољшања пропусног опсега, брзина преноса ће увек бити оскудан ресурс за висококвалитетне апликације. С друге стране, из дана у дан расте потреба за испоруком мултимедијалних података, као што су стриминг и видео на захтев. Под таквим околностима јасно је да видео кодеци морају бити у стању да реше различите проблеме који настају током преноса мултимедијалних података.

4.4.1. Систем видео комуникације

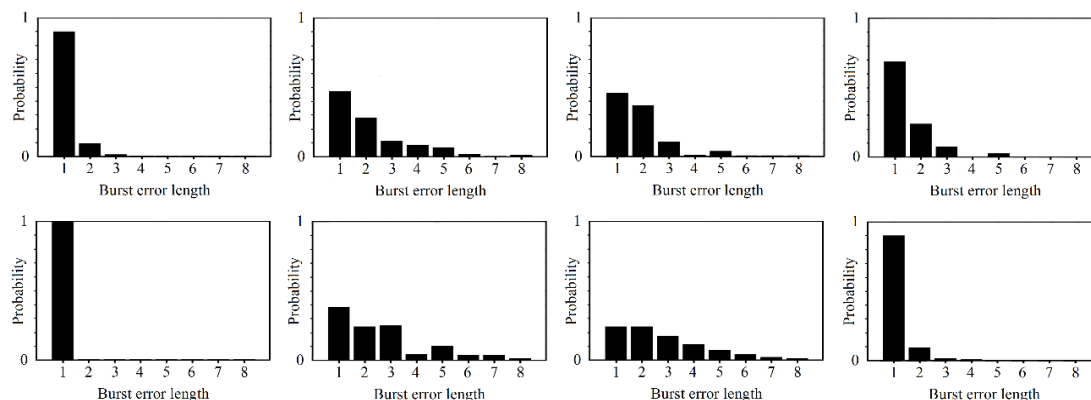
На слици 4.6 приказана је уопштена шема система за пренос видео сигнала преко Интернета. За разлику од фотографија, видео оквири укључују релативне информације о времену, које се морају одржавати да би се осигурала савршена реконструкција на пријему [28]. Осим тога, видео кодери користе и технике компресије како би смањили стварну количину пренетих података. У општем случају, видео енкодер генерише битове података који се чувају у баферу и који представљају компресовани видео ток. Током преноса, систем може закаснити, изгубити или оштетити један или више бита. На пријемној страни, бафер декодера компензује варијације кашњења како би се пренос података одржао константним.



Слика 4.6. Систем за пренос мултимедијалних података.

4.4.2. Битске грешке у преносу података: проблеми и постојећа решења

Наведени начин комуникације може да се одвија путем жичних или бежичних линкова. У бежичним мрежама, фејдинг и сметње узрокују појаву вишеструких узастопних битских грешака, док се у жичним мрежама, поред загушења, јављају и каналне грешке које захватају мали број суседних бита [29]-[34] (слика 4.7).



Слика 4.7. Битске грешке које се јављају у жичаним мрежама.

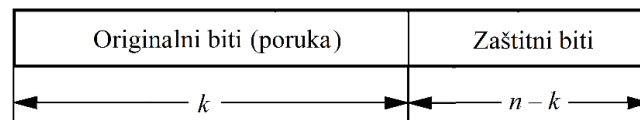
Главни проблем, с тим у вези, јесте чињеница да мултимедијалне апликације, на транспортном слоју, користе UDP. Овај протокол, као што је већ наведено, обезбеђује брз и непоуздан пренос података по принципу *"провери да ли пакет има битске грешке: ако их нема, прихвати га, ако их има, одбаци га"*. То значи да на пријем неће стићи сви пакети, већ само они који нису оштећени. Стога се поставља питање како надоместити недостајуће пакете, а да корисник то не примети? Ово питање навело је инжењере да решење за проблем недостајућих пакета *"потраже"* на апликативном слоју. То је урађено тако што су у све познате видео кодеке (MPEG-2, MPEG-4, H.264/AVC, итд.) интегрисани алгоритми који могу да реконструишу неколико изгубљених пакета. Ови алгоритми се често зову AL-FEC кодовима [28] (енг. *Application Layer Forward Error Correction Codes*) и деле се на:

1. Дводимензионалне кодове за проверу парности који користећи минималну редундансу и једноставне операције могу да реконструишу 1 изгубљени пакет.
2. Рид-Соломон кодове, који користећи минималну редундансу и релативно сложене операције могу да реконструишу m изгубљених пакета ($m > 1$).
3. Степеничасте LDPC кодове, који користећи већу редундансу и релативно једноставне операције могу да реконструишу m изгубљених пакета ($m > 1$).

У свим наведеним случајевима, под редундансом се мисли на додатне пакете који се на основу *"правих"* пакета генеришу на пријему и који се заједно са њима шаљу ка предајнику. Јасно је да оваква стратегија може изазвати велико загушење саобраћаја, поготово у жичаним мрежама.

5. Целобројни заштитни кодови

Задатак сваког комуникационог система је да што верније пренесе поруку од предајника до пријемника. Како би то било могуће, порука се мора претворити у низ бита, који се потом, применом алгоритама заштитног кодовања, "припремају" за слање кроз физички медијум (жични или бежични). Суштина те "припреме" јесте да се на основу k оригиналних бита (порука) генерише $n - k$ заштитних бита који се додају на крај поруке и који омогућавају пријемнику да детектује или коригује грешке које се могу десити током преноса података (слика 5.1).

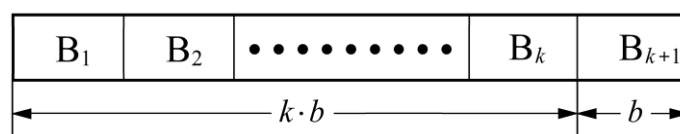


Слика 5.1. Структура пакета која се шаље од предајника ка пријемнику.

Литература која се бави заштитним кодовањем доминантно је фокусирана на линеарне заштитне кодове [35], који користе матричне операције, што их чини неприкладним за софтверску имплементацију (без обзира на врсту процесора) [36], [37]. У овој дисертацији биће приказани потпуно другачији заштитни кодови, који се зову целобројни заштитни кодови. Име су добили по томе што користе целобројне операције које подржавају сви модерни процесори.

5.1. Заједничке особине свих целобројних заштитних кодова

За разлику од линеарних заштитних кодова, целобројни могу да исправљају битске грешке датог типа. То значи да је могуће изабрати одређене врсте грешака (нпр. оне које се најчешће јављају током преноса података), а потом конструисати целобројне заштитне кодове који могу да их исправе. Друга добра особина свих целобројних заштитних кодова јесте да имају исту структуру као чексам кодови. Другим речима, кодна реч се састоји од k b -битних бајтова података и једног b -битног контролног бајта (слика 5.2). У случају целобројних заштитних кодова, контролни бајт се израчунава као збир производа целобројних вредности бајтова података и коефицијената C_i , док се синдром примљене кодне речи S рачуна на исти начин као и код кодова за детекцију грешке, односно као разлика вредности контролног бајта на предаји и на пријему [38]. Обе ове чињенице су истакнуте у следећим дефиницијама.



Слика 5.2. Структура кодне речи код целобројних заштитних кодова.

Дефиниција 1. Нека је $Z_{2^b-1} = \{0, 1, \dots, 2^b - 2\}$ прстен целих бројева по модулу $2^b - 1$ и нека је $B_i = \sum_{n=0}^{b-1} a_n \cdot 2^n$ целобројна репрезентација b -битног бајта, где је $a_n \in \{0, 1\}$ и $1 \leq i \leq k$. Тада је код $C(b, k, c)$, дефинисан као

$$C(b, k, c) = \left\{ x \in Z_{2^b-1}^{k+1} : \sum_{i=1}^k C_i \cdot B_i \equiv B_{k+1} \pmod{2^b-1} \right\} \quad (5.1)$$

целобројни $(kb + b, kb)$ заштитни код, при чему је $x = (B_1, B_2, \dots, B_k, B_{k+1}) \in Z_{2^b-1}^{k+1}$ вектор кодне речи, $c = (C_1, C_2, \dots, C_k, 1) \in Z_{2^b-1}^{k+1}$ је вектор коефицијената, док је $B_{k+1} \in Z_{2^b-1}$ цео број.

Дефиниција 2. Нека су предата кодна реч, примљена кодна реч и вектор грешке означени са $x = (B_1, B_2, \dots, B_k, B_{k+1}) \in Z_{2^b-1}^{k+1}$, $y = (\underline{B}_1, \underline{B}_2, \dots, \underline{B}_k, \underline{B}_{k+1}) \in Z_{2^b-1}^{k+1}$ и $e = (\underline{B}_1 - B_1, \underline{B}_2 - B_2, \dots, \underline{B}_k - B_k, B_{k+1} - \underline{B}_{k+1}) = (e_1, e_2, \dots, e_k, e_{k+1}) \in Z_{2^b-1}^{k+1}$, респективно. У том случају, синдром S примљене кодне речи је дефинисан као

$$S = \sum_{i=1}^k C_i \cdot \underline{B}_i - \underline{B}_{k+1} \pmod{2^b-1} = \sum_{i=1}^{k+1} e_i \cdot C_i \pmod{2^b-1} \quad (5.2)$$

Из (5.2) је јасно да ненулта вредност S указује на присуство једне или више грешака унутар t b -битних бајтова ($1 \leq t < k + 1$). Декодер ће моћи да исправи ове грешке ако се одговарајући код конструише на следећи начин.

1. *Одређивање типа грешке коју код треба да детектује и/или да исправи.* Овај корак се, у суштини, своди на дефинисање вредности t и e_i . На пример, ако желимо конструисати кодове који могу да исправе једноструке битске грешке у оквиру једног b -битног бајта, вредности t и e_i ће бити $t = 1$ и $e_i = \pm 2^r$, где је $0 \leq r \leq b - 1$. С друге стране, ако треба конструисати целобројне заштитне кодове који могу да исправе једноструке битске грешке у оквиру два b -битна бајта, вредности t и e_i ће бити $t = 2$ и $e_i = \pm 2^r$, где је $0 \leq r \leq b - 1$.

2. *Дефинисање скупа синдрома који одговарају грешкама које код треба/мора да исправи.* У општем случају, овај скуп је дефинисан као

$$\zeta = \bigcup_{h=1}^t s_h \quad (5.3)$$

при чему је

$$s_1 = \{ e_{i_1} \cdot C_{i_1} \pmod{2^b-1} : 1 \leq i_1 \leq k+1 \} \quad (5.4)$$

$$s_2 = \{ e_{i_1} \cdot C_{i_1} + e_{i_2} \cdot C_{i_2} \pmod{2^b-1} : 1 \leq i_1 < i_2 \leq k+1 \} \quad (5.5)$$

$\vdots$

$$s_t = \{ e_{i_1} \cdot C_{i_1} + e_{i_2} \cdot C_{i_2} + \dots + e_{i_t} \cdot C_{i_t} \pmod{2^b-1} : 1 \leq i_1 < i_2 < \dots < i_t \leq k+1 \} \quad (5.6)$$

3. *Проналажење коефицијената C_i .* За сваку дужину бајта ($b \geq 2$) потребно је извршити компјутерску претрагу како би се пронашли коефицијенти C_i . Иако се број коефицијената повећава са повећањем b , горња теоријска граница (k_{max}) се, у

општем случају, не може се одредити (k_{max} зависи од "снаге" кодова који се желе конструисати: што су кодови "јачи", то је вредност k_{max} мања). Без обзира на ту чињеницу, вредности коефицијената C_i морају бити такве да

$$s_1 \cap s_2 \cap \dots \cap s_t = \emptyset,$$

$$|\xi| = \sum_{h=1}^t |s_h| \cdot \binom{k+1}{h},$$

где $|X|$ означава кардиналност X .

4. *Избор параметара кода и генерисање синдром табеле.* Број пронађених коефицијената C_i одређује број бајтова који се могу заштитити. Одабиром да ли ће се користити сви коефицијенти или не, одређује се величина кодне речи као и величина синдром-табеле (СТ). Ова табеле увек има $|\xi|$ уноса и генерише се на основу вредности t, b, k, e_i и C_i . Сврха сваког уноса је да опише однос између синдрома ненулте вредности, локација грешке и вредности грешке (слика 5.3).

Element of $\xi(S)$	Error location (i_1)	Error value (E_1)	Error location (i_2)	Error value (E_2)		Error location (i_t)	Error value (E_t)
$\xleftarrow{\quad b \quad}$	$\xleftarrow{\quad \lceil \log_2(k+1) \rceil \quad}$	$\xleftarrow{\quad b \quad}$	$\xleftarrow{\quad \lceil \log_2(k+1) \rceil \quad}$	$\xleftarrow{\quad b \quad}$		$\xleftarrow{\quad \lceil \log_2(k+1) \rceil \quad}$	$\xleftarrow{\quad b \quad}$

Слика 5.3. Битска величина једног уноса који се смешта у СТ.

Из горњих корака, јасно је да је процес изградње целобројних заштитних кодова независан од процедура кодовања/декодовања. Међутим, комплетности ради потребно је истаћи да комуникација између крајњих тачака почиње тек када се СТ генерише и ускладишти у локалним меморијама удаљених рачунара. У том случају, за сваку примљену кодну реч, декодер ће израчунати синдром S . Ако је његова вредност једнака нули, декодер ће претпоставити да је кодна реч без грешке. Међутим, ако је вредност S различита од нуле, декодер ће претражити СТ како би пронашао унос који садржи информације о локацији и вредности грешке (ради се о уносу код којег су првих b бита исти као и код синдрома S). Уколико такав унос постоји, декодер ће (у паралели) извести операције:

$$B_{i_1} = \underline{B}_{i_1} + E_1 \pmod{2^b - 1} \quad (5.7)$$

$$B_{i_2} = \underline{B}_{i_2} + E_2 \pmod{2^b - 1} \quad (5.8)$$

$\vdots$

$$B_{i_t} = \underline{B}_{i_t} + E_t \pmod{2^b - 1} \quad (5.9)$$

У супротном, прогласиће грешку која се не може исправити.

У досадашњим радовима [39]-[50] је показано да се целобројни заштитни кодови могу користити не само у различитим комуникационим системима, већ и за потребе заштите података у рачунарским меморијама. У наставку ће бити дат кратак преглед најзначајних класа целобројних заштитних кодова.

5.1.1. Целобројни SEC-B_{l/b}ED кодови

Целобројни заштитни кодови који имају способност да коригују једноструке битске грешке и да детектују l -битне рафалне грешке унутар једног b -битног бајта (једноструке l/b рафалне грешке) (целобројни SEC-B_{l/b}ED кодови) презентовани су у [39]. У раду је прво показано да целобројна вредност појединачне грешке износи $e_i = \pm 2^r$, где је $0 \leq r \leq b-1$, док је целобројна вредност l/b рафалне грешке једнака $e_i = \pm 2^r \cdot (2m-1)$, где је $0 \leq r \leq b-l$, $1 \leq m \leq 2^{t-1}$ и $1 \leq t \leq l$. Након тога, дате су следеће дефиниције и теорема.

Дефиниција 3. Скуп свих синдрома који одговара једноструким битским грешкама је дефинисан као

$$s_1 = \bigcup_{i=1}^{k+1} \{ \pm 2^r \cdot C_i \pmod{2^b-1} : 0 \leq r \leq b-1 \} \quad (5.10)$$

Дефиниција 4. Скуп свих синдрома који одговара једноструким l/b рафалним грешкама је дефинисан као

$$s_2 = \bigcup_{r=0}^{b-x} \bigcup_{u=1}^{2^{x-1}} \bigcup_{x=1}^l \bigcup_{i=1}^{k+1} \{ \pm 2^r \cdot (2u-1) \cdot C_i \pmod{2^b-1} \} \quad (5.11)$$

Теорема 1. Кардиналност скупа s_1 је

$$|s_1| = 2 \cdot b \cdot (k+1).$$

Доказ. Доказ је дат у [39].

Теорема 2. Кодови дефинисани са (5.1) могу да коригују једноструке битске грешке и да детектују једноструке l/b рафалне грешке ако постоји k различитих коефицијената C_i таквих да је

1. $|\xi| = |s_1| = 2 \cdot b \cdot (k+1)$,
2. $s_1 \cap s_2 = \emptyset$,
3. $s_2 \subseteq Z_{2^{b-1}} \setminus \{0,1\}$.

Доказ. Доказ је дат у [39].

Последњи корак у поступку конструисања целобројних SEC-B_{l/b}ED кодова јесте проналазак коефицијената C_i који задовољавају услове из Теореме 1. Пошто се овај задатак није могао извести без коришћења рачунара, било је неопходно написати одговарајући рачунарски програм. С обзиром да су целобројни SEC-B_{l/b}ED кодови намењени за потребе заштите рачунарских меморија, вредности параметра b , коришћене у рачунарским претрагама [39], биле су мале (табела 5.1).

Табела 5.1. Коефицијенти за неке целобројне SEC-B_{3/b}ED кодове.

$b = 8$											
9	13	19	21	25	43						
$b = 9$											
9	11	15	19	23	29	35	37				
$b = 10$											
9	11	13	19	21	23	35	47	49	51	59	71

5.1.2. Целобројни SEC-SB_{l/b}ED-DB_{h/b}ED кодови

Први корак у конструисању целобројних кодова који могу да коригују једноструке битске грешке и да детектују једноструке l/b рафалне грешке и h -битне рафалне грешке унутар два b -битна бајта (двоструке h/b рафалне грешке) јесте одређивање целобројних вредности наведених грешака [40]. Када је реч о првој врсти грешке, њена целобројна вредност је једнака $e_i = \pm 2^r$, где је $0 \leq r \leq b-1$ и $1 \leq i \leq k+1$. С друге стране, целобројна вредност једноструке l/b рафалне грешке једнака је $e_i = \pm 2^r \cdot (2u-1)$, где је $0 \leq r \leq b-x$, $1 \leq x \leq l$, $1 \leq u \leq 2^{x-1}$ и $1 \leq i \leq k+1$. Ако се две такве грешке догоде истовремено, јасно је да ће оне променити целобројне вредности два b -битна бајта за $e_i = \pm 2^s \cdot (2v-1)$ и $e_j = \pm 2^t \cdot (2w-1)$, где је $0 \leq s \leq b-y$, $1 \leq y \leq h$, $1 \leq v \leq 2^{y-1}$, $0 \leq t \leq b-z$, $1 \leq z \leq h$, $1 \leq w \leq 2^{z-1}$ и $1 \leq i < j \leq k+1$. На основу овога, дате су следеће дефиниције и теореме.

Дефиниција 5. Скуп синдрома који одговара једноструким битским грешкама је дефинисан као

$$s_1 = \bigcup_{r=0}^{b-1} \bigcup_{i=1}^{k+1} \{ \pm 2^r \cdot C_i \pmod{2^b-1} \} \quad (5.15)$$

Дефиниција 6. Скуп синдрома који одговара једноструким l/b рафалним грешкама је дефинисан као

$$s_2 = \bigcup_{r=0}^{b-x} \bigcup_{x=1}^l \bigcup_{u=1}^{2^{x-1}} \bigcup_{i=1}^{k+1} \{ \pm 2^r \cdot (2u-1) \cdot C_i \pmod{2^b-1} \} \quad (5.16)$$

Дефиниција 7. Скуп синдрома који одговара двоструким h/b рафалним грешкама је дефинисан као

$$s_2 = \bigcup_{s=0}^{b-y} \bigcup_{y=1}^h \bigcup_{v=1}^{2^{y-1}} \bigcup_{i=1}^k \bigcup_{t=0}^{b-z} \bigcup_{z=1}^h \bigcup_{w=1}^{2^{z-1}} \bigcup_{j=i+1}^{k+1} \{ \pm 2^s \cdot (2v-1) \cdot C_i \pm 2^t \cdot (2w-1) \cdot C_j \pmod{2^b-1} \} \quad (5.17)$$

Теорема 3. Кардиналност скупа s_1 је

$$|s_1| = 2 \cdot b \cdot (k+1).$$

Доказ. Доказ је дат у [40].

Теорема 4. Кодови дефинисани са (5.1) могу исправити једноструке битске грешке и детектовати једноструке l/b рафалне грешке и двоструке h/b рафалне грешке ако постоји k различитих коефицијената C_i таквих да је

1. $|\zeta| = |s_1| = 2 \cdot b \cdot (k+1)$,
2. $s_1 \cap s_2 = \emptyset$,
3. $s_1 \cap s_3 = \emptyset$,
4. $s_2, s_3 \subseteq Z_{2^{b-1}} \setminus \{0, 1\}$.

Доказ. Доказ је дат у [40].

Претходна теорема имплицира да је број елемената s_1 унапред познат, али не и њихове вредности. Оне се могу одредити само уз помоћ рачунара, што значи да је било потребно претражити коефицијенте C_i из прстена целих бројева по модулу $2^b - 1$. У [40] су приказани резултати који омогућавају конструкцију практичних SEC-SB $_{l/b}$ ED-DB $_{h/b}$ ED кодова (табела 5.2).

Табела 5.2. Коефицијенти за неке целобројне SEC-SB $_{l/32}$ ED-DB $_{4/32}$ ED кодове.

$l = 9$														
529	579	621	857	971	1119	3493	3593	3905	5689	6451	7349	8523	8921	9201
12923	13993	14941	14987	17825	18747	20925	21205	26879	29347	31205	32701	34469	36207	36437
$l = 10$														
1041	1079	1125	1163	1273	1475	1633	2197	2349	3623	3921	8063	9995	11159	12133
14317	14775	16721	17143	17363	19049	20195	21507	22215	23041	24329	26793	31477	36167	36961

5.1.3. Целобројни SB $_{h/b}$ EC-SB $_{l/b}$ ED кодови

Први корак у конструкцији целобројних кодова који могу га коригују једноструке h/b рафалне грешке и да детектују једноструке l/b рафалне грешке јесте одређивање целобројних вредности поменутих грешака. Из [41] се зна да је целобројна вредност једноструке h/b рафалне грешке једнака $e_i = \pm 2^{r_1} \cdot (2u_1 - 1)$, где је $1 \leq x_1 \leq h$, $0 \leq r_1 \leq b - x_1$, $1 \leq u_1 \leq 2^{x_1-1}$ и $1 \leq i \leq k + 1$. На основу овога, лако је закључити да је целобројна вредност једноструке h/b рафалне грешке једнака $e_i = \pm 2^{r_2} \cdot (2u_2 - 1)$, где је $1 \leq x_2 \leq l$, $0 \leq r_2 \leq b - x_2$, $1 \leq u_2 \leq 2^{x_2-1}$ и $1 \leq i \leq k + 1$. Међутим, да би разлика између ових грешака била јаснија, у једноструке l/b рафалне грешке су убрајане само оне чија дужина варира од $h + 1$ до $b - 1$.

Дефиниција 8. Скуп синдрома који одговара једноструким h/b рафалним грешкама је дефинисан као

$$s_1 = \bigcup_{r_1=0}^{b-x_1} \bigcup_{u_1=1}^{2^{x_1-1}} \bigcup_{x_1=1}^h \bigcup_{i=1}^{k+1} \{ \pm 2^{r_1} \cdot (2u_1 - 1) \cdot C_i \pmod{2^b - 1} \} \quad (5.18)$$

Дефиниција 9. Скуп синдрома који одговара једноструким l/b рафалним грешкама (искључујући једноструке h/b рафалне грешке) је дефинисан као

$$s_2 = \bigcup_{r_2=0}^{b-x_2} \bigcup_{u_2=1}^{2^{x_2-1}} \bigcup_{x_2=h+1}^l \bigcup_{i=1}^{k+1} \{ \pm 2^{r_2} \cdot (2u_2 - 1) \cdot C_i \pmod{2^b - 1} \} \quad (5.19)$$

Теорема 5. Кардиналност скупа s_1 је

$$|s_1| = [2^h \cdot (b - h + 2) - 2] \cdot (k + 1).$$

Доказ. Доказ је дат у [41].

Теорема 6. Кодови дефинисани са (5.1) могу да исправе једноструке h/b рафалне грешке и да детектују једноструке l/b рафалне грешке ако постоји k различитих коефицијената C_i таквих да је

1. $|\xi| = |s_1| = [2^h \cdot (b - h + 2) - 2] \cdot (k + 1)$,
2. $s_1 \cap s_2 = \emptyset$,
3. $s_2 \subseteq Z_{2^{b-1}} \setminus \{0, 1\}$.

Доказ. Доказ је дат у [41].

Последњи корак у конструисању целобројних $SB_{h/b}EC-SB_{l/b}ED$ кодова јесте проналажење коефицијената који задовољавају услове из теореме 6. У табели 5.3 су приказани коефицијенти за 32-битне кодове.

Табела 5.3. Коефицијенти за неке целобројне $SB_{3/32}EC-SB_{l/32}ED$ кодове.

$l = 10$															
1025	1027	1029	1031	1033	1035	1037	1039	1041	1043	1045	1047	1049	1051	1053	1055
1057	1059	1061	1063	1065	1067	1069	1071	1073	1075	1077	1079	1081	1083	1085	1087
$l = 11$															
2049	2051	2053	2055	2057	2059	2061	2063	2065	2067	2069	2071	2073	2075	2077	2079
2081	2083	2085	2087	2089	2091	2093	2095	2097	2099	2101	2103	2105	2107	2109	2111

5.1.4. Целобројни $SB_{h/b}EC-SB_{l/b}ED-DB_{d/b}ED$ кодови

Конструкција целобројних $SB_{h/b}EC-SB_{l/b}ED-DB_{d/b}ED$ кодова је сложенија од целобројних $SB_{h/b}EC-SB_{l/b}ED$ кодова, пошто се морају узети у обзир и двоструке d/b рафалне грешке [41]. Добро је, међутим, то што се целобројна вредност ове врсте грешке може лако одредити. Наиме, пошто се зна целобројна вредност једноструке h/b рафалне грешке, јасно је да двострука рафална d/b грешка мења целобројне вредности два b -битна бајта за $e_i = \pm 2^{r_3} \cdot (2u_3 - 1)$ и $e_j = \pm 2^{r_4} \cdot (2u_4 - 1)$, где је $1 \leq x_3 \leq d$, $0 \leq r_3 \leq b - x_3$, $1 \leq u_3 \leq 2^{x_3-1}$, $1 \leq x_4 \leq d$, $0 \leq r_4 \leq b - x_4$, $1 \leq u_4 \leq 2^{x_4-1}$ и $1 \leq i < j \leq k + 1$. На основу овога, дате су следећа дефиниција и теорема.

Дефиниција 10. Скуп синдрома који одговарају двоструким d/b рафалним грешкама је дефинисан као

$$s_3 = \bigcup_{r_3=0}^{b-x_3} \bigcup_{u_3=1}^{2^{x_3-1}} \bigcup_{x_3=1}^d \bigcup_{i=1}^k \bigcup_{r_4=0}^{b-x_4} \bigcup_{u_4=1}^{2^{x_4-1}} \bigcup_{x_4=1}^d \bigcup_{j=i+1}^{k+1} \left\{ \pm 2^{r_3} \cdot (2u_3 - 1) \cdot C_i \pm 2^{r_4} \cdot (2u_4 - 1) \cdot C_j \pmod{2^b - 1} \right\} \quad (5.20)$$

Теорема 7. Кодови дефинисани са (5.1) могу исправити једноструке h/b рафалне грешке и детектовати једноструке l/b рафалне грешке и двоструке d/b рафалне грешке ако постоји k различитих коефицијената C_i таквих да је

1. $|\xi| = |s_1| = [2^h \cdot (b - h + 2) - 2] \cdot (k + 1)$,
2. $s_1 \cap s_2 = \emptyset$,
3. $s_1 \cap s_3 = \emptyset$,
4. $s_2, s_3 \subseteq Z_{2^{b-1}} \setminus \{0, 1\}$.

Доказ. Доказ је дат у [41].

Да би се могли конструисати целобројни $SB_{h/b}EC-SB_{l/b}ED-DB_{d/b}ED$ кодови потребно је пронаћи коефицијенте који задовољавају услове теореме 7. Као и у случају целобројних $SB_{h/b}EC-SB_{l/b}EC$ кодова овај задатак је обављен уз употребу рачунара. Фокус је био стављен на 32-битне кодове и неке практичне вредности параметара h, l, d и b (табела 5.4).

Табела 5.4. Коефицијенти за неке целобројне $SB_{3/32}EC-SB_{l/32}ED-DB_{4/32}ED$ кодове.

$l = 10$															
1041	1079	1133	1839	1973	2317	3799	5877	6121	6295	6699	10387	12017	14257	18567	20953
21867	22963	23635	24421	26917	34123	35547	37379	38773	40903	47621	53929	54749	57575	57911	58197
$l = 11$															
2065	2103	2149	2227	2319	2491	2791	3809	6563	6807	8775	10091	10669	11987	13777	13847
18523	18655	21433	28309	29219	30437	33123	35381	39503	41617	42363	44023	44381	45221	46097	46187

5.1.5. Целобројни $SB_{l/b}EC-DED$ кодови

Први корак у конструкцији целобројних кодова који могу га коригују једноструке l/b рафалне грешке и да детектују двоструке битске грешке јесте одређивање целобројних вредности поменутих грешака [42]. Када је у питању прва грешка, њена целобројна вредност је позната из [39]. С друге стране, зна се да двоструке битске грешке могу да оштете један или два бајта. Ако је оштећен један бајт, његова целобројна вредност биће промењена за $e_i = \pm 2^r \pm 2^s$, где је $0 \leq r < s \leq b-1$ и $1 \leq i \leq k+1$. Насупрот томе, ако су два бајта оштећена, њихове целобројне вредности ће бити промењене за $e_i = e_j = \pm 2^t$, где је $0 \leq t \leq b-1$ и $1 \leq i < j \leq k+1$. На основу овога, дате су следеће дефиниције и теорема.

Дефиниција 11. Скуп синдрома који одговара једноструким l/b рафалним грешкама је дефинисан као

$$s_2 = \bigcup_{r=0}^{b-l} \bigcup_{m=1}^{2^{t-1}} \bigcup_{t=1}^l \bigcup_{i=1}^{k+1} \{ \pm 2^r \cdot (2m-1) \cdot C_i \pmod{2^b-1} \} \quad (5.21)$$

Дефиниција 12. Скуп синдрома који одговара двоструким битским грешкама је дефинисан као

$$s_2 = d_1 \cup d_2 \quad (5.22)$$

где је

$$d_1 = \bigcup_{r=0}^{b-2} \bigcup_{s=r+1}^{b-1} \bigcup_{i=1}^{k+1} \{ (\pm 2^r \pm 2^s) \cdot C_i \pmod{2^b-1} \} \quad (5.23)$$

$$d_2 = \bigcup_{r=0}^{b-1} \bigcup_{s=0}^{b-1} \bigcup_{i=1}^k \bigcup_{j=i+1}^{k+1} \{ \pm 2^r \cdot C_i \pm 2^s \cdot C_j \pmod{2^b-1} \} \quad (5.24)$$

Теорема 8. Кардиналност скупа s_1 је

$$|s_1| = [2^l \cdot (b-l+2) - 2] \cdot (k+1)$$

Доказ. Доказ је дат у [42].

Теорема 9. Кодови дефинисани са (5.1) могу исправити једноструке l/b рафалне грешке и детектовати двоструке битске грешке ако постоји k различитих коефицијената C_i таквих да је

$$1. |\xi| = |s_1| = [2^l \cdot (b - l + 2) - 2] \cdot (k + 1)$$

$$2. s_1 \cap s_2 = \emptyset$$

$$3. s_2 \subseteq Z_{2^{b-1}} \setminus \{0, 1\}$$

Доказ. Доказ је дат у [42].

Да би се могли пронаћи коефицијенти који задовољавају услове из теореме 9 било је потребно написати одговарајући рачунарски програм. Помоћу њега пронађени су C_i који омогућавају конструкцију неких практичних кодова (табела 5.5).

Табела 5.5. Коефицијенти за неке целобројне $SB_{l/32}EC-DED$ кодове.

$l = 8$									
1283	2309	2839	3449	5419	5969	6607	7261	7923	8777
9575	11099	11637	13473	14071	14829	15459	16751	18033	19667
20921	22983	24397	25035	25641	28431	30235	31333	32167	34173
35615	36533	37953	38851	41483	42265	45135	46227	50391	52421
54829	55515	59823	60733	63923	65941	68813	69855	72567	74427
76957	81379	91143	94657	96511	97965	101993	104069	106051	113393
115475	131481	135639	140923						
$l = 9$									
2563	4613	5655	6745	10795	11853	12975	14193	15251	17013
19079	22107	23137	24287	27493	28919	31049	32203	33453	34639
35729	37443	39189	40423	41881	46877	49089	50135	52843	55749
58531	59577	62651	63793	68639	70057	72317	76885	79599	82573
85543	88969	91685	93203	95871	99099	101469	103287	108715	110829
114119	118003	119825	123069	124719	129737	135425	142069	148579	150807
153193	154975	160557	169483						

5.1.6. Целобројни $R_{l/b}EC$ кодови

Да би се конструисали целобројни кодови који могу да коригују до t битских грешака унутар једног b -битног бајта потребно је одредити целобројну вредност поменуте грешке. У [43] је анализиран случај $t \leq 2$, одакле се лако закључује да је $e_i \in \{\pm 2^r, \pm 2^s \pm 2^t: 0 \leq r \leq b-1, 0 \leq s < t \leq b-1\}$. Због тога су дате следећа дефиниција и теорема.

Дефиниција 13. Скуп синдрома који одговара једноструким и двоструким битским грешкама унутар једног b -битног бајта је дефинисан као

$$\xi = s_1 \cup s_2 \quad (5.25)$$

где је

$$s_1 = \bigcup_{r=0}^{b-1} \bigcup_{i=1}^{k+1} \{ \pm 2^r \cdot C_i \pmod{2^b-1} \} \quad (5.26)$$

$$s_2 = \bigcup_{r=0}^{b-2} \bigcup_{s=r+1}^{b-1} \bigcup_{i=1}^{k+1} \{ (\pm 2^r \pm 2^s) \cdot C_i \pmod{2^b-1} \} \quad (5.27)$$

Теорема 10. Кодови дефинисани са (5.1) могу исправити једноструке и двоструке битске грешке унутар једног b -битног бајта ако постоји k различитих коефицијената C_i таквих да је

$$|\xi| = |s_2| = [2 \cdot (b-1)^2 - 2] \cdot (k+1).$$

Доказ. Доказ је дат у [43].

За илустрацију применљивости теореме 10, извршена је компјутерска претрага чији је циљ био да се пронађу коефицијенти C_i за 32-битне кодове (табела 5.6). Овакво ограничење, у погледу претраге, је уведено због чињенице да су 32-битни кодови савршено прикладни за имплементацију на модерним процесорима.

Табела 5.6. Коефицијенти за неке целобројне $R_{3/32}EC$ кодове.

19	23	25	27	29	37	39	41	47	49	53	59	61	67	71	77
79	83	89	97	101	103	107	109	113	121	131	137	139	149	151	157
163	167	173	179	181	191	193	197	199	211	223	227	229	233	239	251
263	269	271	277	281	283	289	293	307	311	313	317	331	337	347	349
353	357	359	361	365	367	373	379	383	389	397	401	409	419	421	431
433	437	439	443	449	457	461	463	465	467	475	479	487	491	499	503
521	523	529	541	547	551	557	563	569	571	575	577	587	593	599	601
607	613	617	619	621	625	631	641	643	647	653	659	661	667	673	675

5.1.7. Целобројни $D_{1/b}EC$ кодови

Да би се конструисали целобројни кодови који могу да коригују једноструке битске грешке унутар два b -битна бајта (двострука $1/b$ битска грешка) потребно је одредити целобројну вредност поменуте грешке [44]. С обзиром да је целобројна вредност једноструке битске грешке једнака $e_i = \pm 2^r$, где је $0 \leq r \leq b-1$ и $1 \leq i \leq k+1$, јасно је да ће двострука $1/b$ битска грешка променити целобројне вредности два b -битна бајта за $e_i = e_j = \pm 2^r$, где је $0 \leq r \leq b-1$ и $1 \leq i < j \leq k+1$. На основу тога, дате су следеће дефиниције и теореме.

Дефиниција 14. Скуп синдрома који одговара једноструким битским грешкама је дефинисан као

$$s_1 = \bigcup_{r=0}^{b-1} \bigcup_{i=1}^{k+1} \{ \pm 2^r \cdot C_i \pmod{2^b-1} \} \quad (5.28)$$

Дефиниција 15. Скуп синдрома који одговара двоструким $1/b$ битским грешкама је дефинисан као

$$s_2 = \bigcup_{r=0}^{b-1} \bigcup_{s=0}^{b-1} \bigcup_{i=1}^k \bigcup_{j=i+1}^{k+1} \{ \pm 2^r \cdot C_i \pm 2^s \cdot C_j \pmod{2^b-1} \} \quad (5.29)$$

Теорема 11. Кодови дефинисани са (5.1) могу исправити једноструке битске грешке и двоструке $1/b$ битске грешке ако постоји k различитих коефицијената C_i таквих да је

$$1. |s_1| = 2 \cdot b \cdot (k+1)$$

$$2. |s_2| = 2 \cdot b^2 \cdot k \cdot (k+1)$$

$$3. s_1 \cap s_2 = \emptyset$$

Доказ. Доказ је дат у [44].

Теорема 12. Нека је $\zeta = s_1 \cup s_2$ скуп грешака које могу исправити целобројни $D_{1/b}EC$ кодови. У том случају, важи да је

$$|\zeta| = |s_1| + |s_2| = 2 \cdot b \cdot (b \cdot k + 1) \cdot (k+1).$$

Доказ. Доказ је дат у [44].

Последњи корак у конструкцији целобројних $D_{1/b}EC$ кодова јесте проналажење коефицијената који задовољавају услове теореме 11. У табели 5.7 приказани су делимични резултати за 32-битне кодове [44].

Табела 5.7. Коефицијенти за неке целобројне $D_{1/32}EC$ кодове.

19	213	377	667	1905	3927	4387	6251
8885	9603	11453	14335	14707	22503	25869	29893
31985	36665	43669	67325	69505	69705	81097	86685
95069	98609	103547	122631	132627	159785	195623	210897

5.1.8. Целобројни $DEC-(TAEC)_b$ кодови

Први корак у конструкцији целобројних кодова који могу га коригују једноструке и двоструке битске грешке, као и троструке суседне битске унутар једног b -битна бајта (ТСБ_б грешке) јесте одређивање целобројних вредности поменутих грешака. Анализа из [45] је показала да ове вредности припадају скупу $e_i \in \{\pm 2^r, \pm 2^s \pm 2^t, (\pm 2^2 \pm 2^1 \pm 2^0) \cdot 2^m : 0 \leq r \leq b-1, 0 \leq s < t \leq b-1, 0 \leq m \leq b-3\}$. Сходно томе, дате су следеће дефиниције и теорема.

Дефиниција 16. Скуп синдрома који одговара једноструким битским грешкама је дефинисан као

$$s_1 = \bigcup_{r=0}^{b-1} \bigcup_{i=1}^{k+1} \{ \pm 2^r \cdot C_i \pmod{2^b-1} \} \quad (5.30)$$

Дефиниција 17. Скуп синдрома који одговара двоструким битским грешкама је дефинисан као

$$s_2 = d_1 \cup d_2 \quad (5.31)$$

где је

$$d_1 = \bigcup_{r=0}^{b-2} \bigcup_{s=r+1}^{b-1} \bigcup_{i=1}^{k+1} \{ (\pm 2^r \pm 2^s) \cdot C_i \pmod{2^b-1} \} \quad (5.32)$$

$$d_2 = \bigcup_{r=0}^{b-1} \bigcup_{s=0}^{b-1} \bigcup_{i=1}^k \bigcup_{j=i+1}^{k+1} \{ \pm 2^r \cdot C_i \pm 2^s \cdot C_j \pmod{2^b-1} \} \quad (5.33)$$

Дефиниција 18. Скуп синдрома који одговара ТСБ_б грешкама је дефинисан као

$$s_3 = \bigcup_{i=1}^{k+1} \bigcup_{m=0}^{b-3} \{ (\pm 2^2 \pm 2^1 \pm 2^0) \cdot 2^m \cdot C_i \pmod{2^b-1} \} \quad (5.34)$$

У [45] је показано да је s_3 подскуп од d_1 , па је стога дата следећа теорема.

Теорема 13. Кодови дефинисани са (5.1) могу исправити једноструке битске грешке, двоструке битске грешке и ТСБ_б грешке ако постоји k различитих коефицијената C_i таквих да је

1. $|d_1| = [2 \cdot (b-1)^2 - 2] \cdot (k+1)$
2. $|d_2| = 2 \cdot b^2 \cdot k \cdot (k+1)$
3. $d_1 \cap d_2 = \emptyset$

Доказ. Доказ је дат у [45].

Теорема 14. Нека је $\xi = d_1 \cup d_2$ скуп грешака које могу исправити целобројни DEC-(ТАЕС)_б кодови. У том случају, важи да је

$$|\xi| = |d_1| + |d_2| = 2 \cdot [b \cdot (k+1) - 1]^2 - 2.$$

Доказ. Доказ је дат у [45].

За потребе илустрације применљивости горње теорије, извршена је исцрпна компјутерска претрага. Циљ је био да се нађу коефицијенти C_i помоћу којих је могуће конструисати неке практичне кодове (табела 5.8).

Табела 5.8. Коефицијенти за неке целобројне DEC-(ТАЕС)₃₂ кодове.

45	201	477	1109	1319	3129	3453	4847
9581	10117	11837	15411	17897	18439	23781	29749
34757	36419	44865	46009	51889	68223	81619	93047
108053	112279	113181	117189	164183	167119	169211	196783
201311	209395	256657	264427	275489	282429	310935	354225
386703	428269	432035	446911	514953	599285	634607	690403
748103	774457	834335	892067	893141	1013237	1067671	1087365
1103047	1122449	1191261	1248189	1297563	1342681	1458509	1570785
1701685	1789337	1904439	2007495	2143923	2240111	2300025	2332779
2346995	2583965	2597467	3176613	3200875	3218123	3333741	3677993
3759663	4008735	4245743	4301929	4539051	4637371	5001505	5168905
5659385	6365959	7104157	7276563	8858289	8994505	9282467	9856101

5.1.9. Целобројни SB_{l/b}ЕС-DB_{h/b}ЕС кодови

Да би се конструисали целобројни кодови који могу да коригују једноструке l/b рафалне грешке и двоструке h/b рафалне грешке, потребно је одредити целобројне вредности поменутих грешака. У [46] је показано да је целобројна вредност једноструке l/b рафалне грешке једнака $e_i = \pm 2^r \cdot (2u - 1)$, где је $0 \leq r \leq b - x$, $1 \leq u \leq 2^{x-1}$, $1 \leq x \leq l$ и $1 \leq i \leq k + 1$. Самим тим, лако је било закључити да ће двострука h/b рафална грешка променити целобројне вредности два b -битна бајта за $e_i = \pm 2^s \cdot (2v - 1)$ и $e_j = \pm 2^t \cdot (2w - 1)$, где је $0 \leq s \leq b - y$, $1 \leq v \leq 2^{y-1}$, $1 \leq y \leq h$, $0 \leq t \leq b - z$, $1 \leq w \leq 2^{z-1}$, $1 \leq z \leq h$ и $1 \leq i < j \leq k + 1$. На основу ових чињеница, дате су следеће дефиниције и теореме.

Дефиниција 19. Скуп синдрома који одговара једноструким l/b рафалним грешкама је дефинисан као

$$s_1 = \bigcup_{r=0}^{b-l} \bigcup_{m=1}^{2^{v-1}} \bigcup_{t=1}^l \bigcup_{i=1}^{k+1} \{ \pm 2^r \cdot (2m-1) \cdot C_i \pmod{2^b-1} \} \quad (5.35)$$

Дефиниција 20. Скуп синдрома који одговара двоструким h/b рафалним грешкама је дефинисан као

$$s_2 = \bigcup_{s=0}^{b-y} \bigcup_{v=1}^{2^{v-1}} \bigcup_{y=1}^h \bigcup_{i_1=1}^k \bigcup_{t=0}^{b-z} \bigcup_{w=1}^{2^{z-1}} \bigcup_{z=1}^h \bigcup_{i=j+1}^{k+1} \{ \pm 2^s \cdot (2v-1) \cdot C_i \pm 2^t \cdot (2w-1) \cdot C_j \pmod{2^b-1} \} \quad (5.36)$$

Теорема 15. Кодови дефинисани са (5.1) могу да исправе једноструке l/b рафалне грешке и двоструке h/b рафалне грешке ако постоји k различитих коефицијената C_i таквих да је

1. $|s_1| = [2^l \cdot (b-l+2) - 2] \cdot (k+1)$
2. $|s_2| = [2^h \cdot (b-h+2) - 2]^2 \cdot (k+1) \cdot (k/2)$
3. $s_1 \cap s_2 = \emptyset$

Доказ. Доказ је дат у [46].

Теорема 16. Нека је $\xi = s_1 \cup s_2$ скуп грешака које могу исправити целобројни $SB_{l/b}EC$ - $DB_{h/b}EC$ кодови. У том случају, важи да је

$$|\xi| = |s_1| + |s_2| = (k+1) \cdot \{ 2^l \cdot (b-l+2) - 2 + [2^h \cdot (b-h+2) - 2]^2 \cdot (k/2) \}.$$

Доказ. Доказ следи из претходне теореме.

Да би се проверила исправност изнете теорије, извршена је детаљна рачунарска претрага. Главни циљ је био да пронаћи неколико коефицијената за 32-битне кодове (табела 5.9), јер су они најпогоднији за софтверску имплементацију.

Табела 5.9. Коефицијенти за неке целобројне $SB_{l/32}EC$ - $DB_{3/32}EC$ кодове.

$l = 6$							
1099	8483	20863	43115	55239	122093	156689	282249
$l = 7$							
2187	9437	20267	37263	73997	81761	170223	239435
$l = 8$							
4361	9055	18385	40301	49897	93391	182553	204921

5.1.10. Целобројни $SB_{l/b}EC$ - $DB_{l/b}EC$ кодови

Први корак у конструкцији целобројних кодова који могу га коригују једноструке l/b рафалне грешке и двоструке $1/b$ битске грешке јесте одређивање целобројних вредности поменутих грешака. У [47] је показано да је целобројна вредност једноструке l/b рафалне грешке једнака $e_i = \pm 2^r \cdot (2m-1)$, где је $0 \leq r \leq b-l$, $1 \leq m \leq 2^{v-1}$, $1 \leq v \leq l$ и $1 \leq i \leq k+1$. Поред тога, показано је и да двоструке $1/b$ битске грешке мењају целобројне вредности два b -битна бајта за $e_i = e_j = \pm 2^r$, где је $0 \leq r \leq b-1$ и $1 \leq i < j \leq k+1$. Стога су дате следеће дефиниције и теореме.

Дефиниција 21. Скуп синдрома који одговара једноструким l/b рафалним грешкама је дефинисан као

$$s_2 = \bigcup_{r=0}^{b-l} \bigcup_{m=1}^{2^{t-1}} \bigcup_{i=1}^l \bigcup_{k=1}^{k+1} \{ \pm 2^r \cdot (2m-1) \cdot C_i \pmod{2^b-1} \} \quad (5.37)$$

Дефиниција 22. Скуп синдрома који одговара двоструким $1/b$ битским грешкама је дефинисан као

$$s_2 = \bigcup_{r=0}^{b-1} \bigcup_{s=0}^{b-1} \bigcup_{i=1}^k \bigcup_{j=i+1}^{k+1} \{ \pm 2^r \cdot C_i \pm 2^s \cdot C_j \pmod{2^b-1} \} \quad (5.38)$$

Теорема 17. Кодови дефинисани са (5.1) могу да исправе једноструке l/b рафалне грешке и двоструке h/b $1/b$ битске грешке ако постоји k различитих коефицијената C_i таквих да је

1. $|s_1| = [2^l \cdot (b-l+2) - 2] \cdot (k+1)$
2. $|s_2| = 2 \cdot b^2 \cdot k \cdot (k+1)$
3. $s_1 \cap s_2 = \emptyset$

Доказ. Доказ је дат у [47].

Теорема 18. Нека је $\xi = s_1 \cup s_2$ скуп грешака које могу исправити целобројни $SB_{l/b}EC-D_{1/b}EC$ кодови. У том случају, важи да је

$$|\xi| = |s_1| + |s_2| = [2^l \cdot (b-l+2) + 2 \cdot b^2 \cdot k - 2] \cdot (k+1) \quad (5.39)$$

Доказ. Доказ је дат у [47].

Иако је кардиналност скупа ξ унапред позната, његови елементи се не могу одредити без употребе рачунара. Због тога је било потребно написати одговарајући рачунарски програм. Помоћу њега пронађени су коефицијенти за 32-битне кодове (табела 5.10), који су најпогоднији за софтверску имплементацију.

Табела 5.10. Коефицијенти за неке целобројне $SB_{l/32}EC-D_{1/32}EC$ кодове.

$l = 9$												
2563	5653	6711	8729	11823	13957	15051	18089	19235	21101	24263	25441	28467
30709	33723	37021	39183	42929	46631	49189	51961	55179	73377	82751	87689	100591
112429	117803	128759	150973	166733	178585							
$l = 10$												
5123	11285	13371	17433	23599	27789	29899	35975	38049	42277	48411	50697	56719
60883	63037	66711	73145	77429	79659	86055	88301	92713	102641	104943	114949	124109
128353	135509	137671	151011	157257	161567							

5.1.11. Целобројни DAEC кодови

Да би могао да исправи једноструке битске асиметричне ($1 \rightarrow 0$) грешке и двоструке битске асиметричне ($1 \rightarrow 0$) грешке, декодер мора знати које вредности синдрома S одговарају овим грешкама. У [48] је показано да целобројне вредности поменутих грешака припадају скупу $e_i \in \{-2^r, -2^s - 2^t; 0 \leq r \leq b-1, 0 \leq s < t \leq b-1\}$. Због тога су дате следеће дефиниције и теорема.

Дефиниција 23. Скуп синдрома који одговара једноструким битским асиметричним грешкама је дефинисан као

$$\varepsilon_1 = s_1 \cup s_2 \quad (5.40)$$

где је

$$s_1 = \bigcup_{r=0}^{b-1} \{2^r \pmod{2^b-1}\} \quad (5.41)$$

$$s_2 = \bigcup_{r=0}^{b-1} \bigcup_{i=1}^k \{-2^r \cdot C_i \pmod{2^b-1}\} \quad (5.42)$$

Дефиниција 24. Скуп синдрома који одговара двоструким битским асиметричним грешкама је дефинисан као

$$\varepsilon_2 = d_1 \cup d_2 \cup d_3 \cup d_4 \quad (5.43)$$

где је

$$d_1 = \bigcup_{r=0}^{b-2} \bigcup_{s=r+1}^{b-1} \{2^r + 2^s \pmod{2^b-1}\} \quad (5.44)$$

$$d_2 = \bigcup_{r=0}^{b-2} \bigcup_{s=r+1}^{b-1} \bigcup_{i=1}^k \{(-2^r - 2^s) \cdot C_i \pmod{2^b-1}\} \quad (5.45)$$

$$d_3 = \bigcup_{r=0}^{b-1} \bigcup_{s=0}^{b-1} \bigcup_{i=1}^k \{-2^r \cdot C_i + 2^s \pmod{2^b-1}\} \quad (5.46)$$

$$d_4 = \bigcup_{r=0}^{b-1} \bigcup_{s=0}^{b-1} \bigcup_{i=1}^k \bigcup_{j=i+1}^{k+1} \{-2^r \cdot C_i - 2^s \cdot C_j \pmod{2^b-1}\} \quad (5.47)$$

Теорема 19. Кодови дефинисани са (5.1) могу да исправе једноструке битске асиметричне грешке и двоструке битске асиметричне грешке ако постоји k различитих коефицијената C_i таквих да је

1. $|s_1| = b$
2. $|s_2| = b \cdot k$
3. $|d_1| = (b/2) \cdot (b-1)$
4. $|d_2| = (b/2) \cdot (b-1) \cdot k$
5. $|d_3| = b^2 \cdot k$
6. $|d_4| = (b^2/2) \cdot (k-1) \cdot k$
7. $s_1 \cap s_2 \cap d_1 \cap d_2 \cap d_3 \cap d_4 = \emptyset$

Доказ. Доказ је дат у [48].

Теорема 20. Нека је $\xi = \varepsilon_1 \cup \varepsilon_2$ скуп грешака које могу исправити целобројни ДАЕС кодови. У том случају, важи да је

$$|\xi| = \frac{b}{2} \cdot (k+1) + \frac{b^2}{2} \cdot (k+1)^2$$

Доказ. Доказ је дат у [48].

За потребе илустрације применљивости теореме 19, у табели 5.11 приказани су резултати рачунарске претраге за кодове са параметрима $b \leq 32$ и $k \leq 64$.

Табела 5.11. Коефицијенти за неке целобројне DAEC кодове.

$b = 24$					
19	93	197	485	1111	1771
2247	2625	3949	5337	7011	7931
9591	13231	15711	21675	24077	27837
30461	34073	43167	45451	47377	72245
104847	148427	182693	211017		
$b = 32$					
19	93	197	485	1111	1771
2247	2625	3923	4721	6459	7463
10941	13277	14329	15263	20183	21329
26267	31609	36597	40453	44993	47465
63449	65371	82467	86767	92563	95947
105581	108297	110849	131365	138807	150609
152801	177615	197791	203213	229677	246945
263135	278391	303707	305785	332895	341235
352515	388459	428269	435293	457423	478493
490611	504469	545203	571633	590825	595143
674535	715827	740695	742475		

5.1.12. Целобројни $SB_{l/b}AEC$ - $R_{l/b}AEC$ кодови

Да би могао да исправи једноструке l/b рафалне асиметричне ($1 \rightarrow 0$) грешке и t битских асиметричних ($1 \rightarrow 0$) грешака унутар једног b -битног бајта $[(BA\Gamma)_b$ грешке], декодер мора да зна које вредности синдрома S одговарају овим грешкама. С обзиром да неке $(BA\Gamma)_b$ грешке генеришу исте синдрома као и једноструке l/b рафалне асиметричне грешке, у обзир су узете само $(BA\Gamma)_b$ грешке код којих растојање између оштећених бита износи d ($l < d < b$) [дуге $(BA\Gamma)_b$ грешке]. Стога су у [49] дате следеће дефиниције и теореме.

Дефиниција 25. Нека су $e_1 = \{1\}$ и $e_w = \{2^{w-1} + 1, 2^{w-1} + 3, \dots, 2^w - 1\}$ скупови непарних целих бројева и нека је $2 \leq w \leq l < b$. Тада је скуп синдрома који одговара једноструким l/b рафалним асиметричним грешкама дефинисан као

$$\varepsilon_1 = s_1 \cup s_2 \quad (5.48)$$

где је

$$s_1 = \bigcup_{m=1}^l \bigcup_{r=0}^{b-m} \{2^r \cdot e_m \pmod{2^b-1}\} \quad (5.49)$$

$$s_2 = \bigcup_{m=1}^l \bigcup_{r=0}^{b-m} \bigcup_{i=1}^k \{-2^r \cdot e_m \cdot C_i \pmod{2^b-1}\} \quad (5.50)$$

Дефиниција 27. Нека су $f_2 = \{2^s + 2^z\}$ и $f_v = \{2^s + 2^{x_1} + 2^{x_2} + \dots + 2^{x_{v-2}} + 2^z\}$ скупови целих бројева, где је $3 \leq v < l < b$, $0 \leq s < x_1 < \dots < x_{v-2} < z$ и $s + l \leq z \leq b - 1$. Тада је скуп синдрома који одговара дугим $(BA\Gamma)_b$ грешкама дефинисан као

$$\varepsilon_2 = s_3 \cup s_4 \quad (5.51)$$

где је

$$s_3 = \bigcup_{n=2}^t \bigcup_{t=2}^l \{ f_n \pmod{2^b-1} \} \quad (5.52)$$

$$s_4 = \bigcup_{n=2}^t \bigcup_{t=2}^l \bigcup_{i=1}^k \{ -f_n \cdot C_i \pmod{2^b-1} \} \quad (5.53)$$

Теорема 20. Кодови дефинисани са (5.1) могу да исправе једноструке l/b рафалне асиметричне грешке и $(\text{БАГ})_b$ грешке ако постоји k различитих коефицијената C_i таквих да је

1. $|\varepsilon_1| = (k+1) \cdot [2^{l-1} \cdot (b-l+2) - 1]$
2. $|\varepsilon_2| = (k+1) \cdot \sum_{i=0}^{b-l-1} \sum_{j=2}^t (b-l-i) \cdot \binom{l+i-1}{j-2}$
3. $\varepsilon_1 \cap \varepsilon_2 = \emptyset$

Доказ. Доказ је дат у [49].

Теорема 21. Нека је $t = 2$ и нека је $\xi = \varepsilon_1 \cup \varepsilon_2$ скуп грешака које могу исправити целобројни $\text{SB}_{l/b}\text{AEC-R}_{t/b}\text{AEC}$ кодови. У том случају, важи да је

$$|\xi| = |\varepsilon_1| + |\varepsilon_2| = \left[2^{l-1} \cdot (b-l+2) - 1 + \frac{(b-l)^2 + b-l}{2} \right] \cdot (k+1)$$

Доказ. Доказ је дат у [49].

За потребе илустрације применљивости теореме 20, у табели 5.12 приказани су резултати рачунарске претраге за кодове са параметрима $b = 32$, $t = 2$ и $8 \leq l \leq 9$.

Табела 5.12. Коефицијенти за неке целобројне $\text{SB}_{l/32}\text{AEC-R}_{2/32}\text{AEC}$ кодове.

$l = 8$															
2	259	261	263	265	269	271	277	281	283	289	293	299	307	311	313
317	337	341	347	349	353	359	361	367	373	379	383	389	397	401	409
419	421	431	433	439	443	449	457	461	463	467	479	487	491	499	503
509	521	523	541	547	557	563	569	571	577	587	593	599	601	607	613
617	619	631	641	643	647	653	659	661	673	677	691	701	709	719	727
733	739	743	751	757	761	773	787	797	809	811	821	823	827	829	839
853	857	859	863	877	881	883	887	907	911	919	929	937	941	947	953
967	971	977	983	991	997	1009	1013	1019	1021	1039	1049	1051	1061	1063	1069
$l = 9$															
2	515	517	519	521	523	527	529	533	541	547	551	553	557	563	569
571	577	587	593	599	601	607	613	617	619	631	643	647	653	659	661
673	677	691	701	709	719	727	733	739	743	751	757	761	769	773	787
797	809	811	821	823	827	829	839	853	857	859	863	877	881	883	887
907	911	919	929	937	941	947	953	967	971	977	983	991	1009	1013	1019
1021	1031	1039	1049	1051	1061	1063	1069	1087	1091	1093	1097	1103	1109	1117	1123
1129	1151	1163	1171	1181	1187	1193	1201	1213	1217	1223	1229	1231	1237	1249	1259
1277	1279	1289	1291	1297	1301	1303	1307	1319	1321	1327	1361	1367	1369	1373	1381

5.1.13. Целобројни $\text{SB}_{l/b}\text{AEC-DAEC}$ кодови

Да би могао да исправи једноструке l/b рафалне асиметричне $(1 \rightarrow 0)$ грешке и двоструке битске асиметричне $(1 \rightarrow 0)$ грешке, декодер мора да зна које вредности синдрома S одговарају овим грешкама. У [36] је показано да целобројне вредности поменутих грешака припадају скупу $e_i \in \{-2^r \cdot (2m-1), -2^s - 2^t; 0 \leq r \leq b -$

$l, 1 \leq m \leq 2^{v-1}, 1 \leq v \leq l, 0 \leq r \leq b-1, 0 \leq s < t \leq b-1\}$. Стога су у [50] дате следеће дефиниције и теореме.

Дефиниција 28. Скуп синдрома који одговара једноструким l/b рафалним асиметричним грешкама је дефинисан као

$$\varepsilon_1 = s_1 \cup s_2 \quad (5.54)$$

где је

$$s_1 = \bigcup_{r=0}^{b-l} \bigcup_{m=1}^{2^{v-1}} \bigcup_{v=1}^l \bigcup_{i=1}^k \{2^r \cdot (2m-1) \cdot C_i \pmod{2^b-1}\} \quad (5.55)$$

$$s_2 = \bigcup_{r=0}^{b-l} \bigcup_{m=1}^{2^{v-1}} \bigcup_{v=1}^l \{-2^r \cdot (2m-1) \pmod{2^b-1}\} \quad (5.56)$$

Дефиниција 29. Скуп синдрома који одговара двоструким битским асиметричним грешкама које оштећују два b -битна бајта је дефинисан као

$$\varepsilon_2 = s_3 \cup s_4 \quad (5.57)$$

где је

$$s_3 = \bigcup_{r=0}^{b-1} \bigcup_{s=0}^{b-1} \bigcup_{i=1}^k \bigcup_{j=i+1}^{k+1} \{-2^r \cdot C_i - 2^s \cdot C_j \pmod{2^b-1}\} \quad (5.58)$$

$$s_4 = \bigcup_{r=0}^{b-1} \bigcup_{s=0}^{b-1} \bigcup_{i=1}^k \{-2^r \cdot C_i + 2^s \pmod{2^b-1}\} \quad (5.59)$$

Дефиниција 30. Скуп синдрома који одговара двоструким битским асиметричним грешкама које оштећују један b -битни бајт (искључујући једноструке l/b рафалне асиметричне грешке) је дефинисан као

$$\varepsilon_3 = s_5 \cup s_6 \quad (5.60)$$

где је

$$s_5 = \bigcup_{i=1}^k \{-2^r \cdot (2^{s-r} + 1) \cdot C_i \pmod{2^b-1} : l \leq r+l \leq s \leq b-1\} \quad (5.61)$$

$$s_6 = \{2^r \cdot (2^{s-r} + 1) \pmod{2^b-1} : l \leq r+l \leq s \leq b-1\} \quad (5.62)$$

Теорема 22. Кодови дефинисани са (5.1) могу да исправе једноструке l/b рафалне асиметричне грешке и двоструке битске асиметричне грешке ако постоји k различитих коефицијената C_i таквих да је

$$1. |\varepsilon_1| = [2^{l-1} \cdot (b-l+2) - 1] \cdot (k+1)$$

$$2. |\varepsilon_2| = \frac{b^2 \cdot k}{2} \cdot (k+1)$$

$$3. |\varepsilon_3| = \frac{(b-l+1) \cdot (b-l)}{2} \cdot (k+1)$$

$$4. \varepsilon_1 \cap \varepsilon_2 \cap \varepsilon_3 = \emptyset$$

Доказ. Доказ је дат у [50].

Теорема 23. Нека је $\xi = \varepsilon_1 \cup \varepsilon_2 \cup \varepsilon_3$ скуп грешака које могу исправити целобројни $SB_{l/b}AEC$ - $DAEC$ кодови. У том случају, важи да је

$$|\xi| = |\varepsilon_1| + |\varepsilon_2| + |\varepsilon_3| = \left\lfloor \frac{2^l \cdot (b-l+2) + b^2 \cdot k + (b-l+1) \cdot (b-l) - 2}{2} \right\rfloor \cdot (k+1).$$

Доказ. Доказ је дат у [50].

За потребе илустрације применљивости теореме 22, у табели 5.13 приказани су резултати рачунарске претраге за кодове са параметрима $b = 32$ и $8 \leq l \leq 9$.

Табела 5.13. Неки коефицијенти за целобројне $B_{l/32}AEC$ - $DAEC$ кодове.

$l = 8$							
515	533	553	603	719	1153	1263	1317
4159	4747	5811	7557	9121	13679	18557	19741
23951	30511	31223	44615	45017	49263	52075	54421
56299	69621	80371	102001	105277	112425	114387	144093
$l = 9$							
1027	1037	1081	1167	1217	1385	1483	2213
2551	3339	5295	7411	10997	11365	13233	16795
18617	26351	27609	30417	39257	43611	46701	55825
64389	77159	89799	99699	104851	107481	115583	122463

5.2. Целобројни SEC- $DAEC$ кодови

Иако наведени целобројни кодови могу да исправе различите врсте грешака, ниједан од њих не може да реши проблем губитака пакета које шаљу Интернет камере. Разлог за то не лежи само у врстама грешака које се јављају током преноса података (слика 4.7), већ и због тога што ниједан поменути целобројни код не може ефикасно да замени Интернет чексуму, која је "главни кривац" за одбацивање оштећених пакета. Због тога је било потребно дизајнирати нову класу целобројних кодова која би могла овај задатак успешно да реши.

5.2.1. Мотивација

Традиционалне Интернет апликације, као што су пренос датотека или и-мејл, захтевају поуздану испоруку података. Због тога оне, на транспортном слоју (ТС), користе ТСР протокол. Главна особина овог протокола, као што је већ напоменуто у четвртом поглављу, јесте да осигура да сви пакети података буду примљени без грешке. Уколико се током преноса неки пакет оштети, предајник ће га изнова и изнова слати све док пријемник не потврди да га је примио без грешке.

За разлику од наведених апликација, мултимедијалне, као што су глас и видео, користе другачију филозофију. Тачније, оне радије прихватају оштећене пакете него да чекају на њихово поновно слање. Разлог је тај што мултимедијални алгоритми обезбеђују одређени степен отпорности на грешке, тако да пријемник има користи од оштећених пакета (с друге стране, поново послати пакет су бескорисни

за мултимедијалне апликације). Упркос томе, UDP протокол, као најчешће коришћени мултимедијални ТС протокол, не прослеђује ниједан оштећени пакет. Штавише, његова улога се, као што је описано у четвртом поглављу, своди на детектовање грешака у преносу и одбацивање свих оштећених пакета. Такво понашање је последица чињенице да су подаци заштићени Интернет чексумом, која може неке грешке да открије, али не и да их исправи.

С обзиром да се Интернет чексума не може заменити ниједним линеарним кодом (харверска замена би захтевала модификовање милијарди уређаја, док би софтверска замена драстично смањила пропусни опсег мреже), неколико истраживача је предложило да се UDP пакет "подели" на осетљиви и неосетљиви део, при чему је само осетљиви део заштићен Интернет чексумом [51], [52]. Експерименти који су уследили [53], [54] показали су да се оваквим приступом значајно смањује број одбачених пакета, што је мотивисало IETF да га имплементира у нови ТС протокол који је назван UDP-Lite [55]. Стандардизација овог протокола је, међутим, отворила нови проблем, а то је некомпатибилност са UDP/IP стеком (поље "дужина" означава дужину осетљивог дела, а не величину UDP пакета).

Имајући ово у виду, као и чињеницу да 87% до 99% свих грешака које настају у преносу чине једноструке битске грешке и двоструке суседне битске грешке (слика 4.7), у [56] је представљено решење које значајно ублажава проблем одбацивања пакета. Суштина предложене идеје јесте да се Интернет чексума замени са 16-битним целобројним кодовима који могу да исправе једноструке битске грешке и двоструке суседне битске грешке (целобројни SEC-DAEC кодови). Употреба таквих кодова не само би омогућила пријемнику да поправи велику већину оштећених пакета, већ он не би морао да претрпи било какве хардверске промене, пошто целобројни SEC-DAEC кодови представљају генерализацију Интернет чексуме.

5.2.2. Конструкција

У литератури која се бави заштитним кодовањем може се наћи много линеарних кодова који могу да исправе једноструке битске грешке и двоструке суседне битске грешке [57]-[59]. Заједничка карактеристика за све ове кодове је да користе операције над пољем Галоа, које не подржавају процесори опште намене (ПОН). Због овог недостатка, софтверски базирани декодери су спори и изузетно сложени (ПОН-ови морају да емулирају операције над пољем Галоа, а самим тим да користе неколико десетина/стотина операција по једном биту [36], [37]). За разлику од линеарних кодова, предложени користе целобројне операције које подржавају сви ПОН-ови. Самим тим, они се могу ефикасно имплементирати у софтверу.

Међутим, пре него што се то објасни, потребно је описати начин конструкције целобројних SEC-DAEC кодова. Први корак, у том погледу, је исти као и код

осталих целобројних кодова: потребно је дефинисати целобројне вредности грешака које треба исправити. У нашем случају, овај корак је поједностављен, пошто је из [39] познато да је целобројна вредност једноструке битске грешке једнака $e_i = \pm 2^r$, где је $0 \leq r \leq b-2$ и $1 \leq i \leq k+1$. Када су у питању двоструке суседне битске грешке, знамо да оне могу оштетити један или два b -битна бајта. Из дефиниција 1 и 2, се лако може показати да ће, у првом сценарију, целобројна вредност једног b -битног бајта бити промењена за $e_i = \pm 2^{r+1} \pm 2^r$, где је $0 \leq r \leq b-2$ и $1 \leq i \leq k+1$. Из истих дефиниција се, такође, може показати да ће, у другом сценарију, целобројне вредности два b -битна бајта бити промењене за $e_i = \pm 2^0$ и $e_{i+1} = \pm 2^{b-1}$, где је $1 \leq i \leq k$. Имајући ово у виду, у [56] су наведене следеће дефиниције.

Дефиниција 31. Скуп синдрома који генеришу једноструке битске грешке је дефинисан као

$$s_1 = \bigcup_{i=1}^{k+1} \{ \pm 2^r \cdot C_i \pmod{2^b-1} : 0 \leq r \leq b-1 \} \quad (5.63)$$

Дефиниција 32. Скуп синдрома који генеришу двоструке суседне битске грешке је дефинисан као

$$s_2 = d_1 \cup d_2 \cup d_3 \quad (5.64)$$

где је

$$d_1 = \bigcup_{i=1}^{k+1} \{ \pm (2^1 + 2^0) \cdot 2^r \cdot C_i \pmod{2^b-1} : 0 \leq r \leq b-2 \} \quad (5.65)$$

$$d_2 = \bigcup_{i=1}^{k+1} \{ \pm (2^1 - 2^0) \cdot 2^r \cdot C_i \pmod{2^b-1} : 0 \leq r \leq b-2 \} \quad (5.66)$$

$$d_3 = \bigcup_{i=1}^k \{ \pm 2^0 \cdot C_i \pm 2^{b-1} \cdot C_{i+1} \pmod{2^b-1} \} \quad (5.67)$$

Ако упоредимо скупове s_1 и d_2 приметићемо да је $d_2 \subseteq s_1$. Ова чињеница нам омогућава да изнесемо следеће теореме.

Теорема 24. Кодови дефинисани са (5.1) ће моћи да исправе све једноструке битске грешке и све двоструке суседне битске грешке ако постоји k међусобно различитих коефицијената C_i таквих да је

1. $|s_1| = 2 \cdot b \cdot (k+1)$,
2. $|d_1| = 2 \cdot (b-1) \cdot (k+1)$,
3. $|d_3| = 4 \cdot k$,
4. $s_1 \cap d_1 \cap d_3 = \emptyset$,

Доказ. Из [39] знамо да је услов 1 неопходан и довољан услов за исправљање

једноструких грешака. С друге стране, услов 2 наводи да двоструке суседне битске грешке, ограничене на један b -битни бајт, генеришу $2 \cdot (b - 1) \cdot (k + 1)$ ненултих синдрома. Да бисмо ово доказали, поћићемо од чињенице да се скуп d_1 може изразити као

$$d_1 = \bigcup_{i=1}^{k+1} n_i$$

где је

$$n_1 = \{ \pm 3 \cdot 2^r \cdot C_1 \pmod{2^b-1} : 0 \leq r \leq b-2 \}$$

$\vdots$

$$n_k = \{ \pm 3 \cdot 2^r \cdot C_k \pmod{2^b-1} : 0 \leq r \leq b-2 \}$$

$$n_{k+1} = \{ \mp 3 \cdot 2^r \pmod{2^b-1} : 0 \leq r \leq b-2 \}$$

Ако су горњим изразима коефицијенти C_i изабрани тако да је задовољен услов

$$n_1 \cap \dots \cap n_k \cap n_{k+1} = \emptyset$$

$$|n_1| = \dots = |n_k| = |n_{k+1}|$$

скуп d_1 ће имати

$$|d_1| = |n_1| + \dots + |n_k| + |n_{k+1}| = |n_{k+1}| \cdot (k+1) = 2 \cdot (b-1) \cdot (k+1)$$

елемената различитих од нуле. На сличан начин, услов 3 имплицира да двоструке суседне битске грешке, ограничене на два b -битна бајта, генеришу $4 \cdot b$ ненултих синдрома. Да бисмо ово доказали, приметимо да се скуп d_2 може изразити као

$$d_2 = \bigcup_{i=1}^k u_i$$

где је

$$u_1 = \{ \pm 2^0 \cdot C_1 \pm 2^{b-1} \cdot C_2 \pmod{2^b-1} \}$$

$\vdots$

$$u_{k-1} = \{ \pm 2^0 \cdot C_{k-1} \pm 2^{b-1} \cdot C_k \pmod{2^b-1} \}$$

$$u_k = \{ \pm 2^0 \cdot C_k \mp 2^{b-1} \pmod{2^b-1} \}$$

Ако су горњим изразима коефицијенти C_i изабрани тако да је задовољен услов

$$u_1 \cap \dots \cap u_{k-1} \cap u_k = \emptyset$$

$$|u_1| = \dots = |u_{k-1}| = |u_k|$$

скуп d_2 ће имати

$$|d_2| = |u_1| + \dots + |u_{k-1}| + |u_k| = |u_k| \cdot k = 4 \cdot k$$

ненултих елемената. Коначно, услов 4 осигурава да су синдроми које проузрокују

једноструке битске грешке различити од нуле и различити од оних које генеришу двоструке суседне битске грешке. Самим тим, кодови који задовољавају услове од 1 до 4 су $(kb + b, kb)$ целобројни SEC-DAEC кодови.

Теорема 25. Нека је $\xi = s_1 \cup d_1 \cup d_3$ скуп грешака које могу исправити целобројни SEC-DAEC кодови. У том случају, важи да је

$$|\xi| = |s_1| + |d_1| + |d_3| = 4 \cdot b \cdot (k+1) + 2 \cdot (k-1).$$

Доказ. Доказ директно произилази из теореме 24.

Теорема 26. За сваки $(kb + b, kb)$ целобројни SEC-DAEC код важи да

$$k \leq \left\lfloor \frac{2^{b-1} - 2 \cdot b}{2 \cdot b - 1} \right\rfloor.$$

Доказ. Теорема 25 нам говори да скуп ξ има $4 \cdot b \cdot (k+1) + 2 \cdot (k-1)$ ненултих елемената, док Дефиниција 1 указује да је број ненултих синдрома једнак $2^b - 2$. Стога, као резултат, имамо неједнакост

$$4 \cdot b \cdot (k+1) + 2 \cdot (k-1) \leq 2^b - 2$$

одакле следи да је

$$k \leq \left\lfloor \frac{2^{b-1} - 2 \cdot b}{2 \cdot b - 1} \right\rfloor.$$

Последњи корак у конструисању целобројних SEC-DAEC кодова јесте проналазак коефицијената који задовољавају услове теореме 24. Овај задатак се не може извршити без употребе одговарајућег рачунарског програма (Прилог 3). Неки од резултата, добијени његовом употребом, приказани су у табелама 5.14 и 5.15.

5.2.3. Процедура корекције грешака

Процедура корекције грешака за целобројне SEC-DAEC кодове је слична оној из [44], [45]. Укратко, по пријему оштећене кодне речи ($S \neq 0$), декодер ће претражити СТ како би пронашао унос који садржи податке о локацији и вредности грешке (слика 5.4). Након тога, декодер ће извршити операције

Element of $\xi(S)$	Error location (i)	Error value (E_1)	Error location (j)	Error value (E_2)
$\longleftarrow b \longrightarrow$	$\longleftarrow \lceil \log_2(k+1) \rceil \longrightarrow$	$\longleftarrow b \longrightarrow$	$\longleftarrow \lceil \log_2(k+1) \rceil \longrightarrow$	$\longleftarrow b \longrightarrow$

Слика. 5.4. Битска величина једног СТ уноса за целобројне SEC-DAEC кодове.

$$B_i = \underline{B}_i + E_1 \pmod{2^b - 1} \tag{5.68}$$

$$B_j = \underline{B}_j + E_2 \pmod{2^b - 1}, \tag{5.69}$$

где је $E_1 \in \{\mp 2^r \pmod{2^b - 1}: 0 \leq r \leq b-1\} \cup \{\mp 3 \cdot 2^r \pmod{2^b - 1}: 0 \leq r \leq b-2\}$,

$E_2 \in \{\mp 2^{b-1} \pmod{2^b - 1}\} \cup \{0\}$, $1 \leq i \leq k$ и $j = i + 1$.

Табела 5.14. Број коефицијената за неке целобројне SEC-DAEC кодове.

	Дужина бајта (у битима)											
	5	6	7	8	9	10	11	12	13	14	15	16
Теоријска граница	0	1	3	7	14	25	47	88	162	302	563	1056
Експеримент. резултати	0	0	2	2	6	9	21	32	85	118	284	429

Табела 5.15. Коефицијенти за 15-битне и 16-битне целобројне SEC-DAEC кодове.

$b = 15$															
5	13	19	23	29	37	43	53	55	66	67	71	79	85	89	95
107	109	113	121	133	135	141	146	157	161	169	175	179	185	189	195
197	205	210	223	225	231	233	243	245	261	265	273	283	286	287	291
287	301	306	307	310	317	323	331	334	353	357	365	371	381	385	393
407	409	414	422	445	447	451	461	467	475	478	485	487	491	497	502
523	531	541	542	545	550	599	601	605	611	627	631	649	651	655	665
701	711	718	747	749	755	779	781	785	790	837	841	851	857	862	870
883	887	909	915	917	923	935	939	953	955	971	973	977	986	989	1042
1065	1079	1081	1099	1101	1105	1125	1131	1139	1165	1166	1173	1177	1186	1190	1243
1245	1249	1257	1263	1267	1285	1307	1309	1314	1389	1419	1421	1427	1435	1437	1445
1446	1489	1490	1495	1559	1571	1573	1589	1591	1610	1614	1619	1627	1630	1637	1654
1658	1689	1699	1757	1765	1767	1813	1817	1831	1835	1851	1873	1890	1943	1945	1967
2006	2125	2187	2193	2201	2215	2231	2246	2279	2283	2329	2338	2374	2422	2515	2629
2659	2662	2699	2765	2777	2779	2885	2915	2934	2966	2994	3001	3227	3239	3255	3353
3383	3429	3438	3477	3483	3484	3493	3509	3511	3515	3533	3694	4247	4403	4563	4722
4942	4966	5026	5194	5239	5355	5395	5705	5805	5831	5914	5994	6358	6444	6507	6596
6708	6731	6812	6821	6926	7208	7236	7289	7495	8587	8724	9305	9307	9414	9937	10086
10892	11505	11558	12668	14279	19549	19665	20761	21275	22811	29266	29661				
$b = 16$															
5	13	19	23	29	37	43	53	55	66	67	71	79	95	101	109
118	121	125	131	133	139	141	146	157	161	166	182	189	191	195	203
215	223	233	235	238	281	283	287	293	294	310	323	325	329	334	337
343	347	350	358	365	367	370	391	395	401	403	409	413	419	427	439
442	450	459	461	467	475	481	485	490	494	502	503	505	526	534	557
558	565	569	577	578	602	607	613	614	622	635	641	643	649	651	659
670	673	693	701	710	715	721	722	737	743	745	751	755	761	762	788
799	803	809	811	817	819	827	828	839	841	845	865	871	886	898	907
909	913	919	931	941	945	953	954	1007	1009	1018	1049	1051	1059	1069	1071
1081	1093	1097	1100	1113	1115	1119	1122	1133	1137	1145	1157	1159	1165	1166	1179
1186	1187	1189	1206	1213	1219	1229	1231	1234	1246	1249	1254	1274	1289	1299	1307
1314	1337	1339	1354	1429	1436	1459	1463	1466	1471	1475	1481	1483	1489	1491	1507
1540	1543	1554	1561	1562	1570	1581	1582	1586	1587	1589	1593	1603	1607	1619	1627
1637	1639	1651	1653	1657	1667	1669	1675	1687	1689	1735	1743	1753	1765	1766	1807
1813	1817	1823	1827	1837	1843	1853	1859	1874	1915	1930	1942	1970	1982	2097	2099
2129	2141	2143	2161	2164	2189	2195	2203	2245	2251	2270	2278	2281	2291	2319	2321
2329	2335	2350	2375	2413	2415	2419	2447	2459	2469	2485	2487	2511	2513	2519	2529
2583	2591	2601	2605	2629	2647	2675	2677	2702	2709	2787	2798	2838	2842	2854	2867
2871	2921	2925	2935	2979	2985	2995	3011	3125	3127	3141	3142	3150	3170	3175	3187
3215	3219	3227	3258	3269	3309	3313	3335	3358	3385	3443	3447	3448	3469	3474	3529
3541	3559	3623	2629	3647	3659	3689	3692	3697	3701	3721	3727	3731	3794	3814	4311
4313	4381	4387	4407	4411	4495	4510	4517	4519	4718	4934	4972	5018	5197	5261	5385
5521	5677	5714	5747	5902	5906	5923	5988	6004	6242	6247	6311	6365	6389	6391	6420
6551	6553	6603	6605	6610	6730	7003	7026	7213	7269	7300	7315	7335	7475	7759	7778
7956	8033	8948	9053	9103	9555	9698	9734	9817	9926	10286	10525	11092	11173	11450	11675
12601	12709	12990	13082	13459	17974	18350	18650	21038	21554	21608	21956	22051	22324	22382	25691
26188	30302	30521	32134	41773	43561	46597	46901	54377	56294	56428	60694	63398			

У ранијим радовима [39]-[50] је показано да сваки декодер мора да изврши m упита у СТ и m поређења ($1 \leq m \leq \lfloor \log_2 |\zeta| \rfloor + 2$) како би пронашао унос који садржи

податке о локацији и вредности грешке. У [56] је, међутим, коришћен приступ који омогућава декодеру да директно приступи траженом уносу ($m = 1$). Ова идеја се заснива на савршеном хеширању [60], односно повећању величине СТ додавањем $M = 2^b - 2 - |\zeta|$ уноса који одговарају грешкама које се не могу исправити. Ови уноси морају бити облика $(S, 0, 0, 0, 0)$ како би се разликовали од оних који садрже податке о локацији и вредности грешке. Како би овај концепт био јаснији, биће илустрован на примеру $(21, 14)$ целобројног SEC-DAEC кода.

Табела 5.16. Комплетна СТ за $(21, 14)$ целобројни SEC-DAEC код.

	S	i	E ₁	j	E ₂		S	i	E ₁	j	E ₂		S	i	E ₁	j	E ₂		S	i	E ₁	j	E ₂
1	1	3	1	0	0	33	33	1	95	0	0	65	65	0	0	0	0	97	97	1	6	0	0
2	2	3	2	0	0	34	34	0	0	0	0	66	66	1	63	0	0	98	98	0	0	0	0
3	3	3	3	0	0	35	35	2	121	0	0	67	67	1	12	0	0	99	99	1	31	0	0
4	4	3	4	0	0	36	36	2	1	3	63	68	68	0	0	0	0	100	100	2	1	0	0
5	5	1	126	0	0	37	37	2	1	3	64	69	69	0	0	0	0	101	101	2	48	0	0
6	6	3	6	0	0	38	38	2	8	0	0	70	70	2	115	0	0	102	102	2	95	0	0
7	7	1	24	0	0	39	39	0	0	0	0	71	71	0	0	0	0	103	103	3	103	0	0
8	8	3	8	0	0	40	40	1	119	0	0	72	72	1	1	2	63	104	104	0	0	0	0
9	9	0	0	0	0	41	41	0	0	0	0	73	73	2	2	0	0	105	105	0	0	0	0
10	10	1	125	0	0	42	42	0	0	0	0	74	74	0	0	0	0	106	106	0	0	0	0
11	11	0	0	0	0	43	43	0	0	0	0	75	75	2	96	0	0	107	107	1	4	0	0
12	12	3	12	0	0	44	44	0	0	0	0	76	76	2	16	0	0	108	108	2	123	0	0
13	13	2	103	0	0	45	45	1	1	2	64	77	77	2	63	0	0	109	109	0	0	0	0
14	14	1	48	0	0	46	46	2	3	0	0	78	78	0	0	0	0	110	110	0	0	0	0
15	15	1	124	0	0	47	47	1	16	0	0	79	79	3	79	0	0	111	111	3	111	0	0
16	16	3	16	0	0	48	48	3	48	0	0	80	80	1	111	0	0	112	112	1	3	0	0
17	17	0	0	0	0	49	49	0	0	0	0	81	81	2	124	0	0	113	113	1	79	0	0
18	18	0	0	0	0	50	50	2	64	0	0	82	82	1	126	2	63	114	114	2	24	0	0
19	19	2	4	0	0	51	51	2	111	0	0	83	83	0	0	0	0	115	115	3	115	0	0
20	20	1	123	0	0	52	52	2	31	0	0	84	84	0	0	0	0	116	116	0	0	0	0
21	21	0	0	0	0	53	53	0	0	0	0	85	85	0	0	0	0	117	117	1	2	0	0
22	22	0	0	0	0	54	54	2	125	0	0	86	86	0	0	0	0	118	118	0	0	0	0
23	23	0	0	0	0	55	55	1	126	2	64	87	87	1	8	0	0	119	119	3	119	0	0
24	24	3	24	0	0	56	56	0	0	0	0	88	88	0	0	0	0	120	120	1	103	0	0
25	25	2	32	0	0	57	57	2	12	0	0	89	89	2	119	0	0	121	121	3	121	0	0
26	26	2	79	0	0	58	58	0	0	0	0	90	90	2	126	3	63	122	122	1	1	0	0
27	27	2	126	0	0	59	59	0	0	0	0	91	91	2	126	3	64	123	123	3	123	0	0
28	28	1	96	0	0	60	60	1	115	0	0	92	92	2	6	0	0	124	124	3	124	0	0
29	29	0	0	0	0	61	61	1	64	0	0	93	93	0	0	0	0	125	125	3	125	0	0
30	30	1	121	0	0	62	62	0	0	0	0	94	94	1	32	0	0	126	126	3	126	0	0
31	31	3	31	0	0	63	63	3	63	0	0	95	95	3	95	0	0						
32	32	3	32	0	0	64	64	3	64	0	0	96	96	3	96	0	0						

Пример 1. Нека је $b = 7$, $k = 2$, $C_1 = 5$ и $C_2 = 27$. Из теореме 25 знамо да СТ има $|\zeta| = 86$ уноса који садрже податке о локацији и вредности грешке. Да бисмо убрзали процедуру корекције грешака, проширићемо СТ додавањем $M = 2^7 - 2 - 86 = 40$ уноса који одговарају грешкама које се не могу кориговати (табела 5.16). Претпоставимо сада да је потребно пренети 14 бита података, 1101011 0011101. У том случају, кодер ће генерисати чек-бајт $B_3 = C_1 \cdot B_1 + C_2 \cdot B_2 \pmod{127} = 5 \cdot 107 + 27 \cdot 29 \pmod{127} = 48 = 0110000_2$, тако да ће кодна реч ће имати облик $x = (B_1, B_2, B_3) = (107, 29, 48) = (1101011_2, 0011101_2, 0110000_2)$.

Сценарио 1: Претпоставимо да је током преноса података оштећен 12-и бит, $y = (\underline{B}_1, \underline{B}_2, \underline{B}_3) = (1101011_2, 0011\underline{0}01_2, 0110000_2) = (107, 25, 48)$. У том случају, вре-

дност синдрома S биће једнака $S = 5 \cdot 107 + 27 \cdot 25 - 48 \pmod{127} = 19$. У следећем кораку, декодер ће прочитати 19-и унос из СТ (табела 5.17) и извршити операцију $B_2 = 25 + 4 \pmod{127} = 29$.

Сценарио 2: Претпоставимо да су током преноса података оштећени 14-и и 15-и бит, $y = (\underline{B}_1, \underline{B}_2, \underline{B}_3) = (1101011_2, 0011100_2, 1110000_2) = (107, 28, 112)$. У том случају, вредност синдрома S биће једнака $S = 5 \cdot 107 + 27 \cdot 28 - 112 \pmod{127} = 36$. Декодер ће, стога, прочитати 36-и унос из СТ (табела 5.17) и извршити две (паралелне) операције: $B_2 = 28 + 1 \pmod{127} = 29$ и $B_3 = 112 + 63 \pmod{127} = 48$.

Сценарио 3: Коначно, претпоставимо да су током преноса података оштећени 9-и, 10-и, 11-и, 12-и и 14-и бит, $y = (\underline{B}_1, \underline{B}_2, \underline{B}_3) = (1101011_2, 0100000_2, 0110000_2) = (107, 64, 48)$. У том случају, декодер ће израчунати $S = C_1 \cdot \underline{B}_1 + C_2 \cdot \underline{B}_2 - \underline{B}_3 \pmod{127} = 5 \cdot 107 + 27 \cdot 64 - 48 \pmod{127} = 56$ и закључити да вредност $S = 56$ указује на грешку које није могуће исправити (табела 5.17). Стога ће декодирање прогласити неуспешним.

Као и остали декодери, и предложени може извршити погрешну корекцију. Она ће се десити сваки пут када нека грешка, различита од једноструке битске грешке и двоструке суседне битске грешке, генерише синдром чија вредност припада скупу ζ . Да бисмо израчунали вероватноћу таквог исхода, претпоставићемо да се оптичка веза може моделовати као канал у којем се могу јавити само t -битне рафалне грешке ($t \leq 3$). На основу ове претпоставке, која се заснива на експерименталним резултатима (слика 4.7), можемо изнети следећу теорему.

Теорема 27. Вероватноћа погрешног целобројног SEC-DAEC декодирања за канал у којем се јављају t -битне рафалне грешке ($t \leq 3$) је једнака

$$P_{ID} = \frac{2 \cdot b \cdot (k+1) + k - 1}{2^{b-1} - 1} \cdot \left\{ (k+1) \cdot 4 \cdot (b-2) \cdot \varepsilon^3 \cdot (1-\varepsilon)^{b-3} \cdot [1 - 4 \cdot (b-2) \cdot \varepsilon^3 \cdot (1-\varepsilon)^{b-3}]^k + \right. \\ \left. k \cdot 8 \cdot \varepsilon^2 \cdot (1-\varepsilon)^{2 \cdot b - 2} \cdot [1 - 8 \cdot \varepsilon^2 \cdot (1-\varepsilon)^{2 \cdot b - 2}]^{k-1} + k \cdot 8 \cdot \varepsilon^3 \cdot (1-\varepsilon)^{2 \cdot b - 3} \cdot [1 - 8 \cdot \varepsilon^3 \cdot (1-\varepsilon)^{2 \cdot b - 3}]^{k-1} \right\}$$

где ε означава вероватностопа појаве грешке.

Доказ. У поменутом каналу, сходно претпоставци, рафалне грешке не могу оштетити више од три узастопна бита. С друге стране, знамо да целобројни SEC-DAEC кодови могу исправити све једноструке битске грешке и све двоструке суседне битске грешке. Дакле, ако је кодна реч оштећена троструком суседном битском (ТСБ) грешком, декодер ће или прогласити грешку у декодовању ($S \notin \zeta$) или ће извршити погрешну корекцију ($S \in \zeta$). Узимајући у обзир да су вишеструке ТСБ грешке за неколико редова величине мање вероватне од појединачних ТСБ грешака, израз за вероватноћу нетачног декодирања гласи

$$P_{ID} = (N_{CS}/N_{TS}) \cdot P_{TB} = [|\zeta|/(2^b - 2)] \cdot P_{TB}$$

при чему N_{CS} означава број синдрома који се може исправити, N_{TS} укупан број синдрома, док P_{TB} означава вероватноћу појаве ТСБ грешке. Приметимо да се вероватноћа P_{TB} може представити као $P_{TB} = P_1 + P_2$, где је P_1 вероватноћа појаве ТСБ грешке која захвата један b -битни бајт (ТСБ₁ грешка), док P_2 означава вероватноћу појаве ТСБ грешке која захвата два суседна b -битна бајта (ТСБ₂ грешка). Да бисмо израчунали вероватноћу P_1 , требамо имати на уму да ТСБ₁ грешке генеришу $4 \cdot (b - 2)$ синдрома по b -битном бајту [45]. Ако овоме додамо чињеницу да кодна реч има $k + 1$ бајтова, јасно је да се вероватноћа P_1 израчунава као $P_1 = (k + 1) \cdot p_1 \cdot (1 - p_1)^k$, где $p_1 = 4 \cdot (b - 2) \cdot \varepsilon^3 \cdot (1 - \varepsilon)^{b-3}$ означава вероватноћу да је један b -битни бајт оштећен ТСБ₁ грешком. Да бисмо израчунали вероватноћу P_2 , морамо знати број синдрома који генеришу ТСБ₂ грешке. С тим у вези, треба имати у виду да је скуп синдрома који одговарају ТВ₂ грешкама дефинисан као

$$t = \bigcup_{i=1}^6 Q_i$$

где је

$$Q_1 = \bigcup_{i=1}^k \left\{ \pm 2^0 \cdot C_i \pm (2^{b-1} + 2^{b-2}) \cdot C_{i+1} \pmod{2^b - 1} \right\}$$

$$Q_2 = \bigcup_{i=1}^k \left\{ \pm 2^0 \cdot C_i \pm (2^{b-1} - 2^{b-2}) \cdot C_{i+1} \pmod{2^b - 1} \right\}$$

$$Q_3 = \bigcup_{i=1}^k \left\{ \pm 2^0 \cdot C_i \pm 2^{b-2} \cdot C_{i+1} \pmod{2^b - 1} \right\}$$

$$Q_4 = \bigcup_{i=1}^k \left\{ \pm (2^1 + 2^0) \cdot C_i \pm 2^{b-1} \cdot C_{i+1} \pmod{2^b - 1} \right\}$$

$$Q_5 = \bigcup_{i=1}^k \left\{ \pm (2^1 - 2^0) \cdot C_i \pm 2^{b-1} \cdot C_{i+1} \pmod{2^b - 1} \right\}$$

$$Q_6 = \bigcup_{i=1}^k \left\{ \pm 2^1 \cdot C_i \pm 2^{b-1} \cdot C_{i+1} \pmod{2^b - 1} \right\}$$

Ако упоредимо скупове Q_2 , Q_3 , Q_5 и d_3 , приметимо да је $Q_2 \equiv Q_3$ и $Q_5 \equiv d_3$. Из овога је јасно да ТСБ₂ грешке генеришу 16 синдрома, тј. 8 синдрома када захвате два бита и 8 синдрома када захвате три бита. С обзиром да се ТСБ₂ грешке могу јавити на k различитих позиција, вероватноћа P_2 се израчунава као $P_2 = k \cdot p_2 \cdot (1 - p_2)^{k-1} + k \cdot p_3 \cdot (1 - p_2)^{k-1}$, где $p_2 = 8 \cdot \varepsilon^2 \cdot (1 - \varepsilon)^{2b-2}$ означава вероватноћу да су два бита у оквиру два суседна b -битна бајта оштећена ТСБ₂ грешком, док $p_3 = 8 \cdot \varepsilon^3 \cdot (1 - \varepsilon)^{2b-3}$ означава вероватноћу да су три бита у оквиру два суседна b -битна бајта оштећена ТСБ₂ грешком. Самим тим, сабирањем вероватноћа P_1 и P_2 добијамо

$$P_{ID} = \frac{2 \cdot b \cdot (k+1) + k - 1}{2^{b-1} - 1} \cdot \left\{ (k+1) \cdot 4 \cdot (b-2) \cdot \varepsilon^3 \cdot (1-\varepsilon)^{b-3} \cdot [1 - 4 \cdot (b-2) \cdot \varepsilon^3 \cdot (1-\varepsilon)^{b-3}]^k + \right.$$

$$k \cdot 8 \cdot \varepsilon^2 \cdot (1-\varepsilon)^{2 \cdot b-2} \cdot [1 - 8 \cdot \varepsilon^2 \cdot (1-\varepsilon)^{2 \cdot b-2}]^{k-1} + k \cdot 8 \cdot \varepsilon^3 \cdot (1-\varepsilon)^{2 \cdot b-3} \cdot [1 - 8 \cdot \varepsilon^3 \cdot (1-\varepsilon)^{2 \cdot b-3}]^{k-1} \Bigg\}$$

Користећи теорему 27 можемо успоставити везу између вредности ε и P_{ID} . Из добијених резултата (табела 5.17) видимо да 16-битни целобројни SEC-DAEC кодови имају малу вероватноћу погрешног декодирања.

Табела 5.17. Вероватноћа погрешног декодовања целобројних SEC-DAEC кодова у случају канала у којем се јављају t -битне рафалне грешке ($t \leq 3$).

(272, 256) код		(528, 512) код		(1040, 1024) код		(2064, 2048) код		(4112, 4096) код	
ε	P_{ID}	ε	P_{ID}	ε	P_{ID}	ε	P_{ID}	ε	P_{ID}
10^{-3}	$2.14 \cdot 10^{-6}$	10^{-3}	$8.31 \cdot 10^{-6}$	10^{-3}	$3.27 \cdot 10^{-5}$	10^{-3}	$1.30 \cdot 10^{-4}$	10^{-3}	$5.17 \cdot 10^{-4}$
10^{-4}	$2.18 \cdot 10^{-8}$	10^{-4}	$8.47 \cdot 10^{-8}$	10^{-4}	$3.34 \cdot 10^{-7}$	10^{-4}	$1.33 \cdot 10^{-6}$	10^{-4}	$5.29 \cdot 10^{-6}$
10^{-5}	$2.18 \cdot 10^{-10}$	10^{-5}	$8.49 \cdot 10^{-10}$	10^{-5}	$3.35 \cdot 10^{-9}$	10^{-5}	$1.30 \cdot 10^{-8}$	10^{-5}	$5.30 \cdot 10^{-8}$
10^{-6}	$2.18 \cdot 10^{-12}$	10^{-6}	$8.49 \cdot 10^{-12}$	10^{-6}	$3.35 \cdot 10^{-11}$	10^{-6}	$1.30 \cdot 10^{-10}$	10^{-6}	$5.30 \cdot 10^{-10}$

5.2.4. Поређење са Интернет чексумом

Предности исправљања грешака које оштећују мултимедијалне пакете уочене су пре 30 година. Из тог разлога, дизајнери АТМ протокола омогућили су пријемнику да исправи једноструке битске грешке које оштећују заглавље пакета [61]. За разлику од АТМ-а, UDP је дизајниран у време када није било потребе за масовном мултимедијалном комуникацијом. Из тог разлога његови дизајнери су одлучили да све податке треба заштитити Интернет чексумом. У наставку ћемо видети које су предности и мане коришћења 16-битних целобројних SEC-DAEC кодова у односу на Интернет чексуму.

1. Могућности контроле грешака

Интернет чексума се може дефинисати као

$$IC(k) = \left\{ x \in Z_{2^{16}-1}^{k+1} : \sum_{i=1}^k B_i \equiv B_{k+1} \pmod{2^{16}-1} \right\} \quad (5.70)$$

где је $x = (B_1, B_2, \dots, B_k, B_{k+1}) \in Z_{2^{16}-1}^{k+1}$ и $k \geq 1$. С друге стране, 16-битни целобројни SEC-DAEC кодови су дефинисани као

$$C(16, k, c) = \left\{ x \in Z_{2^{16}-1}^{k+1} : \sum_{i=1}^k C_i \cdot B_i \equiv B_{k+1} \pmod{2^{16}-1} \right\} \quad (5.71)$$

где је $x = (B_1, B_2, \dots, B_k, B_{k+1}) \in Z_{2^{16}-1}^{k+1}$, $c = (C_1, C_2, \dots, C_k, 1) \in Z_{2^{16}-1}^{k+1}$ и $1 \leq k \leq 429$ (табела 5.16). Иако су конструкцијске разлике између Интернет чексума и целобројних предложених кодова минималне, оне имају огроман утицај на перформансе оба алгорита. Прецизније, Интернет чексума ће детектовати све једноструке рафалне грешке дужине до 15 бита, док ће 16-битни целобројни SEC-DAEC кодови исправити све једноструке битске грешке и све двоструке суседне битске грешке унутар N -битне кодне речи ($N \leq 6880$).

2. Заштита мултимедијалних пакета од грешака

Величина мултимедијалних пакета зависи од верзије IP-а који се користи [24]. Конкретно, ако се гласовни подаци (VoIP пакети) испоручују преко IPv4 мреже, Интернет чексума штити између 60 и 280 бајтова (укључујући заглавље RTP протокола) (слика 5.5), док у случају преноса података преко IPv6 мреже штити између 84 до 304 бајта. Када је у питању телевизијски садржај (IPTV пакети), Интернет чексума штити 228 или више бајтова, ако се пренос обавља преко IPv4 мреже, односно 252 или више, бајтова ако се користи IPv6 мрежа. Све ове чињенице упућују на закључак да 16-битни целобројни SEC-DAEC кодови могу заштитити све VoIP пакете, као и све IPTV пакете чија величина не прелази 752 бајта ($1 \leq M \leq 4$).

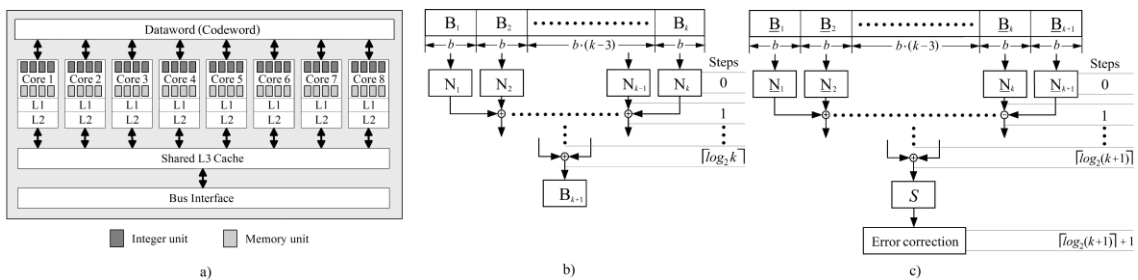
IPv4/IPv6 Header	UDPv4/UDPv6 Header	App Header	VoIP Data
20/40 bytes	8/12 bytes	12 bytes	20 to 240 bytes

IPv4/IPv6 Header	UDPv4/UDPv6 Header	App Header	IPTV Data
20/40 bytes	8/12 bytes	12 bytes	$M \cdot 188$ bytes ($1 \leq M \leq 7$)

Слика 5.5. Заглавља мултимедијалних пакета заштићених Интернет чексумом.

3. Процедуре кодовања и декодовања

Најбржи начин за кодовање података заснован је на коришћењу концепта бинарног стабла. У том случају, кодер (ПОН) мора паралелно израчунати производе $N_i = C_i \cdot B_i$, након чега ће извршити паралелно сабирање k целих бројева (слика 5.6). Дакле, једино додатно кашњење, у поређењу са Интернет чексумом, је време потребно за израчунавање производа два цела броја (Интернет чексума не изводи корак 0).



Слика 5.6. (а) Блок дијаграм осмојезгарног ПОН-а и илустрација процедура (б) кодовања и (с) декодовања.

Слична ситуација је и по питању декодовања. Наиме, декодер (ПОН) мора паралелно израчунати производе $N_i = C_i \cdot B_i$ и паралелно сабрати (одузети) $k + 1$ целих бројева (слика 5.6). Након тога, ако је $S \neq 0$, декодер ће извршити један упит у СТ и једно или два паралелна сабирања. Дакле, у поређењу са Интернет чексумом, постоје три додатне операције: једно целобројно множење, једно целобројно

сабирање и један упит (Интернет чексума не изводи кораке 0 и $\lceil \log_2(k+1) \rceil + 1$). Све ове операције се могу извршити ако је резервисано $|2^b - 2| \times |3 \cdot b + 2 \cdot \log_2 \lceil k+1 \rceil| = |65534| \times |48 + 2 \cdot \log_2 \lceil 430 \rceil| \approx 0.54$ МВ меморије за чување СТ (табела 5.18). То, међутим, није проблем за данашње ПОН-ове, који имају неколико МВ кеш меморије, као и четири целобројне јединице по језгру [62] (слика 5.6).

Табела 5.18. Поређење Интернет чексуме и целобројних SEC-DAEC кодова.

Главне карактеристике	Интернет чексума	Целобројни SEC-DAEC кодови
Способности у погледу детекције и корекције грешака	Детекција једноструких рафалних грешака дужине до 15 бита	Корекција једноструких битских грешака и двоструких суседних битских
Број операција током кодовања	$\lceil \log_2 k \rceil$ целобројних сабирања	$\lceil \log_2 k \rceil$ целобројних сабирања и једно целобројно множење
Број операција током декодовања	$\lceil \log_2(k+1) \rceil$ целобројних сабирања	$\lceil \log_2(k+1) \rceil + 1$ целобројних сабирања, једно целобројно множење и један упит у СТ
Величина пакета који се може заштити	Неограничено	До 6864 бита
Редунданса	16 бита	16 бита
Меморијски захтеви	Не постоје	0.54 МВ
Врста имплементације	Софтверска	Софтверска

6. Примена вештачке интелигенције у СПОВ-у

У овом поглављу биће представљене две дубоке неуронске мреже: CNN (енг. *Convolution Neural Network*) и MLP (енг. *Multi Layer Perceptron*). Прва мрежа је употребљена за детекцију превентивних кварова, као што је подмазивање кугличних лежајева пумпи. Установљено је, на пример, да је превише подмазивања лежајева штетније од неподмазивања. CNN мрежа треба да развије могућност предвиђања потребе замене кугличних лежајева како би се правовремено извршила акција одржавања и како не би дошло до хаварије постројења. Друга мрежа, MLP, употребљена је за регулисање дозирања активних супстанци у отпадној води како би се дошло до идеалних резултата рада постројења.

6.1. Употреба CNN-а

CNN је мрежа је дубока неуронска мрежа обучена за детекцију кварова на пумпама које служе за претакање отпадних вода и њихово испуштање у реципијент. Рад поменутих пумпи је од кључног значаја за функционалност система. Уколико би оне престале са радом, дошло би до проблема пребацивања отпадне воде из пријемне каде у каде за неутрализацију, односно даље у ламиниране сепараторе и у реципијент. Пошто се СПОВ налази испод нивоа фабрике, услед отказа пумпе, дошло би и до плављења постројења. Једно решење приказано је на слици 6.1: постављене су две паралелне пумпе, тако да случају отказа прве долази до аутоматске замене другом пумпом.



Слика 6.1. Пумпе за избацивање воде.

У теорији, ово би требало да да добре резултате. Међутим, у пракси се показало да, по отказу прве пумпе, често затаји и друга пумпа, па самим тим долази до изливања воде из када. Разлог за ово је што обично једна пумпа увек ради, док је друга резервна. Услед великих концентрација киселина и база у води долази до пропадања заптивки-семеринга које штите кугличне лежајеве пумпе. Када вода продре до кугличних лежајева, долази до корозије осовине и блокаде ротора пумпи. Приликом укључења аутоматике долази до активирања моторне заштите због оптерећења пумпе која повлачи знатно већу струју од номинално задате и пумпа не може кренути са радом. У периоду истраживања (2017. година) дошло је чак три пута до ове врсте отказа. Због тога се, као решење, појавила идеја да се кроз дијагностику ових пумпи изврши благовремено превентивно одржавање како би се осигурао поуздан рад система.

6.1.1. Прикупљање података за обуку CNN-а

Први корак у обуци сваке CNN мреже јесте прикупљање информација. То, у конкретном случају, значи да је потребно извршити прикупљање и класификацију звукова приликом рада мотора пумпе. Први корак у том смеру јесте постављање микрофона на одговарајућа места. Као мерни уређај изабран је микрофон компаније NTi Audio, модел M2211 [63]. Ради се о микрофону класе тачности 1 (највиша мерна класа), који је испоручен као фабрички калибрисан уређај. Микрофон је постављен на 20 cm од извора звука, односно мотора пумпе (слика 6.2).



Слика 6.2. Мерно место на којем је постављен микрофон.

По постављању микрофона, вршено је снимање великог броја узорак. За потребе класификације стања мотора пумпе (да ли мотор ради добро или не) коришћени су параметри MotorOK и MotorNOK. Вршено је снимање 281 узорка када мотор пумпе ради добро и 387 узорак када ради лоше. Програм који је коришћен за снимање и одсецање звука на 5 секунди представљен је у Прилогу 4. Снимање ових звучних карактеристика трајало је више од једне године, тачније од марта 2017. године, па до децембра 2018. године. Након извршеног снимања приступљено је претварању снимљених карактеристика звука у слику која се даље користила за обуку CNN мреже (у Прилогу 5 приказан је програмски код који је коришћен у ту сврху).

6.2. Детекција висине звука на бази цепструма

Обзиром да је висина тона или основна фреквенција један од најзначајнијих аспеката говора, откривање висине тона је свеprisутан задатак који се налази у многим истраживачким студијама везаним за говор. Тако се детекција висине тона може наћи у различитим системима за препознавање и идентификацију говорника, системима за синтезу говора, телекомуникационим системима итд. [64]. Поред тога, висина је једна од аудио карактеристика (атрибута) која се користи за класификацију звука, детекцију и препознавање применом машинског или дубоког учења [65].

Висина тона може бити детектована у временском или фреквенцијском домену. Најважније методе у временском домену су типично засноване на аутокорељацији користећи хипотезу да је функција аутокорељације периодичног сигнала такође периодична и да су ова два периода коинцидентна [66]. Метода заснована на цепструму је широко призната као једна од најпопуларнијих техника за детекцију висине тона у фреквенцијском домену. Коришћење цепструма, сложеног цепструма и хомоморфне деконволуције протеже се у различитим областима, као што су аудио обрада, обрада говора, геофизика, радар и медицинско снимање. И цепструм и сложени цепструм нуде вредне примене, укључујући рестаурацију историјских снимака фонографа, детекцију висине тона кроз анализу цепструма, као и препознавање говора и идентификацију звучника. Из цепструма се може извршити процена периодичности повезане са тоном присутне у сигналу. У поређењу са неким другим методама за детекцију висине тона, метода цепструма може да пружи тачне и робусне резултате, али је рачунарски сложена [66].

У наставку рада представљени су потенцијали коришћења различитих карактеристика звука и фреквентног домена за потребе детекције кварова на индустријским машинама (мотори пумпи, семеринзи, куглични лежајеви, итд). Задатак

истраживања је утврдити да ли је могуће направити разлику између разних врста и типова електромотора, мотора пумпи и њихових компоненти, кугличних лежајева, семеринга, одредити њихове висине тона коришћењем алгоритма заснованог на цепструму (наравно модификованог тако да се у добијеном цепструму примењује локатор врха), одредити однос звука за различите врсте брзина мотора, мотора пумпи, за различите ротације. Мисли се на промену смерова (смер мотор лево или смер мотор десно). Још један од задатака истраживања је и да се ови принципи могу примењивати у различитим гранама индустрије.

6.2.1. Алгоритми за детекцију висине тона

Висина тона је важан атрибут говорних сигнала. Вибрације гласних жица током говора доводе до таласног облика који показује одређену периодичност. Ова периодичност је представљена као „корак висине“ у временском домену и „основна фреквенција“ или висина тона у фреквенцијском домену [68]. Међутим, висина тона може бити од значаја у машинском и дубоком учењу примењеном на различите аудио сигнале, укључујући оне који садрже звукове индустријских производа (мотори једносмерне струје, кућни апарати или мотори са унутрашњим сагоревањем путничких возила, итд.) [69].

Данас се примењују различити алгоритми који могу да разликују висине тона и деле се према разним критеријумима. Тако, на пример, постоје алгоритми засновани на блоковима и алгоритми засновани на догађајима [70]. Код алгоритама заснованим на блоковима, сигнал је исечен на мале делове под претпоставком да је висина тона континуирана током тих делова и обрнуто. С друге стране, код алгоритама утемељеним на догађајима користи се обележавање висине тона или откривање епохе, при чему се не полази од претпоставке да је нагиб константан током неколико циклуса. Због тога су ови алгоритми у стању да прате брзе промене тона чак и током сегмената [71].

Према домену у којем се примењују, алгоритми се могу поделити у три групе: алгоритми у временском домену, алгоритми у фреквенцијском домену и хибридни алгоритми (комбинација временског и фреквентног домена) [72]. Алгоритми у временском домену користе карактеристике сигнала у временском домену, при чему се користе метода паралелне обраде [73], метода редукције података, модификована аутокорељација и просечна функција разлике магнитуде [74]. С друге стране, основа алгоритма у фреквенцијском домену лежи у концепту да се периодично понашање посматрано у временском домену манифестује као низ пикова (импулса) у фреквенцијском домену. У оквиру ове групе алгоритама, значајне методе укључују спектар хармонијског производа, детекцију висине тона засновану на анализи цепструма, линеарно предиктивно кодирање и процену висине тона

инспирисане таласним обликом [75]. Посебна врста детектора комбинује алгоритме временског и фреквенцијског домена. Алгоритми у домену фреквенције имају могућност давања привремених спектрално усклађених звучних таласа, а потом се могу применити методе аутокорелације које имају способност одређивања периода висине тона. У хибридне методе детекције тона спадају PEFAC [76], YAAPT [77], MBSC [78] и BaNa [79].

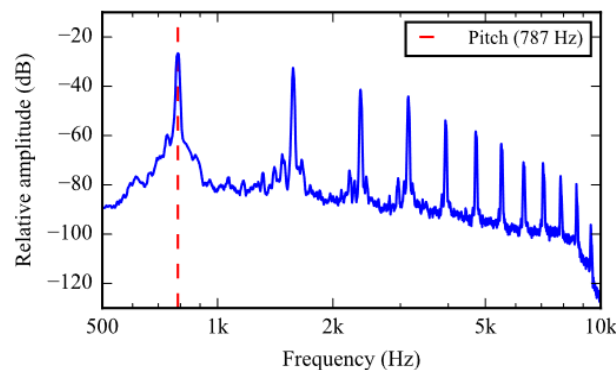
6.2.2. Примењена метода

За потребе овог истраживања одабрани су аудио сигнали са различитим спектралним садржајем. Примењен је алгоритам који је заснован на цепструму као метод у фреквенцијском домену. Разлози за примену ове методе и овог алгоритма је да је ово најпопуларнији метод и да се најчешће користи у практичним применама што је и задатак овог истраживања. Анализом цепструма може се идентификовати максимална вредност унутар дефинисаног сегмента, омогућавајући процену основне фреквенције и висине тона. У овом истраживању коришћен је модификован алгоритам заснован на цепструму за процену висине тонова индустријских мотора – мотора пумпе. Добијене вредности се могу користити за идентификовање различитих радних услова ових мотора, као што су разлике у погледу броја обртаја, промене смера ротације мотора, итд.

Применом инверзне Фуријеове трансформације на логаритам Фуријеове трансформације циљног сигнала $s(n)$ може се добити цепструм $C(m)$ као [80]:

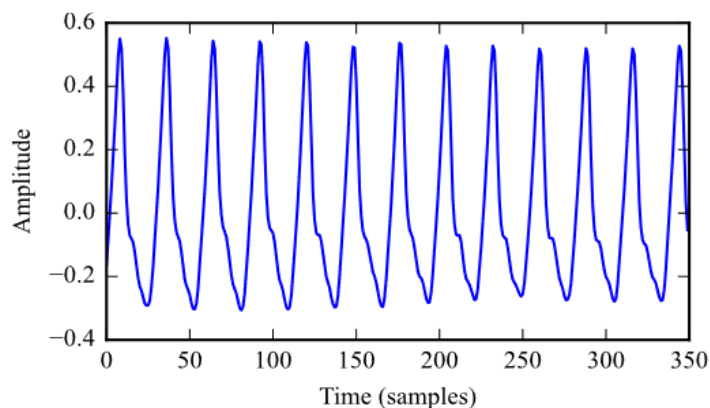
$$C(m) = \frac{1}{N} \left\{ \sum_{n=0}^{N-1} s(n) \cdot \exp^{j \frac{2\pi}{N} mn} \right\} \quad (6.1)$$

Из формуле 6.1 израчуната је висина максимума цепструма. Током истраживања доказано је да је метода утемељена на цепструму сензибилна на шум сигнала у циљаном опсегу.



Слика 6.3. „Спектар звука металног дувачког инструмента који се састоји од фундаменталног (f_0) на фреквенцији од 787 Hz и парцијалне које имају хармонијску дистрибуцију у односу на фундаментално“ [80].

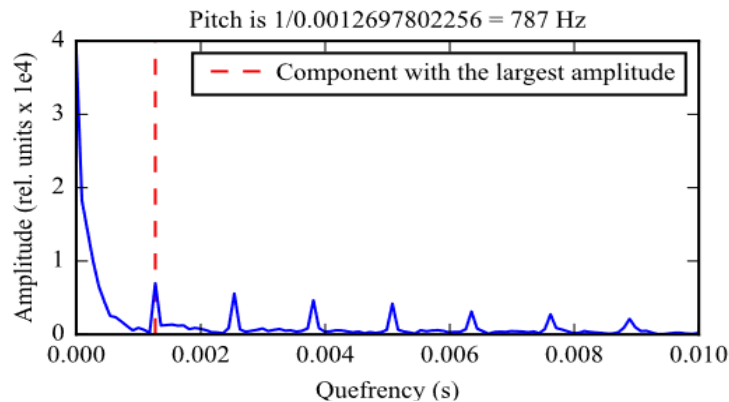
На слици 6.3 приказан је спектар звука металног дувачког инструмента који служи као илустрација сигнала са унапред одређеним распоредом висине и хармоника. С друге стране, постоје сигнали који обухватају звукове одређених индустријских производа са непознатим параметрима. Звук металног дувачког инструмента, на пример, је тонски звук са јасном висином, а његова примарна карактеристика је периодичност, као што се види у временском домену приказаном на слици 6.4 [80].



Слика 6.4. Звук металног дувачког инструмента приказан у временском домену.

Сигнали периодичности се користе у детекцији висине тона на бази цепструма. Сигнал садржи талас специфичне дужине који се стално понавља. Дискретни спектар показује дистрибуцију истакнутих пикова који су равномерно расути, одржавајући природу феномена у оквиру спектра фреквенције.

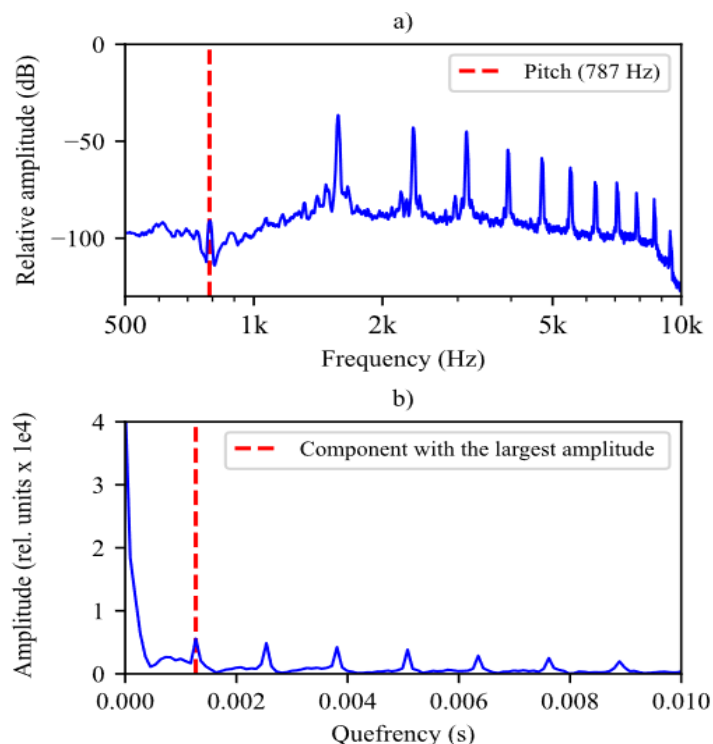
Алгоритам за детекцију висине тона заснован на цепстралној анализи укључује детаљан опис његових пет примарних корака. Почетна фаза подразумева претварање сигнала из временског у спектрални домен. Користећи релевантну функцију у Python-у могу се добити две променљиве: спектар (представљен комплексним бројевима) и вектор фреквенције. Затим се израчунава логаритам величине спектра. Техника цепструма има за циљ да пренесе периодичност уочену у логаритамском спектру у временском домену. Да би се то постигло примењује се инверзна Фуријеова трансформација на податке логаритамске спектралне величине. На слици 6.5 приказан је цепструм звука металног дувачког инструмента добијен описаном методом. Вертикална или x -оса означена је као q -фреквенција, што је преуређени термин за фреквенцију и повезана је са временском скалом. Овај распоред је резултат инверзне Фуријеове трансформације, где је фреквенција инвертована. Сходно томе, више фреквенције су позициониране на почетку x -осе, а ниже позициониране на крају. Да би се фреквенције претвориле у q -фреквенције узима се реципрочна вредност фреквенције, односно $1/\text{фреквенција}$ [80].



Слика 6.5. „Цепструм звука металног дувачког инструмента“ [80].

У следећој фази фокус је на идентификацији максималне вредности цепструма, што подразумева прецизно одређивање вредности q -фреквенције која одговара врхунцу цепструма. У већини случаја, највећа амплитуда струјног цепструма налази се у почетној тачки x -осе, било на нултој фреквенцији или у њеној близини. Овај конкретан сценарио представља значајан изазов за аутоматску детекцију висине тона. Постоји неколико приступа за превазилажење овог проблема. Један од њих је ограничавање опсега q -фреквенције у коме се врши претрага максимума цепструма. Процена висине за одређене звукове, попут оних које производе индустријских производа, може бити нетачна због несигурности око очекиваног фреквентног опсега. Алтернативни приступ укључује занемаривање максималне вредности на нултој фреквенцији у цепструму и уместо тога коришћење друге највеће вредности за процену висине тона. Међутим, овај метод може довести до грешке у детекцији висине тона ако цепструм не садржи максималну вредност на нултој фреквенцији [80].

У овом раду примењен је алтернативни приступ решавању поменутог проблема. Коришћење функције за проналажење врхова из Python `scipy` библиотеке је следеће. Ова функција упоређује суседне вредности како би открила све локалне максимуме и омогућава спецификацију различитих услова за својства пикова. Међутим, важно је напоменути да се максимум на нултој фреквенцији у цепструму или у његовој непосредној близини не сматра аутоматски врхом, јер нема суседних вредности са његове леве стране. Процењена висина тона се добија претварањем вредности q -фреквенције изабраног максималног цепструма у одговарајућу фреквенцију, израчунату као реципрочна вредност q -фреквенције максимума цепструма ($\text{pitch} = 1/(\text{quefreny of cepstrum maximum})$). Да би се потврдио описани алгоритам, основна фреквенција звука металног дувачког инструмента се елиминисе филтрирањем. Максимум цепструма налази се на истој позицији q -фреквенције, како пре, тако и после уклањања основне компоненте (слика 6.6) [80].

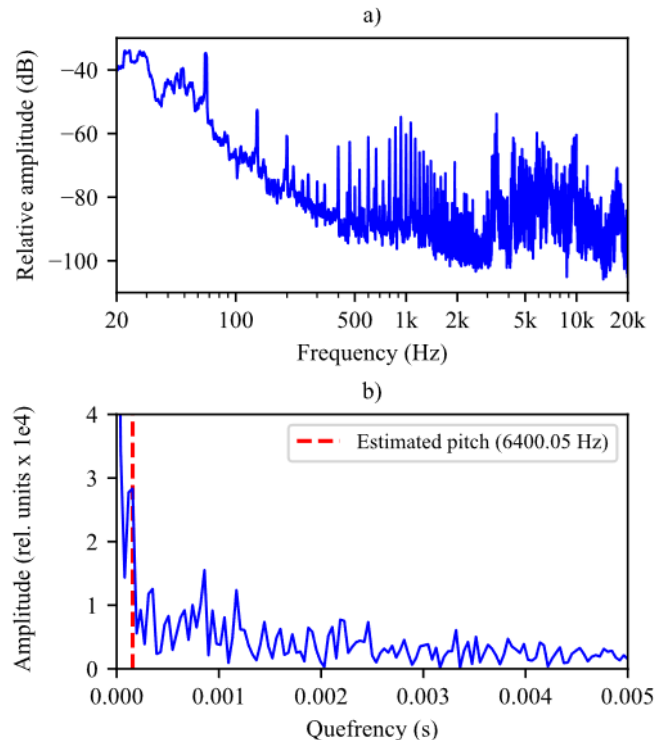


Слика 6.6. „Спектар а) и цепструм б) звука трубе након уклањања основне компоненте која се налази на 787 Hz“ [80].

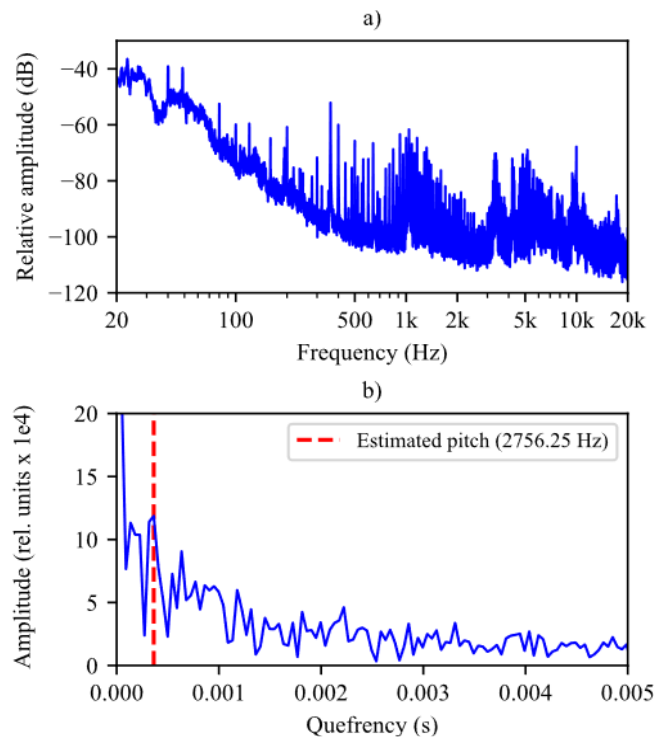
У овом истраживању модификовани алгоритам заснован на цепструму примењен је за процену висине тона користећи различите индустријске звукове, укључујући моторе пумпи и разне моторе. Звук мотора пумпе сниман је у полуанехогеној комори током три циклуса рада са различитим обртајима (4000 rpm, 2400 rpm и 1300 rpm). Слично томе, звуци мотора су снимљени у полуанехогеној комори на два различита типа мотора (тип А и тип Б), у два различита смера ротације (смер 1 и смер 2) и под два различита услова кварова (исправни и са одређеним неисправностима). Анализа снимљених аудио сигнала је вршена коришћењем скрипти развијених у Python 3.8 [80].

6.2.3. Анализа детективног кола

Први предмет анализе били су мотори пумпи. Циљ је био да се утврди да ли је могуће направити разлику између три различита радна услова мотора пумпи, односно три различита броја обртаја у минути: 4000, 2400 и 1300. Слика 6.7 приказује спектар и цепструм мотора пумпи који раде на 4000 rpm. Процењена вредност висине добијене описаним поступком износи 6400,05 Hz [80]. Када се број обртаја промени са 4000 на 2400, долази до специфичних промена у спектру и у целокупном обрасцу, што последично утиче на детектовани тон. Слика 6.8 приказује спектар и цепструм звука мотора пумпи на 2400 rpm, где је процењена висина



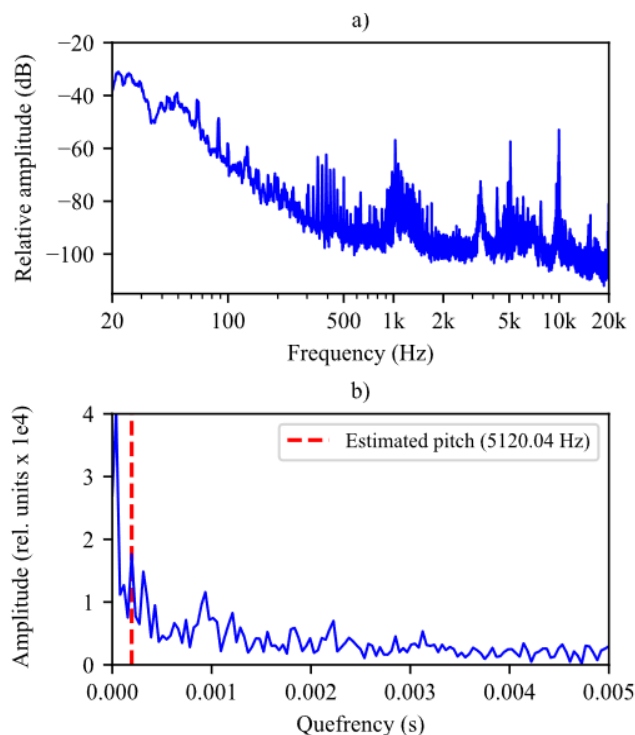
Слика 6.7. „Спектар а) и цепструм б) звука пумпе мотора на 4000 грт“ [80].



Слика 6.8. „Спектар а) и цепструм б) звука пумпе мотора на 2400 грт“ [80].

тона 2756,25 Hz [80]. Након промене броја обртаја на 1300, откривена висина се мења на 5120,04 Hz. Слика 6.9 приказује спектар и цепструм за овај одређени број

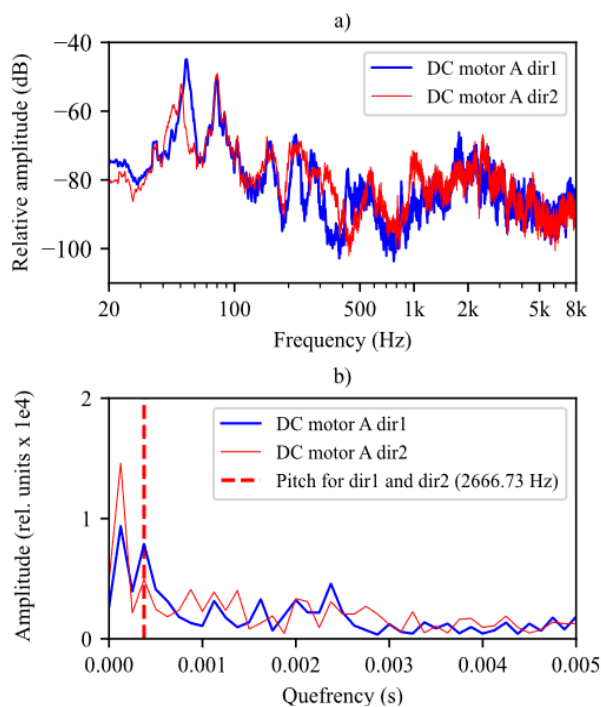
обртаја. Са наведених слика примећује се да се релевантни максимум цепструма мења. Поред тога, и његов образац се трансформише са варијацијом ротације. Стога би било разумно увести бар један додатни атрибут који може обухватити разноликост узорка цепструма. Овај атрибут може бити повезан са распадом омотача цепструма, са енергијом цепструма унутар одређених граница q -фреквенције или са вектором који обухвата вредности цепструма у оквиру унапред дефинисаних граница q -фреквенције [80].



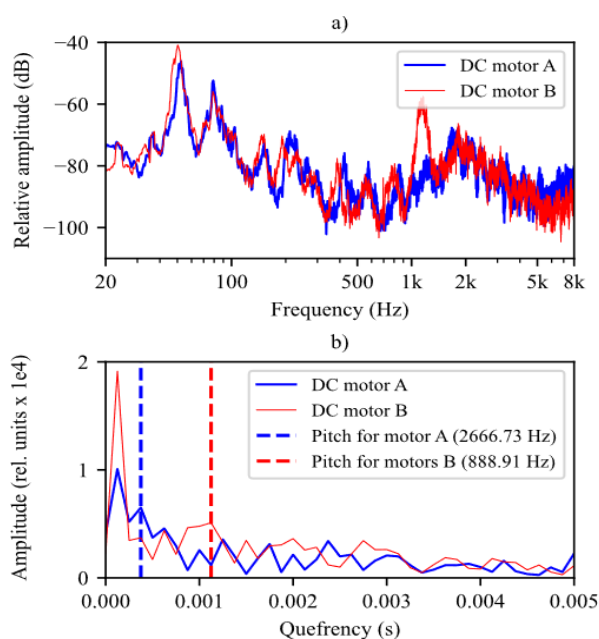
Слика 6.9. „Спектар а) и цепструм б) звука пумпе мотора на 1300 грм“ [80].

Следећа фаза испитивања обухватала је моторе. Слика 6.10 илуструје спектар и цепструм једносмерног мотора типа А у два контрастна смера ротације. Анализа спектра открива да општи образац остаје конзистентан за оба смера ротације. Поред тога, обрасци цепструма показују одређене сличности, иако постоје разлике између различитих праваца ротације. Процењена висина тона за оба смера је идентична (2666,33 Hz), пошто се највећа вредност у цепструму јавља на истој фреквенцији. Слика 6.11 илуструје поређење између спектра и „цепстра“ звукова два различита мотора пумпи означена као типови А и Б. Сlike 6.10а и 6.11а приказују облике спектра, док слике 6.10б и 6.11б приказују цепструме. Ови визуелни прикази откривају сличности и разлике које произилазе из употребе различитих машина истог типа А за анализу. Приметно је да процењени корак за два мотора типа А остаје константан (2666,73 Hz), док мотор типа Б има процењени нагиб од

888,91 Hz. Када се испитују детектоване вредности корака за моторе са и без кvara, варијације између мотора су зависне од специфичног квара. У одређеним случајевима, квар изазива промену у периодичности или псеудопериодичности звучног таласа, што последично утиче на процењену висину тона [80]. Слика 6.12

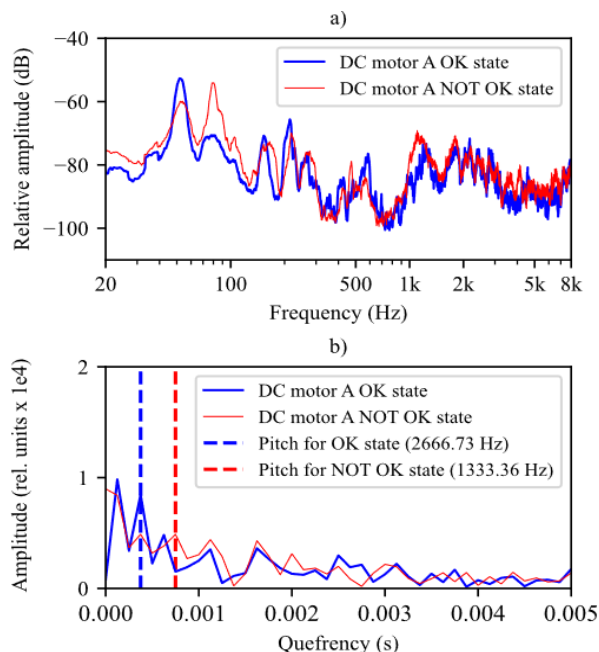


Слика 6.10. „Спектар а) и цепструм б) мотора А у два супротна смера ротације“ [80].



Слика 6.11. „Спектар а) и цепструм б) различитих мотора једносмерне струје (мотори типа А и Б) и смер ротације 1“ [80].

илуструје један од таквих случајева који представља спектре и “цепстре” мотора без квара (MotorOK) и са отказом (MotorNOK). Процењени корак за MotorOK је 2666,73 Hz, а за MotorNOK је 1333,36 Hz. Примећујемо да су, у овом случају, разлике између спектра и цепстра веће у односу на претходно анализирана два случаја мотора [80].



Слика 6.12. „Спектар а) и цепструм б) мотора без отказа (MotorOK) и мотора са кваром (MotorNOK)“ [80].

6.3. Методе анализе звука применом мере сличности слике

6.3.1. Уводна разматрања

Обрада слика је привукла велику пажњу у последњих неколико деценија. Ова техника обухвата различите задатке, међу којима се издваја процена квалитета или деградације слике, односно сличности слике [81]-[83]. Током овог периода развијен је низ алгоритама за поређење слика [84]. Сличност, генерално, представља степен огледања у односу на везу између две компоненте, односно две слике [85]. Сличност игра виталну улогу у различитим областима, као што су метеорологија и климатологија [86], медицинска дијагностика [87], аутономна возња, обрада говора и анализа звука [88], итд. Конкретно, анализа сличности слика се примењује за решавање проблема као што су као детекција објеката и упаривање слика [89], [90], препознавање лица, претрага слика, компресија и процена квалитета слике, анализа структуре и текстуре слике [91], синтеза и реконструкција слике [92], праћење објеката [93], итд.

Процена сличности слика је сложен подухват који захтева примену различитих метода за поређење слика. Ове методе се ослањају на различите метрике или мере сличности слика, које су познате као функције удаљености или мере изобличења. Ове метрике служе за квантификацију нивоа сличности између две слике. Неке од ових мера укључују коришћење односа пик сигнал-шум (енг. *peak-signal-to-nois ratio-PSNR*), Еуклидско растојање (енг. *Euclid distance - ED*), ED на квадрат (L_2 норма) и средња апсолутна грешка (енг. *Mean Absolute Error - MAE*; L_1 норма), меру индекса структурне сличности (енг. *Structural Similarity Index Measure - SSIM*), меру индекса сличности карактеристика (енг. *Feature Similarity Index Measure - FSIM*) [81]-[83], [85], [90], [93], [94], итд. Неке од наведених мера су једноставне мере ниског нивоа, на пример, L_1 и L_2 норме (удаљености), које имају ограничен капацитет да схвате дубоку конотацију слике, као што је препознавање објекта на основу њиховог семантичког садржаја (људи, цвеће, итд). С друге стране, неке друге мере, као што су оне засноване на дистрибуцији вероватноће интегритета (заједничке и маргиналне ентропије), прилично су сложене [95].

Захтеви за перформансе специфичне апликације диктирају захтеве перформанси за ове мере сличности слика. Дакле, важно својство компресије слике је монотонички однос између измереног и уоченог изобличења, док у проналажењу слике способност разликовања сличних и различитих слика може бити довољна. У препознавању слике пожељно је користити меру која је отпорнија на померање и ротацију, док је у регистрацији и праћењу пожељно користити меру која показује бољу локализацију и толеранцију на шум [96].

Уназад неколико година, методе класификације, препознавања и детекције (енг. *Classification, Recognition and Detection - CRD*) засноване на звуку и вибрацијама постале све раширеније. Ове методе се могу применити у бројним применама, као што је предвиђање преосталог века употребе лежаја [97], откривање кварова у турбовентилаторском мотору, праћење редуктора, идентификација оштећених лежајева у високо софистицираним мотора као што су авионски мотори [98] и аутономно акустичко праћење процеса и догађаја у производном погону [99]. У поређењу са сликовним CRD методама, које су веома истражене, аудио CRD методе су знатно мање зреле [100]. У аудио CRD методама могу се идентификовати три општа приступа: једна која се ослања на ручно дизајниране карактеристике, друга која користи приступ заснован на слици и трећа која користи сирови аудио сигнал. Други приступ је добио значајно прихватање, посебно у светлу брзог напретка у технологији DNN-а (енг. *Deep Neural Network*), посебно CNN и њихових побољшаних могућности, заједно са повећаном доступношћу јединица за обраду. На овај начин, побољшање перформанси од обраде слике може се потенци-

јално применити на аудио сигнале, иако високе перформансе DNN-а или CNN-а још увек нису достигнуте у аудио пољу [100].

Аудио сигнали могу бити визуелно представљени на више начина. Међутим, критички преглед сродних техника представљања сигнала недостаје у литератури [101]. Звук се обично визуелно приказује испитивањем флукуација у фреквентном спектру током времена, често коришћењем спектрограма. Коришћење класичног спектрограма пуне резолуције омогућава идентификацију различитих карактеристика унутар аудио сигнала. Међутим, важно је истаћи да рачунски трошкови CNN-а ескалирају са већим бројем тачака података. Ово се може превазићи коришћењем група фреквенцијских филтера, као што су покретни просек и мел-филтери, који се могу користити за израчунавање енергије у одређене фреквентне опсеге за смањење броја тачака на датој фреквенцији [101].

Аудио и вибрациони сигнали поседују различите обрасце који се могу ефикасно превести у одговарајуће слике. Сходно томе, технике које се користе у обради слике, као што је анализа сличности слика, такође се могу користити за анализу аудио и вибрационих сигнала. Дакле, матрица копојављивања оријентисана на анализу текстуре слике се користи за препознавање звука у аудио надзору, где се централни моменти екстраховани из слике спектрограма користе као карактеристике [102]. Комбиновањем ненадзиране анализе сличности, посебно косинусне сличности, са CNN-ом постаје могуће идентификовати и пратити индустријске процесе анализом акустичких потписа спектрограма. Спектар снаге централних коефицијената мел-фреквенције (енг. *Mel-frequency Cepstral Coefficients* - MFCC) је комбинован са сликом спектрограма за праћење емпатије у рекламама у реалном времену [103]. Квалитет аудио сигнала након компресије се процењује помоћу мере структурне сличности као што је средња структурна сличност (енг. *Mean Structural Similarity* - MSSIM), која се примењује на фиксне оквире временског домена аудио секвенце и на 2Д временско-фреквентне мапе [104]. Праћење стања и дијагностика кварова система лежајева се спроводе упоређивањем спектралних слика података о нормалним вибрацијама са тестираним подацима помоћу SSIM-а, као што је изведено у [105], или коришћењем PSNR-а, као што је изведено у [98]. У другим студијама, дијагностика квара лежајева се такође врши коришћењем спектрограма и спектра података о вибрацијама, као што је наведено у [98].

Бројне представљене студије и случајеви употребе реализовани у пракси показали су значајне могућности у погледу коришћења звука за различите задатке, као што су класификација индустријских производа или откривање кварова производа. Да би се искористиле предности обраде слике, почетни корак укључује ма-

пирање звука на одговарајуће слике. Након тога, ове слике се користе за даљу обраду, као што је примећено у DNN или CNN класификаторима. Тренутно не постоји стандардизована метода за повезивање звука са сликом, што доводи до примене различитих алтернатива. Штавише, постоји недостатак успостављених техника за процену информација које слика преноси. Циљ овог истраживања јесте да се поменути јаз превазиђе коришћењем три мере сличности слике: Еуклидске удаљености (ED), Пирсоновог коефицијента корелације (PCC) и индекса структуралне сличности (SSIM). Специфични циљ је да се истражи да ли уобичајено коришћена репрезентација звука, тј. мел-спектрограм, садржи адекватне дескриптивне информације за разликовање класа кроз процену сличности слике. Мере сличности слика које се овде користе су оне које се често користе у пракси (на пример [101], [106], [107]), и које се односе на различите аспекте сличности слика (растојање, корелатив и структурне карактеристике). Штавише, циљ је пружити увид у значај ових метрика сличности у процени сличности између слика које поседују различите атрибуте и оних природних слика.

Обухватајући звукове које производе индустријски производи, истраживање се спроводи у седам специфичних категорија: вентилатори, мењачи, пумпе, клизне шине, аутићи (означени као ToyCar), играчке-возови (означени као ToyTrain) и вентили. Ови звуци су доступни као скупови података у оквиру изазова „Детекција и класификација акустичних сцена и догађаја“ (енг. *Detection and Classification of Acoustic Scenes and Events - DCASE*) [108]-[110]. Звуци се прво мапирају у слике мел-спектрограма. Након тога, три горе поменуте мере сличности слика се израчунавају за сваку од слика. Ове мере се, затим, користе за испитивање звукова индустријских производа коришћењем приступа обраде слике. Анализа обухвата коришћење матрица сличности и топлотних мапа, уз спроведену додатну статистичку анализу. Примарни нагласак је стављен на истраживање сличности унутар класе и између различитих класа, са циљем да се илуструје степен сличности између звукова који припадају истој класи и оних из различитих класа.

Јединственост овог истраживања може се описати на следећи начин:

1. Мел-спектрограми садрже информације које омогућавају процену сличности звукова које производе индустријски производи. Анализом слика генерисаних из ових спектрограма може се утврдити да ли одређена слика ефективно представља јединствене карактеристике одређеног звука.

2. Слике мел-спектрограма се анализирају како би се испитале сличности унутар и међу класама, чинећи основу за креирање параметарског класификатора.

3. Процењиване су перформансе три често коришћене мере сличности слика за процену сличности слика са својствима која се разликују од природних слика,

као што су својства која се тичу различитих значења оса (време и фреквенција, тј. ширина и висина), транспарентност слике и нелокално распоређени објекти [111].

6.3.2. Мере за процену сличности слика

У практичним применама коришћене су разне методе за процену сличности и квалитета слике. Ове методе обухватају широк опсег, почевши од конвенционалних мера као што су MSE и $PSNR$, које процењују сличност слике на бази пиксел по пиксел, до напреднијих мера као што су метрика квалитета перцепције. Постоје и неке друге мере које припадају статистичким методама (израчунавање статистике слике) и спектралним методама (користи се Фуријеова трансформација или декомпозиција подопсега) [91]. Такође, одређене технике се фокусирају на процену структурне сличности узимајући у обзир напредне карактеристике људског визуелног система. Употреба основних мера разлике тачка по тачка је опала и замењена је мерама заснованим на теорији информација, као што је узајамно информисање (енг. *Mutual Information* - MI). MI служи као свеобухватна мера информационог садржаја, нудећи већу флексибилност [112]. Нажалост, ове предности долазе на рачун повећаних рачунарских захтева. Ипак, треба напоменути да је најстарија и најједноставнија мера за процену сличности слика MSE [113], [114]. Прорачун се може добити коришћењем једнодимензионалних (1Д) или дводимензионалних (2Д) података, при чему се први користе на следећи начин:

$$MSE = \frac{1}{N} \sum_{i=1}^N (x_i - y_i)^2 \quad (6.6)$$

где су x_i и y_i 1Д подаци, док је N број пиксела ових слика. MSE , нажалост, не разматра ни распоред пиксела слике (просторну дистрибуцију), нити њихову корелацију [114]. Штавише, сви пиксели слике имају исти значај у погледу израчунавања MSE -а. Поред тога, на средњу квадратну грешку (MSE) значајно утиче просторна просечна вредност слика које се пореде. Ова карактеристика представља значајне недостатке за задатке као што је процена квалитета слике, где распоред и састав пиксела директно утичу на укупан квалитет слике због високог нивоа структуре присутне у сликама. Вреди напоменути да се MSE такође може израчунати из дводимензионалних података користећи следећи приступ:

$$MSE_{2D} = \frac{1}{W \cdot H} \sum_{i=1}^W \sum_{j=1}^H (x_{i,j} - y_{i,j})^2 \quad (6.7)$$

Димензије 2Д података, као што је слика, могу бити представљене са W и H , што указује на број редова и колона, респективно. Ово мерење нуди апроксимацију апсолутне сличности и такође је рачунарски ефикасна. Друга мера блиско повезана са средњом квадратном грешком (MSE) је Еуклидска удаљеност (ED), која

израчунава растојање између одговарајућих тачака два 1Д вектора, x и y , у еуклидском простору као

$$ED = \sqrt{\sum_{i=1}^N (x_i - y_i)^2} = \sqrt{N \cdot MSE} \quad (6.8)$$

Једна од најчешће коришћених мера корелације је PCC, која одражава степен линеарне везе између две варијабле [115]. PCC је статистички индекс и може се израчунати на следећи начин [116]:

$$PCC = \frac{\sum_{i=1}^N (x_i - \mu_x)(y_i - \mu_y)}{\sqrt{\sum_{i=1}^N (x_i - \mu_x)^2 \sum_{i=1}^N (y_i - \mu_y)^2}} = \frac{\sigma_{xy}}{\sigma_x \sigma_y} \quad (6.9)$$

У датом сценарију, вредности i -тог пиксела на сликама x и y , означене као x_i и y_i респективно, се трансформишу у 1Д векторе дужине N . Средње вредности слика x и y су представљене са μ_x и μ_y , респективно. Коваријанса између x и y је означена као σ_{xy} , док су стандардне девијације x и y представљене са σ_x и σ_y , респективно. Коефицијент PCC, дефинисан у једначини (6.9), означава скаларно мерење глобалне сличности између поређених слика. Његове вредности се крећу од 1 до -1, што указује на савршену позитивност (највећа сличност или савршено подударење) и негативну линеарну везу. Ова мера се обично користи за подударење шаблона и препознавање образаца. У поређењу са другим мерама сличности као што су MSE и MAE, PCC је мање осетљив на промене у осветљењу. Погодно је користити PCC када се промене у осветљењу могу апроксимирати линеарном трансформацијом. Међутим, ако то није случај, препоручује се да се размотри употреба MI или друге мере ординалне сличности као боље алтернативе.

6.3.3. Структурна сличност

Својство високе структуре је значајна карактеристика природних слика, где просторно суседни пиксели показују значајан степен корелације. Ово својство се може искористити за добијање значајних информација о распореду објеката унутар визуелне сцене. Штавише, сличност у структури између слика може послужити као поуздана апроксимација њихове перципиране сличности. SSIM индекс се појављује као најраспрострањенија мера за квантификацију структуре слике. Када се користи за процену сличности између две слике, овај задатак се даље дели на три подзadatка: упоређивање осветљености, контраста и структуре. Током овог поређења, локална статистика и SSIM индекс се израчунавају унутар локалног прозора. У суштини, SSIM вредност се добија за сваки прозор слике коришћењем технике клизног прозора која систематски анализира слику пиксел по пиксел. Ова

карактеристика SSIM мере показује предност, јер признаје неуједначена статистичка својства слике у различитим регионима. Поред тога, с обзиром на аспект сложености, извођење SSIM израчунавања на сегментима података показује се као предност. Штавише, овај приступ омогућава неједнако пондерисање сегмената.

Осветљеност закрпа се може одредити израчунавањем просечног интензитета пиксела унутар сваког простора. Ова мера даје индикацију укупне светлости или таме региона слике представљених са p и r :

$$l(p, r) = (2\mu_p\mu_r + C_1) / (\mu_p^2 + \mu_r^2 + C_1) \quad (6.10)$$

$$c(p, r) = (2\sigma_p\sigma_r + C_2) / (\sigma_p^2 + \sigma_r^2 + C_2) \quad (6.11)$$

$$s(p, r) = (\sigma_{pr} + C_3) / (\sigma_p\sigma_r + C_3) \quad (6.12)$$

У датом контексту, μ_p и μ_r симболизују средње вредности пиксела унутар локалних прозора p и r ; респективно. Слично, σ_p и σ_r представљају стандардну девијацију вредности пиксела унутар ових локалних прозора. Коваријанса (унакрсна корелација) између p и r прозора је означена са σ_{pr} . Да би се спречиле нестабилности када се суме у имениоцима претходне три једначине приближавају нули, уводе се константе C_1 , C_2 и C_3 . Величина локалног прозора може бити 11×11 , као што је коришћено у [92]. Када је у питању примена SSIM-а на аудио сигнале, поређење осветљености није посебно релевантно због чињенице да дуже аудио секвенце обично имају средњу вредност нула. Као резултат, подаци осветљености и контрасту су одвојени од структурних информација, али се све ове компоненте комбинују да би се утврдила сличност слика. Овде су структурне информације везане за облик објеката [114]. Даље, осим константе C_3 , термин структуре означава коефицијент унакрсне корелације између два локална прозора. Термини осветљености и контраста, који су подешени средњим вредностима и вредностима варијансе, крећу се од 0 до 1. С друге стране, термин структуре спада у интервал од -1 до 1 . SSIM индекс се постиже комбиновањем ова три термина на следећи начин:

$$SSIM(p, r) = [l(p, r)]^\alpha [c(p, r)]^\beta [s(p, r)]^\gamma \quad (6.13)$$

Променљиве α , β и γ су параметри који одређују релативну важност сваке компоненте и често им се додељује вредност 1. SSIM креће се од 0 до 1, где вредност 0 означава потпуно несличност између слике, а вредност 1 означава да су слике идентичне. Укупан SSIM индекс, који процењује глобалну сличност слика на свим просторним локацијама, добија се израчунавањем средње вредности SSIM-а у локалним прозорима. У литератури се наводи да SSIM објективно предвиђа субјективну оцену квалитета слике [91], осим код слика са шумом и замућења, као и

код слика близу прага препознавања [86]. SSIM индекс помера сличност слике и процену квалитета са нивоа заснованог на пикселима на фазу засновану на структури [117]. Област обраде слика се често ослања на метрику засновану на SSIM-у. Као резултат тога, SSIM је постао широко коришћен алат у различитим студијама као незванични стандард за процену квалитета слике, посебно када је присутна референтна слика, као што је у домену препознавања слике.

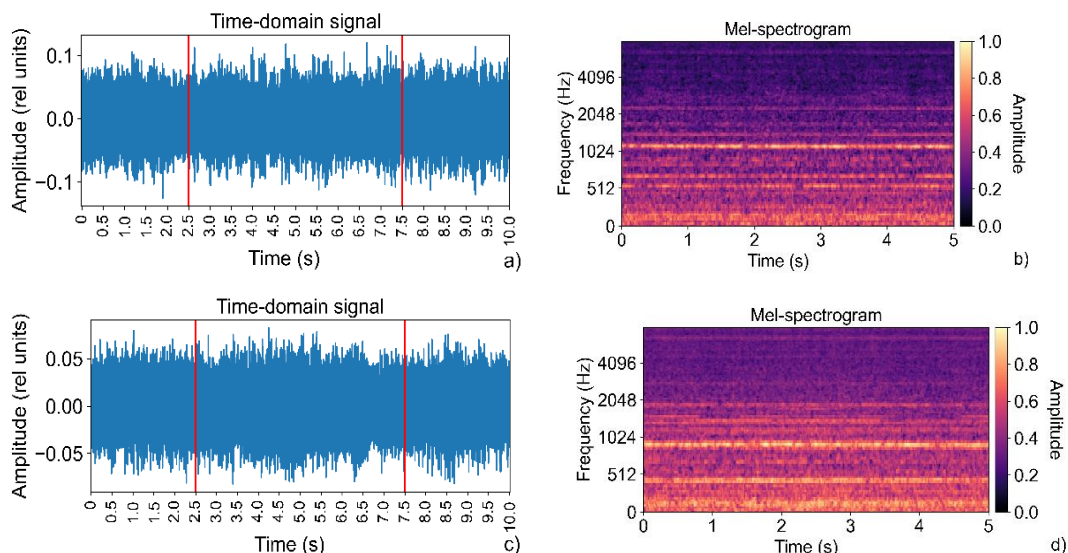
6.4. Тестирање

6.4.1. Скуп података

Звучни узорци (аудио сигнали) који су коришћени у овом истраживању узети су из [108]. Да би се решио изазов који је постављен, скуп података је састављен спајањем информација које потичу из два различита извора: MIMII DUE [109] и TOYADMOS2 [110]. Прикупљени скуп података обухвата низ звукова који се генеришу током рада седам типова машина (индустријских производа): вентилатора, мењача, пумпе, клизача, аутомобили, воза и вентила. Процедура снимања користила је доследан приступ за снимање звукова са свих седам типова машина. Ова метода је обезбедила да добијени узорци буду уједначени и да се могу лако упоредити током накнадне анализе. Примарни циљ генерисања овог скупа података био је да се истражи неисправност индустријских машина олакшају инспекције које укључују промену између изворног и циљног домена. Ова промена настаје услед промена у условима рада и животне средине. Изворни домен представља почетно стање које карактерише велики број звучних узорака доступних за обуку. Насупрот томе, циљни домен одговара измењеном стању, где је само ограничен број звучних узорака био доступан за обуку. Разлике између изворног и циљног домена, познате као померање домена, обухватају различите факторе као што су радна брзина, оптерећење машине, амбијентална бука, температура грејања, однос сигнал-шум SNR и још много тога. Вреди напоменути да се ова померања домена дешавају чак и када се користи исти тип машине, иако са различитим моделима машина и конфигурације делова, као и варијацијама у постављању микрофона. Скуп података који се помиње у овом опису је помно дизајниран да послужи за идентификацију аномалних звукова, посебно у сврху аутоматског откривања кварова у машинама.

Достављени звучни узорци су снимци поменутих машина у трајању од по 10 секунди. Ови снимци се чувају у “*wave*” формату и једноканалном 16-битном формату. Сем хватања звука циљне машине, узорци обухватају и звукове пратеће опреме и звукове околине. Снимљени сигнали имају фреквенцију узорковања од 16 kHz. Снимања су вршена или у безехогеној комори, обичним просторијама или

кабинама за звучну изолацију. Позадинска бука из различитих фабрика снимљена је одвојено, а мешана је са звуцима машина у фази постпроцесирања. Два примера звучних узорака у временском и фреквентном домену приказана су на слици 6.13.



Слика 6.13. Звукови два насумично одабрана узорка из [109], [110].

Снимљени звучни узорци су подељени у три скупа података: скупови података за развој, скуп података за додатну обуку и скуп података евалуацију. Развојни скуп података састоји се од три секције, од којих свака садржи 1003 звучна узорка за сваку машину. Сваки одељак приближно одговара једном производу. Унутар сваке секције укључени су само нормални оперативни аудио узорци из два домена, који се састоје од 1000 узорака из изворног домена и 3 узорка из циљног домена. Ови узорци показују различите услове, као што су промене у радној брзини и амбијенталној буци. Дакле, укупно 3009 звучних узорака је доступно по машини. У овој студији користи се овај скуп развојних података који укључује узорке звука и из изворног и из циљног домена, и који ради у нормалном радном режиму и означава се као комплетан скуп података. Издвојени скуп података, који је мањи подскуп који користи у овом истраживању, креиран је по истој методологији као и цео скуп података. Међутим, скуп података састоји се само од 300 звучних узорака за сваку машину. Штавише, екстраховани скуп података укључује искључиво звучне узорке из одговарајућег одељка.

6.4.2. Прорачун и анализа мера сличности слика

Анализа сличности слика (у овом случају мел-спектрограми) врши се помоћу метрика сличности: ED, PCC и SSIM. ED и PCC се израчунавају применом израза (6.8) и (6.9), док се SSIM израчунава применом израза (6.10) до (6.13). Програмски

језик Python се користи за спровођење комплетне процедуре обраде, која обухвата обраду аудио сигнала, као и израчунавање и процену мера сличности слика. Да би се измерио ниво сличности унутар одређене класе, било да се ради о индустријском производу или одређеној врсти машине (нпр. вентилатор), мера сличности се одређује за све могуће комбинације (парове) слика које припадају тој специфичној класи (производу). С обзиром да редослед слика не утиче на израчунавање три мере сличности које се користе у овом контексту, све комбинације без икаквих пермутација су укључене у анализу. Да би се истражила сличност између различитих класа, израчунавају се три различите мере сличности. Ово укључује одабир једне слике из једне класе и друге слике из друге класе и израчунавање сличности између њих. У овим прорачунима се разматрају све могуће комбинације парова слика, без разматрања пермутација. Сходно томе, генерише се матрица за сваку специфичну класу и меру сличности, која представља сличност унутар класе. С друге стране, за међукласну сличност, постоји шест матрица за сваку класу и меру сличности, која обухвата поређења између слика.

Да би се стекао даљи увид у матрице сличности унутар разреда и међу класама, генеришу се топлотне мапе и спроводи се статистичка анализа. Средња вредност, минимум, максимум, медијана и стандардна девијација се затим израчунавају из ових матрица као уобичајене статистичке мере. Ове израчунате вредности играју кључну улогу у поређењу три одабране мере сличности које се истражују у овој студији.

6.4.3. Резултати

Описани поступак је укључивао израчунавање три различите мере сличности слика (ED, PCC и SSIM) за велики број парова узорака. Ови парови су се састојали од мел-спектрограма машина које су припадале седам наведених класа. Добивене вредности сличности су затим организоване у квадратне матрице, са димензијама од 3009 x 3009 или 300 x 300, у зависности од величине скупа података који се користи (пун или екстрахован). Важно је напоменути да редослед којим су узорци распоређени не утиче на израчунате вредности сличности за било коју од три мере. Сходно томе, ове матрице показују симетрију у односу на главну дијагоналу. Испитивањем било ког реда или колоне ових матрица, могу се уочити вредности сличности између специфичног узорка (машине) и свих других узорака из исте класе (унутаркласна сличност) или из различитих класа (међукласна сличност). Вредности на главној дијагонали представљају самосличност сваког узорка, са вредношћу од 0 за ED или 1 за PCC и SSIM.

Налази који се односе на унутаркласну сличност представљени су кроз серију од 21 матрице за комплетан скуп података, као и једнак број матрица за екстра-

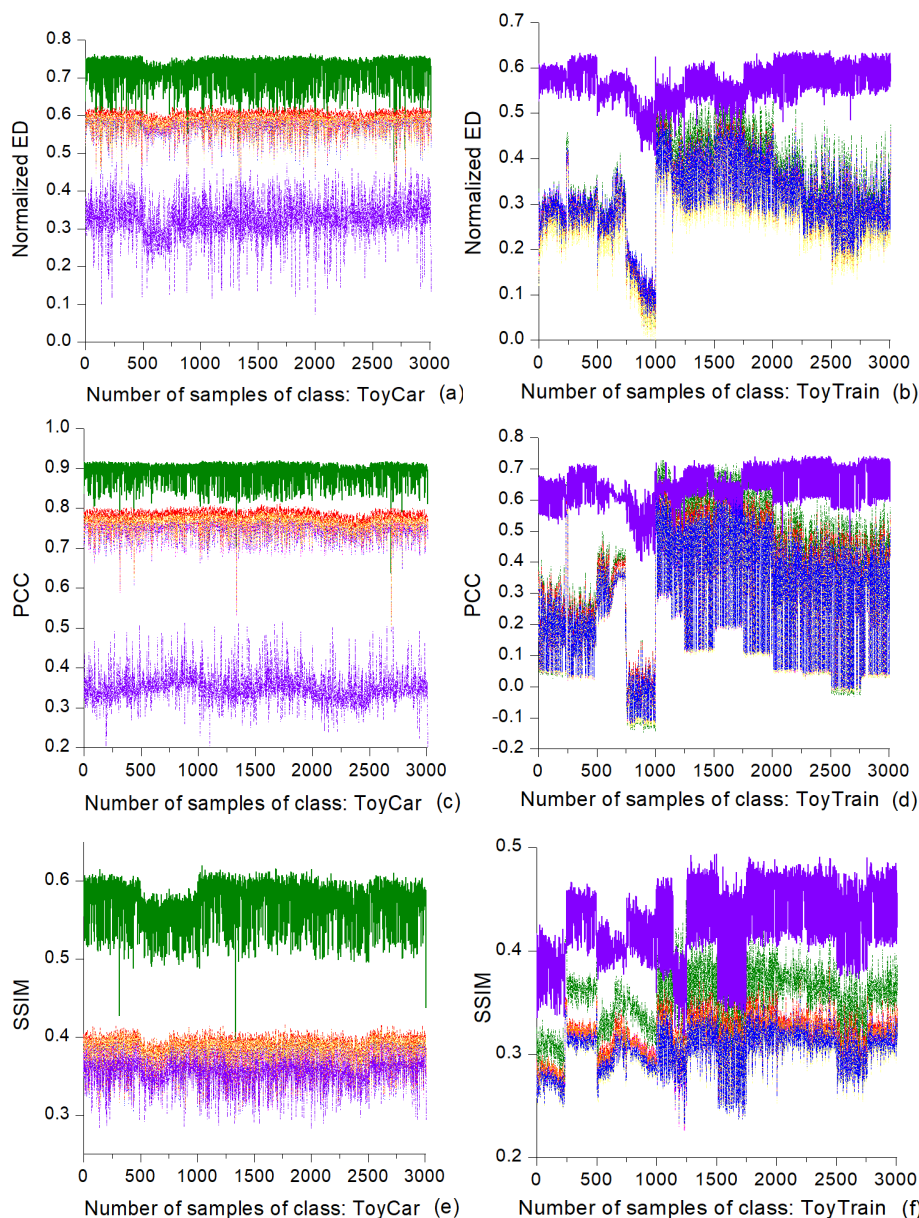
ховани скуп података. Унутар ових матрица, свака мера сличности је представљена са седам матрица, по једна за сваки тип машине или класу, за оба скупа података. Насупрот томе, међукласна сличност је приказана у већем броју матрица у поређењу са сличношћу унутар класа. Као резултат, за сваки скуп података постоји укупно 3 x 21 матрица (узимајући у обзир све могуће комбинације узорака, али и занемарујући њихове пермутације). Добијене вредности сличности су детаљно анализиране из различитих перспектива, док су резултати представљени на начин који одговара напредовању од општег прегледа, па до сложеније анализе.

6.4.4. Анализа на нивоу мера сличности слике усредњених по узорку

На овом детаљнијем нивоу анализе, фокус је на средњој вредности мера сличности израчунатих за сваки појединачни узорак (машину) у матрицама сличности. Ове матрице представљају вредности сличности између специфичног узорака и свих других узорака из исте или различитих класа. Вредности сличности за сваки узорак су организоване у редове или колоне повезане са тим узорком. Да би се израчунала вредност узорака одређује се просек свих вредности унутар одређеног реда или колоне. Овај процес генерише вектор са 3009 елемената, који представљају средње вредности по узорку за унутаркласну и међукласну сличност. Испитивањем ових средњих вредности, може се стећи даљи увид у понашање сваког узорака, јер оне пружају детаљније разумевање изван општих трендова на које указују саме средње вредности.

Слика 6.14 приказује графиконе који приказују средње вредности по узорку за две различите категорије, аутомобил и воз, заједно са све три мере сличности. Из графикона је видљиво да средња линија по узорку за унутаркласну сличност доследно показује веће вредности у поређењу са линијама сличности међу класама у свим илустрованим сценаријима. Фокусирајући се на нормализоване дијаграме ED линија за класу аутомобила играчке, они се могу категорисати у три групе на основу њихових вредности, распоређених у опадајућем редоследу. Прво, средње вредности унутар класе имају највишу позицију. Друго, средње вредности између класа за све могуће комбинације аутића са било којом класом из групе од пет класа заузимају другу позицију. Најзад, међукласна средства између класа аутомобил и воз-играчка су позиционирани на најнижем нивоу. Овај распоред је представљен на слици 6.14а.

Слика 6.14б илуструје разлику у средњим вредностима узорака међукласних нормализованих ED линија између класе воз и класе аутомобил. Конкретно, узорци средстава класе играчке-воз показују посебан образац. Овде се узорци класе возова играчака могу поделити на 12 подкласа, при чему је раздвајање подкласа

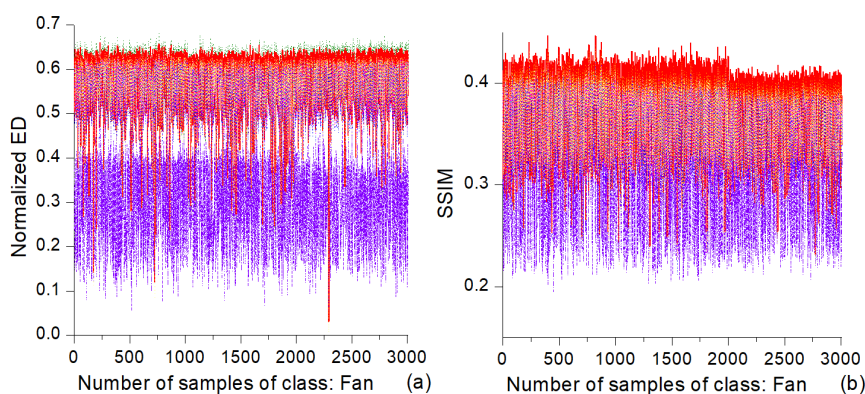


Слика 6.14. Дијаграми средњих линија по узорку за три различите мере сличности унутар разреда (пуна линија) и између класа (испрекидане линије) за цео скуп података: (а) ED нормализован за класу аутомобила-играчака, (б) ED нормализован за класу возова играчака, (ц) PCC за класу аутомобила играчака, (д) PCC за класу возова играчака, (е) SSIM за класу аутомобила играчке и (ф) SSIM за класу возова играчака (легенда је иста за све графичке податке) [118].

представљено постепеним обликом линијских дијаграма унутар разреда и међу разредима [118]. Упоредивањем слике 6.14а–д, која представља нормализоване резултате ED и PCC, постаје очигледно да они показују висок степен сличности, што указује да су примарни обрасци конзистентни. Међутим, постоје неке мање разлике које вреди напоменути. Прво, релативни односи одређених линија унутар класе и међукласних линија нису прецизно усклађени. Друго, ширење вредности

линија међу класама за класу воз-играчка је уже у нормализованом ED (слика 6.14б) у поређењу са PCC (слика 6.14д).

Исходи за нормализовани ED и PCC одржавају главне трендове уочене у средњим вредностима $SSIM$ по узорку. Сходно томе, класе аутића и возова показују највеће средње вредности по узорку у смислу сличности унутар класа. Поред тога, класа играчка-воз показује различито понашање манифестујући се као распоређени образац и унутаркласних и међукласних линија повезаних са различитим подкласама. Важно је напоменути да се релативни односи вредности сличности за ове ове подкласе разликују у различитим мерама сличности. Линије приказане на слици 6.14 показују одређени степен преклапања, посебно унутар групе од пет класа. Као резултат тога, постаје тешко разазнати сваку линију појединачно. Међутим, ово није примарни фокус овде. Главна намера је да се илуструје образац и дистрибуција вредности сличности за средње вредности узорака на различитим нивоима. Ови нивои обухватају (а) све линије које представљају или унутаркласну или међукласну сличност, (б) све линије које одговарају одређеној класи и (ц) све линије повезане са одређеном мером сличности.



Слика 6.15. Дијаграми просечних линија по узорку за унутаркласну (пуна линија) и међукласну сличност (испрекидане линије) за комплетан скуп података и узорке вентилатора: (а) нормализовани ED и (б) $SSIM$ [118].

Преклапање линијских графикана је још веће у класама које припадају групи од пет класа (слика 6.15). Средње вредности сличности по узорку у појединачним линијама за узорке вентилатора показују веће ширење у поређењу са узорцима колица и воза, у односу на све три мере сличности. Овај диспаритет у ширењу објашњава релативно мање уочљиву разлику између линија које припадају различитим класама. Приметно је да међукласна линија за класу аутомобила играчака показује највеће вредности, мало надмашујући линију унутар класе. Вредности унутарразредне линије сличне су међуразредним линијама у свим класама унутар групе од пет класа. Супротно томе, линија међу класама за класу играчка-воз при-

казује најмање вредности. Када се упореде резултати за нормализовани ED и SSIM, постаје евидентно да примарни трендови остају доследни, иако постоје варијације у опсегу вредности за појединачне линије и њихове односе.

6.5. Класификација звука мотора пумпи помоћу DNN-а

6.5.1. Уводна разматрања

Иако су многи алгоритми дубоког учења позајмљени из обраде слика и примењени у различитим доменама, кључно је признати фундаменталне разлике између ових домена. Ово признање наглашава потребу за специјализованом перспективом на аудију, што резултира развојем аудио-специфичних решења. Баш као и у обради слике, аудио домен такође има користи од слагања вишеструких слојева унапред, конволуционих и повратних информација како би се побољшале могућности моделирања [119]-[134]. Избор специфичне архитектуре дубоке неуронске мреже (енг. *Deep Neural Network* - DNN) у вештачкој интелигенцији (енг. *Artificial Intelligence* - AI) зависи од конкретног задатка. Дакле, најчешће коришћени DNN за класификацију аудио сцене је CNN [120], док алгоритми за детекцију аудио догађаја углавном користе конволуциону рекурентну неуронску мрежу.

Што се тиче оптимизације структуре мреже, она се обично ради на ручни начин [121]. Коришћењем грешке валидације као параметра, CNN архитектура се бира експериментисањем. Овај приступ пружа смернице за процес пројектовања, укључујући коришћење мањег броја параметара за мање скупове података и прилагођавање броја канала и величина мапе обележја у наредним конволуционим слојевима. Главни циљ је да се предвиди радно стање тестираног мотора, односно разврставање у унапред дефинисане класе: исправни и неисправни мотори. Овај задатак се може решити применом различитих приступа укључујући и онај заснован на звуку [122], [134]. Обзиром да се DNN у великој мери користи у аудио домену за апликације дубоког учења, он је пожељан избор у овом конкретном случају. Да би се испунио овај захтев, креиран је прилагођени CNN да прецизно класификује моторе на основу њихових јединствених карактеристика звука [134].

6.5.2. DNN и CNN

Са својом способношћу да ефикасно карактеришу и класификују аудио сигнале, DNN-ови нуде значајну робусност у аудио домену, захваљујући карактеристичним спектро-временским облицима аудио догађаја. Међутим, ослањање само на DNN, као црну кутију за обраду звука, је широко препознато као незадовољавајуће. Да би се развили најнапреднији модели и мрежне архитектуре, укључујући одређивање хиперпараметара, императив је свеобухватно разумевање како и

зашто DNN-ови уче из аудио сигнала и њихових репрезентација, као што су слике засноване на спектрограму [134].

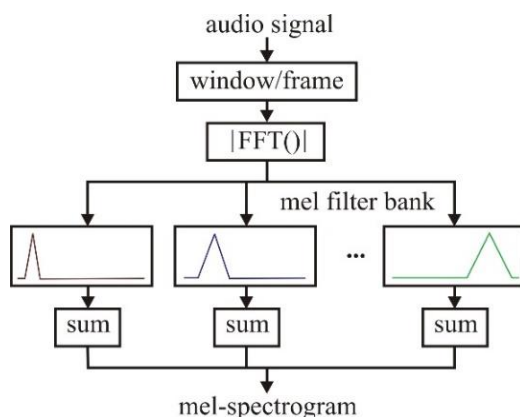
Једна од најчешће коришћених мрежа за обраду звука утемељену на вештачкој интелигенцији је CNN. У неким истраживањима о моделима дубоког учења из литературе, показано је да CNN надмашују друге моделе у обради слика и видео података [123]. Због тога CNN добија на популарности у обради звука заснованој на вештачкој интелигенцији [124]-[127] за коју се може сматрати да је неосетљива на позицију узорка на сликама заснованим на спектрограму [128] и препозната као погодна техника за класификацију карактеристика слике спектрограма. CNN могу да искористе инваријантност која постоји у спектрограмима за своје конволуционе и операције удруживања [129]. Поред тога, у стању су да постигну транслациону инваријантност и толеранцију на мање разлике у подацима у обрасцима [130].

Велики број параметара потпуно повезаних неуронских мрежа може се смањити применом CNN-а [131]. Оскудна повезаност и дељене тежине у CNN-има олакшавају оптимизацију мреже смањењем броја параметара и ризика од преоптерећења [132]. У CNN-у постоје два примарна слоја: конволуцијски и обједињавајући слојеви, при чему су завршни слојеви обично потпуно повезани слојеви. Када се разматра обједињавање слика заснованог на спектрограму, генерално се закључује да се ефекти разликују у зависности од тога да ли се спајање врши дуж фреквенцијске или временске осе. Показало се да перформансе обично деградирају када се врши обједињавање фреквенција [133], [134].

6.5.3. Мапирање аудио сигнала у слике

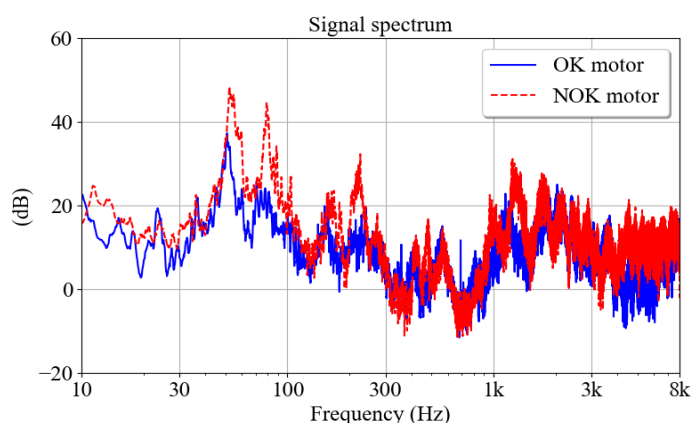
Процес класификације укључује коришћење звукова мотора који су снимљени унутар безехогеног сандука у кутији који се налази у постројењу за прераду и пречишћавање отпадних вода. Мерни микрофон је постављен на приближно 20 cm испод испитиваног мотора, који се налазио на свом постољу. Мотори су покретани у два различита смера ротације, а трајање рада у сваком смеру било је приближно 8 секунди [134]. Након снимања са фреквенцијом узорковања од 16 kHz, аудио сигнали пролазе кроз почетну фазу обраде. Ова фаза укључује издвајање средњег дела сваког правца ротације, док се искључују прелазни региони на почетку и на крају сигнала у трајању овог дела од 5 секунди. Екстраховани аудио сигнали се затим трансформишу у слику налик спектрограму. Да би се ово постигло, изабран је log-mel спектрограм, који је изведен из групе троугластих филтера који се преклапају и израчунавају спектралну енергију унутар сваког мел опсега (слика 6.16). Да би се добили мел-спектрограми аудио сигнала мотора, коришћене

су величине оквира од 1024 узорка и величине скокова од 256 узорака, заједно са 96 мел-опсега [134].



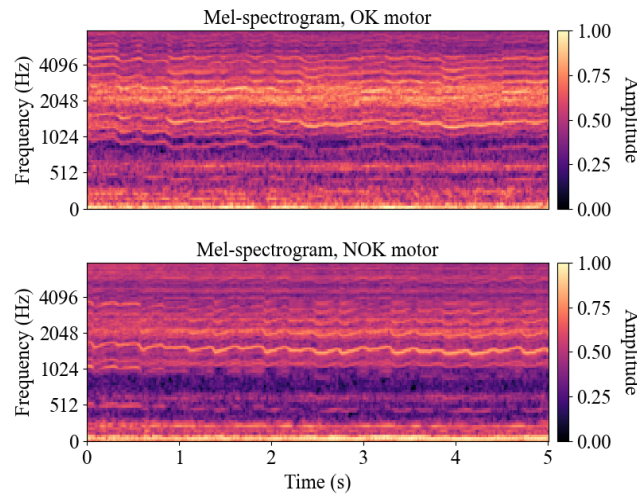
Слика 6.16. Добијање мел-спектрограма из аудио сигнала, где је брза Фуријеова трансформација означена са FFT.

Раздвајање између неисправних и исправних мотора извршило је искусно особље произвођача мотора слушајући звук мотора. Пре започињања процеса класификације, неопходно је анализирати аудио сигнале мотора у временском и фреквенцијском домену. У оптималном сценарију, јасан диспаритет између неисправног мотора (MotorNOK) и мотора који добро функционише (MotorOK) био би очигледан. Међутим, пошто су ови мотори потпуно нови, у већини случајева постоје само мале разлике међу њиховим звуковима. Аудио сигнали у временском домену ове две класе мотора запањујуће личе један на други. Исто тако њихови спектри показују сличности, иако са неким различитим катрактеристикама (слика 6.17) [134]. Дакле, постоје неки пикови, у спектру MotorNOK, који нису толико изражени у спектру MotorOK (нпр. они на око 80 Hz или 220 Hz).



Слика 6.17. Спектар насумично изабраног појединачног ОК и НОК моторног аудио сигнала у трајању од 5 секунди.

Превођење својства ових сигнала из временског и фреквентног домена су у домен мел-спектрограма илустровано је на слици 6.18. Међутим, ова репрезентација такође обухвата и друге значајне карактеристике аудио сигнала, као што је флукуација појединих компоненти фреквенције током времена. Поред тога, види се да MotorNOK има нешто слабије компоненте у опсегу између 300 Hz и 900 Hz, односно да MotorOK има наглашеније компоненте између 2 kHz и 2,6 kHz [134].

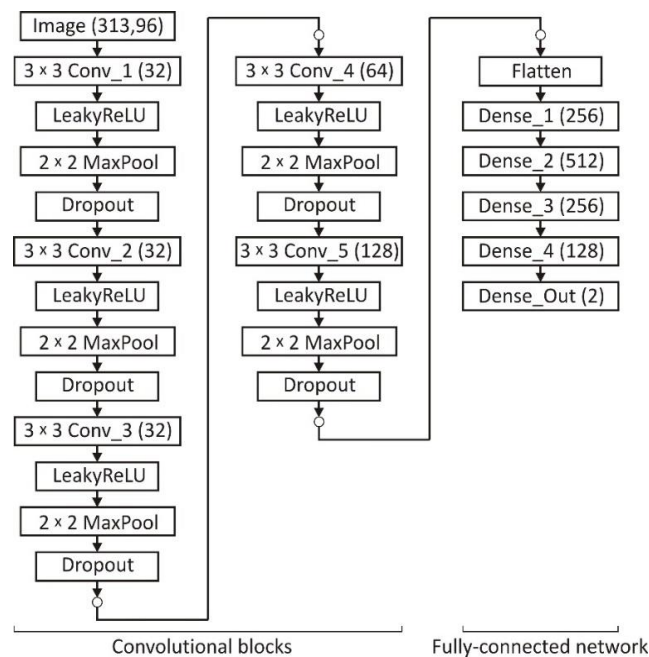


Слика 6.18. Лог-мел спектрограм насумично одабраног појединачног ОК и NOK аудио сигнала мотора у трајању од 5 секунди.

6.5.4. Развијена CNN

Да би се мотори класификовали у две категорије (MotorOK и MotorNOK), пажљиво је развијена и имплементирана CNN са секвенцијалним процесним цевоводом. Овај CNN, конструисан као линеарни сноп слојева, програмиран је помоћу језика Python. Мрежа се састоји од конволуционих, активационих, обједињених и испадајућих слојева у конволуционом блоку мреже који прати потпуно повезана мрежа (слика 6.19) [134]. На слици 6.20 приказан је свеобухватан преглед конволуционих мрежа, максималног обједињавања, изравнавања и потпуно повезаних слојева у развијеном CNN-у, укључујући придружене бројеве параметара. Оваква мрежна архитектура је изабрана због наредних фаза истраживања где ће се резолуција параметара смањити специфичним алгоритмима квантизације.

Лог-мел спектрограм, који служи као мрежни улаз, је димензија 96×313 . То означава присуство 96 мел опсега и 313 временских оквира. Конволуцијски блок, кључна компонента мреже, састоји се од пет конволуционих слојева. Сваки слој користи величину филтера 3×3 , док број филтера варира између слојева, постављајући се на 32, 32, 32, 64 и 128, респективно. *Padding* се примењује како би се осигурало да улазно-излазне величине остану конзистентне. Након сваког конво-



Слика 6.19. Архитектура CNN-а развијена за циљни случај употребе класификовања мотора [134].

Layer (type)	Output Shape	Param #
conv2d (Conv2D)	(None, 96, 313, 32)	320
dropout (Dropout)	(None, 48, 157, 32)	0
conv2d_1 (Conv2D)	(None, 48, 157, 32)	9248
dropout_1 (Dropout)	(None, 24, 79, 32)	0
conv2d_2 (Conv2D)	(None, 24, 79, 32)	9248
dropout_2 (Dropout)	(None, 12, 40, 32)	0
conv2d_3 (Conv2D)	(None, 12, 40, 64)	18496
leaky_re_lu_3 (LeakyReLU)	(None, 12, 40, 64)	0
conv2d_4 (Conv2D)	(None, 6, 20, 128)	73856
leaky_re_lu_4 (LeakyReLU)	(None, 6, 20, 128)	0
dense (Dense)	(None, 256)	983296
dense_1 (Dense)	(None, 512)	131584
dense_2 (Dense)	(None, 256)	131328
dense_3 (Dense)	(None, 128)	32896
dense_4 (Dense)	(None, 2)	258

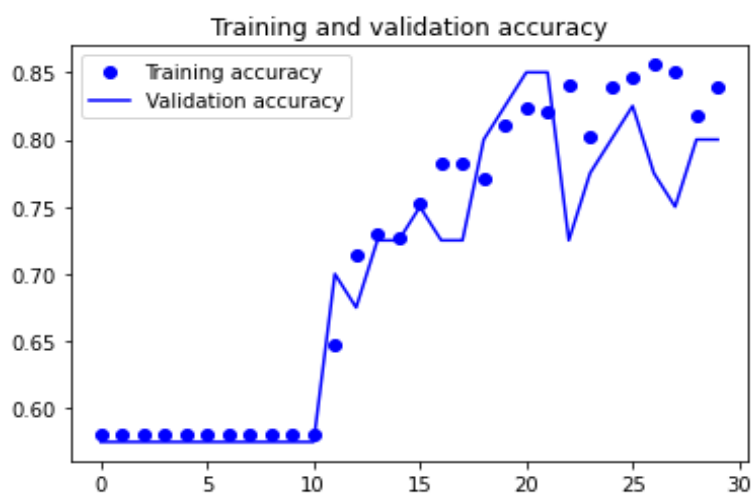
Слика 6.20. Слојеви, излазни облици и бројеви параметара развијене CNN добијене директно од Python-а.

луционог слоја уграђен је активациони слој, који користи функцију активације ReLu која цури са коефицијентом нагиба од 0,03. Након тога, уводи се слој максималног обједињавања, који користи облик прозора за обједињавање од 2×2 . Пре-

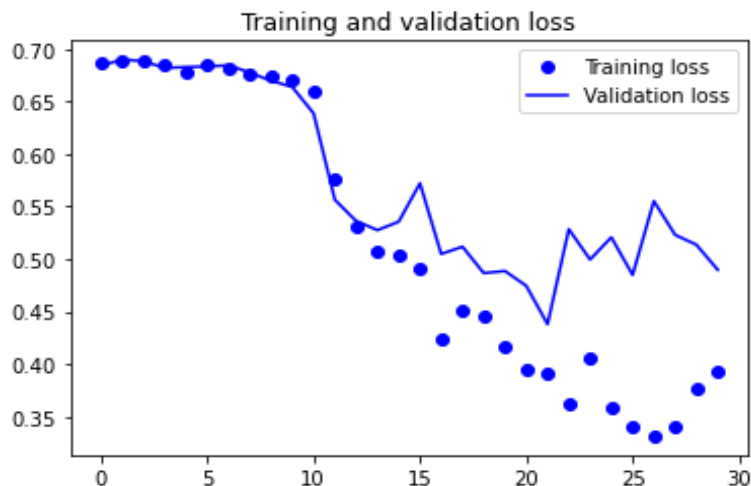
тходно следећем конволуционом слоју, претпоследњи слој је слој који се испушта, где је 10% чворова насумично постављено на нулу. Ова комбинација слојева, односно слојева конволуције, активације, максималног обједињавања и испадања, понавља се пет пута, чиме се формира конволуцијски блок CNN архитектуре [134]. Када овај блок заврши своју обраду, резултујући излаз се пропушта кроз слој за изравнавање како би се обезбедила префињена репрезентација. Након тога, изглађени излаз се доводи у потпуно повезану мрежу. Ова мрежа се састоји од неколико скривених слојева, од којих је сваки у потпуности повезан са следећим. Конкретно, постоје четири скривена слоја са 256, 512, 256 и 128 веза. Последњи слој ове мреже је у потпуности повезан са излазним слојем који је дизајниран да генерише два излаза за смештај две различите класе мотора [134].

6.5.5. Резултати класификације оштећења мотора пумпе

Развијена CNN је примењена на означене податке добијених аудио сигнала мотора пумпе који садрже укупно 668 узорака, односно 281 узорак типа MotorOK и 387 узорака типа и MotorNOK. Да би се модел ефикасно обучио, коришћен је низ од 64 и 30 епоха. Због релативно мале величине скупа података, примењена је *k-fold* унакрсна валидација, са вредношћу *k* једнаким 10. Сlike 6.21 и 6.22 пружају увид у тачност обуке и валидације, као и губитак модела са добрим перформансама и на подацима за обуку и валидацију. Из ових бројки је евидентно да постоји приметан тренд побољшања и тачности обуке и валидације, заједно са смањењем губитака и за скупове података за обуку и за валидацију. Тестирањем модела на посебном тест сету који се састоји од 94 узорака постигнута је тачност од 0,89, док је одговарајући губитак теста измерен на 0,325.



Слика 6.21. Проценат тачности учења за моторе пумпи МОК и MNOK.



Слика 6.22. Губитак тачности учења за моторе пумпи МОК и MNOK.

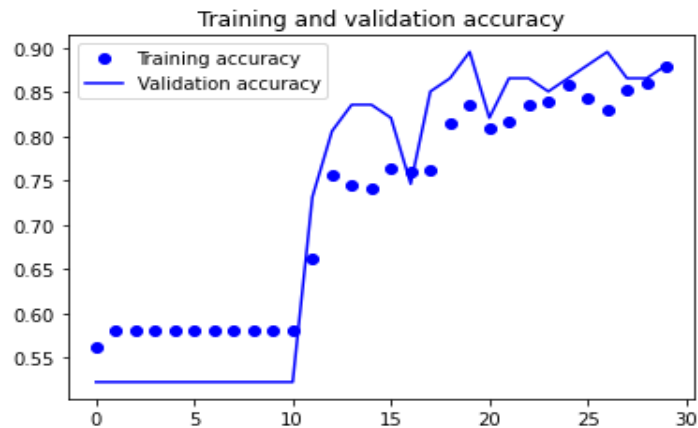
Слика 6.23 приказује додатне метрике перформанси, као што су прецизност, опозив, F1 резултат и подршка (број узорака из тестног скупа). Метрике „Macro avg“ и „Weighted avg“ се односе на непондерисане и пондерисане мере учинка, респективно [134].

Класа	Прецизност	Опозив	F1-оцена	Подршка
Класа 0	0.98	0.83	0.90	54
Класа 1	0.81	0.97	0.89	40
Тачност		0.89		94
Macro avg	0.90	0.90	0.89	94
Weighted avg	0.91	0.89	0.89	94

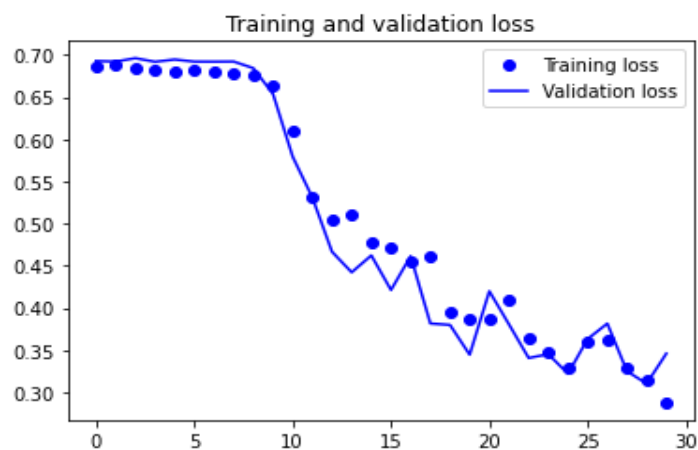
Слика 6.23. Прецизност, опозив, F1 оцена и број тест узорака постигнути применом развијене CNN на набављени сет означених узорака звука мотора (класа 0 је везана за MNOK моторе, а класа 1 је за МОК моторе).

6.5.6. Резултати класификације оштећења семеринга

Означени подаци аудио сигнала семеринга мотора пумпе анализирани су коришћењем развијеног CNN модела. Скуп података се састојао од укупно 568 узорака, од којих је 180 узорака категорисано као SOK (семеринг ОК) и 388 узорака категорисано као тип SNOK (семеринг NOK). Модел је обучен коришћењем серије од 64 и 30 епоха. Да би се решила ограничена величина скупа података, коришћена је „*k*-структура“ унакрсна валидација са $k = 10$. Сlike 6.24 и 6.25 приказују тачност обуке и валидације, као и губитак модела са добрим перформансама и на подацима за обуку и валидацију. Тачност постигнута на тест сету, који се састоји од 67 узорака, износи 0,82, док је одговарајући губитак теста 0,33. Слика 6.26 даје резиме прецизности, опозива, F1 резултата и подршке (број узорака из тестног скупа).



Слика 6.24. Проценат тачности учења за моторе пумпи за класу семеринга SOK и SNOK.



Слика 6.25. Губитак тачности учења за моторе пумпи за класу семеринга SOK и SNOK.

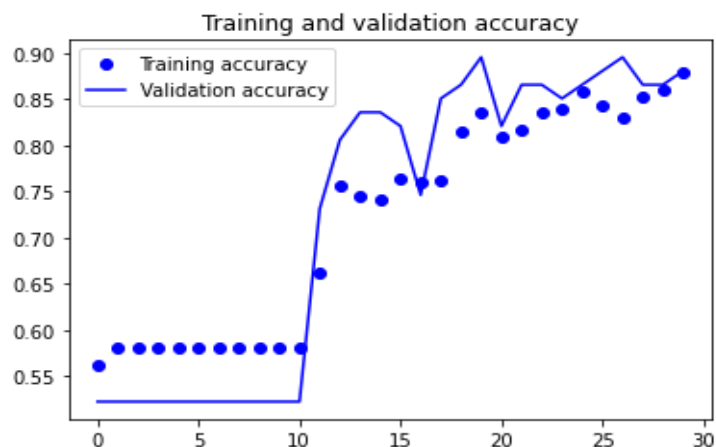
Класа	Прецизност	Опозив	F1-оцена	Подршка
Класа 0	0.92	0.79	0.85	42
Класа 1	0.71	0.88	0.79	25
Тачност		0.82		67
Macro avg	0.81	0.83	0.82	67
Weighted avg	0.84	0.82	0.82	67

Слика 6.26. Прецизност, опозив, F1 оцена и број тест узорак постигнути применом развијене CNN на набављени сет означених узорака звука пумпи мотора семеринга (класа 0 је везана за SNOK моторе, а класа 1 за SOK моторе).

6.5.7. Резултати класификације оштећења лежајева

Да би се испитали означени подаци аудио сигнала кугличних лежајева мотора пумпи коришћен је развијени модел CNN мреже. Овај скуп података обухватао је укупно 628 узорака, са 232 узорка категоризована као тип је LOK (лежајеви ОК) и 396 узорака категоризованих као тип LNOK (лежајеви NOK). Модел је обучен

коришћењем серије од 64 и 30 епоха, а да би се решила ограничена величина скупа података, коришћена је „*k*-структура“ унакрсна валидација са $k = 10$. Тачност обуке и валидације, као и губитак, приказани су на сликама 6.27 и 6.28, док слика 6.29 даје резиме прецизности, опозива, F1 резултата и подршке за скуп тестова.



Слика 6.27. Проценат тачности учења за моторе пумпи за класу кугличних лежајева LOK и LNOK.



Слика 6.28. Губитак тачности учења за моторе пумпи за класу кугличних лежајева LOK и LNOK.

Класа	Прецизност	Опозив	F1-оцена	Подршка
Класа 0	0.88	0.88	0.88	42
Класа 1	0.80	0.80	0.80	25
Тачност		0.85		67
Macro avg	0.84	0.84	0.84	67
Weighted avg	0.85	0.85	0.85	67

Слика 6.29. Прецизност, опозив, F1 оцена и број тест узорака постигнути применом развијене CNN на набављени сет означених узорака звука пумпи мотора кугличних лежајева (класа 0 је везана за LNOK моторе, а класа 1 за LOK моторе).

6.5.8. Закључна разматрања

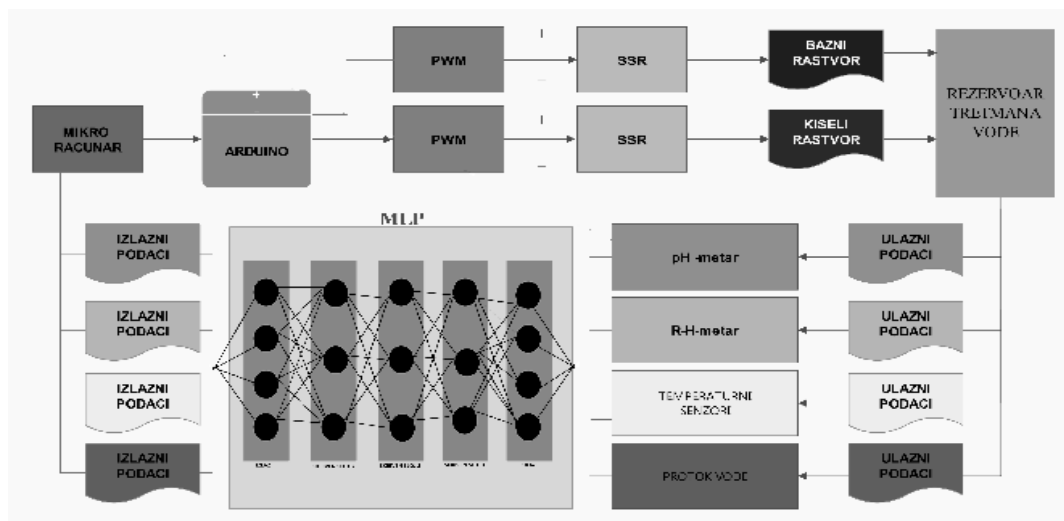
У претходним поглављима појашњене су категоризације за исправност мотора, пумпи, семеринга мотора пумпи и кугличних лежајева мотора пумпи, без кварова и са кваровима. Ова класификација је заснована на звуку који производе ови мотори пумпи док су у раду. Да би се постигла ова класификација аудио сигнали који „хватају“ 5 секунди звука мотора пумпи су трансформисани у мел-спектрограме. Они су, затим, коришћени као улаз за развијени CNN класификатор. Избор мел-спектрограма као одговарајућих слика мотивисан је њиховом способношћу да успоставе равнотежу између величине слике (резолације) и очувања кључних карактеристика аудио сигнала. Овај приступ осигурава да се важне информације задржавају уз одржавање величине слике којом се може управљати.

Развијена CNN мрежа која је конструисана за класификације мотора, пумпи семеринга, лежајева састоји се од два главна блока: конволуционог блока и потпуно повезаног блока. Почетни блок се састоји од пет узастопних секвенци слојева конволуције, активације, удруживања и испадања. С друге стране, други блок се састоји од пет потпуно повезаних слојева. Упркос ограниченој количини узорака, постигнута је импресивна стопа тачности од 89% за моторе пумпи, 82% за семеринге пумпи и 85% за кугличне лежајеве мотора пумпи. У овом истраживњу, процес обележавања субјективно спроводе оператери у постројењу који се ослањају на сопствену способност да перципирају разлике у звуковима мотора. Накнадна анализа означавања, класификације биће спроведена са највећом пажњом. Да би се решио проблем буке у звучном узорку, биће истражени различити приступи. Један такав приступ укључује ажурирање циљане оизнаке комбиновањем тренутних предвиђања модела са ознаком циља са буком користећи конвексну комбинацију.

6.6. Унапређење процеса прераде отпадних вода помоћу MLP-а

6.6.1. Уводна разматрања

У овом делу дисертације представљена је примена вештачке интелигенције на једном од најважнијих процеса у преради и пречишћавању отпадних вода: регулацију рН вредности воде (слика 6.30). Акценат је на примењеној мрежи вештачке интелигенције MLP (енг. *Multi Layer Perceptron*) која је употребљена у развоју алгоритма за контролу рада уређаја. MLP мрежа има минимално један скривени слој (енг. *hidden layer*), док је за овај пројекат је коришћена мрежа са три скривена слоја који директно утичу на алгоритме за одлучивање [135]. То, конкретно, значи да скривени слојеви добијају информације из истрениране базе података у коју су подаци унесени након експерименталних мерења приликом ради у фази пројекто-



Слика 6.30. Шема регулације рН вредности воде помоћу MLP мреже.

вања и истраживања. Параметри су добијени за различите вредности воде (рН, Rh, температуре и боја) и класификовани у две групе: параметри добрих вредности (H_2OOK) и параметри лоших вредности (H_2ONOTOK). Вредности су типа float са децималним зарезом на трећој децимали и као такве су уписане у MySQL базу. Ова база је првенствено изабрана због директног уписа вредности са мерних елемената које се путем Python апликације прослеђују PHP апликацији која врши уносе података у базу.

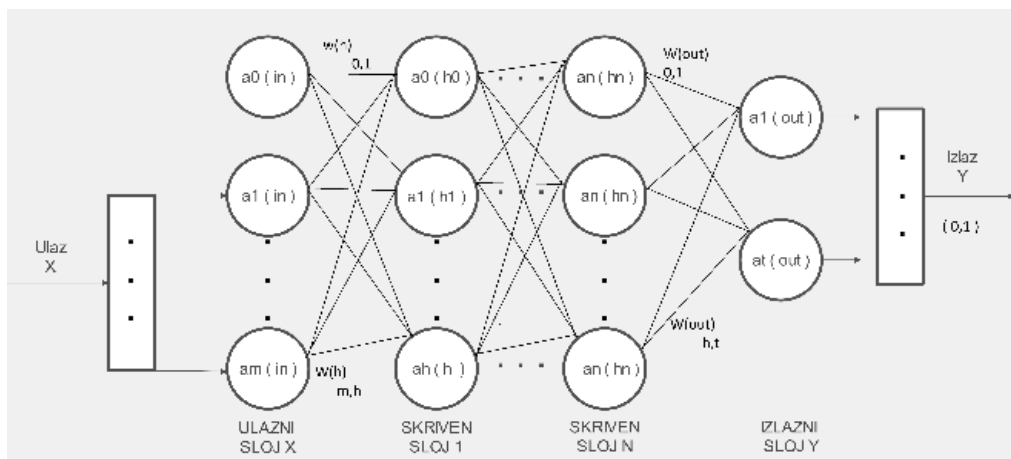
6.6.2. Архитектура MLP-а

MLP је, у основи, проширење неуронске мреже која се састоји од три различита слоја: улазних, излазних и скривених слојева. Улазни слој прима улазни сигнал који захтева обраду. Излазни слој је одговоран за извршавање задатака као што су предвиђање и класификација. Служећи као примарни рачунарски механизам MLP-а, произвољан број скривених слојева се налази између улазног и излазног слоја. Ток података се одвија једносмерно од улазног до излазног слоја, при чему се неурони обучавају коришћењем алгоритама учења повратног ширења (слика 6.31). MLP-ови су конструисани за предвиђање било које функције у континуитету и имају могућност решавања проблема који нису линеарно одвојиви. Главни случајеви употребе MLP-а су класификација образаца, препознавање, предвиђање и апроксимација [136]-[140]. Прорачуни, који се обављају у неуронима излазних и скривених слојева, дати су са [140]

$$o(x) = G(b(2) + W(2)h(x)) \quad (6.14)$$

$$h(x) = \Phi(x) = s(b(1) + W(1)x) \quad (6.15)$$

где су $b(1)$ и $b(2)$ биас вектори, $V(1)$ и $V(2)$ матрице тежине, док G и s означавају



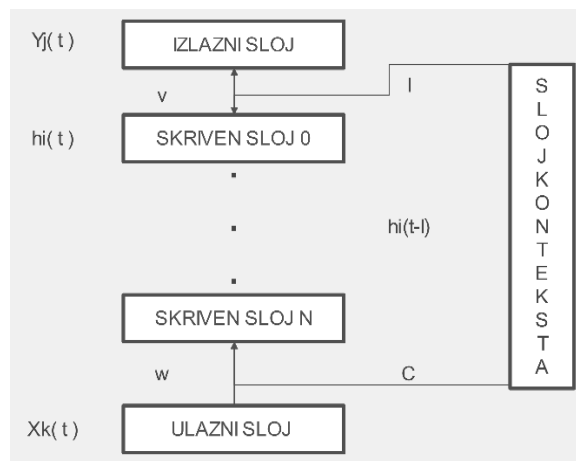
Слика 6.31. Приказ вишеслојног MLP-а.

активационе функције [140]. Скуп параметара за учење је скуп

$$th = \{V(1), b(1), V(2), b(2)\} \quad (6.16)$$

MLP је најраспрострањенија структура неуронске мреже, нарочито двослојна структура (у овој структури улазне јединице и излазни слој су међусобно повезани међускривеним слојем). Сваки неурон у мрежи је опремљен функцијом нелинеарне активације која је способна за диференцијацију. Мрежа има могућност статичког мапирања између улазног и излазног простора. RNN (енг. *Recurrent Neural Network*), као врста неуронске мреже која је дизајнирана за руковање секвенцијалним серијама, ради и на улазном и на унутрашњем простору стања. Међу различитим типовима рекурентних неуронских мрежа Елманов RNN се истиче због укључивања додатног слоја названог контекстни слој. Овај слој се састоји од чворова који делују као елементи кашњења у једном кораку, неприметно интегрисани унутар локалних путања повратне спреге (слика 6.32). Елманов RNN модел обухвата рекурентне везе које крећу почев од скривених неурона и протежу се као слоју контекстуалних јединица. Ове контекстуалне јединице служе као складиште за излазе скривених неурона, чувајући их са тежином од 1 за један временски корак, а затим их поново уводе у улазни слој. Скривени неурони на тај начин имају записе о својим претходним активацијама, што мрежи пружа могућност обављања задатака учења [137], [140]. Улазно-излазно мапирање за двослојну Елманову мрежу са n улазних чворова (индекс k), m скривених чворова (индекс i), t контекстних јединица (индекс u) и p излазних чворова (индекс j) се описује са

$$\begin{cases} y_j(t) = f \sum_{i=1}^m h_i(t) v_{ji} + b_j \\ h_i(t) = g \sum_{k=1}^n x_k(t) w_{ik} + \sum_{u=1}^m h_i(t-1) c_{iu} + b_i \end{cases} \quad (6.17)$$



Слика 6.32. Двослојни Елманов RNN са придруженом матрицом тежине [140].

Активација тренутног стања $h_i(t)$ се комбинује са претходном активацијом $h_i(t-1)$ кроз контекстни слој. Означавају се одговарајуће матрице тежине у мрежи $V_{m \times n}$, $C_{m \times m}$, $W_{p \times m}$, одговарајуће преносне функције за скривене (g) и излазне (f) слојеве и термин за биас b [138]. Приступ учењу се ослања на присуство означеног скупа података за обуку, који садржи N улазно-излазних примера који се сматрају тачним:

$$T = \{(x_i, d_i)\}_{i=1}^N \quad (6.18)$$

За израчунавање слободних параметара неуронске мреже неопходно је узети у обзир улазни вектор x_i за сваки пример i , као и жељени одговор d_i . Ради једноставности у презентацији претпоставља се да је жељени одговор d_i скаларна вредност. Штавише, величина узорака је представљена са N . Циљ је да се утврде вредности слободних параметара који ће обезбедити довољно блиско подударање између стварног излаза неуронске мреже, генерисаног улазним вектором x и жељеног одговора d_i за сваки пример i , у статистичком смислу. У том случају може се користити једначина за средњу квадратну грешку

$$E(n) = \frac{1}{N} \sum_{i=1}^N (d_i - y_i)^2 \quad (6.19)$$

Насупрот томе, улазне карактеристике се преносе на улазну функцију u , која израчунава пондерисани збир улазних карактеристика [140]:

$$u(x) = \sum_{i=1}^n w_i x_i \quad (6.20)$$

Након израчунавања, резултујућа вредност се преноси на функцију активације f , која производи излаз перцептрона. У традиционалном перцептрону, функција f је дефинисана као функција корака:

$$y = f(u(x)) = \begin{cases} 1, & \text{ако је } u(x) > 0 \\ 0, & \text{супротно од претходног} \end{cases} \quad (6.21)$$

6.6.3. Резултати обуке и тестирања MLP мреже

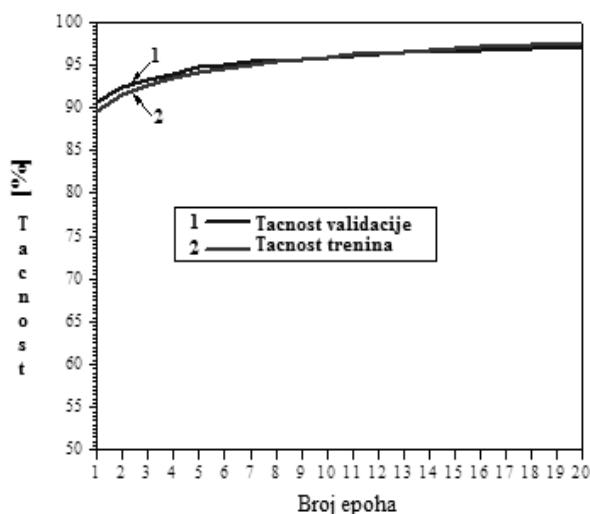
У овом потпоглављу биће приказани су резултати тестирања и обуке MLP мреже. Изабрани параметри (број епоха и број скривених слојева) су подешени како мрежа не би била премало или превише обучавана, односно број скривених слојева је постављен тако да би се извршење обуке обавило у што краћем временском периоду, а како резултати не би утицали на тачност и валидност обуке.

MLP је обучен и тестиран на MNIST са 60.000 слика у бази података у сивој скали ручно писаних цифара коришћених за обуку мреже и 10.000 слика за тестирање величине 28 x 28 пиксела. Улазни слој се састоји од 784 чвора, што представља величину слике од 28 x 28 пиксела. Скривени слој се састоји од 128 чворова, док излазни слој има 10 чворова, што одговара 10 класа (цифара). У скривеним и излазним слојевима користе се функције *ReLU* и *Softmax*. Вишеслојни перцептрон користи специфичне по-ставке хиперпараметара, укључујући стопу регуларизације од 0,01, брзину учења од 0,0005 и величину серије од 128 (табела 6.1) [140].

Табела 6.1. Релевантни параметри за дату MLP неуронску мрежу.

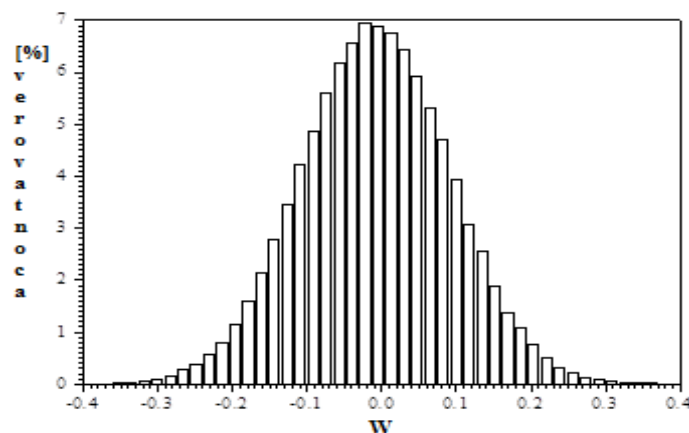
Параметри	$b^{(h)}$	$b^{(out)}$	$w^{(h)}$	$w^{(out)}$
Укупно	128	10	100352	1280

Слика 6.33 илуструје криве учења за обуку MLP неуронске мреже кроз 20 епоха. Очигледно је да се тачност обуке и валидације повећава до тачке стабилности и да постоји минималан јаз између ових вредности.



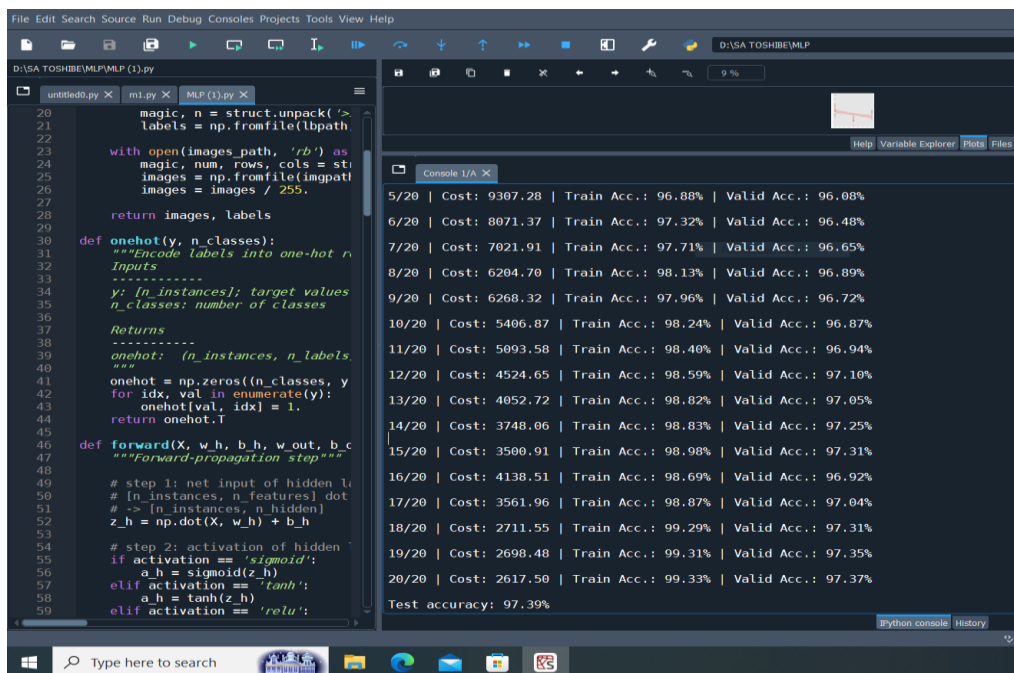
Слика 6.33. Крива учења за дату MLP неуронску мрежу.

Слика 6.34 представља хистограм тежина између улазног и скривених слојева истрениране MLP мреже. На у оси приказана је вероватноћа у процентима, док су на х оси приказане тежине. Са хистограма може се видети да максимална фреквенција не прелази 7, а минимална 0,2 %. Хистограм је нормалне дистрибуције.



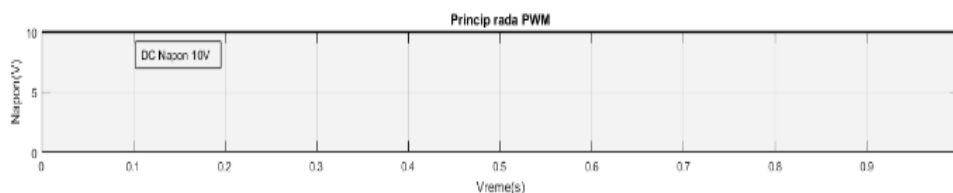
Слика 6.34. Хистограм тежина (између улазног и скривеног слоја) трениране MLP неуронске мреже.

На слици 6.35 приказан је изглед тренинга за MLP неуронску мрежу са скупом из MNIST базе података, која се обучава за 20 епоха. Тест тачности је 97,39%. Кроз сваку епоху приказана је тачност тренинга и тачност валидације.

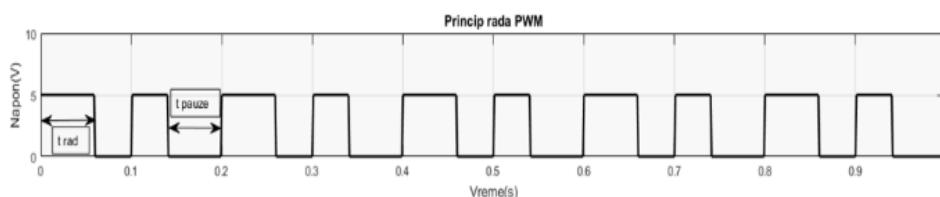


Слика 6.35. Пример тренинга и валидације тачности.

За регулацију рН параметара употребљена је PWM, односно пулсна модулација. Принцип рада пулсне модулације је објашњен графиконима који следе. На слици 6.36 приказан је график за једносмерни DC напон напајања актуатора од 10V. Смањивањем времена t -рада и повећањем времена t -паузе на PWM добија се смањење напона на жељену вредност од 5V (слика 6.37). PWM, коришћена за потребе екперимента, ради на принципу ултра брзог прекидача који у десетинама делова милисекунде врши паљење, односно гашење напона. На регулатору рН је примењена PWM [141] модулација за регулисање отварања/затварања актуатора са базним раствором, односно актуатора са киселинама у зависности од вредности рН. Као PWM генератор сигнала на рН регулатору је коришћен PLC Wecon због индустријских стандарда.



Слика 6.36. Напон актуатора 10V DC.



Слика 6.37. Одмеравање циклуса PWM модулације.

6.6.4. Закључна разматрања

Примењено решење може дати мали допринос смањењу загађења рецепијената (речних водотокова), као и уштеди енергије и људских ресурса. Употребом MLP алгоритама вештачке интелигенције, као и коришћењем нове генерације микрорачунара, микроконтролера и електронских компоненти добијено је решење које смањује загађење испусне воде за 2,8% у односу на тренутно коришћена решења у индустријској употреби, а за 80% смањује потребу за људским ресурсима. Описано решење примењује се и на друге процесе у фабрици за пречишћавање отпадне воде, као што су регулација Redox потенцијала, дозирање флокуланата, згушњивача који сакупљају штетне материје, односно тешке метале (хром, цинк и манган). Даљи развој односи се на употребу нове врсте филтрације која се првенствено састоји од филтрације активним угљем, где се очекује повећање ефикасности постројења за додатних 3%. Након засићења филтера са активним угљем мо-

гуће је спаљивање у енергани коју је могуће изградити у постројењу. Овим би се избегли трошкови складиштења и уништавања опасних материја које се до сада обављало у иностранству, а чији су трошкови били значајни. Ако би се направила енергана, због значајног капацитета постројења за прераду и већих количина активног угља који се користи у филтрацији, инвестиција је исплатива већ након две године. Енергија која би се добила сагоревањем могла би се користити за грејање воде или прављење водене паре која се већ користи у процесу производње фабрике. Овиме би се значајно повећала ефикасност и исплативост постројења.

7. Стратегија за унапређење рада СПОВ-а

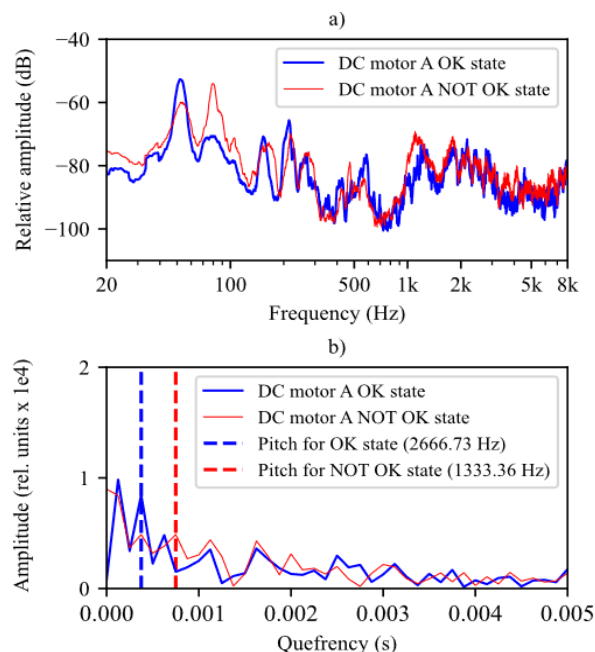
7.1. Видео надзор када

У петом поглављу детаљно су описани целобројни SEC-DAEC кодови који могу да исправе једноструке битске грешке и двоструке суседне битске грешке. Иако у постројењу није вршена провера квалитета UTP везе, на основу анализе дневних активности постројења дошло се до закључка да главни разлог због којег долази до повремених губитака видео пакета лежи у несавршености UTP кабла. Како је праћење нивоа воде у кадама један од битнијих задатака постројења, у наредном периоду биће предложено имплементирање целобројних SEC-DAEC кодова. За очекивати је, да се након њихове, број изгубљених видео пакета смањи за 8 до 10 пута.

7.2. Предвиђање квара индустријских уређаја

Резултати из шестог поглавља показују да висине тонова дефинишу различите услове и стања уређаја. Међутим, постоје случајеви када се из висине тона не могу уочити разлике између стања. Због тога се, као додатна аудио карактеристика, може увести мера израчуната из цепструма. На слици 7.1а могу се приметити релативне амплитуде у децибелима и одређене фреквенције цепструма за исправан мотор који је на графику приказан плавом бојом и неисправан који је представљен црвеном бојом. На слици 7.1б представљене су карактеристике исправног мотора (график плаве боје) односно неисправног (график црвене боје). Врх је представљен испрекиданом плавом линијом на 2666,73 Hz за исправан мотор, односно испрекиданом црвеном линијом на 1333,36 Hz за неисправан мотор. Из свега овога се може закључити да се на основу разлике цепструма може успешно вршити класификација различитог стања више врста различитих електромотора.

Примарни значај анализе није у мерама сличности слике. Сlike које се проучавају у овом раду нису природне слике; уместо тога, оне се стварају кроз специфичан процес који укључује трансформацију или мапирање звука индустријског производа (машине) у мел-спектрограм. У шестом поглављу, слика 6.14 представља графиконе који приказују средње вредности по узорку за две класе (аутомобил и воз) и све три мере сличности. Резултати добијени из нормализованог ED и PCC показују изузетну сличност, што сугерише да примарни трендови остају доследни. Када се ED и PCC користе уместо SSIM мера, уочавају се поузданији резултати и јасније разлике између класа. Треба напоменути да постоји значајна корелација између ових сувишних информационих мера, јер све три мере показују исте фундаменталне трендове у добијеним резултатима.



Слика 7.1. „Крива спектра а) и крива цепструма б) за моторе без отказа и моторе са отказима“ [80].

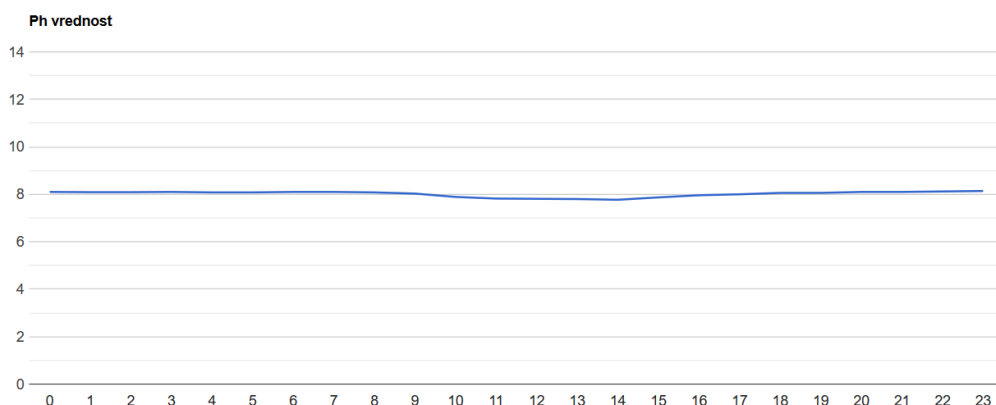
Ипак, очигледно је да ослањање само на једну меру сличности слике, у њеном постојећем стању, не би понудило довољно информација за конструисање потпуно оперативног параметарског класификатора који се заснива на вредностима сличности слика. Да би се направио класификатор способан да произведе резултате упоредиве са онима које постижу дубоке неуронске мреже DNN, постаје неопходно добити поузданије и конзистентније информације кроз процену сличности слика [118].

Категоризација мотора у две категорије је одређена звуком који ови мотори производе током рада. Да би их класификовали аудио снимци који снимају 5 секунди релевантног звука мотора се трансформишу у мел-спектрограме, који служе као улаз за CNN класификатор који је развијен. Избор мел-спектрограма као одговарајућег приказа слике заснива се на његовој способности да успостави равнотежу између величине слике и очувања значајних карактеристика аудио сигнала [134]. Из добијених резултата (слике 6.22 - 6.29) може се видети тачност обуке и валидације за 30 епоха, односно губитак обуке и валидације за 30 епоха. Са слика 6.21, 6.24 и 6.27 може се видети да је тачност обуке за класификацију за мотор ОК и мотор NOK износила 89%, за класификацију оштећења семеринга SOK и SNOK 82%, а за класификацију оштећења кугличних лежајева LOK и LNOCK 85%. Истраживање и обука CNN мреже, и поред малог броја исправних-неисправних класификатора, показала је одличне резултате. Примена ове мреже могућа је и у

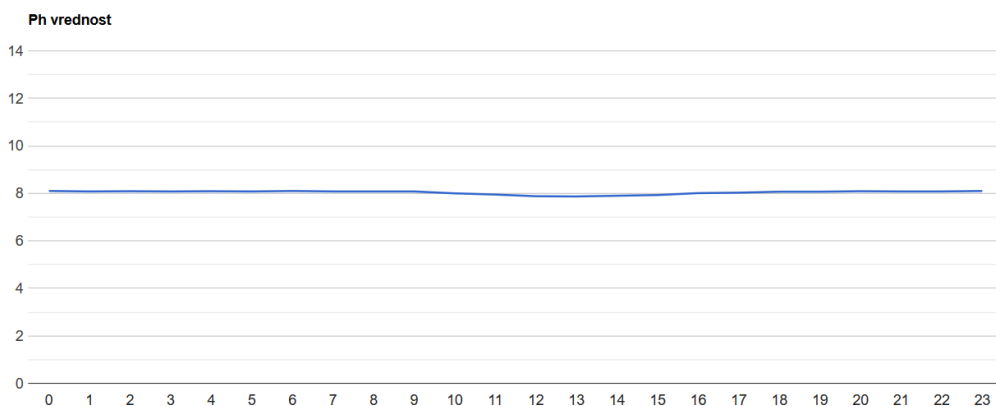
другим гранама индустрије где се, на основу звука, може направити разлика између исправног и неисправног рада машине.

7.3. Регулација рН вредности отпадне воде

Након примењених решења за аутоматско управљање у постројењу и употребом MLP неуронске мреже за управљање и дозирање база и киселина, добијени су резултати на дневном и месечном нивоу. Графици, представљени сликама 7.2 - 7.9, генерисани су из базе података помоћу RHP апликације. На сликама 7.2 и 7.3 приказане су месечне вредности рН параметара отпадне воде након корекција, односно дозирања киселине или кречног млека. На слици 7.2 приказане су рН вредности када дозирање врши администратор постројења, док су на слици 7.3 приказане рН вредности када контролу има вештачка интелигенција (под надзором оператора постројења). Са слика се може приметити да су криве приближних особина. Иначе, жељена рН вредност треба да се креће око 8.

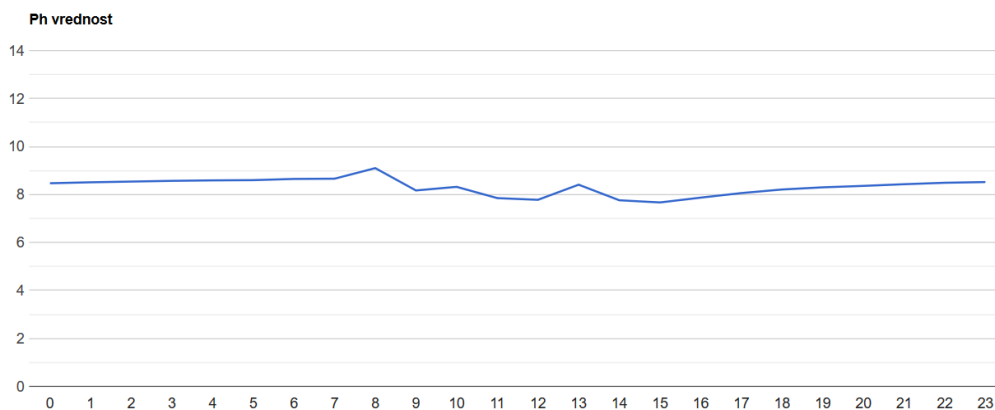


Слика 7.2. рН вредност отпадне воде након корекције (дозирање је вршио оператор).

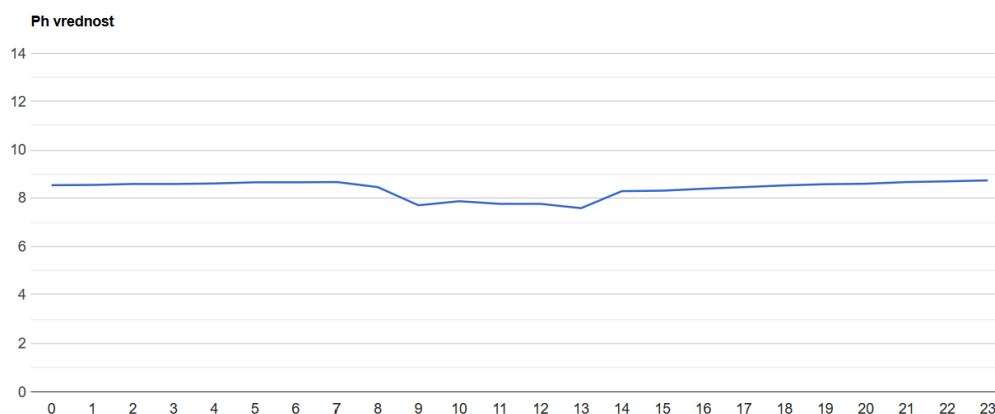


Слика 7.3. рН вредност отпадне воде након корекције (дозирање је вршила вештачка интелигенција под надзором оператора).

На сликама 7.4 и 7.5 приказани су графици вредности рН параметара у сепаратору. Као и на претходним графицима може се приметити блага предност употребљеног аутоматизованог решења употребом вештачке интелигенције у односу на управљање оператора постројењем.



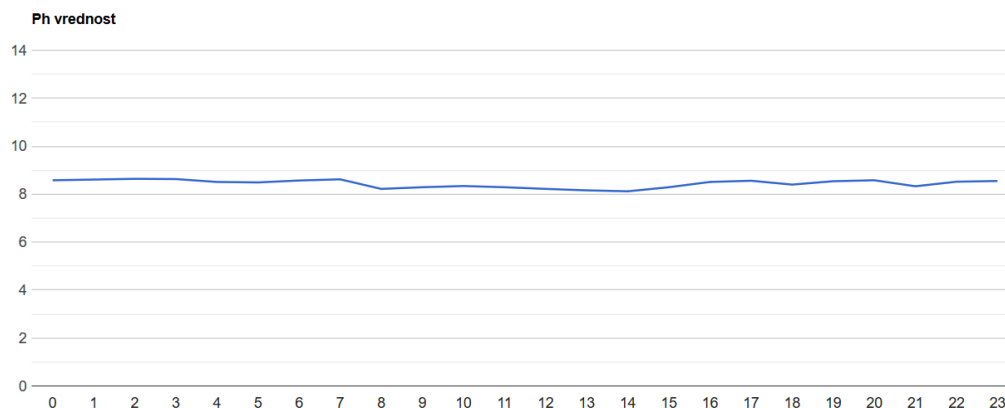
Слика 7.4. рН вредност отпадне воде у сепаратору (дозирање је вршио оператор).



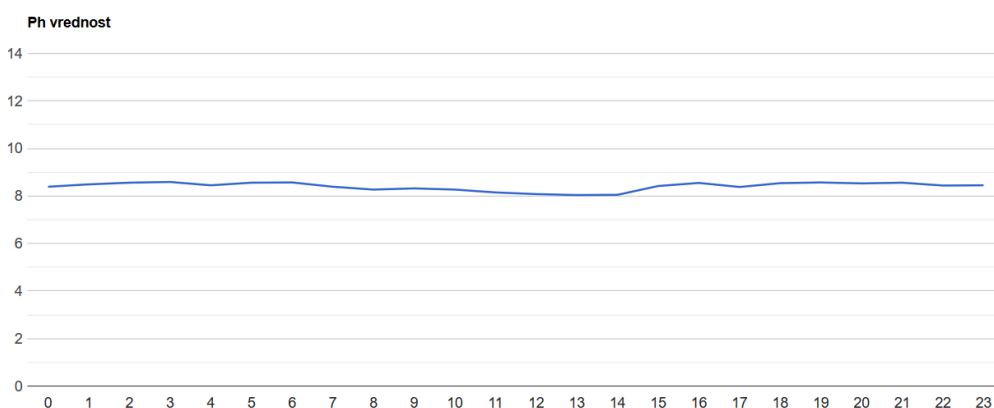
Слика 7.5. рН вредност отпадне воде у сепаратору (дозирање је вршила вештачка интелигенција под надзором оператора).

На сликама 7.6 и 7.7 приказани су графици на излазу из постројења према реципијенту. И у овом случају видимо да управљање, које је вршила вештачка интелигенција (под надзором оператора), даје боље резултате.

Из свега приказаног јасно је да примењено решење може дати допринос смањењу загађења реципијената, као и уштеди енергије и људских ресурса. Употребом MLP алгоритама вештачке интелигенције, као и употребом нове генерације микрорачунара и електронских компоненти добијено је решење које смањује загађење испусне воде за 2,8% у односу на тренутно коришћена решења у индустријској употреби, а за 80% смањује потребу за људским ресурсима. Описано решење



Слика 7.6. рН вредност отпадне воде на излазу из постројења (сва дозирања је вршила вештачка интелигенција под надзором оператора).



Слика 7.7. рН вредност отпадне воде на излазу из постројења (сва дозирања је вршио оператор).

може се применити и на друге процесе у фабрици за пречишћавање отпадне воде, као што су регулација Redox потенцијала, дозирање флокуланата, згушњивача који сакупљају штетне материје, односно тешке метале. Даљи развој односи се на употребу нове врсте филтрације која се првенствено састоји од филтрације активним угљем, где се очекује повећање ефикасности постројења за додатних 3%. Након засићења филтера са активним угљем могуће је спаљивање у енергани коју је могуће изградити у фабрици чиме би се избегли трошкови складиштења и уништавања опасних материја које се до сада обављало у иностранству. Осим тога, енергија која би се добила сагоревањем могла би се користити за грејање воде или прављење водене паре која се већ користи у процесу производње фабрике. Тиме би се додатно повећала ефикасност и исплативост постројења.

7.4. Вредности параметара отпадне воде у унапређеном СПОВ-у

Обавезном контролом граничних вредности отпадних вода, једном месечно у акредитованој лабораторији (законска обавеза), остварени су готово идеални резултати [21]. Конкретно, анализе су показале да су, уз примену вештачке интелигенције, вредности параметара отпадне воде на излазу из постројења за 20% ниже од вредности које је прописао законодавац (слика 7.8).

Резултати мерења акредитоване лабораторије													
Параметри	Јединица мере	Број процеса											
		1	2	3	4	5	6	7	8	9	10	11	12
Алуминијум	mg/l	2	-	2	-	-	-	-	-	1	2	1	2
Азот из амонијака	mg/l	80	-	-	-	25	45	-	-	15	20	-	-
НРК	mgO ₂ /l	300	-	90	-	180	3500	-	-	90	300	320	200
Гвожђе	mg/l	1	-	-	-	2	-	-	-	2	2	2	2
Флуориди	mg/l	40	-	40	-	40	-	-	-	40	20	-	-
Азот нитрати	mg/l	-	-	4	-	-	4	-	-	3	4	-	-
Угљоводоници	mg/l	8	-	9	-	7	9	-	-	8	8	7	9
Фосфор	mg/l	1	-	1	-	1	1	-	-	1	1	1	1
Токсичност за рибе		5	-	1	-	4	5	-	-	3	4	4	5
АОХ	mg/l	0,67	-	0,62	-	0,59	0,61	-	-	0,62	0,66	0,66	0,6
Арсен	mg/l	0,06	-	-	-	-	-	-	-	-	-	-	-
Баријум	mg/l	-	-	-	-	-	1,33	-	-	-	-	-	-
Олово	mg/l	0,33	-	-	-	0,32	-	-	-	0,34	0,33	-	0,33
Кадмијум	mg/l	0,13				0,07			-	0,14	0,05		0,17
	kg/t	0,2							-				
Слободни хлор	mg/l	0,33	-		-		0,32				0,33		
Укупни хром	mg/l	0,33	-	0,33	-			-		0,33	0,33	0,5	0,32
Хром	mg/l	0,06	-	0,06	-			-		0,05	0,05		0,08
Кобалт	mg/l			0,65						0,66			
Цијанид	mg/l	0,13					0,66	-			0,14		
Бакар	mg/l	0,33	-					-	-	0,33	0,33	0,5	0,31
Никл	mg/l	0,33	-		-			-	-	0,33	0,34	0,5	0,32
Жива	mg/l								-				
	kg/t								-				
Селен	mg/l									0,67			
Сребро	mg/l	0,06						-	-				
Сулфиди	mg/l	0,67	-		-			-	-	0,67			
Калај	mg/l	1,33		1,38		1,31		-					
Цинк	mg/l	1,33	-	1,36		1,37			-	1,35	1,33	1,32	1,344

Слика 7.8. Вредности параметара отпадних вода на излазу из постројења (након примене вештачке интелигенције) које су измерене у овлашћеној лабораторији у фебруару 2018. године.

8. Закључак

Вода је, као што је у уводу дисертације наведено, основна потреба и главни елемент за живот човека. Кроз дугогодишњи научно-истраживачки рад, аутор је покушао да образложи своје идеје који би помогле да се спречи даље загађење воде. Неке од њих изложене су и објашњене у овој дисертацији. Конкретно, аутор је свој истраживачки рад фокусирао у три правца: употреба целобројних SEC-DAEC кодова, примена CNN мреже и примена MLP мреже.

У раду је показано да целобројни SEC-DAEC кодови могу играти важну улогу у погледу одржавања континуираног рада постројења. Главна намена поменутих кодова јесте заштита видео података које шаљу камере за снимање и праћење нивоа воде у кадама. Аутор је, током своје научно-истраживачке каријере, неколико пута био сведок изливања воде у постројењу, која је, због присуства велике количине база и киселина, оштећивала бројну опрему. Из потребе да се на удаљени начин надзире ниво воде у разним кадама, родила се и идеја о примени целобројних SEC-DAEC кодова у СПОВ-у уместо Интернет чексуме.

Осим коришћења поменутих кодова, рад постројења се може унапредити и на друге начине. Један од њих, описан у дисертацији, тиче се коришћења вештачке интелигенције. Током дугогодишњег научно-истраживачки рада, аутор се бавио и проблемом предвиђања кварова у СПОВ-у. У ту сврху вршена су снимања звукова разних индустријских уређаја како би се препознало да ли они раде правилно или не. Снимци звукова су претварани у мел-спектрограме да би се потом вршила анализа на бази цепструма и применом мере сличности слике. Кључна улога, у том погледу, додељена је CNN мрежи, која је обучена да детектује кварове и отказе.

Друга врста неуронске мреже, MLP, је експериментално тестирана у постројењу. У раду је показано да је MLP успела да одржава рН вредност отпадне воде на нивоу од 8.0, док је оператер (човек) имао до 18% лошији учинак. За разлику од CNN-а, ова мрежа је обучена да прати вредности параметара отпадне воде у кадама (рН вредност, Redox потенцијал, итд.) и да их одржава у оквиру законски прописаних граница.

На крају треба истаћи и то да се приказани резултати могу применити и у другим индустријским погонима. Контролом рада, праћењем процеса, прикупљањем података и информација, адекватном анализом и обуком мрежа вештачке интелигенције могуће је унапредити многе процесе рада, смањити или спречити оштећења и губитке, благовремено реаговати на неповољан развој ситуације, смањити трошкове и грешке у одлучивању, итд. Аутор се нада да ће нека од предложених решења наћи своју практичну примену како би се водени ресурси сачували и за будућа покољења.

9. Литература

- [1] Službeni glasnik Republike Srbije, "Uredba o graničnim vrednostima emisije zagađujućih materija u vodama i rokovima za njihovo dostizanje: br. 67/2011, 48/2012 i 1/2016", Beograd, 2016.
- [2] L. Knetević, „Rekuperacija materijalnih resursa iz komunalnog čvrstog otpada-reciklaža“, Zbornik radova „Otpadne vode i ostali otpadi“, Vrnjačka Banja, 1995.
- [3] D.. Reis, S. Filho, C. Grotto, M. Bilhain, D. Pereira, „Carboxymethylcellulose and cellulose xanthate matrices as potential adsorbent material for potentially toxic Cr³⁺, Cu²⁺ and Cd²⁺ metal ions: a theoretical study“, Theoretical Chemistry Accounts, vol. 139, no. 6, pp. 1-8, 2020.
- [4] Y. Huang, D. Wu, X. Wang, „Removal of heavy metals from water using polyvinylamine by polymer-enhanced ultrafiltration and flocculation“, Separation and Purification Technology, vol. 158, pp. 124-136, 2016.
- [5] J. Shao, S. Qin, J. Davidson, „Recovery of nickel from aqueous solutions by complexation-ultrafiltration process with sodium polyacrylate and polyethylenimine“, J. Hazardous Materials, vol. 244-245, pp. 472-477, 2013.
- [6] J.K. Pi, H.C. Yang, L.S. Wan, „Polypropylene microfiltration membranes modified with TiO₂ nanoparticles for surface wettability and antifouling property“, Journal of Membrane Science, vol. 500, pp. 8-15, 2016.
- [7] D.J. Ennigrou, M. Ben Sik Ali, M. Dhahbi, „Copper and Zinc Removal from Aqueous Solutions by Polyacrylic Acid Assisted-Ultrafiltration“, Desalination, vol. 343, pp. 82-87, 2014.
- [8] J. Gao, Y. Qiu, B. Hou, Q. Zhang, X. Zhang, „Treatment of Wastewater Containing Nickel by Complexation- Ultrafiltration Using Sodium Polyacrylate and the Stability of PAA-Ni Complex in the Shear Field“, Chemical Engineering Journal, vol. 334, pp. 1878-1885, 2018.
- [9] A. Yürüm, A. Taralp, N. Biçak, H.Ö. Özbelge, L. Yilmaz, „High Performance Ligands for the Removal of Aqueous Boron Species by Continuous Polymer Enhanced Ultrafiltration“, Desalination, vol. 320, pp. 33-39, 2013.
- [10] M.A. Khosa, S.S. Shah, X. Feng, „Metal Sericin Complexation and Ultrafiltration of Heavy Metals from Aqueous Solution“, Chemical Engineering Journal, vol. 244, pp. 446-456, 2014.
- [11] B. Lam B, S. Déon, N. Morin-Crini, G. Crini, P. Fievet, „Polymer-Enhanced Ultrafiltration for Heavy Metal Removal: Influence of Chitosan and Carboxymethyl Cellulose on Filtration Performances“, Journal of Cleaner Production, vol. 171, pp. 927-933, 2018.
- [12] J. Huang, F. Yuan, G. Zeng, X. Li, Y. Gu, L. Shi, W. Liu, Y. Shi, „Influence of Ph on Heavy Metal Speciation and Removal from Wastewater Using Micellar-Enhanced Ultrafiltration“, Chemosphere, vol. 173, pp. 199-206, 2017.
- [13] T. Urošević, K. Trivunac, „Achievements in low-pressure membrane processes microfiltration (MF) and ultrafiltration (UF) for wastewater and water treatment. Current Trends and Future Developments on (Bio-) Membranes“, Recent Achievements in Wastewater and Water Treatments, Elsevier Inc. Ch.3, pp. 67-107, 2020.
- [14] M. Vandenbossche, M. Jimenez, M. Casetta, M. Traisnel, „Remediation of Heavy Metal by Biomolecules: A review“, Critical Review in Environmental Science and Technology, vol. 45, pp. 1644-1704, 2015.

- [15] S. Gaćeša, „Osnove tehnologije vode i otpadnih voda“, Novi Sad, 1980.
- [16] D. Ljubisavljević, A. Đukić, B. Babić, „Prečišćavanje otpadnih voda“, Građevinski fakultet Beograd, 2004.
- [17] B. Vučijak, A. Čerić, I. Silajdžić, S. Midžić-Kurtagić, „Voda za život: osnove integralnog upravljanja vodnim resursima“, Institut za hidrotehniku Građevinskog fakulteta, Sarajevo, 2011.
- [18] <https://www.jumo.net/attachments/JUMO/attachmentdownload?id=6029>
- [19] <https://www.zalivnisistemi.rs/automatika/elektromagnetni-ventil-pgv-100/>
- [20] M. Miletić, R. Cvejić, V. Dedić, „Napredne tehnike prečišćavanja otpadnih voda“, Peti naučno-stručni skup sa međunarodnim učešćem „Zaštita voda u zelenoj industrijskoj revoluciji“, Zbornik radova, Beograd, 2021, ISBN 978-86-81400-60-9.
- [21] M. Miletic, M. Ilic, T. Markovic and B. Mihajlovic, “Application of IT Technology In Wastewater Treatment”, 10th Int. Sci. Conf. On Defensive Technologies (OTEH 2022), pp. 506-513, Oct. 2022.
- [22] https://www.we-con.com.cn/en/product_page/220.html
- [23] https://www.we-con.com.cn/en/product_page/244.html
- [24] L. Peterson, B. Davie, “Computer Networks: A Systems Approach”, 6th Edition, Elsevier Science, 2021.
- [25] J. Postel, Transmission Control Protocol (TCP), IETF RFC 793, 1981.
- [26] J. Postel, Internet Protocol (IP), IETF RFC 791, 1981.
- [27] J. Postel, User Datagram Protocol, IETF RFC 768, 1980.
- [28] D. Wu, Y. T. Hou, Y.-Q. Zhang, “Transporting Real-Time Video over the Internet: Challenges and Approaches,” Proc. IEEE, vol. 88, no. 12, pp. 1855–1877, 2000.
- [29] D. Mello, E. Offer, J. Reichert, “Error Arrival Statistics for FEC Design in Four-Wave Mixing Limited Systems,” Proc. Opt. Fiber Commun. Conf., pp. 529-530, 2003.
- [30] L. James, “Error Behaviour in Optical Networks”, PhD thesis, Dept. of Engineering, Univ. Cambridge, 2005.
- [31] P. Anslow, O. Ishida, “Error Distribution in Optical Links”, IEEE 802.3 HSSG Interim Meeting, Atlanta, 2007.
- [32] S. Yao, X. Liu, D. Liu, “The Impact of Differential Precoding on 25-Gbit/s EDB, NRZ and NRZ-NFC,” IEEE 802.3ca Task Force, 2016.
- [33] D. van Veen, V. Houtsma, “Proposals for Cost-Effectively Upgrading Passive Optical Networks To a 25G Line Rate,” J. Lightw. Tech., vol. 35, no. 6, pp. 1180-1187, 2017.
- [34] R. Strobel, “Error Correction Coding for Passive Optical Networks”, Oberpfaffenhofen Workshop on High Throughput Coding (OWHTC), Feb. 2019.
- [35] E. Fujiwara, “Code Design for Dependable Systems: Theory and Practical Applications,” John Wiley & Sons, Inc., 2006.
- [36] Z. Wu, C. Gong, D. Liu, “Computational Complexity Analysis of FEC Decoding on SDR platforms,” J. Signal Process. Syst., vol. 89, no. 2, pp. 209-224, 2017.
- [37] W. Qiao, D. Liu, S. Liu, “QFEC ASIP: A Flexible Quad-Mode FEC ASIP for Polar, LDPC, Turbo and Convolutional Code Decoding,” IEEE Access, vol. 6, pp. 72189-72200, 2018.
- [38] A. Radonjic, V. Vujicic, “Logarithmic Time Encoding and Decoding of Integer Error Control Codes,” Eng. Rep., vol. 5, no. 1, e12675, Nov. 2023.

- [39] A. Radonjic, "Integer Codes Correcting Single Errors and Detecting Burst Errors within a Byte," *IEEE Trans. Dev. Mater. Rel.*, vol. 20, no. 4, pp. 748-753, 2020.
- [40] A. Radonjic, I. Ristic, I. Scepanovic, "Integer Codes Correcting Single Errors and Detecting Burst Errors within Two Bytes," *IET Commun.*, vol. 17, no.11, pp. 1255-1261, 2023.
- [41] A. Radonjic, V. Vujicic, "Integer Codes Correcting/Detecting Burst Errors within One Byte and Detecting Burst Errors within Two Bytes," *Comput. Appl. Math.*, accepted for publication, <https://doi.org/10.1007/s40314-023-02435-2>.
- [42] A. Radonjic, V. Vujicic, "Integer Codes Correcting Burst Errors within a Byte and Detecting Double Errors," *China Commun.*, accepted for publication.
- [43] A. Radonjic, V. Vujicic, "Integer Codes Correcting Sparse Byte Errors," *Cryptogr. Commun.*, vol. 11, no. 5, pp. 1069-1077, 2019.
- [44] A. Radonjic, "Integer Codes Correcting Single Errors within Two Bytes," *J. Circ. Syst. Comput.*, vol. 30, no. 14, 2150260, 2021.
- [45] A. Radonjic, "Integer Codes Correcting Double Errors and Triple-Adjacent Errors within a Byte," *IEEE Trans. VLSI Syst.*, vol. 28, no. 8, pp. 1901-1908, 2020.
- [46] A. Radonjic, N. Pokhrel, P. Das "Integer Codes Correcting Burst Errors within Two Bytes," *Comput. Appl. Math.*, vol. 42, no. 3, 128, 2023.
- [47] A. Radonjic, P. Das, V. Vujicic, "Integer Codes Correcting Burst Errors within One Byte and Single Errors within Two Bytes," *Cryptogr. Commun.*, accepted for publication. <https://doi.org/10.1007/s12095-023-00687-7>.
- [48] A. Radonjic, K. Bala, V. Vujicic, "Integer Codes Correcting Double Asymmetric Errors," *IET Commun.*, vol. 10, no. 14, pp. 1691-1696, 2016.
- [49] A. Radonjic, V. Vujicic, "Integer Codes Correcting Burst and Random Asymmetric Errors within a Byte," *J. Franklin Inst.*, vol. 355, no. 2, pp. 981-996, 2018.
- [50] A. Radonjic, V. Vujicic, "Integer Codes Correcting Burst Asymmetric Errors within a Byte and Double Asymmetric Errors," *Cryptogr. Commun.*, vol. 12, no. 2, pp. 221-230, 2020.
- [51] R. Braden, D. Borman, C. Partridge, Computing the Internet Checksum, IETF RFC 1071, 1988.
- [52] L.-A. Larzon, M. Degermark, S. Pink, "UDP Lite for Real Time Applications," HP Laboratories, Tech. Rep. HPL-IRI-1999-001, 1999.
- [53] A. Singh, A. Konrad, A. Joseph, "Performance Evaluation of UDP-Lite for Cellular Video," *Proc. ACM NOSSDAV*, pp. 117-124, 2001.
- [54] E. Ryu, H. Yoo, "Audio Streaming System on Mobile Phone Using UDP-Lite Protocol," *Proc. KIPS Fall Conf.*, pp. 1357-1360, 2002.
- [55] L. Larzon et al., The Lightweight User Datagram Protocol (UDP-Lite), IETF RFC 3828, 2004.
- [56] M. Miletic, A. Radonjic, "Integer Codes Correcting Single Errors and Double Adjacent Errors," *Electron. Lett.*, vol. 59, no. 20, e12973, 2023.
- [57] R. Razavi, M. Fleury, M. Ghanbari, "Correct Bluetooth EDR FEC Performance with SEC-DAEC Decoding," *Electron. Lett.*, vol. 43, no. 22, pp. 1-2, 2007.
- [58] S. Cha, H. Yoon, "Single-Error-Correction and Double-Adjacent-Error-Correction Code for Simultaneous Testing of Data Bit and Check Bit Arrays in Memories," *IEEE Trans. Dev. Mater. Rel.*, vol. 14, no. 1, pp. 529-535, 2014.

- [59] S. Tripathi, J. Jana, J. Bhaumik, "New Low Power and Fast SEC-DAEC and SEC-DAEC-TAEC Codes for Memories in Space Application," *Integration*, vol. 89, pp. 56-72, 2023.
- [60] K. Mehlhorn, P. Sanders, "Algorithms and Data Structures: The Basic Toolbox," Springer, 2008.
- [61] S. Dravida, R. Damodaram, "Error Detection and Correction Options for Data Services in B-ISDN," *IEEE J. Set. Areas Commun.*, vol. 9, no. 9, pp. 1484-1495, 1991.
- [62] <https://www.agner.org/optimize/microarchitecture.pdf>
- [63] <https://www.nti-audio.com/en/products/measurement-microphonec>
- [64] N. Miller, "Pitch Detection by Data Reduction," *IEEE Trans. Acoust. Speech Sign. Process.*, vol. 23, no. 1, pp. 72-79, 1975.
- [65] A. B. Nielsen, L. K. Hansen, U. Kjems, "Pitch based Sound Classification," *Proc. IEEE Int. Conf. on Acoustics, Speech and Signal Processing*, pp. 788-791, 2006.
- [66] K. Pratibha, H.M. Chandrashekar, "Estimation and tracking of pitch for noisy speech signals using EMD based autocorrelation function algorithm," *Proc. 2nd IEEE Int. Conf. on Recent Trends in Electronics, Information & Communication Technology (RTEICT)*, pp. 2071-2075, 2017.
- [67] T.G. Stockham, T.M. Cannon, R.B. Ingebretsen, "Blind deconvolution through digital signal processing," *Proc. IEEE*, vol. 63, no. 4, pp. 678-692, 1975.
- [68] A. V. Oppenheim, R. W. Schaffer, "From frequency to quefrency: A history of the cepstrum," *IEEE Signal Process. Mag.*, vol. 21, no. 5, pp. 95-106, 2004.
- [69] M. Milivojević, D. Ćirić, "Extracting relevant acoustic features of the internal combustion engines," *Proc. 7th Int. Conf. on Electrical, Electronic and Computing Engineering - IcETRAN*, pp. 49-53, 2020.
- [70] H. Chander, B. Singh, R. Khanna, "Short time cepstrum analysis method for pitch estimation of an arbitrary speech signal using MATLAB," *Int. J. Res. Electron. Comput. Eng.*, vol. 5, no. 3, pp. 1-4, 2017.
- [71] T. Ananthapadmanabha, B. Yegnanarayana, "Epoch extraction of voiced speech," *IEEE Trans. Ac. Speech Sign. Proc.*, vol. 23, no. 6, pp. 562-575, 1975.
- [72] L. Sukhostat, Y. Imamverdiyev, "A comparative analysis of pitch detection methods under the influence of different noise conditions," *J. Voice*, vol. 29, no. 4, pp. 410-417, 2015.
- [73] B. Gold, L. Rabiner, "Parallel processing techniques for estimating pitch periods of speech in the time domain," *J. Acoust. Soc. Amer.*, vol. 46, no. 2, pp. 442-448, 1969.
- [74] A. de Cheveigne, H. Kawahara, "YIN, a fundamental frequency estimator for speech and music," *J. Acoust. Soc. Amer.*, vol. 111, no. 4, pp. 1917-1930, 2002.
- [75] A. Camacho, "SWIPE: A sawtooth waveform inspired pitch estimator for speech and music," Ph.D. dissertation, University of Florida, USA, 2007.
- [76] S. Gonzalez, M. Brookes, "A pitch estimation filter robust to high levels of noise (PEFAC)," *Proc. 19th Europ. Sign. Process. Conf.*, pp. 451-455, 2011.
- [77] K. Kasi, S. A. Zahorian, "Yet another algorithm for pitch tracking," *Proc. IEEE Int. Conf. Acoust. Speech Sign. Process.*, pp. 13-17, 2002.
- [78] L. N. Tan, A. Alwan, "Multi-band summary correlogram-based pitch detection for noisy speech," *Speech Commun.*, vol. 55, pp. 841-856, 2013.

- [79] H. Ba, N. Yang, I. Demirkol, W. Heinzelman, “BaNa: a hybrid approach for noise resilient pitch detection,” *Proc. IEEE Stat. Sign. Process. Workshop*, pp. 4-7, 2012.
- [80] D. Ćirić, M. Janković, M. Milenković, M. Miletić, “Cepstrum-based pitch Detection of industrial product sound”, *Proc. of 8th Int. Conf. on Electrical, Electronic and Computing Engineering - IcETRAN 2021*, pp. 45-50, 2021.
- [81] Q. Liu, N. Li, H. Jia, Q. Qi, L. Abualigah, “Modified remora optimization algorithm for global optimization and multilevel thresholding image segmentation”, *Mathematics*, vol. 10, 1014, 2022.
- [82] G. Batchuluun, N. Baek, K. Park, “Enlargement of the field of view based on image region prediction using thermal videos”, *Mathematics*, vol. 9, 2379, 2021.
- [83] N. Ortega-Sánchez, D. Oliva, E. Cuevas, M. Pérez-Cisneros, A. Juan, “An evolutionary approach to improve the halftoning process”, *Mathematics*, vol. 8, 1636, 2020.
- [84] V. Starovoitov, E. Eldarova, K. Iskakov, “Comparative analysis of the SSIM index and the Pearson coefficient as a criterion for image similarity”, *Eurasian J. Math. Comput. Appl.*, vol. 8, pp. 76-90, 2020.
- [85] Z. Almahdawie, Z. Hussain, “Performance of signal similarity measures under 1/f noise”, *J. Theor. Appl. Inf. Technol.*, vol. 96, pp. 8399-8409, 2018.
- [86] T. Sato, H. Kusaka, “Statistical intercomparison of similarity metrics in sea level pressure pattern classification”, *J. Met. Soc. Japan*, vol. 99, pp. 993-1001, 2021.
- [87] G. Renieblas, A. Nogués, A. González, N. Gómez-Leon, E. Guibelalde del Castillo, “Structural similarity index family for image quality assessment in radiological images”, *J. Med. Imag.*, vol. 4, 035501, 2017.
- [88] L. Nanni, A. Rigo, A. Lumini, S. Brahmam, “Spectrogram classification using dissimilarity space”, *Appl. Sci.* vol. 10, 4176, 2020.
- [89] P. Sinha, R. Russell, “A perceptually based comparison of image similarity metrics”, *Perception*, vol. 40, pp. 1269-1281, 2011.
- [90] G. Batchuluun, J. Koo, Y. Kim, K. Park, “Image region prediction from thermal videos based on image prediction generative adversarial network”, *Mathematics*, vol. 9, 1053, 2021.
- [91] J. Zujovic, T. Pappas, D. Neuhoff, “Structural texture similarity metrics for image analysis and retrieval”, *IEEE Trans. Image Process*, vol. 22, pp. 2545-2558, 2013.
- [92] J. Snell, K. Ridgeway, R. Liao, B. Roads, M. Mozer, R. Zemel, “Learning to Generate Images with Perceptual Similarity Metrics”, *Proc. of the IEEE Int. Conf. on Image Processing*, Beijing, pp. 4277-4281, 2017.
- [93] H. Yang, J. Wang, Y. Miao, Y. Yang, Z. Zhao, Z. Wang, Q. Sun, D. Wu, “Combining spatio-temporal context and Kalman filtering for visual tracking”, *Mathematics*, vol. 7, 1059, 2019.
- [94] A. Ewees, L. Abualigah, D. Yousri, A. Sahlol, M. Al-Qaness, S. Alshathri, M. Elaziz, “Modified artificial ecosystem-based optimization for multilevel thresholding image segmentation”, *Mathematics*, vol. 9, 2363, 2021.
- [95] A. Melbourne, G. Ridgway, D. Hawkes, “Image Similarity Metrics in Image Registration”, *Proc. of the Medical Imaging 2010: Image Processing*, pp. 962–971, 2010.
- [96] A. Nakhmani, A. Tannenbaum, “A new distance measure based on generalized image normalized cross-correlation for robust video tracking and image recognition”, *Pattern Recognit. Lett.*, vol. 34, pp. 315–321, 2013.

- [97] B. Wu, B. Zhang, W. Li, F. Jiang, "A novel method for remaining useful life prediction of bearing based on spectrum image similarity measures", *Mathematics*, vol. 10, 2209, 2022.
- [98] P. Arun, S. Madhukumar, P. Careena, "A Method for the Investigation of Bearing Vibration Based on Spectrogram Image Comparison", *Proc. of the IOP Conf. Series: Materials Science and Engineering*, vol. 396, 012044, 2018.
- [99] J. Bynum, G. Earle, D. Lattanzi, M. DeJong, J. Schooling, G. Viggiani, "A Convolutional Neural Network Approach to the Semi-Supervised Acoustic Monitoring of Industrial Facilities", *Proc. of the Int. Conf. Smart Infrastructure and Construction*, pp. 4277-4281, 2019.
- [100] B. Zhang, J. Leitner, S. Thornton, "Audio Recognition Using MEL Spectrograms and Convolution Neural Networks", *Noiselab University of California*, 2019.
- [101] R. Sharan, H. Xiong, S. Berkovsky, "Benchmarking audio signal representation techniques for classification with convolutional neural networks", *Sensors*, vol. 21, 3434, 2021.
- [102] R. Sharan, T. Moir, "Robust Audio Surveillance Using Spectrogram Image Texture Feature", *Proc. of the IEEE Int. Conf. on Acoustics, Speech and Signal Processing*, pp. 1956-1960, 2015.
- [103] J. Zhang, X. Wen, A. Cho, M. Whang, "An empathy evaluation system using spectrogram image features of audio", *Sensors*, vol. 21, 7111, 2021.
- [104] S. Kandadai, J. Hardin, C. Creusere, "Audio Quality Assessment Using the Mean Structural Similarity Measure", *Proc. of the IEEE Int. Conf. Acoustics, Speech, and Signal Processing*, pp. 221-224, 2008.
- [105] P. Arun, S. Lincon, N. Prabhakaran, "An automated method for the analysis of bearing vibration based on spectrogram pattern matching", *J. Appl. Res. Technol.*, vol. 17, pp. 126-136, 2019.
- [106] D. Van der Weken, M. Nachtegaal, E. Kerre, "An Overview of Similarity Measures for Images", *Proc. of the IEEE Int. Conf. Acoustics, Speech and Signal Processing*, pp. 3317-3320, 2002.
- [107] L. Wang, Y. Zhang, J. Feng, "On the Euclidean distance of images", *IEEE Tran. Pattern Anal. Mach. Intell.*, vol. 27, pp. 1334-1339, 2005.
- [108] <https://dcase.community/challenge2021/index>
- [109] R. Tanabe, H. Purohit, K. Dohi, T. Endo, Y. Nikaido, T. Nakamura, Y. Kawaguchi, "MIMII DUE: Sound Dataset for Malfunctioning Industrial Machine Investigation and Inspection with Domain Shifts Due to Changes in Operational and Environmental Conditions", *Proc. of the IEEE Workshop on Applications of Signal Processing to Audio and Acoustics*, pp. 21-25, 2021.
- [110] N. Harada, D. Niizumi, D. Takeuchi, Y. Ohishi, M. Yasuda, S. Saito, "ToyADMOS2: Another dataset of miniature-machine operating sounds for anomalous sound detection under domain shift conditions", *arXiv:2106.02369*, 2021.
- [111] H. Purwins, B. Li, T. Virtanen, J. Schlüter, S. Chang, T. Sainath, "Deep learning for audio signal processing", *J. Sel. Top. Sig. Proc.*, vol. 18, pp. 206-219, 2019.
- [112] M. Valero, S. Verstockt, C. Mata, D. Jimenez, L. Queen, O. Rios, E. Pastor, E. Planas, "Image similarity metrics suitable for infrared video stabilization during active wildfire monitoring: A comparative analysis", *Remote Sens.*, vol. 12, 540, 2020.

- [113] H. Mitchell, "Image similarity measures", Image Fusion Theories, Techniques and Applications, 1st ed., pp. 167-185, 2010.
- [114] A. Zaric, M. Loncaric, D. Tralic, M. Brzica, E. Domic, S. Grgic, "Image Quality Assessment - Comparison of Objective Measures with Results of Subjective Test", Proc. of the 52nd Int. Symp. ELMAR, pp. 113-118, 2010.
- [115] S. Li, Z. Li, J. Gong, "Multivariate statistical analysis of measures for assessing the quality of image fusion", Int. J. Image Data Fusion, vol. 1, pp. 47-66, 2010.
- [116] D. Ciric, Z. Peric, M. Milenkovic, N. Vucic, "Evaluating similarity of spectrogram-like images of DC motor sounds by Pearson correlation coefficient", Elektron. Ir Elektrotehnika, vol. 28, pp. 37-44, 2022.
- [117] W. Lu, C. Li, Y. Shi, X. Sun, "Image Quality Assessment Based on SIFT and SSIM", Proc. of the Chinese Conf. on Image and Graphics Technologies 2014: Advances in Image and Graphics Technologies pp. 1-7, 2014.
- [118] D. Ćirić, Z. Perić, N. Vučić, M. Miletić, "Analysis of Industrial Product Sound by Applying Image Similarity Measures", Mathematics, vol. 11, 498, 2023.
- [119] H. Purwins, B. Li, T. Virtanen, J. Schlüter, S. Chang, T. Sainath, "Deep Learning for Audio Signal Processing", J. Selected Topics Sig. Proc., vol. 13, no. 2, pp. 206-219, 2019.
- [120] J. Abeßer, "A Review of Deep Learning Based Methods for Acoustic Scene Classification", Appl. Sci., vol. 10, no. 6, 2020.
- [121] C. Roletscheck, T. Watzka, A. Seiderer, D. Schiller, E. André, "Using an Evolutionary Approach to Explore Convolutional Neural Networks for Acoustic Scene Classification", Detection Class. Acoust. Scenes Events Workshop, pp. 1-5, 2019.
- [122] O. AlShorman, F. Alkahatni, M. Masadeh, M. Irfan, A. Glowacz, F. Althobiani, J. Kozik, W. Glowacz, "Sounds and Acoustic Emission-Based Early Fault Diagnosis of Induction Motor", Adv. Mech. Eng., vol. 13, no. 2, pp. 1-19, 2021.
- [123] I. W. Liu, Z. Wang, X. Liu, N. Zeng, Y. Liu, F. E. Alsaadi, "A Survey of Deep Neural Network Architectures and Their Applications", Neurocomputing, vol. 234, pp. 11-25, 2017.
- [124] A. Khamparia, D. Gupta, N. G. Nguyen, A. Khanna, B. Pandey, P. Tiwari, "Sound Classification Using Convolutional Neural Network and Tensor Deep Stacking Network", IEEE Access, vol. 7, pp. 7717-7727, 2019.
- [125] J. Lee, J. Park, K. L. Kim, J. Nam, "Sample-level Deep Convolutional Neural Networks for Music Auto-tagging Using Raw Waveforms", Sound Music Comput. Conf., Espoo, pp. 1-7, 2017.
- [126] J. Schlüter S. Böck, "Improved Musical Onset Detection with Convolutional Neural Networks", IEEE Int. Conf. Acoust. Speech Sig. Proc., pp. 6979-6983, 2014.
- [127] Z. Ouyang, H. Yu, W-P. Zhu, B. Champagne, "A Fully Convolutional Neural Network for Complex Spectrogram Processing in Speech Enhancement", IEEE Int. Conf. Acoust. Speech Sig. Proc., pp. 5756-5760, 2019.
- [128] Y. Zeng, H. Mao, D. Peng, Z. Yi, "Spectrogram Based Multi-Task Audio Classification", Multimedia Tools App., vol. 78, no. 3, pp. 3705-3722, 2019.
- [129] J. Pons, O. Slizovskaia, R. Gong, E. Gomez, X. Serra, "Timbre Analysis of Music Audio Signals with Convolutional Neural Networks", arXiv:1703.06697, 2017.

- [130] O. Abdel-Hamid, A. Mohamed, H. Jiang, L. Deng, G. Penn, D. Yu, "Convolutional Neural Networks for Speech Recognition", *IEEE/ACM Trans. Audio Speech Lang. Proc.*, vol. 22, no. 10, pp. 1533-1545, 2014.
- [131] Y. LeCun, L. Bottou, Y. Bengio, P. Haffner, "Gradient-based Learning Applied to Document Recognition", *Proc IEEE*, vol. 86, no. 11, pp. 2278-2324, 1998.
- [132] I. Goodfellow, Y. Bengio, A. Courville, "Deep Learning (Adaptive Computation and Machine Learning Series)", MIT Press, 2016.
- [133] M. Espi, M. Fujimoto, K. Kinoshita, T. Nakatani, "Exploiting Spectro-temporal Locality in Deep Learning Based Acoustic Event Detection", *EURASIP J. Aud. Speech Music Proc.*, article no. 26, 2015.
- [134] D. Ćirić, M. Janković, M. Miletić, "Sound Based DC Motor Classification by a Convolution Neural Network", *Proc. of 57rd Int. Sci. Conference on Information, Communication and Energy Systems and Technologies*, pp. 1-4, 2022.
- [135] L. Almeida, "Multilayer perceptrons (Handbook of Neural Computation)", IOP Publishing Ltd. and Oxford University Press, 1997.
- [136] M. Popescu, V. Balas, O. Olaru, N. Mastorakis, "The Backpropagation Algorithm Functions for the Multilayer Perceptron", *Proc. of the 11th WSEAS Int. Conf. on Sustainability in Science Engineering*, pp. 28-31, 2009.
- [137] Y. LeCun, B. Boser, J. Denker, D. Henderson, R. Howard, W. Hibbard, L. Jackel, "Backpropagation applied to handwritten zip code recognition", *Neural Comput.*, vol. 1, no. 4, 1989.
- [138] X. Glorot, Y. Bengio, "Understanding the difficulty of training deep feedforward neural networks", *J. Machine Learn. Res.*, vol. 9, pp. 249-256, 2010.
- [139] R. Rojas, „Neural Networks: A Systematic Introduction“, Springer Verlag, 1996.
- [140] C. Gershenson, „Artificial Neural Networks for Beginners“, Computer Science, ArXiv, 2003.
- [141] M. Miletić, M. Rančić, "Primena veštačke inteligencije u procesu prerade otpadnih voda", *Naučno-stručni časopis DIT*, no. 38, pp. 75-81, 2022, ISSN 0354-7140, COBISS.SR-ID 105108999.

10. Прилози

Прилог 1. Python код за читање вредности са PLC-а и унос вредности температуре у MySQL базу података

```
import socket, sys, time
import threading
import random
import ctypes
def timeMs():
    t = int(round(time.time() * 1000))
    return t
def timePassedMs(time):
    return (timeMs() - time)
def createEncKey(pin):
    tmp = pin.split("-")
    key = 0
    if len(tmp) == 8:
        i = 0
        for p in tmp:
            key |= ((int(p) & 0x04) >> 2) << (7 - i)
            i += 1
        return key
    else:
        print "!!!Invalid pin!!!"
        pass
def sendAllEncrypt(s, data, pin):
    key = createEncKey(pin)
    i = 0
    sendData = chr(ord(data[i]) ^ key)
    for i in range(1, len(data)):
        sendData += chr(ord(data[i]) ^ key)
    s.sendall(sendData)
class HmiHiDistanceCommunication(object):
    def __init__(self):
        self.Host = ""
        self.Port = 10001
        self.txOn = False
        self.t = None
        self.values = ""
    def setDeviceIp(self, Ip, Pin, Port):
        self.Host = Ip
        self.Pin = Pin
        if Port == "PLC":
            self.Port = 10001
        else:
            self.Port = int(Port)
    def attachToHost(self):
        self.s = socket.socket(socket.AF_INET, socket.SOCK_DGRAM)
        try:
            port = int(self.Port)
        except ValueError:
            port = socket.getservbyname(self.Port, 'udp')
        self.s.settimeout(0)
        self.s.connect((self.Host, port))
        print "logOut:", self.Host, port
    def getHostValues(self):
        if not self.txOn:
            self.txOn = True
            if self.s != None:
                data = chr(0xfe)
                data += chr(0x04)
```

```

data += chr(0xAF)
data += chr(0xFE)
t = timeMs()
while(self.txOn):
    try:
        buf = self.s.recv(256)
    except:
        buf = ""
    if (len(buf) == 128):
        self.values = []
        i = 0
        while i < 128:
            number = int(ord(buf[i]) + (ord(buf[i + 1]) << 8) + (ord(buf[i + 2]) << 16) + (ord(buf[i + 3]) << 24))
            self.values.append(ctypes.c_long(number).value)
            i = i + 4
        if((timePassedMs(t) > 350)):
            t = timeMs()
            sendAllEncrypt(self.s, data, self.Pin)
        self.s.close()
def commStart(self):
    print "commStart"
    if self.t == None:
        self.t = threading.Thread(target = self.getHostValues, args = ())
        self.t.daemon = True
        self.t.start()
def commStop(self):
    self.txOn = False

```

Прилог 2. Python код за унос измерених вредности температура и pH параметара у MySQL базу података

```
def upisubazu(t1, t2, t3):
    import MySQLdb
    db= MySQLdb.connect(host="localhost", user="root", passwd="*****",
    db="trayalotpadnevide")
    try:
        cursor = db.cursor()
        stmt = "INSERT INTO temperature (datum, temperatura1, temperatura2, temperatura3, pHkada1, pHkada2,
        pHkada3,Redox1, Redox2) VALUES ("
        stmt = stmt + "2017-04-23 16:33:26"
        stmt = stmt + ", "
        stmt = stmt + t1
        stmt = stmt + ", "
        stmt = stmt + t2
        stmt = stmt + ", "
        stmt = stmt + t3
        stmt = stmt + ", "
        stmt = stmt + pH1
        stmt = stmt + ", "
        stmt = stmt + pH2
        stmt = stmt + ", "
        stmt = stmt + pH3
        stmt = stmt + ", "
        stmt = stmt + Rx1
        stmt = stmt + ", "
        stmt = stmt + Rx2
        stmt = stmt + ");"
        print stmt
        cursor.execute(stmt)
        print "Record added!"
        cursor.close ()
    except Exception, e:
        print e
        print "error"
        pass
def GetLadderValuesTest():
    host = "192.168.1.50"
    buff = ""
    comm = HmiHiDistaceCommunication()
    comm.setDeviceIp(host, "96-49-220-181-171-37-165-174", "PLC")
    comm.attachToHost()
    comm.commStart()
    t = 0
    while(1):
        print "--",comm.values
        if len(comm.values) > 0:
            # $ varijable
            tmp1=str(comm.values[0])
            tmp2=str(comm.values[1])
            tmp3=str(comm.values[2])
            pH1=str(comm.values[3])
            pH2=str(comm.values[4])
            pH3=str(comm.values[5])
            Rx1=str(comm.values[6])
            Rx2=str(comm.values[7])
            upisubazu(tmp1, tmp2,tmp3,pH1,pH2,pH3,Rx1,Rx2)
            break
        time.sleep(0.3)
        t += 1;
    if t > 20:
        comm.commStop()
        break
```

```
def main():
    GetLadderValuesTest();
if __name__ == '__main__':
    main()
```

Прилог 3. MATLAB код за проналажење коефицијената C_i

```

function [ ] = Coefficients
B=input('\nPlease, specify a byte length (in bits): \nB = ');
Q=2^B-1;
Z2=[];
Z1=[];
s=0;
for i=0:B-2
    for z1=-1:2:1
        for z2=-1:2:1
            s=s+1;
            Z2(s)=mod((2^i)*z1+(2^(i+1))*z2,Q);
        end
    end
end
z=0;
for i=0:B-1
    for z3=-1:2:1
        z=z+1;
        Z1(z)=mod((2^i)*z3,Q);
    end
end
X=union(Z1,Z2);
L=length(X);
X1=X;
Coeff=[1];
r=1;
for m=2:Q-1
    A=[];
    z=0;
    for i=1:L
        A(i)=mod(m*X(i),Q);
    end
    A1=setdiff(A,0);
    if (nnz(A1)==nnz(X))&(nnz(setdiff(A1,X1))==nnz(A1))
        A2=[];
        A3=[];
        t2=0;
        for i=0
            for j=B-1
                for r1=-1:2:1
                    for r2=-1:2:1
                        t2=t2+1;
                        A2(t2)=mod(m*r1*(2^i)+Coeff(r)*r2*(2^j),Q);
                        A3(t2)=mod(m*r1*(2^i)+Coeff(1)*r2*(2^j),Q);
                    end
                end
            end
        end
        if(nnz(setdiff(A2,0))==nnz(A2))&(nnz(setdiff(A2,X1))==nnz(A2))&(nnz(setdiff(A2,A1))==nnz(A2))&
            (nnz(setdiff(A3,0))==nnz(A3))& (nnz(setdiff(A3,X1))==nnz(A3))&(nnz(setdiff(A3,A1))==nnz(A3))
            r=r+1;
            Coeff(r)=m;
            X1=union(X1,A1);
            X1=union(X1,A2);
        end
    end
end
Ci=setdiff(Coeff,1);
fprintf(1,'The following coefficients are found: %d\n',Ci);

```

Прилог 4. Python код за снимање и одсецање звукова мотора

```
import pyaudio
import wave

chunk = 1024 # Record in chunks of 1024 samples
sample_format = pyaudio.paInt16 # 16 bits per sample
channels = 2
fs = 44100 # Record at 44100 samples per second
seconds = 5
#filename = "recordmicrophone.wav"
filename = "m17_190113_134640_dir1.wav"

p = pyaudio.PyAudio() # Create an interface to PortAudio

print('Recording')

stream = p.open(format=sample_format,
                channels=channels,
                rate=fs,
                frames_per_buffer=chunk,
                input=True)

frames = [] # Initialize array to store frames

# Store data in chunks for 3 seconds
for i in range(0, int(fs / chunk * seconds)):
    data = stream.read(chunk)
    frames.append(data)

# Stop and close the stream
stream.stop_stream()
stream.close()
# Terminate the PortAudio interface
p.terminate()

print('Finished recording')

# Save the recorded data as a WAV file
wf = wave.open(filename, 'wb')
wf.setnchannels(channels)
wf.setsampwidth(p.get_sample_size(sample_format))
wf.setframerate(fs)
wf.writeframes(b''.join(frames))
wf.close()
```

Прилог 5. Python код за конверзију звука у слику

```
import inspect
import os
import soundfile
from scipy import signal
import librosa.display
# import matplotlib.pyplot as plt
import numpy as np
# import scipy.io.wavfile as wf
# import addcopyfighandler

def calcSpecFromWav(inputt, nMels, sampleRate, frmSz, frmStp, topDb):
    specS = np.abs(librosa.stft(y=inputt, n_fft=frmSz, hop_length=frmStp, win_length=frmSz,
window=signal.windows.hann))
    melSpecS = librosa.feature.melspectrogram(S=specS, sr=sampleRate, n_mels=nMels)
    logSpecS = librosa.core.power_to_db(melSpecS, top_db=topDb)
    minv = logSpecS.min()
    maxv = logSpecS.max()
    ampl = abs(maxv - minv) + 1e-20
    logSpecS = (logSpecS - minv) / ampl
    return logSpecS

''' START '''
# if not os.path.exists('NpyImages'):
#     os.makedirs('NpyImages')
frmSz = 1024 # 256, 512, 1024, 2048
frmStp = int(frmSz / 4) # / 4
nMels = 96 # 12 #96 #int(frmSz/2) + 1 #24 #96
topDb = 80.0
# plot_type = 'mel' # Chose spectrogram
# MyPath = os.path.dirname(os.path.abspath(inspect.getfile(inspect.currentframe())) + "\\Direction1"
# for audio in os.listdir(MyPath):
#     if (audio.endswith('wav')):
#         y1, sampleRate1 = soundfile.read(MyPath + "\\" + audio)
#         ConvertAndSaveNpy(frmSz, frmStp, nMels, topDb, plot_type, y1, sampleRate1, audio)
audio = "m17_190113_134640_dir1"
y1, sampleRate1 = soundfile.read(audio + ".wav")
# ConvertAndSaveNpy(frmSz, frmStp, nMels, topDb, plot_type, y1, sampleRate1, audio)
MelS1 = calcSpecFromWav(inputt=y1, nMels=nMels, sampleRate=sampleRate1, frmSz=frmSz, frmStp=frmStp,
topDb=topDb)
# MelS1 ovo je slika koja se učitava u CNN mrežu
np.save(audio + 'Image', MelS1)
Industrijske kamere
```