



SAKARYA ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ DERGİSİ
SAKARYA UNIVERSITY
JOURNAL OF SCIENCE



SAKARYA
ÜNİVERSİTESİ

PART A

21(1), 2017



Doç.Dr. Emrah DOĞAN



saujs@sakarya.edu.tr



www.dergipark.gov.tr/saufenbilder

ŞUBAT / FEBRUARY 2017

e-ISSN: 2147-835X

SAKARYA ÜNİVERSİTESİ FEN BİLİMLERİ ENSTİTÜSÜ DERGİSİ EDITÖR LİSTESİ

Editörler: Ahmet Zengin, Cüneyt Bayılmış, Kerem Küçük

Bölüm Editörleri

BİLGİSAYAR MÜH. + SİBER GÜVENLİK		ELEKTRİK-ELEKTRONİK MÜH.		BİYOMEDİKAL MÜH.		BİLİŞİM SİSTEMLERİ	
AHMET ZENGİN	Sakarya Uni.	İHSAN PEHLİVAN	Sakarya Uni.	MUSTAFA BOZKURT	Sakarya Uni.	HALİL YIĞIT	Kocaeli Uni.
CÜNEYT BAYILMIŞ	Sakarya Uni.	YILMAZ UYAROĞLU	Sakarya Uni.	ARIF OZKAN	Düzce Üniv		
KEREM KÜÇÜK	Kocaeli Uni.	SEZGİN KAÇAR	Sakarya Uni.				
DEVRİM AKGÜN	Sakarya Uni.						

A GRUBU

ŞUBAT,
AĞUSTOS

Editörler: Emrah Doğan, Beytullah Eren

Bölüm Editörleri

İNŞAAT MÜH.		JEYFİZİK MÜH.		ÇEVRE MÜH.		MİMARLIK	
EMRAH DOĞAN	Sakarya Uni.	MURAT UTKUCU	Sakarya Uni.	BEYTULLAH EREN	Sakarya Uni.	TAHSİN TURĞAY	Sakarya Uni.
PETER CLAISSE	Coventry Uni.	ALİ PINAR	Boğaziçi Uni.	NİLGÜN BALKAYA	İstanbul Uni.		
JAMEL KHATİB	Uni. Wolverhampton			A.S. ERSES YAY	Sakarya Uni.		
OSMAN KIRTEL	Sakarya Uni.			AHMET AYĞÜN	Bursa Teknik		
ALİ SARIBIYIK	Sakarya Uni.						
İnan Keskin	Karabük Uni.						
ERTAN BOL	Sakarya Uni.						

B GRUBU

NİSAN, EKİM

Editörler: Serkan Zeren

Bölüm Editörleri

MAKİNA TASARIMI		METALURJİ VE MALZEME MÜH.		MEKATRONİK MÜH.		MAKİNA VE ENERJİ	
HÜSEYİN PEHLİVAN	Sakarya Uni.	MUSTAFA KURT	Ahi Evran Uni.	SERKAN ZEREN	Kocaeli Uni.	YUSUF ÇAY	Sakarya Uni.
		ÖZGÜL KELEŞ	İstanbul Teknik Uni	FARUK YALÇIN	Sakarya Uni.		
		ZAFER BARLAS	Sakarya Uni.	BARIŞ BORU	Sakarya Uni.		
OTOMOTİV		YANGIN + APET		ENDÜSTRİ+KALİTE YON			
ÇAN HAŞIMOĞLU	Sakarya Uni.			A. SERHAT DEMİR	Sakarya Uni.		
VEZİR AYHAN	Sakarya Uni.			ÖZER UYGUN	Sakarya Uni.		
İDRİS CESUR	Sakarya Uni.						

C GRUBU

HAZİRAN,
ARALIK

Editörler: Mehmet Nebioğlu, Şevket Gür

Bölüm Editörleri

MATEMATİK		FİZİK		KİMYA		BİYOLOJİ	
MEHMET ÖZEN	Sakarya Uni.	M. van de VENTER	Nelson Mandela M. Uni	MEHMET NEBİOĞLU	Sakarya Uni.	NAZAN DENİZ YÖN ERTUĞ	Sakarya Uni.
S. SALUR	Rochester Uni.	SADIK BAĞCI	Sakarya Uni.	MURAT TUNA	Sakarya Uni.	HÜSEYİN AKSOY	Sakarya Uni.
ŞEVKET GÜR	Sakarya Uni.						
GENETİK MÜH.		NANOBİLİM VE NANO MÜH.		ZİRAAT MÜH.		GIDA MÜH.	
TUEN OGRAS	TÜBİTAK	İLKAY ŞİŞMAN	Sakarya Uni.	RAMAZAN MERAL	Bingöl Uni.	DİLEK ANGIN	Sakarya Uni.
		MAHMUT ÖZACAR	Sakarya Uni.				

SAKARYA ÜNİVERSİTESİ FEN BİLİMLERİ
ENSTİTÜSÜ DERGİSİ YAYIN PLANI

İÇİNDEKİLER

- 1-Artificial Intelligence Library For Html5 Based Games: Dignityai --- Berkan Uslu, Ecir Uğur Küçüksille
- 2-Jenerik Cordic Algoritmasının Fpga'da Donanımsal Gerçeklenmesi --- Suhap Şahin, Burcu Kir Savaş
- 3-Geniş Bant Konum Belirleme Sistemi Performans Analizi Ve İyileştirilmesi --- Kerem Küçük
- 4-PID Denetleyici Uygulamaları İçin Yeni Bir Online Deney Seti Tasarımı --- Sezgin Kaçar, Ali Fuat Boz, Burak Arıcıoğlu, Harun Tekin
- 5-Optik Ağların Afet Sonrası Onarımı İçin Çoklu Tamirci Problemi --- Ferhat Dikbiyik
- 6-Karaciğer Mikroarray Kanseri Verisinin Sınıflandırılması İçin Genetik Algoritma Kullanarak Anfis'in Eğitilmesi --- Bülent Haznedar, Mustafa Turan Arslan, Adem Kalinli
- 7-A Novel Learning Algorithm To Optimize The Fuselage Drag Coefficient --- Tugrul Oktay
- 8-Dış Ortam Kablosuz Networklerde Uygun Sinyal Dağılımının Görünürlük Analizi Yöntemiyle Belirlenmesi --- Mustafa Coşar, Fazli Engin Tomuş, İbrahim Murat Ozulu, Veli İlçi

Artificial intelligence library for html5 based games: DignityAI

Berkan Uslu¹, Doç. Dr. Ecir Uğur Küçüksille^{2*}

21.04.2016 Geliş/Received, 08.08.2016 Kabul/Accepted

doi: 10.16984/saufenbilder.22281

ABSTRACT

Today, acceleration of internet and common use of web pages, revealed the necessity of work with any browser smoothly for each application without of requirement of any plug-in. Generally, HTML5 is a new body of standards which is formed with the combination of CSS and JavaScript. In this context, by analysing game engines developed for HTML5, their features and advantages are investigated. Although, these game engines are close to catch up with the level of popular game engines, it is seen that none of artificial intelligence library was developed for HTML5 based games up to now. In this study, DignityAI artificial intelligence library is developed to fill this deficiency. Developed library has ability to be integrated to all HTML5 games independently from game engine and to add artificial intelligence dynamics to these games.

Keywords: HTML5, WebGL, Artificial Intelligence Library, DignityAI, Computer Games.

Html5 tabanlı oyunlar için yapay zeka kütüphanesi: DignityAI

ÖZ

Günümüzde internetin hızlanması ve web sayfalarının yaygınlaşması, her uygulamanın her tarayıcıda herhangi bir eklenti gerektirmeden sorunsuzca çalışması gerekliliğini ortaya çıkarmıştır. HTML5 genel olarak, CSS ve Javascript'in birleşimi ile oluşan yeni bir standartlar bütünüdür. Bu noktada, HTML5 için geliştirilen oyun motorları incelenerek; özellikleri ve sundukları olanaklar araştırılmıştır. Bu oyun motorları; popüler oyun motorlarının seviyesini yakalamaya yakın olmalarına rağmen, herhangi bir yapay zeka kütüphanesinin şimdiye kadar HTML5 tabanlı oyunlar için geliştirilmemiş olduğu görülmüştür. Bu çalışmada, söz konusu eksikliği gidermek amacı ile DignityAI yapay zeka kütüphanesi geliştirilmiştir. Geliştirilen kütüphane, oyun motorundan bağımsız olarak tüm HTML5 oyunlarına entegre edilebilen ve bu oyunlara yapay zeka dinamiklerini katabilme yeteneğine sahip bir kütüphanedir.

Anahtar kelimeler: HTML5, WebGL, Yapay Zeka Kütüphanesi, DignityAI, Bilgisayar Oyunları.

¹ Süleyman Demirel Üni., Mühendislik Fak., Bilgisayar Müh., ISPARTA – berkan.uslu@gmail.com

² Süleyman Demirel Üni., Mühendislik Fak., Bilgisayar Müh., ISPARTA – ecirkucuksille@gmail.com

* Sorumlu Yazar / Corresponding Author

1. INTRODUCTION

HTML5 and WebGL are among the most popular and latest technologies which are especially focused by the World in recent years. Also, with each research on game developing field, which became stronger by accelerating, increases its invaluable position. Besides, the topic of artificial intelligence, which the large technology companies interested in, perhaps can be shown in the first place in addition to the rising popularity of this valuable research subjects. This situation has gained even more importance by the help of personal assistant with artificial intelligence published by three technology giants such as Apple, Microsoft, and Google.

In this research, first of all, the development of artificial intelligence systems in games and current artificial intelligence libraries are described by considering the history of artificial intelligence in computer games. During the research, all games that can be integrated with the artificial intelligence engine library were analyzed independent of any game engine. However, it was observed that there was no such library that can be used for games based on HTML5. In this point, developed DignityAI library have the feature of being first of its kind. Furthermore, it was observed that each developed artificial intelligence library focuses on a particular type. In this context, the investigations are carried out on the relationship between artificial intelligence and game types. It is centered upon the action (FPS, TPS), role-playing game (RPG), real-time strategy (RTS), adventure, platform, sports and racing from game types. The use of methods of finite state Machine(FSM), hierarchical artificial intelligence, fuzzy state machines (FuSM), behavior trees, path-finding, data driven systems and fuzzy logic artificial intelligence has seen and some information is given.

Subsequently, information about the development of HTML5 and WebGL is given. Then, HTML game engines are examined, the stable and current most popular game engines, which can develop both 2D and 3D games, are handled. Investigated game engines include Construct 2, ImpactJS, PixiJS, PlayCanvas, Three.js, Phaser, Kiwi.js and enchant.js. The sample game in this study is developed by using PlayCanvas game engine.

Finally, information about general structure and all classes of DignityAI artificial intelligence library within the scope of such study is given. In addition, a small game developed by using this library is presented at the end of the study.

2. LITERATURE REVIEW

Wexler, in his work, firstly made an introduction to the history of computer games and examined the development of the artificial intelligence in the computer games. Although the main scrutinized issue was artificial

intelligence in computer games, he studied the artificial intelligence system of Black and White, Lionhead Studios' game, and where can the artificial intelligence come in games [1].

Middleton, in his work, observed the history of artificial intelligence in computer games and development of artificial intelligence systems in the last 25 years of history. In the study, he made detailed examination and comparison from chess artificial intelligence designed by Alan Turing in 1950's to Deep Blue computer that defeated Garry Kasparov, from the famous RPG game Dungeons&Dragons to another legendary RTS game Dune II, from Age of Empires series to artificial intelligence systems of Valve's most popular game in the FPS genre Half Life [2].

Stall, in his work, proceed through a very natural example experienced by him to explain the finite state machines and described the finite state machines in a very clear manner [3].

Schwab, in his work, made a very detailed investigation on the details of developing AI (artificial intelligence) game engine from simple components like navigation, decision-making and inputs to detailed research of game genres and components, from the detailed method analysis such as finite state machines, fuzzy state machines, messaging systems to complete AI game development. Some advanced issues such as genetic algorithms, neural networks, fuzzy logic, behavior trees were described particularly in his book [4].

Lubbers et al., in his work, elaborated on details of HTML programming from HTML and XML which is the basic components of HTML5 to presently JavaScript and CSS that comprise HTML5. He also searched intimately all API libraries of HTML5 which was widely used and had not been examined yet, besides with such review it was given useful examples to learn what browsers are supported by which API [5].

Freeman, in his work, scrutinized JavaScript and CSS that constituent the HTML5 and revealed the detailed guide to use the full power of HTML5 by using such technologies. Focusing on HTML programming he mentioned the methods of forms and form verification and after referred the detailed CSS resource [6].

Parisi, in his work, opened the doors of the world's of WebGL and described the formation and history of the WebGL up to today. Primarily making an introduction to WebGL and then by using Three.js game engine he developed a simple WebGL application. He referred the use of WebGL with graphics, animations, 2D and 3D environments for promotion and developed the sample WebGL game [7].

Rabin, in his work, expressed the artificial intelligence method that depends on the level of detail (LOD) which is a new artificial intelligence approach in RPG games. He mentioned it was an effective technique that speeded up the artificial intelligence system together with the adaptation to artificial intelligence of such technique in graphic programming and that helping to reduce the CPU load [8].

Compton et al., in his work, studied on procedural section design, which is one of the most difficult artificial intelligence subjects, in the platform games and developed the new algorithm for section design improving four-level method [9].

Beirne, in his work, handled the artificial intelligence in the racing games and explained the development of tuning and car modification artificial intelligent with their most used models [10].

Moreover, some projects on github.com has been added within the works in this field by referring some parts of artificial intelligence in HTML5 games.

Gordon, in his work, presents us a framework called Javascript State Machine and focused on finite state machines (FSM) [11].

Cowart, in his work, described behavior trees creating with Machine.js project focused on finite state machines (FSM) similarly [12].

Xu, in his work, just focused on path-finding and studied Pathfinding.js project which is one of the best projects that is specialized in this area by hosting several search algorithms [13].

3. ARTIFICIAL INTELLIGENCE IN COMPUTER GAMES

3.1. History of Artificial Intelligence in Games

Computer games arose with the game named “Tennis for Two” in 1958 which was developed by William Higinbotham, who was a researcher in Brookhaven National Laboratory. This first game developed by William, was only possible to be played with an oscilloscope. The first game that can be played on computer was “Spacewar” developed by Steve Russell from MIT. In 1970s, Nolan Bushnell and Ted Dabney, who will later be the founder of Atari, developed the game named “Computer Space” which would be the first video arcade game. In 1980’s, the first 3D game Battlezone was developed by US government for use of military training and 4 years later namely in 1984, Nintendo game console provided computer games to commercially enter to houses by hitting the market. Afterwards, it became a huge industry together with the acquisition of PlayStation by

Sony and spread of computer games in personal computers [1].

When looking the history of artificial intelligence in computer games, it can be said it was revealed firstly on board game genres. The artificial intelligence for chess game in 1950s developed by Alan Turing and Claude Shannon, who are the fathers of artificial intelligence, is shown as the first example of it. In 1952, the checker game “The Samuel Checkers-playing Program” developed by Arthur Samuel from IBM is known as the first game to be self-learning and considered within the earliest examples of artificial intelligence [2].

In 1990s, the fictionalizing period of more intelligent systems began with the use of artificial intelligence in computer games and developing processor technology. Many games in “Turn-base Strategy/ TBS” genres (Chess, Checker, Go etc.) are considered as the first game genre which the artificial intelligence was used and “Role-Playing Games /RPG” genre follow it. In 1992, with Dune II’s, which is developed by Westwood Studios and known as the first example of the “Real-Time Strategy /RTS” genre, the working requirement of new and faster algorithms on artificial intelligence real-time calculations and decision-making structures appeared.

In 1998, Half-Life game in “First-person shooter /FPS” genre developed by Valve is varied as the one of the innovative games in artificial intelligence. One of the major innovations on Half-Life game is the use of new genre called “schedule-driven state machine” instead of behavior model that is named finite state machine used on artificial intelligent games and composed of limited numbered states, transition between states, actions.

In 1999, the improving requirement of more intelligent “non- player characters” (NPCs) of developed artificial intelligence in RPG, RTS and FPS game genres was noticed, revealing the games such as Age of Empires II and Unreal Tournament and it began the process of coming up today the use of very high level features.

Nowadays, we are moving forward to a system where the navigation algorithms are risen to next levels, where act and tracking sensors responds faster and the game characters that act and think like humans can be programmed. In a computer game to be created in future, it will be possible to create games with super intelligent NPCs in which the artificial intelligent characters take over the game and all of the humanity struggles to beat the artificial intelligent characters in the game.

3.2. Artificial Intelligence Libraries Used in Games

Many different artificial intelligence libraries have been used in games and until now most of them were integrated to game engines and not functional without using the game engine. The main point to study is the artificial intelligence libraries that work independent from game engines. DignityAI developed in this study is designed with completely independent structure from the game engine and can be applied to all game engines.

Kythera artificial intelligence library is a library which is well known in present days and started to be used in large games. Star Citizen is the leading of these games. It gets ahead of other libraries with respect to both usual functions, such as detection, target selection, hiding, group coordination, and dynamic navigation and behavior trees. Kytera, was designed to be a system based on C++ that supports Windows, Mac and Linux with 32-bit and 64-bit processor architecture and to be integrated to all game engines. However, it cannot be integrated with the HTML5.

Havok AI artificial intelligence library is a library that focuses on the main navigation functions like path-finding and path-following. One of its most significant features is to create automatic navigation mesh. In spite of the fact that Havok AI was developed by using C++, it can still give support to all leader platforms. However, there is no integration with the HTML5.

Rain artificial intelligence library is developed for Unity game engine but it is sold and distributed independent of it. It consists of the advanced navigation features like automatic navigation meshing, path-following and path finding. However, it has no HTML5 support. WebGL export option was improved with Unity 5 but it doesn't give native HTML5 output.

Kynapse artificial intelligence library is a large artificial intelligence library which is developed by Autodesk Company and nowadays separated to two products named Gameware Navigation and Gameware Cognition. Gameware navigation is the artificial intelligence library searching for options to the requirements of usual path-finding and navigation. Gameware Cognition is a tool that helps to create behavior trees supporting the visual programming platform.

PathEngine artificial intelligence library is the artificial intelligent library that focuses on the best path-finding algorithms that is improved until now. PathEngine which has a very well designed action model with the features of dynamic navigation mesh creation, obstacle recognition, crush monitoring proved itself by being used by many big projects.

Masa Life artificial intelligence library is an artificial intelligent library that focuses on decision-making. Masa Life which is developed with C++ that works on Windows and is a small integration of Unity Windows integration, approaches mostly to subjects such as decision-making mechanisms, behavior trees and navigation. In addition to this, it has the working fields like reasoning, information provision and processing.

Cyntient artificial intelligence library is an artificial intelligent library developed for the game industry. It is a system that provides virtual characters' to analyze by learning each other's behaviors and to take action. The aim of Cyntient is to develop an open world space game named Galak-Z through providing a realistic experience by creating intelligent and emotional characters.

As it mentioned above, there are many artificial intelligence libraries and additional to these there are integrated artificial intelligence modules inside the game engines. But, because none of these systems can be integrated to technologies such as HTML5 and WebGL at one point they are platform dependent and require advanced systems to improve. In this manner, improved DignityAI library contains many classic artificial intelligent items such as path-finding, decision-making, behavior and task managing that focuses on HTML5 and WebGL.

Dignity AI is completely open source coded and with this aspect it differs from other artificial intelligence libraries. Besides, it is aimed to be use in all browser based systems with the power of HTML5 and designed in architectural structure to be used for nearly all game genres with powerful basis.

3.3. Artificial Intelligence Relationship with Game Genres

Most of game engines designed until now carried out their works following the way based on to develop games belong to specific a genre and to fulfill certain functions within these games. One of the major factors in the issue being specific to genre can be summarized with that the games are very complex compared with the other systems and game and/or game engine developers focuses on the ways to solve the problems by dividing them into pieces.

Game genres customized the artificial intelligence libraries while shaping the game engines. Such an extent that there are unique different mechanics of each game. In following each paragraph, the relationship between artificial intelligence and game genres are described.

Action (FPS, TPS) games are generally developed in game genres of First Person Shooter (FPS) and Third Person Shooter (TPS). The path finding property is one of the main mechanics of action games. The other property is

EnemyAI term it can be called enemy artificial intelligence. In this type of action games, there usually is an enemy that can act differently in certain intelligence levels. Behavior trees must be present for these behaviors. They must know what kind of response they should give against which kind of events and their decision-making mechanism must have been developed. Since the action games were designed for single player, all other items except the player must have artificial intelligence. It is one of the most widely used game genres of artificial intelligence with this aspect.

Role-playing games (RPGs) is emerged as the game genres that have existed since the first release of computers. RPG games can be described as which is played with huge maps and the level reaching of the player by performing certain duties. However, although this game style is defined as simple as this, it evolved into very complex games and by including many mechanics and moving to online side as well it created MMORPG (Massively Multiplayer Online Role Playing Games). The artificial intelligence in RPG games begins with the path-finding of controlled character and going from one place to another. Inventory list of the character exists, many various types of materials can be found in this inventory list such as weapons, foods, clothes. The character learns with trial which weapon to be used when facing against the enemy to be killed, and also enemy can have weapons and spell power. However, these characters can generally manage the basic actions like using weapons and detection by hosting simple behavior trees. The main feature that separates RPG type games from the other types is the multitude of interactive objects.

One of the artificial intelligence techniques used in RPG games is a method named artificial intelligence according to level of detail (LOD) and used inside the games that occurs on huge maps like Baldur's Gate series. CPU overuse can be hindered by using this applied technique preventing the operation of the artificial intelligence of the objects away from the character. Thus, faster and more stable artificial intelligence systems can be designed [8].

Real-time strategy games (RTS) is perhaps one of the game types where the artificial intelligence used most widely. While any army, city etc. that is managed by strategy is controlled by player, items such as the army, city are managed by artificial intelligence. RTS games are one of the most difficult games to develop with regards to artificial intelligence. Because CPU must graphically process the items like characters that are moving and buildings that are formed while it is busy with the artificial intelligence. Additional to optimization difficulties, it is very detailed in terms of diversity of artificial intelligent items. In this game genre, the finite state machines is frequently used for static tasked units. It profits from behavior trees and path-finding abilities. Nevertheless, the structure named Fuzzy State Machine (FuSM) was used in modeling of strategic units. Fussy state machines have the

configuration that make a decision by calculating various situations inside the system. Another case that is used in RTS game genre is the structure named the Hierarchical AI. This structure can be illustrated as follows: when attacking to somewhere it is necessary to fight other enemies that is faced even if they are not the main target. In this case, the new task will be taken hierarchically and it is proceeded to the main target after sorting out.

Adventure games is type that is still kept alive today with Walking Death series and played much despite being a very old kind. In action games, it is not desired players to follow a certain flow, it is required to explore the field and to reach next level by gathering or finding the hidden objects around the field. Adventure games involve the general enemy artificial intelligence, detection systems and classical behavior trees.

Platform games are the game genres that stand the test of time since the earlier times of computer games and has again come up with the developed mobile technologies. Platform games generally do not have an item as path-finding. It is the simplest and least complicated game type with regard to artificial intelligence.

The most complicated artificial intelligence used in platform games is found in map designs of games. Procedurally the creating of stages in platform games is usually done by artificial intelligence. Moreover, the process is even more difficult from the design of stages in strategy games and RPG. Since because changes are so few, a clever artificial intelligent algorithm required in order to design the objects that does not repeat each other and to eliminate the monotony [9].

Sport games are one of areas that the artificial intelligent used most heavily. Artificial intelligence is very complex in sport games; it consists of set of rules and team calculations about applied sport branch. A complex artificial intelligence based on sensors, detection and certain behavior trees is found in sport games. Furthermore, it is required to contain the tactical artificial intelligence for situations such as changes to be done with the progress of the game and the shifting of the tactic system, etc. Additional to use of finite-state machine fuzzy-state machines and data-driven systems are used frequently. The reason is that the data will change during the games and the parameters like performance of player and condition will be monitored and processed by state machines.

Racing games have their own type of artificial intelligence mechanics. The artificial intelligence in racing games can be classified as track display, control and detection the lines of other racers, the control of artificial intelligent character's race car (engine, car, plane, etc.). Moreover, the innovations with artificial intelligence-controlled car modification and tuning can be seen in this area [10].

3.4. HTML5 and WebGL

HTML is one of the oldest text editing languages, whose roots almost goes back to internet and used since 1993. HTML used 2.0, 3.0 and 4.0 versions until 1999 and lastly 4.0.1 version in 1999. These HTML versions were developed by World Wide Web Consortium (W3C) organization's group called the HTML Working Group. Later, stopping their works on HTML, the group diverged to another web standard XML and then XHTML accordingly. Afterwards, a small group of people that wanted a new web standard, gathered under the name of Web Hypertext Application Working Group (WHATWG) in 2004 and published the HTML5 specifications [5].

After these researches, W3C began to work about HTML5 in 2006 and the first operational version was released by them in 2008. HTML5 is a new body of standards which is formed with combination of HTML5, CSS and JavaScript. HTML5 is used with tags and new elements, CSS is used with the visual parts of these tags and elements, JavaScript is used to process the content of all structure and to response the actions of user and to utilize programmatic advantages of HTML5 [6].

HTML5 was published as a standard by W3C on September 28th, 2014. HTML5 brings some new features like Canvas (2D and 3D), Cross Document Messaging, Geolocation Audio and Videos, Forms, SVG, WebSocket API, Local Storage, Offline Web Apps, Drag and Drop, Web Workers, Servers-Sent Events, XMLHttpRequest Level, Local Storage, Offline Web Apps, Drag and Drop, Web Workers, Server-Sent Events, XMLHttpRequest Level 2.

System has become even stronger by developing several API on HTML5. The most important one of these is display of 3D visual through WebGL without requirement of any plug-in installation on browser side. WebGL is essentially OpenGL's portion adapted to HTML5. So, HTML5 can create 3D objects with JavaScript without the necessity of use of any other programming language or can perform animation by importing a character model.

Developers can use all hardware graphic processing power of computer via any browser with WebGL. Before the WebGL developers was able to use hardware graphic processing and other features with games they developed by making users to install a plug-in (such as Adobe Flash Player) or providing them to download and install the application special to own operating system (exe, app, deb, apk, etc.) to their devices [7].

Khronos Group was developed the WebGL similar to other HTML5 APIs by building it on OpenGL ES 2.0. 3D graphic applications became able to use in the regular html elements and low-level DOM interface through this API. Additional to this, 3D web applications can be created just

as upper-level computer games can be developed. The other yield of integration of WebGL on OpenGL ES is to operate on low-power devices and mobile platforms (iPhone, iPad, Android Devices, etc.) with the correct resource management in all leader mobile operation systems by virtue of the fact that OpenGL ES is adapted for embedded systems. However, WebGL is a very low-level library like OpenGL. The use this library requires expertise and is as complex as OpenGL but the use of WebGL has become simpler through some game engines. The best example of this case can be given as PlayCanvas, Three.js and Babylon.js. By these engines, 3D games can be developed just like modern game engines Unity, Unreal Engine, CryEngine and all processes can be performed via an internet browser.

3.5. HTML5 Game Engines

There are two options about game developing with HTML5. The first is Canvas where 2d games can be developed, the second is WebGL where the 3D games are developed. Both 2D and 3D games can be developed under the same platform via game engines. HTML game engines are briefly described in the following paragraphs.

Construct2 game engine developed originally as hobby by a group of student in 2007 and emerged with the name of Construct Classic. Construct 2 was released HTML5-driven in 2011 and many radical changes made on this game engine. It embodies very important features such as multi-platform support, easy learning and visual programming [14]. Construct 2 was designed for development of 2D games and it was mentioned the users can develop game without any knowledge of programming. SDK Template file offered by Construct must be downloaded in order to use DignityAI library inside Construct 2. It can be possible to easily add artificial intelligence features to elements in the game engine through the codes to be transferred into this template.

ImpactJS is one of the popular and paid game engines that was begun to develop by Dominic Szablewski in 2010. One of the biggest advantages of this game engine has Entity structure. It effects the whole by calling update and draw methods in all Entity structures that are connected to draw() and update() methods, which located in the engine.. /libs/game/entities/ directory can be found inside the project, which the Dignity AI will be used. When DignityAI 1.0.0 version is added to this directory at <https://github.com/berkanuslu/dignityai/releases/download/v1.0.0/dignityai.v1.0.0.min.js> address, artificial intelligence features can be added to intended objects by using `<script src="libs/dignityai.v1.0.0.min.js"></script>` in index.html page [15].

PixiJS is a game engine which is built by Good Boy Digital, build on Node.js and distributed free of charge through Github. Contrary to other 2D game engines, PixiJs

renders the game with the WebGL to increase the performance if the browser has WebGL support. However, it carries on the operation via Canvas with the browsers which have no WebGL support. PixiJS is showed as a tool in render parts of most game engines like ImpactJS, Phaser, PandaJS. By creating a /libs/ folder in the project where the DignityAI will be used and when DignityAI 1.0.0 version is added into this folder at <https://github.com/berkanuslu/dignityai/releases/download/v1.0.0/dignityai.v1.0.0.min.js> address, artificial intelligence features can be to intended objects added by using `<script src="libs/dignityai.v1.0.0.min.js"></script>` in index.html page.

PlayCanvas differently from other game engines allows developing the game with a Cloud account on web. Codes can be modified by the help of the editor named PlayCanvas Code Editor. One of the most important properties of PlayCanvas is that it has its own level editor. This editor helps to design the 3D scene and to edit the materials just as big game engines (Unity, Unreal Engine etc.). Another feature that makes PlayCanvas different from other game engines is that the users can use all public projects inside their own project repository by forking. Installation process is completed when DignityAI 1.0.0 version file, which is located in <https://github.com/berkanuslu/dignityai/releases/download/v1.0.0/dignityai.v1.0.0.min.js> address, is downloaded and when this file is linked to an Entity. By this means artificial intelligence features can be added to intended objects.

Three.js might be an engine distributed free of charge and focused on WebGL and took the first step for developing 3D applications with WebGL. Many renderers can be found in Three.js such as WebGL, Canvas, SVG, CSS3D, DOM. It allows many items such as stage, camera, animation, lighting, material and shader. >An editor design is making in Three.js like PlayCanvas and it is planned to realize 3D scenes with the help of this editor. By creating a /libs/ folder in the project where the DignityAI will be used and when DignityAI 1.0.0 version is added into this folder at <https://github.com/berkanuslu/dignityai/releases/download/v1.0.0/dignityai.v1.0.0.min.js> address, artificial intelligence features can be added to intended objects by using `<script src="libs/dignityai.v1.0.0.min.js"></script>` in index.html page.

Phaser is a HTML5 game engine which is distributed free of charge and published as open source. Phaser is concentrated on 2D game developing side. Phaser, which uses PixiS on render side, allows to develop 3D applications by supporting the HTML5 and WebGL. It includes many features such as preloader, physical system, sprite, animation, particle, input, sound, tilemaps, scaling to devices, mobile browser support [16]. There is plug-in logic in Phaser. All features added to application is

developed with the help of these plug-ins. The sample plug-in file that can be found in the <https://github.com/photonstorm/phaser-plugins/blob/master/SamplePlugin/SamplePlugin.js> address is already similar with the DignityAI developing structure. By creating a project inside phaser plug-ins and then transferring DignityAI files into this folder, DignityAI will be suitable for this game engine as well.

Kiwi.js is an open source game engine which was designed to develop mobile and desktop game on HTML5. It has WebGL support for 2D and 3D design features. It benefits from CocoonJS platform to publish the games [17]. It supports the plug-in system similarly to most of games engines like Phaser. By creating a /libs/ folder in the project where the DignityAI will be used and when DignityAI 1.0.0 version is added into this folder at <https://github.com/berkanuslu/dignityai/releases/download/v1.0.0/dignityai.v1.0.0.min.js> address, artificial intelligence features can be added to intended objects by using `<script src="libs/dignityai.v1.0.0.min.js"></script>` in index.html page.

Enchant.js is a considerably simple JavaScript library. It is used to develop game and application. It is developed by researchers Akihabara Research Center, Tokyo, at first in 2011 and shared as open source with the license of MIT. Because of its simple and plain framework, beside game developing it is frequently used for application developing. It supports all design methods like Canvas, DOM and WebGL. By creating a /libs/ folder in the project where the DignityAI will be used and when DignityAI 1.0.0 version is added into this folder at <https://github.com/berkanuslu/dignityai/releases/download/v1.0.0/dignityai.v1.0.0.min.js> address, artificial intelligence features can be added to intended objects by using `<script src="libs/dignityai.v1.0.0.min.js"></script>` in index.html page [18].

Many more HTML5 game engines are being developed and added to this list everyday. Several game engines such as Babylon.js, GameMaker, Turbulenz, Cocos2d-X, Isogenic Engine, Panda.js, Crafty, voxel.js, MelonJS, stage.js has been designed for the ones that want to develop games with HTML5. When looked into these game engines and APIs, it can be seen that lots of items like graphic, audio, video, physics are thought and they can already be used in HTML5 based games. However, it is known there is no game engine or API which hosts any library about artificial intelligence. DignityAI is developed in order to complete the missing parts at this point.

4. DEVELOPED LIBRARY: DignityAI

4.1. General Class Structure of DignityAI

Class diagram of the developed library is given in Figure 1. DignityObject is placed at the top when looked at the

DignityAI general class structure. All the other classes are fundamentally inherited from this class. If we are to give a short information about the classes, what are the functionalities of all the classes and what they are used for will be seen in general terms.

4.2. DignityPath

DignityPath class is the class that handles the basic path-finding. It is specifically designed as to cover the navigation functions both for 2D and 3D games.

All the items inherited from DignityAIBase type has sets of DignityPath by the name of a paths. This sets the paths which the artificial intelligence character will follow in relation with its tasks. More than one path can be defined within the tasks. Defining of the DignityPath class in the DignityAIBase, provides that it can reach previous paths through any task on DignityMission. Thus, any task transforms into a more flexible structure.

Structure is developed in this way by flexible design. For the path given in the DignityPath class, together with the start and finish locations other sub paths which will be used to move to those start and finish locations can be defined. Also, whether the given path will or will not repeat from end to start and move on a random location can also be determined through the same class.

Statements defined by this class are actually used by a move() method for DignityAIBase class and classes inherited from this class. This method provides the object to get into act to follow the path designated in the task definition by reading the paths on the artificial intelligent object. Fundamentally it is mandatory to determine and to fill the finish point; at the end the object is moved towards the finish point. However, some basic algorithms are used for the object to calculate the path it will take and draw a route according to the obstacles on the way. There are many different search algorithms such as A*, IDA*, MA*, Breadth-First-Search, Best-First-Search, Dijkstra, Jump Point Search and many more within these algorithms. From these algorithms, DignityAI uses the most used and most common A* algorithm. However, all the other algorithms or a library such as Pathfinding.js which has a HTML5 library containing all these algorithms can also be easily integrated into DignityAI.

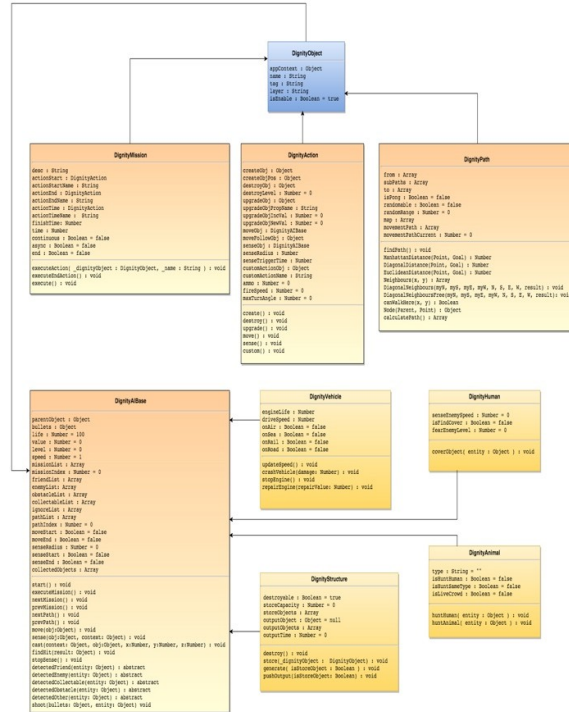


Figure 1: DignityAI Class Diagram

4.3. DignityAction

DignityAction is actually a process part of the DignityAI. All the actions are defined within DignityAction. DignityAction has move(), sense(), destroy(), create(), upgrade() and custom() main functions.

With move() function; the pathList member defined in the DignityAIBase follows the path which is defined on the DignityPath class' object. It finds the shortest path for this process and creates an action. Here the basic condition is; to provide the object, which will be moved with the moveObj property defined in the DignityAction class, to trigger the move method defined in DignityAIBase.

With sense() function; detection according to enemy lists (enemyList) and friends lists (friendList) defined in DignityAIBase is done and main processes are performed according to these detections. This method also provides the object, given to senseObj property and detection to be done, to trigger the sense method defined in DignityAIBase.

With destroy() function; object pointed as target, which is defined with destroyObj property and is from DignityAIBase class or from another class inherited from this, gets damaged. This damage, reduces life feature by the value of destroyLevel for the objects inherited from DignityAIBase class.

With create() function, any object defined with createObj property are created on stage. Also, with the function defined in DignityAction named createObjPos position where the object will be created is given.

upgrade() function is used to increase all the values of the object defined with upgradeObj property. In any way processes, such as level upgrade, health increase are done by this function. For this, name of the object to be upgraded in upgradeObjPropName is given in String. With the upgradeObjIncVal property, information of how much the property, which is defined with upgradeObjPropName, will be increased is given. Thus, given increase value adds to that value. In case a negative value is given, it means that parameter's value is decreased. However, in case a value assignment to be done directly but not as increase/decrease the value of the variable, upgradeObjIncVal property shall be given 0 (zero) and new value of the variable is given to upgradeObjNewVal. An object's value assignment to another can be done by this way.

In the basic custom() function makes required functions of all the objects', which are not defined in DignityAction class, run as actions. It means in case another action apart from move, sense, destroy, create and upgrade actions, custom() function shall be used. This function calls the method, which is given with customActionObjName, of the object' which is given with customActionObj, as action.

4.4. DignityMission

DignityMission class is a class provides the mission managing and can assign mission to each artificial intelligent character. Elementarily is has a structure where missions can be defined as lists, actions for the missions can be defined by DignityAction and these actions can be run as synchronously or asynchronously.

Essentially DignityMission consist of description and action definitions. While description is defined for the cases in which the mission will be shown to users, action is defined by an object assignment in DignityAction class. For this actionStart, actionEnd and actionTime properties are defined in DignityMission class. By means of these properties a start action, end action and a timing action which can be run in any time can be defined. Methods to be used by actions are decided by DignityAction class method names (move, sense, create, destroy, upgrade and custom) which are defined in actionStart, actionEnd and actionTime properties. Also, whether the action will run synchronously or asynchronously is determined by a property named async.

Control of the mission's end in the DignityMission is provided with finishTime variable. If finishTime variable is given 0 (zero) it means the mission will be provided by

temporary actions. If any value is given to this variable, mission has a time to live in terms of seconds. actionEnd action runs at the end of this time and comes the next mission. Another parameter from the mission transition is async variable. By means of this variable, it is decided whether the mission will be synchronous or asynchronous. All the asynchronous missions are run at the same time. But in case the first mission is synchronous, second and third missions are asynchronous, then the first mission runs primarily and then second and third missions are run at the same time after the first mission is finished.

4.5. DignityAIBase

DignityAIBase class is the basic class of the artificial intelligent objects. All the artificial intelligence characters are characters inherited from this class. Classes such as DignityStructure, DignityHuman, DignityVehicle and DignityAnimal are the classes to be used for the artificial intelligent characters customized on DignityAIBase and artificial intelligent characters inherited from this class. Also, as much as customization can be done by adding objects inherited from DignityAIBase.

All the previous classes are classes used to improve the DignityAI basic mechanics and to develop the basic properties to be used in DignityAIBase. Basic use area of these classes is DignityAIBase and classes inherited from it.

Properties named speed, life, value and level are defined in DignityAIBase class. These are the common properties of all the artificial intelligent objects. Together with them artificial intelligent objects must have a mission list (missionList). This mission list starts to run respectively (asynchronous ones are at the same time) to run the mission list. In any case if it comes to a synchronous mission, finish of this synchronous mission is to be waited before running the other asynchronous missions.

Also, there are properties name friendList, obstacleList and collectableList. With these properties detection processes in the sense() method, which is within the DignityAction classes' main methods, are done. sense() function realizes the defined detection processes of the user by looking at these lists and as a result of the detection process detectedFriend(), detectedEnemy(), detectedCollectable() and detectedObstacle() methods are called. Besides, unwanted objects detection can be determined with the ignoreList property. Objects in the ignoreList are not subject to object detection process in any way.

Also, pathList array in the DignityAIBase class, by holding the object in DignityPath type it provides the use of these paths. If there are no path lists added to this artificial intelligent character, it provides it not to be moved to a newly added path on the system. By this way,

access to a single source through all the other classes is provided.

executeMission() method is defined in order to run any existing mission in DignityAIBase. Also nextMission() and prevMission() methods are defined in order to navigate between missions. Likewise for navigation processes in pathList property nextPath() and prevPath() methods exist in order to use in move() method.

There is move() method in DignityAIBase, which uses the DignityPath class for moving the artificial intelligent objects and in order to regulate the objects movements to coordinate points within the existing path. This method respectively provides the movement of the object to all the points in the calculated shortest way of the path data and when the movement process is finished (when reached to final point) to end the action or to move forward to next point if exist.

For detection processes of the artificial intelligence objects, sense() method is defined. In this method, essentially rays are sent from the objects position to the positions in the scanning area by using classic Ray Cast method. cast() method is defined for sending ray process. Objects that hit with the ray (in order to object to detect this hit process object need collision component) falls into findHit() method as the result. And in this method, by looking that the object is not in ignoreList but in which one of the friendList, enemyList, obstacleList, collectableList that list's detection methods triggered. If the found method is in ignoreList then other lists are not checked.

Also, for the other objects, which are inherited from DignityAIBase class, to interpret the detection statuses in different ways, when an object, which does not fall into any of the lists above, is detected detectedOther() method is defined in order to be able to reach these objects and to establish a flexible structure at the detection point. This method is called if the detected object is neither in the ignoreList nor in any of the other lists. Customizations can be done for the objects inherited from DignityAIBase class and override this method.

Last defined method is the shoot() method. According to the detection result, desired objects can be fired at. Objects to be fired (bullet, cannon, bomb, arrow, etc.) are stored in file named bullets.js. Different weapons can be determined according to each object.

4.6. DignityStructure

It is designed to add artificial intelligent features to fixed objects such as building, structure which are inherited from the DignityAIBase. By this class mission and action can be given to a structure. Normally objects such as buildings and structures do not move; but still to moving feature can be added to these structures by using move() action.

There are some customized properties within the DignityStructure. First of these properties is to define the structure to be collapsible, destroyable. Second property is the capacity of the structure. This capacity indicates the object capacity which it can hold, hide inside. If wanted, this capacity can be increased by multiplying with the level feature in DignityAIBase class. Most important property of the structures is them to have a mechanism which gives an output in a given time. By this means they can produce defined objects in given periods and store the outputs if required.

4.7. DignityVehicle

It is developed to create a vehicle model with artificial intelligence. There is a drive() function within this class, which uses the move action of DignityAction class however managing the movement situation specific to the class. By means of this method vehicle's speed is found by calculating the engine condition and driving speed. When the engine condition goes bad, speed reduces automatically. When the vehicle crashes with another vehicle, engineLife property gets reduced by the value of the crash' intensity and speed is calculated again according to this. crashVehicle() method controls the status of crash. There is a repairEngine() method for improving the engineLife value when received an object such as repair kit. Lastly there is a stopEngine() method to stop the vehicle on demand.

4.8. DignityHuman

It is defined to create an artificial intelligence human model. It additionally contains features such as detecting the enemy, to search hiding spots for warrior characters and move towards this spot, getting scared level from enemies.

4.9. DignityAnimal

It is a class designed just as DignityHuman and customized according to animalistic properties and able to create animal objects. DignityAnimal class additionally has the information of animal types, whether it will hunt down the humans or not, whether it will hunt down its own kind or not and whether it lives in packs or not. In the light of these information; detecting humans and hunt them, detecting its own kind and hunt them and also detecting its own kind and go next to them situations are built.

5. A SAMPLE GAME DEVELOPED BY USING DignityAI

When programming a sample game by using DignityAI rather than a visually complete prepared and production oriented structure, a structure which uses the basic mechanics of DignityAI, and completes missions on a

simple map and operates necessary actions as required by using the DignityAIBase and DignityAnimal, DignityStructure, DignityVehicle classes inherited from DignityAIBase was built. Also, while developing the sample game boundaries of the DignityAI library and extra properties, which can be developed, are set to be included in the later studies.

Game to be developed shall define the paths and obstacles on a randomly created 48x48 grid map. Afterwards, a simple tank model is controlled by DignityVehicle class and all the properties such as finding path, detection, mission managing, action managing are handled. Besides that, there is a DignityStructure object named RandomModelCreator which inserts objects that can gather at random points on the map. Lastly there are RedCreator and BlueCreator objects which are DignityStructure objects and produce Red and Blue teams. These objects create tanks on stage periodically and provide these tanks to follow the determined path by assigning coordinates on the random map and meanwhile making them to show different acts against the enemies and collectible objects that they detect.

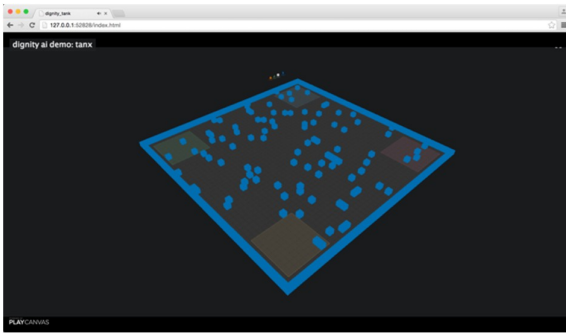


Figure 2. Created random map

As can be seen in Figure 2 the map will be created randomly every time the page is refreshed. After the map is created, object named RandomModelCreator is created and first mission definitions are made in order to create objects that can gather on random points on the map.

```
addRandomModelCreator: function() {
//create random model creator
var _action = new DignityAction();
_action.customActionObj = this;
_action.customActionName = 'createRandomModel';
var _mission = new DignityMission();
_mission.async = true;
_mission.continuous = true;
_mission.actionTime = _action;
_mission.actionTimeName = 'custom';
_mission.time = 10000; //create every 10 seconds
var _creator = new DignityStructure();
_creator.missionList.push(_mission);
_creator.start();
},
createRandomModel: function() {
var model = this.getRandomModel();
model.setPosition(this.getRandomPosition());
},
```

By means of addRandomModelCreator() method, firstly a DignityAction object is defined. createRandomModel() method is defined to this object's customActionObj and customActionName properties, which are in the same script. After the action definition, a DignityMission object is created to manage how this action will work. This object contains a mission, which works asynch and continues constantly. It calls the DignityAction's custom() method in every 10 seconds. And this method also calls the above described relevant method and the model is created in a random point. Figure 3 shows models randomly created on the map.

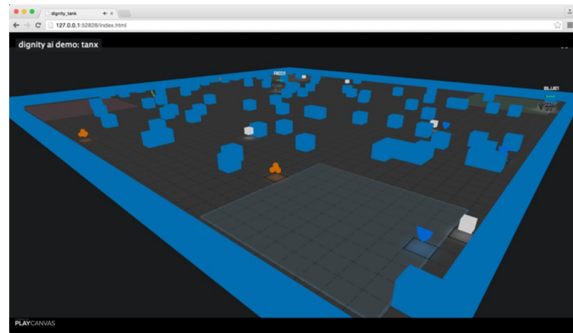


Figure 3. Models randomly created on the map

Together with these one of the 4 models will be randomly added on the map in every ten seconds. After finish the creation of these random creations, it is continued to create tanks one each for red and blue team and to define the artificial intelligence objects and their functions. Process of creating a tank for the red team is as follows.

```
addRedTankCreator: function() {
//create red tank creator
var _action = new DignityAction();
_action.customActionObj = this;
_action.customActionName = 'createRedTankModel';

var _mission = new DignityMission();
_mission.async = true;
_mission.continuous = false;
_mission.actionTime = _action;
_mission.actionTimeName = 'custom';
_mission.time = 10000; //create every 10 seconds

var _creator = new DignityStructure();
_creator.missionList.push(_mission);
_creator.start();
},
createRedTankModel: function() {
var tankName = 'tank_' + new Date().getTime();
var newTank = this.tank.clone();
newTank.setName(tankName);
newTank.enabled = true;
newTank.setPosition(this.getRandomPositionRange(2, 10, 2, 10)); //left bottom
newTank.addComponent('collision', { type: 'box', halfExtents: new pc.Vec3(0.5, 0.5, 0.5) });
this.tanks.addChild(newTank);
var ourPath = new DignityPath();
ourPath.map = this.map.script.createmap.map;
var mapX = Math.round((47.5 - newTank.getPosition().z) / 1.0);
var mapY = Math.round((newTank.getPosition().x - 0.5) / 1.0);
```

```

while(ourPath.movementPath.length == 0) {
  ourPath.from =[mapX,mapY];
  ourPath.randomable = true;
  ourPath.findPath();
}

console.log('red path: ' + ourPath.movementPath);

var vehicle = new DignityVehicle();
vehicle.appContext = app;
vehicle.enemyList.push('tank');
vehicle.enemyList.push('damage');
vehicle.enemyList.push('repair');
vehicle.enemyList.push('shield');
vehicle.enemyList.push('default');
vehicle.ignoreList.push('wall');
vehicle.pathList.push(ourPath);
vehicle.parentObject = newTank;

newTank.script.tank.setHP(10);
newTank.script.tank.ai = vehicle;
newTank.script.tank.hidden(false);
newTank.script.tank.setTankName('red1');

var firstAction = new DignityAction();
firstAction.moveObj = vehicle;

var firstMission = new DignityMission();
firstMission.async = true;
firstMission.actionStart = firstAction;
firstMission.actionStartName = 'move';

var secondAction = new DignityAction();
secondAction.senseObj = vehicle;
secondAction.senseRadius = 10;

var secondMission = new DignityMission();
secondMission.async = true;
secondMission.actionStart = secondAction;
secondMission.actionStartName = 'sense';

vehicle.missionList.push(firstMission);
vehicle.missionList.push(secondMission);
vehicle.start();
},
addBlueTankCreator: function() {
  //create blue tank creator
  var _action = new DignityAction();
  _action.customActionObj = this;
  _action.customActionName = 'createBlueTankModel';

  var _mission = new DignityMission();
  _mission.async = true;
  _mission.continuous = false;
  _mission.actionTime = _action;
  _mission.actionTimeName = 'custom';
  _mission.time = 10000; //create every 10 seconds

  var _creator = new DignityStructure();
  _creator.missionList.push(_mission);
  _creator.start();
},
createBlueTankModel: function() {
  var tankName = 'tank_' + new Date().getTime();
  var newTank = this.tank.clone();
  newTank.setName(tankName);
  newTank.enabled = true;
  newTank.setPosition(this.getRandomPositionRange(37, 45, 2,
10)); //right top
  newTank.addComponent('collision', { type: 'box', halfExtents:
new pc.Vec3(0.5, 0.5, 0.5) });
  this.tanks.addChild(newTank);

  var ourPath = new DignityPath();
  ourPath.map = this.map.script.createmap.map;

  var mapX = Math.round((47.5 -
newTank.script.tank.entity.getPosition().z) / 1.0);
  var mapY = Math.round((newTank.script.tank.entity.getPosition().x - 0.5) / 1.0);

  while(ourPath.movementPath.length == 0) {
    ourPath.from =[mapX,mapY];
    ourPath.randomable = true;
    ourPath.findPath();
  }

  console.log('blue path: ' + ourPath.movementPath);
  var vehicle = new DignityVehicle();
  vehicle.appContext = app;
  vehicle.enemyList.push('tank');
  vehicle.enemyList.push('damage');
  vehicle.enemyList.push('repair');
  vehicle.enemyList.push('shield');
  vehicle.enemyList.push('default');
  vehicle.ignoreList.push('wall');
  vehicle.pathList.push(ourPath);
  vehicle.parentObject = newTank;

  newTank.script.tank.setHP(10);
  newTank.script.tank.ai = vehicle;
  newTank.script.tank.hidden(false);
  newTank.script.tank.setTankName('blue1');

  var firstAction = new DignityAction();
  firstAction.moveObj = vehicle;

  var firstMission = new DignityMission();
  firstMission.async = true;
  firstMission.actionStart = firstAction;
  firstMission.actionStartName = 'move';

  var secondAction = new DignityAction();
  secondAction.senseObj = vehicle;
  secondAction.senseRadius = 10;

  var secondMission = new DignityMission();
  secondMission.async = true;
  secondMission.actionStart = secondAction;
  secondMission.actionStartName = 'sense';

  vehicle.missionList.push(firstMission);
  vehicle.missionList.push(secondMission);
  vehicle.start();
},

```

When looked into the process blog above addRedTankCreator() and addBlueTankCreator(), which firstly definitions of DignityAction, DignityMission are made, can be seen. Here also a customAction definition is made, a mission definition which will work after 10 seconds but won't repeat is made with DignityMission too. 10 seconds waiting after the opening, createRedTankModel() method is called and artificial intelligence tank character is added to scene as shown in Figure 4.

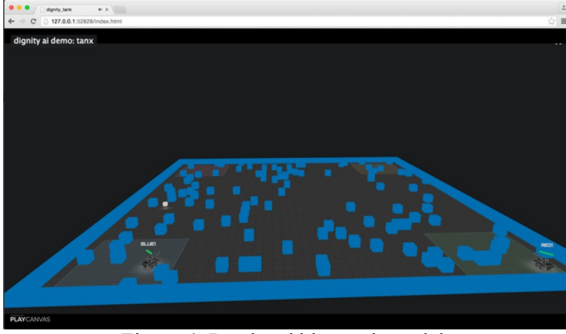


Figure 4. Read and blue tank models

When looked into createRedTankModel method, firstly a tank cloning process is completed. Collision component is described for tank model too. movementPath property gets created by means of findPath() method by linking this object to the created map.

DignityVehicle class is used in order to give artificial intelligence features to tank model. With the help of this class, tank's enemy (enemyList) and friend (friendList) lists are defined. Meanwhile a wall model is added into ignoreList in order to not to detect the wall. Afterwards, previously created DignityPath object is given to pathList as well. Here, parentObject and appContext properties are used to access to game engines's sources.

After the cloning of the tank model, which was created in the game engine and does not have artificial intelligence feature, created DignityVehicle object can use the game engine functions with the ai property in this object.

There are two action and two mission definitions for the tank. First of them is the move action. This action is set to work as asynchronously at the beginning. Second action is the detection function. Here, it is provided for tank to scan a 10-unit area by giving a senseRadius. By adding all these properties to DignityVehicle's missionList, start() method of the DignityVehicle object is called. This allows the activation of the artificial intelligence by running these missions.

After all these processes, two tanks for two different teams detect the enemies and fire at them with the shoot() method as in Figure 5 while moving on randomly given paths.

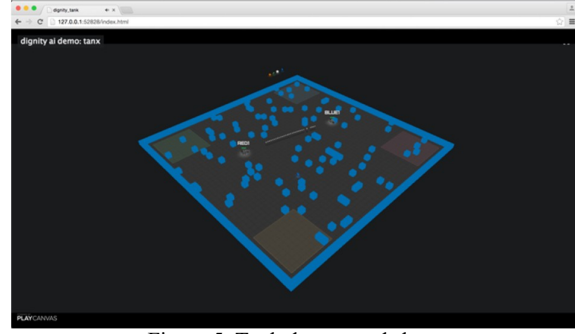


Figure 5. Tank detects and shoot

6. CONCLUSION AND RECOMMENDATIONS

While having an object, which can be used in HTML5 game engines, object oriented structure, the developed artificial intelligence library also contains the necessary items in a HTML5 based artificial intelligence library together with action management and navigation functions.

With a flexible structure, DignityMission class carries out the operation of missions, switching between missions and mission management.

While DignityAction class is able to manage the basic actions by means of move(), sense(), create(), destroy() and upgrade() methods, it can be customized with custom() method and this provides the flexible operation of the system by supporting the actions which can be defined apart from these five actions.

Among the existing methods on small and middle sized maps, with A* algorithm and three basic distance calculation methods, Manhattan and Diagonal and Euclidean, DignityPath class is the one that works simplest and most consistent. However, DignityPath class will be more powerful by adding other algorithms and also different calculation methods.

DignityAIBase class manages the basic artificial intelligence functions as well as being the basic class of the customized classes. With this aspect, it is in the backbone of the system. DignityVehicle, DignityHuman, DignityAnimal and Dignity Structure classes, which are inherited from DignityAIBase, get more powerful and more inclusionary when customized within themselves.

Also, by being open source, developing of DignityAI will move even faster by adapting it to other popular game engines. It will take its place in the literature by being a complete artificial intelligence library, which can carry out all the functions such as situation management, navigation, mission management and action management.

In this regard, first development was made on PlayCanvas game engine. Most important reason to choose PlayCanvas

game engine is because time to start developing game on this one is fairly short and its game engine infrastructure is quite powerful. Also with the help of PlayCanvas' online editor, stage design and 3D game design can be done easily. In addition to all these there are open source games and ready game models in PlayCanvas. These open source games can be forked from the main project as required and keep on developing at its own side. Thus, making it possible to spare more time to DignityAI's own dynamics and powerful structure rather than struggling with the game engine's and game's components.

In the next phase of DignityAI, the target is; to develop the system by adapting it to all artificial engine dynamics and HTML5 game methods and even by moving out of the HTML5 boundaries and contribute to HTML5's development and adapting it to Unity, Unreal Engine and CryEngine game engines, which are more global game engines. For this DignityAI is published as fully open source at social code sharing site Github [19]. Also, the sample application made with DignityAI is published through Github [20].

It is foreseen that the next model of the DignityAI will be a model which can think, add its own missions by itself and choose its own actions and learn.

REFERENCES

- [1] J. Wexler, «Artificial Intelligence in Games: A look at the smarts behind Lionhead Studio's "Black and White" and where it can and will go in the future,» 2002. [Çevrimiçi]. Available: <http://www.cs.rochester.edu/~brown/242/assts/termprojs/games.pdf>. [%1 tarihinde erişilmiştir19 April 2016].
- [2] Z. Middleton, «Case History: The Evolution of Artificial Intelligence in Computer Games,» 2002. [Çevrimiçi]. Available: http://web.stanford.edu/group/htgg/sts145/papers/zmiddleton_2002_1.pdf. [Accessed 19 04 2016].
- [3] M. Stall, «My baby, the finite state machine,» 2006. [Çevrimiçi]. Available: <http://blogs.msdn.com/b/jmstall/archive/2006/09/13/baby-state-machine.aspx>. [Accessed 21 04 2016].
- [4] B. Schwab, AI game engine programming, Boston, Massachusetts: Cengage Learning, 2009.
- [5] P. Lubbers, B. Albers ve F. Salim, Pro HTML5 programming, New York: Apress, 2011.
- [6] A. Freeman, The Definitive Guide to HTML5, New York: Apress, 2011.
- [7] T. Parisi, WebGL: up and running, California: O'Reilly Media, 2012.
- [8] S. Rabin, AI Game Programming Wisdom, Newton Centre: Charles River Media, 2002.
- [9] K. Compton and M. Mateas, "Procedural Level Design for Platform Games," in *In Proceedings Of The Second Artificial Intelligence And Interactive Digital Entertainment Conference*, California, 2006.
- [10] D. Beirne, "Racing Game AI: An Investigation into AI Techniques for Motorsport Simulation Games," 2007. [Online]. Available: <http://www.mygamedemos.com/Abertay/David%20Beirne%20CS%201130A%20Artificial%20Intelligence%20for%20Games%20-%20Racing%20Game%20AI.pdf>. [Accessed 21 04 2016].
- [11] J. Gordon, «A finite state machine javascript micro framework. Github Repository,» 2011. [Çevrimiçi]. Available: <https://github.com/jakesgordon/javascript-state-machine>. [Accessed 21 04 2016].
- [12] J. Cowart, «js ex machina - finite state machines in JavaScript. Github Repository,» 2012. [Çevrimiçi]. Available: <https://github.com/ifandelse/machina.js>. [Accessed 21 04 2016].
- [13] X. Xu, «A comprehensive path-finding library for grid based games. Github Repository,» 2011. [Çevrimiçi]. Available: <https://github.com/qiao/PathFinding.js>. [Accessed 21 04 2016].
- [14] A. Subagio, Learning Construct 2: Design and create your own engaging, extensible and addictive game using Construct 2, Birmingham: Packt Publishing, 2014.
- [15] D. Cielen ve A. Meysman, HTML5 Game Development with ImpactJS: A step-by-step guide to developing your own 2D games, Birmingham: Packt Publishing, 2013.
- [16] B. Bibat, "HTML 5 Shoot'em Up in an Afternoon: Learn (or teach) the basics of Game Programming with this free Phaser tutorial," 2014. [Online]. Available: <https://leanpub.com/html5shootemupinanafternoon/read>. [Accessed 21 04 2016].
- [17] P. Kashyap, «Investigation into the use of HTML 5 game engines to create a responsive social educational game for children,» 2015. [Çevrimiçi]. Available: https://espace.cdu.edu.au/eserv/cdu:46185/This_is_CDU_46185_Kashyap_P.pdf. [Accessed 21 04 2016].
- [18] B. McInnis, R. Shimizu, H. Furukawa, R. Fushimi, R. Tanaka ve K. Kratzer, HTML5

- Game Programming with enchant.js, New York: Apress, 2013.
- [19] B. Uslu, «DignityAI - AI Library for HTML5 Games. Github Repository,» 2015. [Çevrimiçi]. Available: <https://github.com/berkanuslu/dignityai>. [Accessed 21 04 2016].
- [20] B. Uslu, «DignityAI Demo 1: Tanx AI. Github Repository,» 2015. [Çevrimiçi]. Available: https://github.com/berkanuslu/dignity_tank. [Accessed 21 04 2016].

Jenerik Cordic algoritmasının FPGA’da donanımsal gerçekleştirilmesi

Suhap Şahin^{1*}, Burcu Kır Savaş²

07.05.2016Geliş/Received,06.10.2016Kabul/Accepted

doi: 10.16984/saufenbilder.14583

ÖZ

Trigonometrik, logaritmik, hiperbolik vb. matematiksel fonksiyonlarının donanımsal gerçekleştirilmesi sıklıkla kullanılmaktadır. Bu fonksiyonların donanımsal gerçekleştirilmesi yöntemlerinden biri olan CORDIC algoritması donanım kaynağı, güç tüketimi açısından ön plana çıkmaktadır. Çalışmada sinyal işleme uygulamalarında kullanmak amacıyla döndürme ve vektörelmodlarda dairesel açı dönüşümleri kullanan Jenerik CORDIC algoritmasının FPGA’de gerçekleştirilmesi anlatılmıştır. Uygulamada farklı iterasyon değerlerinde ve veri uzunluklarında sentez sonuçlarıyla birlikte gerçekleştirme sonucunda ortalama karesel hata değerleri karşılaştırmalı olarak verilmiştir. Sonuçlarda, sabit çarpan değerinde iterasyon sayısının donanımsal gerçekleştirilmesini etkilemediği ve sabit iterasyon değerinde çarpan değeri arttıkça çıkışta elde edilen sonuçların hata değerlerinin azaldığı gözlemlenmiştir. Gerçeklemede Xilinx firmasına ait Artix-7 FPGA XC7A100T-1CSG324C FPGA entegresi kullanılmıştır.

Anahtar Kelimeler: CORDIC, döndürme modu, vektörel mod, FPGA

Hardware implementation of generic CORDIC algorithm on FPGA

ABSTRACT

Trigonometric, exponential, logarithmic, hyperbolic and several other mathematical functions, are frequently used in hardware implementation applications. CORDIC algorithm, which is a widely used method for hardware implementation of these functions due to its efficient space utilization and low power consumption. In this study, FPGA hardware implementation of rotation angle conversion and circular vector mode CORDIC algorithm for signal processing applications is described. The resulting mean squared error values are provided with respect to different data lengths and different iterations. In this study, the target implementation platform is Xilinx Artix-7 FPGA platform.

Keywords: CORDIC, rotation mod, vectorial mod, FPGA

1. GİRİŞ (INTRODUCTION)

Teknolojinin gelişmesi ve kullanılan cihazların sürekli küçülmesi ile birlikte gömülü sistemler hayatımızın her alanında kullanılmaktadır. Özellikle sinyal ve görüntü işleme, iletişim sistemleri, robotik gibi çeşitli alanlarda ki uygulamaların gerçekleştirilmesinde kullanılan gömülü

sistemler genellikle trigonometrik, logaritmik, hiperbolik gibi temel matematik fonksiyonlarına ihtiyaç duymaktadır [1][2][3][4][5][6]. Bu fonksiyonların donanımsal olarak gerçekleştirilmesi oldukça zor ve maliyetli olması nedeniyle FPGA, ASIC gibi mimariler üzerinde bu fonksiyonların gerçekleştirilebilmesi için farklı matematiksel yaklaşımlara dayalı yöntemleri geliştirilmiştir [7],[8].

* Sorumlu Yazar / Corresponding Author

1 Kocaeli Üniversitesi, Mühendislik Fakültesi Bilgisayar Mühendisliği, Kocaeli - suhapsehin@kocaeli.edu.tr

2 Kocaeli Üniversitesi, Mühendislik Fakültesi Bilgisayar Mühendisliği, Kocaeli- burcu.kir@kocaeli.edu.tr

Literatürde bu mimariler üzerinde ilgili fonksiyonların gerçekleşmesine ilişkin kullanım yoğunluğuna göre 3 donanımsal gerçekleştirme yöntemi ön plana çıkmaktadır [9].

İlk yöntem olan seri açılım ve polinom yaklaşımı donanımsal olarak yoğun iş yükü gerektirmektedir. İkinci yöntem olan bak-oku tablosu yaklaşımında ise uygulamaya ait veri yapısının çözünürlüğünü artırmak için yüksek miktarda hafıza birimi kullanmak gerekmektedir [10].

İlgili fonksiyonların donanımsal olarak gerçekleşmesine yönelik önerilen son yöntem olan CORDIC (COordinateRotationalDIgitalComputer) algoritması [11], kartezyen koordinat sisteminde birim uzunluktaki bir vektörün döndürülerek vektöre ait açı, uzunluk ve yeni kartezyen koordinat bileşenlerinin hesaplanması esasına dayanan bir yaklaşımdır [12].

Hesaplamalarda iki ve ikinin katlarını iteratif olarak kullanan CORDIC algoritması, donanımsal olarak sadece öteleme işlemine ihtiyaç duymaktadır. Bundan dolayı hız ve maliyet olarak avantaj sağlamaktadır [13]. Böylece son zamanlarda literatürde sunulan matematik fonksiyonlarının donanımsal gerçekleşmesine ilişkin çalışmalarda CORDIC algoritması tercih edilmiştir [14][15][16].

Bu makalede, döndürme modda (rotationmode) ve vektörelmodda (vectoringmode) dairesel açı dönüşümü yöntemleri kullanan jenerik CORDIC algoritmasının FPGA’da donanımsal gerçekleşmesi anlatılmıştır. Çalışmada, literatürde sunulan çalışmalardan farklı olarak değişik veri uzunlukları ve iterasyon değerlerinde donanımsal gerçekleştirilen CORDIC algoritmasının çıkış değerlerinde oluşan ortalama karesel hata değerleri ve donanım kaynağı tüketim değerleri gösterilmiştir.

Gerçeklemede Xilinx firmasına ait Artix-7 FPGA XC7A100T-1CSG324C FPGA entegresi kullanılmıştır. Çalışmada sayı formatı olarak tam sayı formatı seçilmiştir.

2. CORDIC ALGORİTMASI (CORDIC ALGORITHM)

Trigonometrik fonksiyonların bilgisayar tarafından hesaplanabilmesi için, 1959 yılında JackVolder tarafından ortaya sürülen CORDIC algoritması, 1971 yılında J.S. Walther tarafından hiperbolik ve üstel fonksiyonlar, logaritma, karekök hesaplamaları yapabilecek şekilde geliştirilmiştir [17][18].

Denklem (1)’de;

$-\frac{\pi}{2} \leq \theta \leq \frac{\pi}{2}$ sınır aralığında tanımlı CORDIC algoritması genel ifadesi verilmiştir.

$$\begin{aligned} x_{i+1} &= x_i - \mu d^i y_i 2^{-i} \\ y_{i+1} &= y_i + \mu d^i x_i 2^{-i} \\ z_{i+1} &= z^i - d^i e_i \end{aligned} \quad (1)$$

Denklem (1)’de tanımlı d^i parametresi, Denklem (2)’de gösterildiği gibi z_i değerinin pozitif veya negatif olmasına göre -1 veya 1 değerini almaktadır. e_i parametresi dairesel, doğrusal ve hiperbolik hesaplamaları yapan her denklem için farklı değer almaktadır. Dairesel dönüşüm için tanımlı eşitlik Denklem (3)’de, doğrusal dönüşüm için tanımlı eşitlik Denklem (4)’de ve hiperbolik dönüşüm için kullanılan eşitlik Denklem (5)’de gösterilmiştir.

$$d_i = \begin{cases} 1, & z_i > 0 \\ -1, & z_i \leq 0 \end{cases} \quad (2)$$

$$e_i = \tan^{-1}(2^{-i}) \quad (3)$$

$$e_i = 2^{-i} \quad (4)$$

$$e_i = \tanh^{-1}(2^{-i}) \quad (5)$$

Denklem (1)’de tanımlı μ parametresi yapılacak hesaplama tekniğine göre değer alan sabit parametredir. Bu değerleri **Hata! Başvuru kaynağı bulunamadı.**’de gösterilmiştir. Her dönüşüm tipi döndürme ve vektörel olmak üzere iki çözüm moduna sahiptir.

Tablo 1. Dönüşüm tipine göre μ parametresinin alacağı değerler (The values which μ parametres will take according to conversion types)

Dönüşüm Tipi	μ
Dairesel	1
Doğrusal	0
Hiperbolik	-1

2.1. Dairesel Açı Dönüşümü (Circular Angle Conversions)

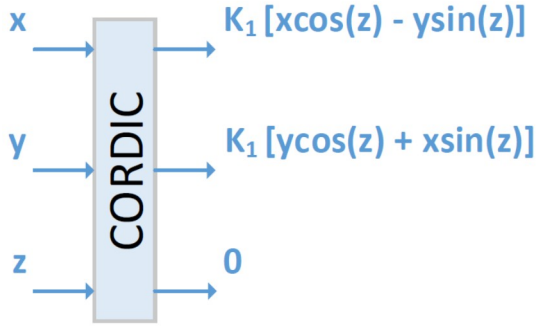
Dairesel açı dönüşüm işlemlerinde, **Hata! Başvuru kaynağı bulunamadı.**’den de görüleceği üzere Denklem (1)’de tanımlanan μ parametre değeri 1 olmaktadır. Dairesel açı dönüşüm işlemlerinde döndürme modu kullanılarak direkt olarak sinüs ve kosinüs değerleri dolaylı olarak ise tanjant ve kotanjant değerleri elde edilebilmektedir. Vektörel modda ise kartezyen

koordinat değerlerinin polar koordinat değerlerine dönüşümü yapılmaktadır.

2.1.1. Döndürme Modu (Rotation Mod)

Döndürme modunda dairesel açı dönüşüm işlemlerinde temel amaç Denklem (1)’de tanımlanan z değişkeni değerini sıfıra yaklaştırmaktadır (Denklem (6), **Hata! Başvuru kaynağı bulunamadı.**).

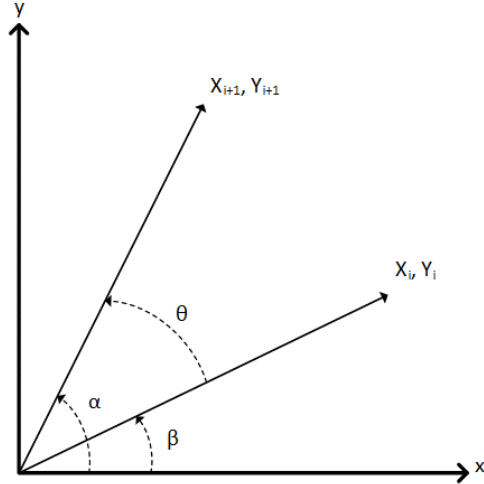
$$\begin{aligned} x_m &= K_1 [x \cos(z) - y \sin(z)] \\ y_m &= K_1 [y \cos(z) + x \sin(z)] \\ z_m &= 0 \end{aligned} \quad (6)$$



Şekil 1. Döndürme modunda Dairesel Açı Dönüşümü ile elde edilebilecek değerler (Values which can be obtained by circular angle rotation in rotation mode)

Döndürme modda dairesel açı dönüşümü, vektörün i . anındaki pozisyonu ile $(i+1)$. anındaki pozisyonu arasındaki açı değeri θ sıfırlanana kadar kaydırma işlemlerinin gerçekleştirilmesiyle yapılmaktadır (Şekil 1). Denklem (7)’de verilen eşitlikle döndürme modda dairesel açı dönüşüm işlemi gerçekleştirilmektedir.

$$\begin{bmatrix} X_{i+1} \\ Y_{i+1} \end{bmatrix} = \cos \theta_i \begin{bmatrix} 1 & -\tan \theta_i \\ \tan \theta_i & 1 \end{bmatrix} \begin{bmatrix} X_i \\ Y_i \end{bmatrix} \quad (7)$$



Şekil 1. Döndürme modunda Dairesel Açı Dönüşümü(Circular angle conversion in rotation mode)

i . adım ile $(i+1)$. adım arasında gerçekleştirilecek döndürme işlemi açı değeri Denklem (8)’deki gibi hesaplanır. Bütün adımlardaki açıların toplamı döndürme açısı θ ’yı vermelidir. Denklem (9)’de tanımlı S_n parametresi $\{-1, 1\}$ değerlerini almaktadır. Bu bilgiler doğrultusunda Denklem (7)’de tanımlanan $\tan \theta_i$ değeri Denklem (10)’daki gibi ifade edilmektedir.

$$\theta_i = \arctan\left(\frac{1}{2^i}\right) \quad (8)$$

$$\sum_{n=0}^{\infty} S_n \theta_n = \theta \quad (9)$$

$$\tan \theta_i = S_i 2^{-i} \quad (10)$$

Denklem (10)’da verilen ifadeyi Denklem (6)’da yerine koyduğumuzda Denklem (11) elde edilir.

$$\begin{bmatrix} X_{i+1} \\ Y_{i+1} \end{bmatrix} = K_i \begin{bmatrix} 1 & S_n 2^{-n} \\ S_n 2^{-n} & 1 \end{bmatrix} \begin{bmatrix} X_i \\ Y_i \end{bmatrix} \quad (11)$$

Denklem (11)’da tanımlanan K_i değişkeni Denklem (12)’de ki gibi hesaplanır.

$$K_i = \frac{1}{\sqrt{1 + 2^{-2i}}} \quad (12)$$

K_i parametresi iteratif süreç içerisinde ihmal edilebilir ve daha sonra bir ölçekleme faktörü olarak uygulanabilir (Denklem 13).

$$K(n) = \prod_{i=0}^{n-1} K_i = \prod_{i=0}^{n-1} \frac{1}{\sqrt{1+2^{-2i}}} \quad (13)$$

2.1.2. Vektörel Mod(Vectorial Mod)

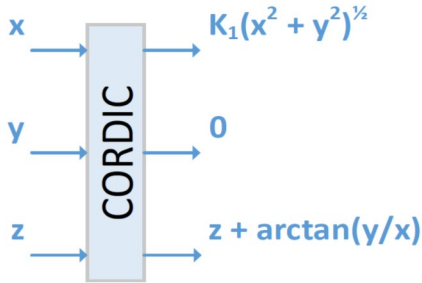
Vektörel dairesel açı dönüşüm işlemlerinde temel amaç Denklem (1)’de tanımlanan y değişkeni değerini sıfıra yaklaştırmaktır (Denklem (14), Şekil 2). Başlangıç değerleri $x_0 = 1$ ve $z_0 = 0$ seçildiği durumda kartezyen koordinat değerlerinin polar koordinat değerlerine dönüşümü yapılmaktadır (Denklem (14)).

Denklem (14)’de tanımlanan K_1 parametresi Denklem (15)’deki gibi hesaplanmaktadır.

$$\begin{aligned} x_m &= K_1 \sqrt{x^2 + y^2} \\ y_m &= 0 \end{aligned} \quad (14)$$

$$z_m = z + \arctan\left(\frac{y}{x}\right)$$

$$K_1 = \prod_{i=0}^{n-1} \sqrt{1+2^{-2i}} \quad (15)$$



Şekil 2. Vektörel Dairesel Açı Dönüşümü ile elde edilebilecek değerler (Values which can be obtained by circular conversion in vectorial mode)

3. CORDIC ALGORİTMASININ DONANIMSAL GERÇEKLENMESİ (HARDWARE IMPLEMENTATION OF CORDIC ALGORITHM)

Bu bölümde, Bölüm 2’de tanıtılan Döndürme ve Vektörel modda Dairesel Açı Dönüşüm işlemlerinin jenerik olarak FPGA tabanlı gerçekleştirilmesi anlatılmaktadır. Şekil 3’de CORDIC algoritmalarının gerçekleştirilmesi aşamasında kullanılan **GENERIC** parametreleri gösterilmektedir. **ITERATION** parametresi algoritmanın istenen değeri hesaplaması için yineleme sayısını ifade etmektedir. Bu değer arttıkça çıkışta elde edilecek sonuçların duyarlılığında da artış gözlemlenecektir. **MUL_COEFF** parametre değeri ise

gerçek sayı formatında hesaplanan tablo değerlerinin tam sayı formatına dönüştürülmesinde kullanılacak 2’nin katı olan katsayı değerini ($2^{\text{MUL_COEFF}}$) ifade etmektedir. **DATA_SIZE** parametresi ise çıkışta elde edilecek verilerin uzunluğunu ifade etmektedir. Bu değer tanımlanması sırasında, **MUL_COEFF** değerine $\text{ceil}(\log_2^{\text{ITERATION}})$ değerinin eklenmesi çıkışta elde edilecek sonuçların doğruluğu açısından önerilmektedir.

```
generic (
    ITERATION : integer;
    MUL_COEFF : integer;
    DATA_SIZE : integer );
```

Şekil 3. Jenerik CORDIC algoritmalarında kullanılan parametreler (Parametres used in Jenerik CORDIC)

3.1 Jenerik Dairesel Açı Dönüşümünün Donanımsal Gerçeklenmesi: Döndürme Modu (Hardware Implementation of Generic Circular Angle Conversions : Rotation Mod)

Dairesel açı dönüşüm işlemlerinin döndürme modda gerçekleştirilmesi aşamasında öncelikle döndürme açı değerlerinin **ITERATION** parametre değerine bağlı olarak bak-oku tablosu oluşturulması gerekmektedir. Dönüşüm işlemleri sırasında her iterasyonda ilgili açı değeri bak-oku tablosundan okunarak sonuç değeri hesaplanmaktadır.

FPGA tabanlı gerçekleştirilmede döndürme açı değerlerinin tutulduğu bak-oku tablosu oluşturma işlemi için gerekli tip ve fonksiyon tanımlama VHDL kodları Şekil 4’de gösterilmiştir. Şekil 4’te **t_Cordic_values** tipi **real** verilerden oluşan **ITERATION** parametre değeri uzunluğunda bir bak-oku tablosu tanımlanmaktadır. **f_Calc_Angels** fonksiyonu, Denklem (7)’yi kullanılarak **ITERATION** parametre değeri uzunluğunda **t_Cordic_values** tipinde bak-oku tablosu üretmektedir. **ITERATION** değeri 12 seçildiği durumda Tablo 2’deki gibi bak-oku tablosu oluşturulmaktadır.

Tablo 2. Denklem (7)’de tanımlı denklem kullanılarak bak-oku tablosu değerlerinin oluşturulması (Formation of look-read table values using equation(7))

i	Denklem (7)	i	Denklem (7)
1	0.78539816339745	7	0.01562372862048
2	0.46364760900081	8	0.00781234106010
3	0.24497866312686	9	0.00390623013197
4	0.12435499454676	10	0.00195312251648
5	0.06241880999596	11	0.00097656218956
6	0.03123983343027	12	0.00048828121119


```

type t_Cordic_values is array(0
to ITERATION - 1) of real;
...
function f_Calc_Angels(ITERATION:
integer) return t_Cordic_values is
variable v_K_angels : t_Cordic_values;
begin
for i in 0 to ITERATION - 1 loop
    v_K_angels(n_i) :=
    arctan(2.0**(-1.0 * real(n_i)));
endloop;
return v_K_angels;
endf_Calc_Angels;

```

Şekil 4. Açık değerlerinin oluşturulmasında kullanılan veri tipi ve fonksiyonun VHDL kodları (Data type used in the formation of angle value and VHDL codes of the function)

Denklem (12)’de tanımlanan $f_Calc_K_values$ parametre değerlerinin hesaplanması için Şekil 6’da gösterilen $f_Calc_K_values$ fonksiyonu kullanılmaktadır. Her bir iterasyon değerleri için ölçekleme faktörü değerleri hesaplanmaktadır.

```

function f_Calc_K_values(ITERATION:
integer) return t_Cordic_values is
variable v_K_values : t_Cordic_values;
begin
    v_K_values(0) := 1.0 / sqrt(2.0);
for i in 1 to ITERATION - 1 loop
    v_K_values(n_i) :=
    v_K_values(n_i - 1) * (1.0 / sqrt(1.0
    + 2.0**(-2.0 * real(n_i))));
endloop;
return v_K_values;
endf_Calc_K_values;

```

Şekil 5. Her bir iterasyon değeri için ölçekleme faktörünün hesaplayan VHDL kodları (VHDL codes calculating the scaling factor for each iteration)

Şekil 4 ve Şekil 5’den da görüleceği üzere f_Calc_Angels ve $f_Calc_K_values$ fonksiyonlarının döndürdüğü değerler *real* veri tipindedir. Bu veri tipi FPGA için sentezlenebilir değildir. Bu nedenle bu değerler normalize edildikten sonra MUL_COEFF parametresi ile çarpılarak *integer* veri tipine $f_Conv_Real_to_Int$ fonksiyonu ile dönüştürülmektedir (Şekil 6).

```

function f_Conv_Real_to_Int(r_Cordic_va
lues : t_Cordic_values; ITERATION,
MUL_COEFF : integer) return t_Int_data
is
    variable v_Int_data : t_Int_data;
begin
for i in 0 to ITERATION - 1 loop
    v_Int_data(n_i) :=
    integer(r_Cordic_values(n_i) *
    real(2**MUL_COEFF));
endloop;
return v_Int_data;
endf_Conv_Real_to_Int;

```

Şekil 6. Gerçek sayı veri tipinden tam sayı veri tipine dönüşüm işlemini gerçekleştiren VHDL kodu (VHDL code processing transformation of real number to whole number)

Örneğin 0.5 değeri MUL_COEFF parametresinin değeri 8 olduğu durumda 4 değerine çevrilmiştir. $f_Conv_Real_to_Int$ fonksiyonu ile elde edilen tam sayı değerleri ile jenerik olarak tanımlanan veri uzunluğunda işlem yapabilmek amacıyla ile tür dönüşümü işleminin gerçekleştirildiği $f_Conv_Int_to_Std_Logic_Vector$ fonksiyonu **Hata! Başvuru kaynağı bulunamadı.**’de gösterilmiştir.

Jenerik CORDIC algoritmasının donanımsal gerçekleştirilmesine ilişkin sözde kod Şekil 8’de verilmiştir. Şekil 9’da tanımlanan 1., 2. ve 3. adımların jenerik parametrelerine bağlı olarak gerçekleştirilmesine ilişkin VHDL kodları Şekil 9, Şekil 10 ve **Hata! Başvuru kaynağı bulunamadı.**’de verilmiştir.

```

function f_Conv_Int_to_Std_Logic_Vector(
r_Int_values : t_Int_data; ITERATION,
DATA_SIZE : integer)
return t_Std_Logic_Vector_data is
    variable v_Std_Logic_Vector_data :
    t_Std_Logic_Vector_data;
begin
for i in 0 to ITERATION - 1 loop
    v_Std_Logic_Vector_data(n_i) :=
    conv_std_logic_vector(
    r_Int_values(n_i), DATA_SIZE);
endloop;
return v_Std_Logic_Vector_data;
endf_Conv_Int_to_Std_Logic_Vector;

```

Şekil 7. Tam sayı veri tipinden std_logic_vector veri tipine dönüşüm işlemini gerçekleştiren VHDL kodu (VHDL code processing transformation of whole number to std_logic_vector number)

Şekil 9’da 1. adımda tanımlanan işlemlerin gerçekleştirilmesine ilişkin VHDL kodları Şekil 10’da verilmiştir. Şekil 10’dan da görüleceği üzere açık değerinin pozitif veya negatif olma durumuna göre atama işlemleri yapılmaktadır.

1. Açık değerine bağlı olarak Denklem (2)’de tanımlanan işlemleri gerçekleştir (Şekil 9).
2. Denklem (10)’da tanımlanan işlemleri gerçekleştir (Şekil 10).
3. Parametreleri güncelle (**Hata! Başvuru kaynağı bulunamadı.**).
4. Tüm iterasyon değerleri için işlemler koşturulduysa 5. adıma, aksi durumda 1. Adıma geç.
5. Denklem (12)’de tanımlanan işlemleri

Şekil 8. CORDIC algoritmasının donanımsal gerçekleştirilmesine ilişkin sözde kod (Pseudo code related to the hardware realization of CORDIC algorithm)

Şekil 8’de 1. adımda tanımlanan işlemlerin gerçekleştirilmesine ilişkin VHDL kodları Şekil 9’da verilmiştir. Şekil 9’da da görüleceği üzere açık değerinin pozitif veya negatif olma durumuna göre atama işlemleri yapılmaktadır.

Şekil 10’de Şekil 8’de 2. adımda tanımlanan işlemlerin gerçekleştirilmesine ilişkin VHDL kodları verilmiştir. Şekil 10’den de görüleceği üzere *CALC_FACTOR* ve *CALC_NEW_V* durumlarında Denklem (10)’da tanımlanan işlemler gerçekleştirilmektedir. *SET_NEW_V* değerinde güncelleme işlemleri yapılarak iterasyondaki yeni değerler, vektördeki yerlerine atanmaktadır.

```

if r_teta < 0 then
r_sigma <= (conv_std_logic_vector(-1 *
2** MUL_COEFF, DATA_SIZE));
else
r_sigma <= (conv_std_logic_vector(1 *
2** MUL_COEFF, DATA_SIZE));
endif;

```

Şekil 9.1. adımda tanımlanan işlemlerin gerçekleştirilmesine ilişkin VHDL kodları (VHDL codes related to realization of processes in the first step)

```

when CALC_FACTOR =>
r_factor <= r_sigma * r_POW_of_2;
r_Cordic_Cntrl <= CALC_NEW_V;

when CALC_NEW_V =>
r_V_new_0 <=
conv_std_logic_vector(2**MUL_COEFF,
DATA_SIZE) * r_V_vector(0) -
r_V_vector(1) * r_factor(MUL_COEFF +
DATA_SIZE - 1 downto MUL_COEFF);
r_V_new_1 <=
conv_std_logic_vector(2**MUL_COEFF,
DATA_SIZE) * r_V_vector(1) +
r_V_vector(0) * r_factor(MUL_COEFF +
DATA_SIZE - 1 downto MUL_COEFF);
r_Cordic_Cntrl <= SET_NEW_V;

when SET_NEW_V =>
r_V_vector(0) <= r_V_new_0(MUL_COEFF
+ DATA_SIZE - 1 downto MUL_COEFF);
r_V_vector(1) <= r_V_new_1(MUL_COEFF +
DATA_SIZE - 1 downto MUL_COEFF);
r_teta_delta <= r_sigma * r_Angel;

```

Şekil 10. 2. adımda tanımlanan işlemlerin gerçekleştirilmesine ilişkin VHDL kodları (VHDL codes related to realization of processes in the second step)

Hata! Başvuru kaynağı bulunamadı.’de Şekil 8’de 3. adımda tanımlanan işlemlerin gerçekleştirilmesine ilişkin VHDL kodları verilmiştir. Bu adımda CORDIC algoritmasında kullanılan parametre güncelleme işlemleri yapılmaktadır.

3.2. Jenerik Dairesel Açık Dönüşümünün Donanımsal Gerçeklenmesi Vektörel Mod(Hardware Implementation of Generic Circular Angle Conversions : Vectorial Mod)

Dairesel açık dönüşüm işlemlerinin vektörel modda gerçekleştirilmesi aşamasında *ITERATION* parametre değerine bağlı olarak bak-okuy tablosu oluşturulması gerekmektedir. Dönüşüm işlemleri sırasında her iterasyonda ilgili değer bak-okuy tablosundan okunarak sonuç değeri hesaplanmaktadır.

```

r_teta <= r_teta -
r_teta_delta(MUL_COEFF + DATA_SIZE - 1
downto MUL_COEFF);
r_POW_of_2 <= '0' &
r_POW_of_2(DATA_SIZE - 1 downto 1);

if n_i + 2 > VALUE_SIZE then
r_Angel <= '0' & r_Angel(DATA_SIZE - 1
downto 1);
else
r_Angel <= r_Angels_SLV_data(n_i + 1);
end if;

```

Şekil 11. 3. adımda tanımlanan işlemlerin gerçekleştirilmesine ilişkin VHDL kodları (VHDL codes related to realization of processes in the third step)

FPGA tabanlı gerçekleştirilmede bak-oku tablosu oluşturma işlemi için gerekli fonksiyon tanımlamaları VHDL kodları Şekil 12’de gösterilmiştir. Şekil 12’de tanımlanan *f_Calc_tp_values* fonksiyonunun döndürdüğü değerlerin giriş olarak verildiği *f_Calc_atantp_values* fonksiyonun kullanarak *ITERATION* parametre değeri uzunluğunda *t_Cordic_values* tipinde bak-oku tablosu üretmektedir.

Denklem (14)’de tanımlanan K_1 parametre değerlerinin hesaplanması için Şekil 13’de gösterilen *f_Calc_K_values* fonksiyonu kullanılmaktadır. Her bir iterasyon değerleri için ölçekleme faktörü değerleri hesaplanmaktadır.

Şekil 14’de vektörel modda CORDIC algoritmasının donanımsal gerçekleştirilmesine ait şematik gösterimi verilmiştir.

3.3. Test Sonuçları (Test Results)

Tablo 3’te sabit çarpan ($MUL_COEFF = 18$) ve veri uzunluğu ($DATA_WIDTH = 24$) değerlerinde farklı iterasyon değerleri için algoritma çıkışında elde edile *kosinüs* ve *sinüs* değerlerinin ortalama karesel hata (OKH) oranları ve FPGA üzerinde kullandığı mandal sayıları gösterilmiştir. Tablo 3’te iterasyon sayısının değişmesi algoritmanın gerçekleştirilmesi esnasında kullanılan alanı etkilemediği ve 20. iterasyondan sonra OKH değerleri değişmediği görülmektedir.

Tablo 4’de sabit iterasyon ($ITERATION = 24$) değerinde farklı çarpan değerleri için algoritma çıkışında elde edile *kosinüs* ve *sinüs* değerlerinin ortalama karesel hata (OKH) oranları ve FPGA üzerinde kullandığı mandal sayıları gösterilmiştir. Tablo 4’de çarpan değerinin artması ile alan kullanımı artmakta fakat OKH değerlerinde azalma görülmektedir.

```

function f_Calc_tp_values(ITERATION:
integer) return t_Cordic_values is
variable v_tp_values : t_Cordic_values;
begin
v_tp_values(0) :=
1.0; v_tp_values(1) :=
1.0; v_tp_values(2) := 1.0;
for n_i in 3 to ITERATION - 1 loop
v_tp_values(n_i) :=
v_tp_values(n_i - 1) / 2.0;
end loop;
return v_tp_values;
end f_Calc_tp_values;
.....
function f_Calc_atantp_values(c_tp_value
s : t_Cordic_values; ITERATION:
integer) return t_Cordic_values is
variable v_atantp_values :
t_Cordic_values;
begin
v_atantp_values(0) := c_PI /
4.0; v_atantp_values(1) := c_PI / 4.0;
v_atantp_values(2) := c_PI / 4.0;
for n_i in 3 to ITERATION - 1 loop
v_atantp_values(n_i) :=
arctan(c_tp_values(n_i));
end loop;
return v_atantp_values;
end f_Calc_atantp_values;

```

Şekil 12. Bak-oku tablosu oluşturmak için kullanılan fonksiyonun VHDL kodları (VHDL codes of the function which is used to form look-uptable)

Tablo 5’de sabit çarpan ($MUL_COEFF = 18$) ve veri uzunluğu ($DATA_WIDTH = 24$) değerlerinde farklı iterasyon değerleri için algoritma çıkışında elde edile *genlik* ve *açı* değerlerinin ortalama karesel hata (OKH) oranları ve FPGA üzerinde kullandığı mandal sayıları gösterilmiştir.

Tablo 5’den iterasyon sayısının değişmesi algoritmanın gerçekleştirilmesi esnasında kullanılan alanı etkilemediği 20. iterasyondan sonra OKH değerleri değişmediği görülmektedir.

Tablo 6. Farklı çarpan değerlerinde kosinüs ve sinüs değerleri için ortalama karesel hata değerleri ve kullanılan mandal sayıları ($ITERASYON = 24$) (Root-mean square values for mean cosine and sine values in different multiplier values and flip-flops that are used- $ITERASYON = 24$)

Mul_c oeff	Okh_genlik	Okh_acı	Mandal	Dsp
10	2.698429×10-8	1.81201 0×10-7	108	3
12	9,482901×10-9	3,60584 9×10-9	120	3
16	1,865137×10-10	9,35340 6×10-11	144	3

20	2,532061×10-13	4,06095 4×10-13	168	6
24	1,108513×10-14	1,84159 2×10-14	227	10

4. SONUÇLAR(CONCLUSION)

Bu çalışmada döndürme modda dairesel açı dönüşümü kullanılarak sinüs ve kosinüs değerlerinin hesaplanması ve vektörel modda dairesel açı dönüşümü kullanılarak da bir vektörün polar koordinat değerlerinin hesaplanması işlemleri FPGA tabanlı donanımsal olarak gerçekleştirilmiştir. Gerçeklemede iterasyon sayısının donanımsal gerçekleştirilmede kullanılan alan tüketimini etkilemediği ve 20. iterasyon değerinden sonra OKH değerlerinin değişmediği Tablo 3 ve Tablo 5’de gösterilmiştir. Bu durumun nedeni olarak seçilen çarpan değerinin bak-oku tablosu değerlerinin tümünü uygun seviyelere ölçekleyememesinden kaynaklanmaktadır. Örneğin döndürme modda *f_Calc_Angels* fonksiyonunun döndürdüğü 20 iterasyon değeri 218 ile çarpma işlemi gerçekleştirdiğimizde elde edilen değer 0 olacaktır. Bu nedenle 20. iterasyon değerinden sonra gerçekleştirilecek olan iterasyonlarda yine fonksiyon 0 değerini döndürecek. İterasyon değerinin sabit tutulup çarpan değerinin artırılması ile OKH değerlerinin değişiklikler gösterdiği Tablo 4 ve **Hata! Yer işareti başvurusu geçersiz.**’da gösterilmiştir. Tablolardan da görüleceği üzere çarpan değeri arttıkça sabit iterasyon değerinde çıkışta elde edilen sonuç değerlerinde hata değerleri azalmaktadır.

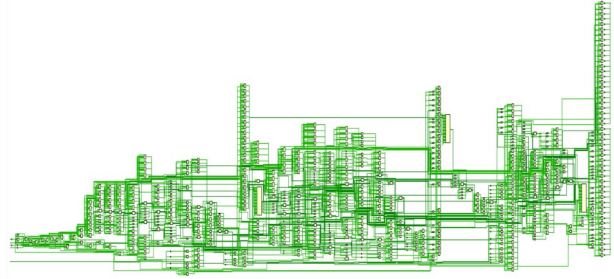
’de sabit iterasyon (ITERATION = 24) değerinde farklı çarpan değerleri için algoritma çıkışında elde edilen *genlik* ve *açı* değerlerinin ortalama karesel hata (OKH) oranları ve FPGA üzerinde kullandığı mandal sayıları gösterilmiştir. Tablo 4’de çarpan değerinin artması ile alan kullanımı artmakta fakat OKH değerlerinde azalma görülmektedir.

```

function f_Calc_K_values(c_atantp_values
: t_Cordic_values; ITERATION: integer)
return realis
variable v_K : real;
begin
    v_K := sqrt(2.0) / 4.0;
    for n_i in 3 to ITERATION - 1 loop
        v_K := v_K *
        cos(c_atantp_values(n_i));
    endloop;
    return v_K;
end f_Calc_K_values;

```

Şekil 13. İterasyon değeri için ölçekleme faktörünü hesaplayan VHDL kodları (VHDL codes calculating the scaling factor for iteration value)



Şekil 14. VektörelModda CORDIC algoritması şematik gösterimi (Schematic display of CORDIC algorithm in Vector mode)

Tablo 3. Farklı iterasyon değerlerinde cosinüs ve sinüs değerleri için ortalama karesel hata değerleri ve kullanılan mandal sayıları (MUL_COEFF = 18, DATA_WIDTH = 24) (Root-mean square values for mean cosine and sine values in different iteration values and flip-flops that are used- MUL_COEFF = 18, DATA_WIDTH = 24)

Iteration	Okh_cos	Okh_sın	Mandal	Dsp
12	2,335501 ×10 ⁻⁸	1,93478 0×10 ⁻⁸	247	10
16	2,346898 ×10 ⁻¹⁰	2,34689 8×10 ⁻¹⁰	247	10
20	2,474788 ×10 ⁻¹⁰	1,26875 2×10 ⁻¹⁰	247	10
24	2,474788 ×10 ⁻¹⁰	1,26875 2×10 ⁻¹⁰	247	10
28	2,474788 ×10 ⁻¹⁰	1,26875 2×10 ⁻¹⁰	247	10

Tablo 4. Farklı çarpan değerlerinde ortalama kosinüs ve sinüs değerleri için ortalama karesel hata değerleri ve kullanılan mandal sayıları (ITERASYON = 24) (Root-mean square values for mean cosine and sine values in different multiplier value sand flip-flops that are used- ITERASYON = 24)

Mul_coef f	Okh_cos	Okh_sın	Mandal	Dsp
10	1,951241 ×10 ⁻⁶	1,951241× 10 ⁻⁶	175	6
12	5,513225 ×10 ⁻⁷	5,115861× 10 ⁻⁷	193	6
16	2,163471 ×10 ⁻⁹	1,697809× 10 ⁻⁹	271	6
20	1,208777 ×10 ⁻¹¹	1,208777× 10 ⁻¹¹	325	16
24	7,503266 ×10 ⁻¹⁴	7,503266× 10 ⁻¹⁴	337	20

Tablo 5. Farklı iterasyon değerlerinde cosinüs ve sinüs değerleri için ortalama karesel hata değerleri ve kullanılan mandal sayıları (MUL_COEFF = 18, DATA_WIDTH = 24) (Root-mean square values for mean cosine and sine values in different iteration value sand flip-flops that are used- MUL_COEFF = 18, DATA_WIDTH = 24)

Iteratio n	Okh_genlik	Okh_acı	Mandal	Dsp
12	8.435011×10 ⁻¹²	1.87307 0×10 ⁻⁷	202	3
16	5.593260×10 ⁻¹²	1.73585 4×10 ⁻⁹	202	3
20	5.593160×10 ⁻¹²	8.47966 6×10 ⁻¹¹	202	3
24	5.593160×10 ⁻¹²	1.20239 0×10 ⁻¹¹	202	3

28	5.593160×10^{-12}	1.20239 0×10^{-11}	202	3
----	----------------------------	--------------------------------	-----	---

Tablo 6. Farklı çarpan değerlerinde kosinüs ve sinüs değerleri için ortalama karasel hata değerleri ve kullanılan mandal sayıları (ITERASYON = 24) (Root-mean square values for mean cosine and sinus values in different multiplier values and flip-flops that are used-ITERASYON = 24)

Mul_c oeff	Okh_genlik	Okh_acı	Mandal	Dsp
10	2.698429×10^{-8}	1.81201 0×10^{-7}	108	3
12	9.482901×10^{-9}	3.60584 9×10^{-9}	120	3
16	1.865137×10^{-10}	9.35340 6×10^{-11}	144	3
20	2.532061×10^{-13}	4.06095 4×10^{-13}	168	6
24	1.108513×10^{-14}	1.84159 2×10^{-14}	227	10

5. SONUÇLAR(CONCLUSION)

Bu çalışmada döndürme modda dairesel açı dönüşümü kullanılarak sinüs ve kosinüs değerlerinin hesaplanması ve vektörel modda dairesel açı dönüşümü kullanılarak da bir vektörün polar koordinat değerlerinin hesaplanma işlemleri FPGA tabanlı donanımsal olarak gerçekleştirilmiştir. Gerçeklemede iterasyon sayısının donanımsal gerçeklemede kullanılan alan tüketimini etkilemediği ve 20. iterasyon değerinden sonra OKH değerlerinin değişmediği Tablo 3 ve Tablo 5’de gösterilmiştir. Bu durumun nedeni olarak seçilen çarpan değerinin bak-oku tablosu değerlerinin tümünü uygun seviyelere ölçekleyememesinden kaynaklanmaktadır. Örneğin döndürme modda *f_Calc_Angels* fonksiyonunun döndürdüğü 20 iterasyon değeri 2^{18} ile çarpma işlemi gerçekleştirdiğimizde elde edilen değer 0 olacaktır. Bu nedenle 20. iterasyon değerinden sonra gerçekleştirilecek olan iterasyonlarda yine fonksiyon 0 değerini döndürecektir. İterasyon değerinin sabit tutulup çarpan değerinin artırılması ile OKH değerlerinin değişiklikler gösterdiği Tablo 4 ve **Hata! Yer işareti başvurusu geçersiz.**’da gösterilmiştir. Tablolardan da görüleceği üzere çarpan değeri arttıkça sabit iterasyon değerinde çıkışta elde edilen sonuç değerlerinde hata değerleri azalmaktadır.

KAYNAKÇA(REFERENCES)

- [1] S. Karthick, P. Priya ve V. S, “CORDIC Based FFT for Signal Processing System”, International Journal of Advanced Research in Electrical, Electronics and Instrumentation Engineering, cilt 1, no. 6, 2012.
- [2] Y. H. Hu ve Z. Wu, “An efficient CORDIC array structure for the implementation of discrete cosine transform”, IEEE Transactions

on Signal Processing , cilt 43, no. 1, pp. 331 - 336, 2002.

- [3] Y. H. Hu, “On the Convergence of the CORDIC Adaptive Lattice Filtering (CALF) Algorithm”, IEEE TRANSACTIONS ON SIGNAL PROCESSING, cilt 46, no. 7, pp. 1861-1871, 1998.
- [4] S. Sharma, P. N. Ravichandran, S. Kulkarni, V. M. ve P. Lakshminarsimahan, “Implementation of Para-CORDIC Algorithm and Its Applications in Satellite Communication”, International Conference on Advances in Recent Technologies in Communication and Computing, 2009 .
- [5] P. Revathi, M. N. Rao ve G. Locharla, “Architecture Design and FPGA Implementation of CORDIC Algorithm for Fingerprint Recognition Applications”, Procedia Technology, cilt 6, p. 371–378, 2012.
- [6] P.Karthikeyan, K.Kavaskar, P.Kirbakaran, A.Manikandan ve R. Sekar, “VLSI Implementation of Cordic Based Robot Navigation Processor”, International Research Journal of Engineering and Technology, cilt 3, no. 2, 2016.
- [7] J. Sujitha ve V. R. Reddy, “Implementation of Log and Exponential Function in FPGA”, International Journal of Engineering Research & Technology, cilt 3, no. 11, 2014.
- [8] L. Deng, C. Chakrabarti, N. Pitsianis ve X. Sun, “Automated optimization of look-up table implementation for function evaluation on FPGAs”, Mathematics for Signal and Information Processing, 2009.
- [9] B. Lakshmi ve A. S. Dhar, “CORDIC Architectures: A Survey”, VLSI Design, cilt 2010, 2010.
- [10] M. A. Çavuşlu, C. Karakuzu ve F. Karakaya, “Neural identification of dynamic systems on FPGA with improved PSO learning”, Applied Soft Computing, cilt 12, no. 9, p. 2707–2718, 2012.
- [11] J. Volder, “The CORDIC Trigonometric Computing Technique”, IRE Trans. Electronic Computers, cilt 8, pp. 330-334, 1959.
- [12] B. Kir, M. Altuncu ve S. Şahin, “FPGA based implementation of CORDIC using different number format”, Technological Advances in Electrical, Electronics and Computer Engineering, 2013.
- [13] J. Sanchez, A. Jimeno, H. Mora, J. Mora ve F. Pujol, “A Cordic-based Architecture for High Performance Decimal Calculations”, IEEE International Symposium on Industrial Electronics, 2007.

- [14] R. Andraka, “A survey of CORDIC algorithms for FPGA based computers”, IEEE Transactions on Computers, cilt 45, no. 3, p. 328–339, 1998.
- [15] K. Kumar, K. Rao ve R. Durga, “FPGA Implementation of DSWG Using CORDIC Algorithm”, International Journal of Innovative Research in Computer and Communication Engineering, cilt 1, no. 7, 2013.
- [16] R. Mehra ve B. Kamboj, “FPGA Implementation of Pipelined CORDIC Sine Cosine Digital Wave Generator”, Int. J. Comp.Tech. Appl, cilt 1, no. 1.
- [17] J. S. Walther, “A unified algorithm for elementary functions”, Proceedings of the AFIPS Spring Joint Computer Conference, 1971.
- [18] J. S. Walther, “The story of Unified CORDIC”, Journal of VLSI Signal Processing, cilt 25, no. 2, p. 107–112, 2000.

Geniş bant konum belirleme sistemi performans analizi ve iyileştirilmesi

Kerem Küçük^{1*}

13.07.2016 Geliş/Received, 28.10.2016 Kabul/Accepted

doi: 10.16984/saufenbilder.64863

ÖZ

Geniş bant spektrumu ve yüksek çözünürlük özellikleri ile çok geniş bant (Ultra-Wide Band, UWB) teknolojisi, birçok kapalı alan konum belirleme probleminde tercih edilmektedir. Bu çalışmada; UWB kablosuz sinyallerini kullanan konum belirleme sisteminin konum belirleme performansının belirlenmesi ve genellikle açık alan konum belirlemede kullanılan yatay hassaslık ölçeğinin (Horizontal Dilution Precision, HDOP) kapalı alanda bu sistem için kullanılmasıyla, konum tahmin hatasının iyileştirilmesi için yeni bir yöntem sunulmaktadır. Bu yöntem kullanışsız olan konum tahmin noktalarının elenmesi temel almaktadır ve UWB konum belirleme sistemi ile elde edilen deneysel konum belirleme sonuçlarının iyileştirilmesini sağlamaktadır. Bu yöntemin performans analizi için UWB sisteminin sağladığı konum verilerine literatürde kullanılan en küçük kareler (Least Squares, LS) ve doğrusal olmayan regresyon (Non-Linear Regression, NLR) ve önerilen HDOP tekniklerinin uygulanması ile elde edilen sonuçlar karşılaştırılmalı olarak sunulmaktadır. Sonuçlar göstermektedir ki; önerilen HDOP tekniği, LS algoritmasının ortalama konum hatasına göre yaklaşık olarak %9 oranında, NLR algoritmasına göre ise %5 oranında daha iyi sonuç vermektedir.

Anahtar Kelimeler: Çok geniş bant, Bina içi konum belirleme, Yatay hassaslık ölçeği

Performance analysis and improvement of the ultra-wide band localization system

ABSTRACT

With the very wide spectrum and high resolution characteristics, ultra-wideband (UWB) communication technique is chosen various indoor localization. This paper presents the localization performance of a positioning system, which uses UWB wireless signals, and a novel method to decrease the localization error using horizontal dilution of precision (HDOP), which uses for outdoor positioning in the literature, for this system. This method focuses to eliminate the unadaptable localization points and provides elimination of UWB experimental localization results. The comparison between the proposed method localizations and the results by elimination of UWB system raw localization data with the least-squares (LS) and the non-linear regression (NLR) techniques are provided for the performance analysis. As the results, the proposed HDOP technique approximately provides 9% the better performance than the LS algorithm. In addition, the technique also provides 5% the better performance than the NLR algorithm.

Keywords: Ultra-wide band, Indoor localization, Horizontal dilution of precision

* Sorumlu Yazar / Corresponding Author

1 Kocaeli Üniversitesi, Mühendislik Fakültesi, Bilgisayar Mühendisliği, Kocaeli - kkucuk@kocaeli.edu.tr

1. GİRİŞ (INTRODUCTION)

Kablosuz haberleşme sistemleri birçok başarılı uygulama ile günümüz iletişim problemlerine kolaylıkla uygulanabilen çözümler sunmaktadır [1]. Bu sistemlerin sunduğu uygulanabilir çözüm yöntemleri sayesinde araştırmacılar son yıllarda çalışmalarını konum tahmini ve gerçek zamanlı kullanıcı takibi problemlerine yoğunlaştırmaktadırlar. Literatürde kullanıcının konumunu tahmin eden ve gerektiğinde takip edebilen farklı ortamlar için çalışmalar mevcuttur [2]. Açık alanlar için sunulan çözümlerden en önemlisi olarak küresel konumlandırma sistemi (Global Positioning System, GPS) [3] kabul edilmesine rağmen, kapalı alanlar için ortak bir kabul gören teknik henüz gerçekleştirilememiştir. Bu alandaki çalışmalar sinyal çözünürlüğü geleneksel radyo teknolojisine göre çok daha iyi olan çok geniş bant (Ultra-Wide Band, UWB) teknolojisi üzerinde yoğunlaşmaktadır [4]. UWB teknolojisini kullanan radyo alıcı vericileri, yüksek bant genişliği ile çoklu yol parazitlerin etkisini azaltarak veriyi daha küçük hata ve yüksek hızlarda iletebilmektedirler. Böylelikle kapalı alanlarda cihazlar arasında güvenli bir kablosuz haberleşme ortamı oluşmaktadır [5].

UWB sistemleri aynı frekans spektrumunda çalışan sistemlerin neden olacağı parazitlerden etkilenmemesi için belirli güç seviyelerinde iletim yapmaktadırlar. Böylelikle kablosuz UWB sinyalleri, özellikle konum belirleme için söz konusu olabilen engellerin içine işleme, kesin konum tahmini, yüksek hızda veri iletimi, düşük maliyetli ve düşük güçlü alıcı-verici tasarımı gibi avantajlar sağlamaktadırlar. Literatürde, UWB haberleşme tekniğini farklı biçimlerde kullanan çok fazla sayıda konum belirleme tekniği mevcuttur [6]. Kapalı alanlar için önerilen teknikler ise radyo sinyalinin parmak izi ve uzaklık tabanlı çözümler olmak üzere ikiye ayrılmaktadırlar. Uzaklık tabanlı konum tahmini yapabilmek için temel yöntem sinyal parametreleri ölçümlerine dayanan nicelikler kümesinin gerçek zamanlı olarak doğrudan çözülmesini öngören geometrik tekniklerdir [6]. Bu nicelikler farklı geometrik şekillerin kesişim kümelerinin bulunmasını sağlayan geometrik modellerden türetilmiştir. Geliş zamanı (Time Of Arrival, TOA) veya alınan sinyal gücü (Received Signal Strength, RSS) ölçümü hedefin pozisyonu için bir daire tanımlayan alıcı ile hedef arasındaki uzaklığın belirlenmesini sağlar. Bu şekilde alıcı ile hedef arasında gerçekleştirilen geliş açısı (Angle of Arrival, AOA) veya geliş zamanı farkı (Time Difference of Arrival, TDOA) ölçümü ile yine hedefin konumu belirlenebilmektedir [7], [8], [9]. Kapalı alanlarda uzaklık tabanlı konum belirleme performansını alıcıların konumu [10], yapılan ölçümün niteliği ve sayısı [11] gibi etkenler değiştirmektedir. Chai ve diğerleri kapalı alanda UWB alıcılarının konumlandırılması ve konumları bilinmeyen

farklı sayıda düğümlerin bağlantılarını kullanan işbirlikçi konum belirleme algoritması önermişlerdir [12]. Bununla birlikte kapalı alanlardaki konum belirleme ihtiyacının uygulamaya yönelik artışı UWB çalışmalarını konum hassasiyetinin artırılmasına yöneltmektedir [13]. Ticari olarak alınabilen bir UWB sistemi ile gerçek zamanlı 3 boyutlu bir konum belirleme sisteminin sunulduğu çalışmada konum verilerinin iyileştirilmesi için triletasyon ve üç izleme filtresi kullanılmıştır [14]. Bununla birlikte GPS için kullanılan GDOP metriğinin kapalı alanlarda da kullanılarak konum belirleme yapılmasına yönelik çalışmalar da mevcuttur [15], [16]. Kapalı alanlarda bu konum belirleme sistemleri için optimizasyon ve istatistiksel yaklaşım teknikleri de çalışılmaktadır [5]. Bu teknikler gürültülü ölçümlerin kullanıldığı parametreler ile elde edilen fazla sayıdaki pozisyon bilgisi ile konum tahmini için teorik çözümler ortaya koyabilmektedirler. Ticari olarak üretilmiş UWB konum belirleme sistemleri günümüzde birçok sağlık, güvenlik, askeri ve acil durum uygulamalarında kullanılabilir [17], [18]. Uygulamalara göre değişen konumlandırma hedeflerinde genelde bir metrenin altında hata beklenmektedir. Bu hedefi doğrultusunda UWB sisteminin kullanacak olduğu konum belirleme tekniği kesinlik, karmaşıklık, kararlılık, ölçeklenebilirlik ve maliyet değerlendirilmesine göre seçilmektedir. Çalışmalara göre UWB konum belirleme ürünleri son gelişmeler ile birlikte yüksek kesinlikte konum belirleme sağlayabilmektedirler [17], [19].

Bu çalışmada, ticari olarak üretilmiş olan bir UWB konum belirleme sisteminin laboratuvar ortamına kurulması ve bu ortam için konum belirleme performansının değerlendirilmesi yapılmaktadır. Bununla beraber, konum belirleme sisteminden elde edilen deneysel konum tahminlerinin, GPS sisteminde açık alanlar için kullanılan yatay hassaslık ölçeği (Horizontal Dilution of Precision, HDOP) [20] ile iyileştirilmesini sağlayan yeni bir yöntem önerilmektedir. Bu amaçla HDOP için eşik değer tanımlaması yapılarak, bir filtre gibi kullanılması sunulmaktadır. Bu yöntemin mevcut teorik çalışmalara göre en önemli avantajı her hangi ek ölçüm parametresine ihtiyaç duymadan, mevcut ölçümlerin kaliteli olanlarının konum tahmini sürecine dâhil edilmesidir. Ayrıca yöntemin birden fazla alıcı içeren tüm UWB sistemleri için uygulanabilir olması diğer bir avantajıdır. Bu tekniğin kullanılabilmesi için ölçümlerin yapılacak olduğu alanın farklı alıcı sayılarına göre HDOP haritalarının çıkarılması gerekmektedir. Literatürde, önerilen HDOP tekniği ile benzer amaçla kullanılan en küçük kareler (Least square, LS) [21] ve doğrusal olmayan regresyon (Non-Linear Regression, NLR) [22] mevcuttur. Bu algoritmalar yüksek hesaplama karmaşıklığına sahiptirler. UWB tekniği kullanan konum belirleme sisteminin ayrıntılı açıklaması ve konumları

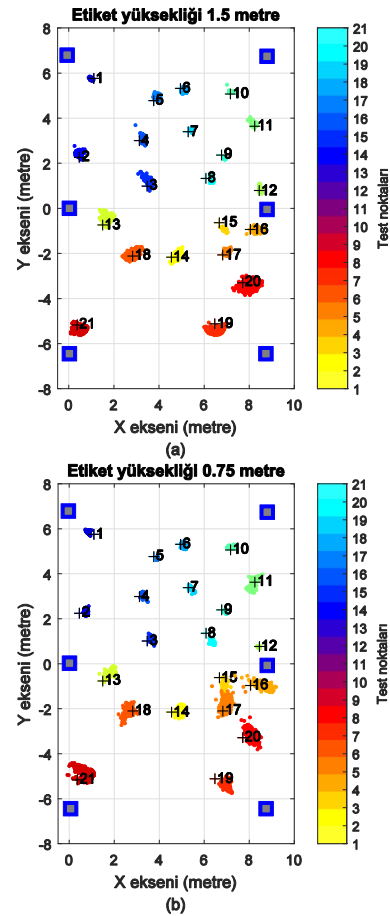
bilinen statik test noktaları için deneysel sonuçların elde edilmesi ve değerlendirilmesi Bölüm 2’de sunulmaktadır. UWB sisteminin sağladığı tahminlerin iyileştirilmesi için kullanılacak olan HDOP yöntemi, LS ve NLR algoritmaları Bölüm 3’te anlatılmaktadır. Önerilen tekniğin literatürde konum belirleme tekniklerinin iyileştirilmesi için kullanılan LS ve NLR algoritmaları ile karşılaştırılmasını kapsayan ayrıntılı konum belirleme performansları Bölüm 4’te sunulmaktadır. Son olarak Bölüm 5’te tartışma ve sonuçlar verilmektedir.

2. UWB KONUM BELİRLEME SİSTEMİ (UWB LOCALIZATION SYSTEM)

Genel bir UWB konum belirleme sisteminde tipik olarak etiket (tag) olarak adlandırılan; kontrol altında tetiklendiğinde UWB sinyallerini yayan bir verici ve UWB sinyallerini alan birden fazla alıcı vardır. Bu sistemler hem statik ölçümleri hem de hareketli ölçümleri gerçekleştirebilmektedir. Piyasada farklı firmaların farklı problemlere yönelik konum belirleme sistemleri mevcuttur [6]. Bu çalışmada kullanılan UWB konum belirleme sistemi, geleneksel tek yönlü zaman bölüşümlü çoklu erişim kontrol kanalı kullanan bir konum belirleme platformudur ve konumlama işlemi için AOA ve TDOA sinyal parametrelerini kullanmaktadır. 6-8.5 GHz frekans bandında çalışan alıcılar bina içinde sabit ve ölçüm yapılacak alana göre yüksek noktalara yerleştirilmiştir. Geleneksel radyo frekans (RF) sinyalinde, bina içerisinde kesin sinyal zamanlaması yapmak zordur. Ancak UWB sinyalleri ile direk yol, yansımalarından ayrılabilirdiği için kesin sinyal zamanlaması yapmak mümkündür. Konumları üç boyutlu olarak tespit edilmiş olan alıcılar, veri kabloları ile merkezi bir bilgisayarında bulunduğu aynı ağa bağlanmışlardır. Bu bilgisayar UWB konum belirleme sisteminin etiket ve alıcılardan sonraki üçüncü bileşeni olan konum belirleme yazılımını içermektedir. Bu yazılım tüm sistemin kurulumu, kalibrasyonu ve kullandığı algoritmalar ile alıcı ölçümlerinden konum tahminlerinin bulunmasını sağlamaktadır. Sistem konum tahmininde her bir alıcı etiket için bir AOA vektörü ve TDOA eğrisini oluşturur. Bu doğru ve eğrinin kesişim noktası etiketin konumunu vermektedir. Tüm alıcıların eş zamanlı çalışabilmeleri ve daha kesin konum belirleme performansı için yapılan kalibrasyonda alıcıların üç boyutlu koordinat sisteminde nereye yerleştiğinin tespiti ve aynı ağa bağlı olan bilgisayarda bulunan yazılıma bu koordinatların biçimde girilmesi gerçekleştirilmiştir.

UWB konum belirleme sistemi 159 m boyutlarındaki laboratuvar ortamında, Şekil 1’de görüldüğü gibi toplam 6 adet alıcıya sahiptir. Alıcıların tamamı yaklaşık olarak 2.30 m yüksekliğindeki sütunların üzerine gerekli kalibrasyon işlemleri yapılarak konumlandırılmışlardır. Bu alıcıların 4 tanesi laboratuvarda birbirlerini görecek

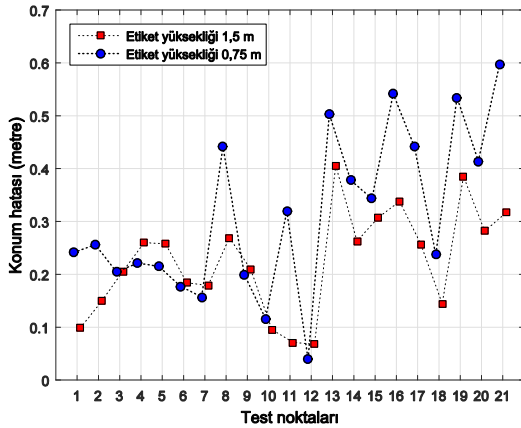
biçimde diğerleri ise koridora yerleştirilmişlerdir. Toplam 21 tane statik ölçüm noktası rasgele belirlenmiştir. Ölçümlerde etiket 0.75 m ve 1.5 m yüksekliğinde üç ayaklı sehpa sabitlenerek konum belirlemesi yapılmıştır. Bu aşamada her bir nokta üzerinde yaklaşık olarak 2.5 dakika beklenerek bir diğer ölçüm noktasına geçilmiştir. Şekil 1’de UWB konum belirleme sisteminin farklı yükseklikteki iki etiket için alıcılardan sağlanan ölçümler ile her bir ölçüm noktası için tespit ettiği tahmin noktaları gösterilmektedir. Etiket yüksekliğinin 1.5 m olduğu konum belirleme tahminlerine bakıldığında birbirini gören alıcıların bulunduğu laboratuvardaki test noktalarındaki hatanın koridorlardaki noktalara göre çok daha küçük olduğu görülmektedir. Etiket yüksekliğinin 0.75 m olduğu ölçümlere bakıldığında da benzer konum tahmin sonuçları görülmektedir. Bu konum belirleme hatasının bu noktalar için fazlalığı koridorlardaki zayıflatıcı etkinin fazla olmasıdır.



Şekil 1. UWB konum belirleme sisteminin farklı yüksekliklerdeki etiketler için tahmin noktaları (The estimated positions of the tags at distinct pitch for the UWB localization system)

UWB konum belirleme sisteminin elde etmiş olduğu verilerden elde edilen konum hatalarının her iki etiket yüksekliği içinde ortalama hataları Şekil 2’de

gösterilmektedir. Ortalama hatalara bakıldığında etiket yüksekliğinin 1.5 m olduğu sonuçlar diğer etiket sonuçlarına göre genel olarak daha iyi sonuçlar vermektedir. Sonuçlara göre ortalama konum hatalarının tamamı her iki etiket yüksekliği için de genel olarak 0.6 m'nin altında kaldığı görülmektedir. Tüm konum tahminlerinin ve test noktalarının ortalamalarında 0.75 m ve 1.5 m etiket yükseklikleri için sırasıyla 0.32 m ve 0.23 m konum hatası elde edilmiştir.



Şekil 2. UWB konum belirleme sisteminin ortalama konum hataları (The mean position errors of the UWB localization system)

3. KULLANILAN YÖNTEMLER (METHODS)

UWB konum belirleme sisteminin sağladığı konum verilerinin iyileştirmesi için önerilen yatay hassaslık ölçeğinin kullanımı ile daha fazla alıcıdan elde edilen konum verilerin tespiti yapılabilmektedir. Daha fazla alıcı ile elde edilmiş konum verisi daha iyi yatay hassasiyete sahiptir. Yatay hassasiyetin iyileşmesi elde edilecek konum verilerinin de iyileşmesi anlamına gelmektedir [20]. Literatürde var olan ve konum verilerinin iyileştirilmesi için de kullanılan LS ve NLR teknikleri ise iteratif işlemler dizisi ile konum verilerinin iyileştirilmesini gerçekleştirmektedirler. Bu yöntemler bu bölümde anlatılmaktadır.

3.1. En Küçük Kareler Yöntemi (Least Mean Squares Method)

En küçük kareler (Least Square, LS) yöntemi, konumu tespit edilmeye çalışılan bir hedefin sistemde kullanılan alıcılara olan uzaklıklarının bir denklem sistemi içerisinde yerleştirilerek, hatanın iyileştirilmesi için kullanılan geleneksel çözüm yöntemlerinde birisidir. Bu yöntem sonsuz uzayların kesiştiğini göz önüne almaktadır. Bu uzayların kesişim noktası en küçük kareler yöntemi ile tespit edilebilmektedir. Bu tanımlamadan yola çıkılarak, alıcı konumlarının (x_i, y_i, z_i) ve (x_j, y_j, z_j) noktaları ve hedef noktanın bu alıcılara olan uzaklıklarının (d_i, d_j)

olduğu varsayıldığında; hedefin konumu için Denklem (1) yazılır.

$$\begin{aligned} (x_i - x)^2 + (y_i - y)^2 + (z_i - z)^2 &= d_i^2 \\ (x_j - x)^2 + (y_j - y)^2 + (z_j - z)^2 &= d_j^2 \end{aligned} \quad (1)$$

Denklem (1)'in varlığında kesişen uzaylar arasındaki fark Denklem (2) ile ifade edilir.

$$\begin{aligned} 2(x_j - x_i)x + 2(y_j - y_i)y + 2(z_j - z_i)z \\ = d_i^2 - d_j^2 - x_i^2 - y_i^2 - z_i^2 + x_j^2 \\ + y_j^2 + z_j^2 \end{aligned} \quad (2)$$

Denklem (2)'nin sürekli ve doğrusal formu Denklem (3) ile gösterilmiştir.

$$\begin{aligned} \mathbf{R}\mathbf{p} = \mathbf{O} \Rightarrow \mathbf{p} = [\hat{x} \quad \hat{y} \quad \hat{z}], \\ \mathbf{R} = \begin{bmatrix} 2(x_0 - x_1) & 2(y_0 - y_1) & 2(z_0 - z_1) \\ 2(x_0 - x_2) & 2(y_0 - y_1) & 2(z_0 - z_1) \\ 2(x_0 - x_3) & 2(y_0 - y_1) & 2(z_0 - z_1) \end{bmatrix}, \end{aligned} \quad (3)$$

$$\mathbf{O} = \begin{bmatrix} d_1^2 - d_0^2 - x_1^2 - y_1^2 - z_1^2 + x_0^2 + y_0^2 + z_0^2 \\ d_2^2 - d_0^2 - x_2^2 - y_2^2 - z_2^2 + x_0^2 + y_0^2 + z_0^2 \\ d_3^2 - d_0^2 - x_3^2 - y_3^2 - z_3^2 + x_0^2 + y_0^2 + z_0^2 \end{bmatrix}$$

Denklem (3)'teki eşitlikler kullanılarak hedef nokta için üç boyutlu konum tespiti Denklem (4) ile ifade edilir. Denklem (4) \mathbf{R} matrisinin uygun Rank değerine sahip olması durumunda tahmin edilecek konum için üç boyutlu konum bilgisi sağlanmaktadır.

$$\mathbf{p} = (\mathbf{R}^T \mathbf{R})^{-1} \mathbf{R}^T \mathbf{O} \quad (4)$$

3.2. Doğrusal Olmayan Regresyon Yöntemi (Non-Linear Regression Method)

Bu yöntem, lineer olmayan bir veya daha fazla sayıda bilinmeyen parametreye dayanan bir tahmin eşitliği ile karakterize edilmektedir. Kalıntı (*Residual*) hataların karelerinin toplamını minimize etmesi sebebiyle; tahmin edilen uzaklıkların kolaylıkla kullanıldığı bu teknik, konum belirleme için çoğu zaman uygun çözümler sunabilmektedir. Doğrusal olmayan regresyonun bir kez uygulanması ile tüm ölçümlere dayanan bir konum tahmini gerçekleştirilir. Alıcının pozisyonu (u, v, w) , etiketin pozisyonu (x_i, y_i, z_i) varsayıldığında; alıcı ile etiket pozisyonu arasında yapılacak ölçümde mutlaka bir ölçüm hatası olmaktadır. Bu iki nokta arasındaki ölçüm hatasını içeren uzaklık Denklem (5) ile gösterilmiştir.

$$r_i = \sqrt{(x_i - u)^2 + (y_i - v)^2 + (z_i - w)^2} + \varepsilon_i \quad (5)$$

Denklem (5)'te ε_i , ölçüm hatasını temsil etmektedir. Ayrıca tahmin edilen noktanın alıcıya olan uzaklığı da Denklem (6) ile verilmiştir.

$$d_i = \sqrt{(x_i - \hat{x}_i)^2 + (y_i - \hat{y}_i)^2 + (z_i - \hat{z}_i)^2} \quad (6)$$

Denklem (5) ve (6) ile konum tahmini Denklem (7)'de verilen standart kalıntı hatasının belirlenmesiyle elde edilir.

$$\sigma^2 = \frac{1}{N} \sum_i^N (r_i - d_i)^2 \quad (7)$$

Denklem (7)'de r_i noktanın alıcı ile arasındaki ölçüm hatasını da içeren uzaklığını, d_i tahmin edilen noktanın alıcıya olan uzaklığı ve N bu nokta için kullanılan tahmin sayısını temsil etmektedir. Elde edilen kalıntı hatasının üstünde kalan ölçümler dışındaki diğer ölçümler hedef konum tahminleri olarak kullanılmaktadırlar.

3.3. Yatay Hassaslık Ölçeği Yöntemi (Horizontal Dilution of Precision Method)

Küresel konumlama sistemi sağlamış olduğu avantajlar sebebiyle; kablosuz haberleşme alanında birçok çalışmaya konu olmuştur. Kesin bir konumlama işlemi için bu sistemin o anki uydu geometrisinin konum belirlemeye uygun olup olmadığına karar vermesi gerekmektedir. Bu kararın verilebilmesi için çalışmalar genellikle hassaslık ölçeğini (Dilution of Precision, DOP) kullanılmaktadırlar. Bu ölçeğe göre yapılacak olan konum belirleme işleminin güvenilirlik seviyesi belirlenmiş olacaktır. Hesaplanan DOP değerinin yüksek çıkması uydu geometrisinin iyi olmadığı, düşük çıkması ise uydu geometrisinin uygun olmadığı anlamına gelmektedir. DOP ölçeği farklı türler ile ifade edilebilmektedir. Bunlardan biri de yatay hassaslık ölçeğidir (HDOP). HDOP ölçeği yatay düzlemde gerçekleştirilen konumlama işlemlerinde kullanılır ve matematiksel olarak uyduların geometrilerinden hesaplanabilmektedir. HDOP değerinin matematiksel olarak hesaplanmasında uyduların bilinen üç boyutlu konumları kullanılarak, konumu tahmin edilen alıcının her bir uyduya olan uzaklıkları tespit edilir ve Denklem (8) ile hesaplanır.

$$R_i = \sqrt{(x_i - R_x)^2 + (y_i - R_y)^2 + (z_i - R_z)^2} \quad (8)$$

Burada R_i tahmin edilen nokta ile uydu arasındaki uzaklığı göstermektedir. R_x , R_y ve R_z konumu tahmin

edilen alıcının üç boyutlu konum bilgilerini, x , y , z terimleri ise uyduların konum bilgilerini temsil etmektedir. Bu uzaklık bilgileri ile beraber Denklem (9) ile verilen formüller kullanılarak her bir uydu için yönlü türevler alınır.

$$\begin{aligned} Dx_i &= \frac{x_i - R_x}{R_i}, Dy_i = \frac{y_i - R_y}{R_i}, \\ Dz_i &= \frac{z_i - R_z}{R_i}, Dt_i = -1 \end{aligned} \quad (9)$$

Burada D_i uyduların zamanları ile ilgili işlem yapılacağıda kullanılabilir (Time DOP, TDOP). Alıcının konum tahmininin doğrulanması için elde edilen bu veriler tüm uydular için sırasıyla bir matrise (G) yerleştirilir. Elde edilen bu matrisin kovaryansı kullanılarak DOP hesaplamaları gerçekleştirilir. DOP hesaplamaları Denklem (10) ile gösterilmektedir.

$$(G^T G)^{-1} = \begin{bmatrix} \sigma_{xx}^2 & \sigma_{xy}^2 & \sigma_{xz}^2 & \sigma_{xt}^2 \\ \sigma_{yx}^2 & \sigma_{yy}^2 & \sigma_{yz}^2 & \sigma_{yt}^2 \\ \sigma_{zx}^2 & \sigma_{zy}^2 & \sigma_{zz}^2 & \sigma_{zt}^2 \\ \sigma_{tx}^2 & \sigma_{ty}^2 & \sigma_{tz}^2 & \sigma_{tt}^2 \end{bmatrix} \quad (10)$$

Bu matrisin köşegen elemanları, her bir eksen ve kullanıcı zamanı için tahmin edilen kullanıcı pozisyonunun varyansını temsil etmektedir. Elde edilen matrisin elemanları kullanılarak HDOP değeri Denklem (11) ile tespit edilmektedir.

$$HDOP = \sqrt{\sigma_{xx}^2 + \sigma_{yy}^2} \quad (11)$$

HDOP ölçeğinin matematiksel olarak hesaplanmasından sonra, bu ölçeğin konum tahminlerinin uygun olup olmadığına karar vermesi için bir seviye tespiti yapılması gerekmektedir. Bunun için algılayıcı alanın tamamına bir HDOP analizi yapılması gerekmektedir. Bu analizin sonucunda Denklem (12) ile verilen matematiksel ifade ile bir HDOP eşik seviyesi ($HDOP_L$) tespit edilebilmektedir.

$$\mu_{HDOP_B} + \sigma_{HDOP_B} \leq HDOP_L \leq \mu_{HDOP_W} + \sigma_{HDOP_W} \quad (12)$$

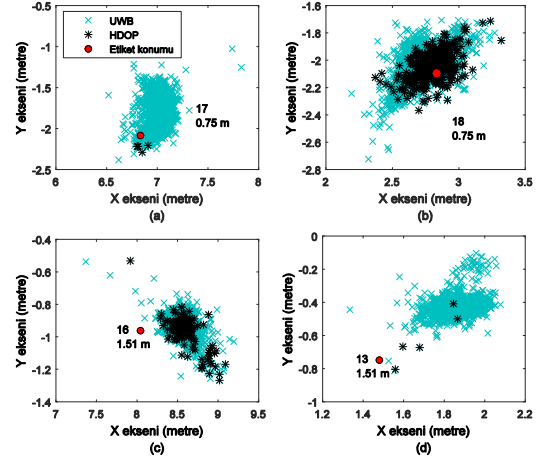
Burada μ_{HDOP_B} ve μ_{HDOP_W} sırasıyla uydu geometrisinin en iyi ve en kötü olduğu durumlarda tüm alan için yapılacak HDOP hesabının ortalamalarını, σ_{HDOP_B} ve σ_{HDOP_W} ise standart sapmalarını temsil etmektedir. Bu önerilen yöntemin kapalı alanda çalışan UWB konum tahmin sistemine uygulanması ile altı alıcının yaptığı ölçümlerin kalitelerinin belirlenmesi ve kalitesiz ölçümlerin elenmesi ile gerçekleştirilmektedir.

4. KONUM BELİRLEME SONUÇLARI (LOCALIZATION RESULTS)

UWB konum belirleme sistemi ile konum belirleme deneysel sonuçları iki farklı etiket yüksekliğinde 21 farklı test noktası için elde edilmiştir. Bu test noktalarının her biri için etiket yüksekliği 1.5 m olduğu durumda UWB konum belirleme sistemi test noktaları için 1500 ile 2200 konum tahmini gerçekleştirmiştir. Elde edilen bu tahmin sayıları yaklaşık olarak 100 ms'de bir okuma ile gerçekleştirilmiştir. Bu elde edilen tahminler konum belirleme algoritmalarının her birinde performans değerlendirilmesi için kullanılmıştır.

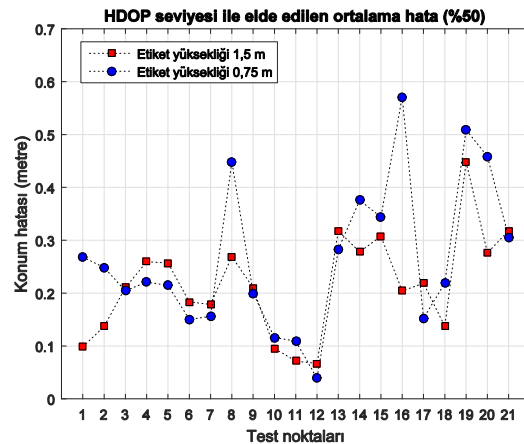
İlk olarak, deneysel sonuçların uygulandığı ve HDOP yöntemi olarak önerilen yöntem için laboratuvar ortamında farklı sayıda alıcılardan veri aldıkları durumda HDOP seviyelerinin tespiti gerçekleştirilmiştir. Buna göre alanın tamamında bütün alıcılardan veri alındığı durumdaki HDOP seviyelerini ile yalnızca üç alıcıdan veri alındığındaki HDOP seviyelerini göz önüne alınarak HDOP eşik değeri seçiminde gerçekleştirilmiştir. Tüm alıcılardan veri alındığı koşulda; bu değer 0.96 - 1 arasında tespit edilmiştir. Bu değer açık alanlarda konum belirlemede de ideal olarak nitelendirilmektedir. Yani tüm alıcılardan veri alındığında gerçekleştirilen konum belirleme işlemindeki alıcı geometrisinin en iyi olduğunu göstermektedir. Farklı kombinasyonlardaki üç değişik alıcıdan veri alındığı koşulda; bu değer 1.5 ile 5 arasında olduğu görülmüştür. Bu seviye kabul edilebilir seviyenin içerisinde olmakla beraber, alıcı geometrisinin her zaman için iyi olmadığı anlamına gelmektedir. Ancak HDOP seviyelerinin düşük olması alıcı geometrilerinin iyi olduğunu göstermektedir. Bu nedenle UWB sisteminin deneysel sonuçlarının iyileştirilmesi aşamasında $HDOP_L$ eşik seviyesi olarak çok iyi seviye olarak kabul edilen 1-2 aralığında 1.5 olarak seçilmiştir.

Şekil 3, test alanında gerçekleştirilen ölçümlerin HDOP değerine göre elimine edilmesi sonucunda kalan ölçümleri göstermektedir. Etiket yüksekliğinin 0.75 m olduğu sonuçlara göre Şekil 3(a)'da 17 numaralı test noktası için ölçümlerinin %99'u elimine edilerek, kalan tahminlerin ortalaması 0.15 m ile %64'lük bir iyileşme gerçekleşmiştir. Şekil 3(b)'de 18 numaralı noktanın test ölçümlerinin %51'ine yakını HDOP seviyesinin üstünde kalmaktadır. Bu ölçümlerin filtrelenmesiyle, ortalama hatada %8'lik bir iyileşme elde edilmiştir. Etiket yüksekliğinin 1.5 m olduğu durumda test noktası 16 için, Şekil 3(c)'de 0.33 m olan ortalama hata 1570 test ölçümünden 1417'sinin elimine edilmesi ile 0.20 m tespit edilmiştir. Son olarak, Şekil 3(d)'de test noktası 13 için 1576 ölçümün 5 tanesi istenilen HDOP seviyesinin altında kalmıştır. Bu 5 ölçümün ortalama hatası ise 0.32 m olarak tespit edilmiştir.



Şekil 3. Test noktalarının HDOP seviyesi ile filtrelenmesi sonucunda kalan ölçümler (The rest measurements of test points by filtering with HDOP level)

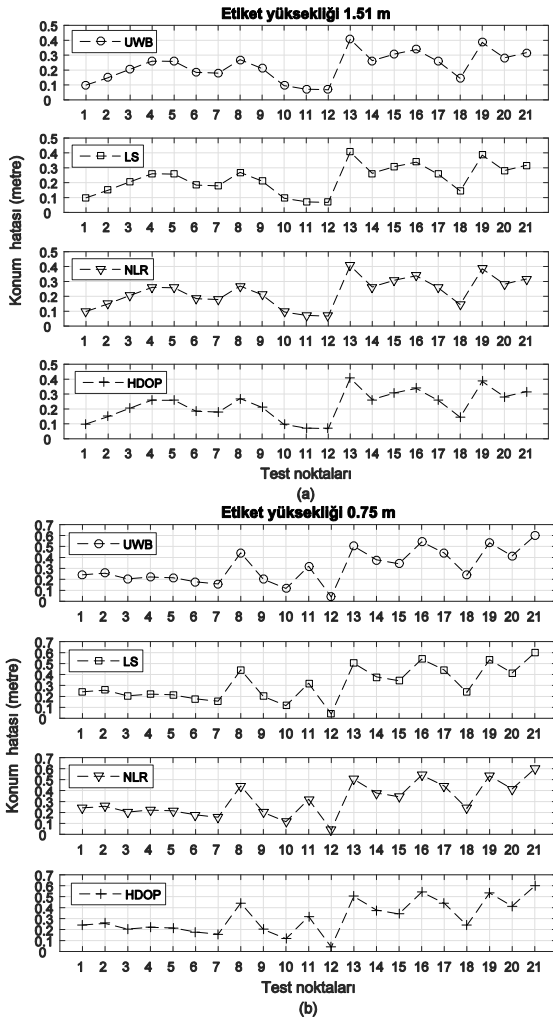
HDOP eşik seviyesinin kullanılarak yapılan filtreleme işlemi sonunda tüm test noktaları ve her iki etiket yüksekliği için elde edilen ortalama pozisyon hataları Şekil 4'te gösterilmektedir. Bu sonuçlara göre etiket yüksekliği 0.75 m için yalnızca 16 ve 19 numaralı test noktaları hariç, diğer tüm test noktalarının ortalama konum hataları 0.5 m'nin altında kalmaktadır. Bu değer 1.5 m yüksekliğindeki etiket sonuçları için ise; 19 numaralı test noktası hariç 0.4 m olarak tespit edilmiştir. Bu sonuçlar genel bir değerlendirme açısından işlenmemiş ölçümler ile kıyaslandığında; tüm test noktaları ve etiket yüksekliği 0.75 m için 0.3133 m olan ortalama konum hatası HDOP eşik seviyesi kullanıldığında 0.2664 m'ye gerilerken, 1.5 m etiket yüksekliğinde ise 0.2261 m iken 0.2159 m olarak elde edilmiştir.



Şekil 4. Test alanındaki tüm noktaların HDOP seviyesi ile filtrelenmesi sonucunda elde edilen ortalama konum hataları (The mean localization errors of all test points obtained by HDOP level)

Şekil 5'te deneysel ölçümlerin, önerilen HDOP tekniği ve diğer algoritmalar ile iyileştirilmesi sonucu elde edilen

ortalama hata performanslarının karşılaştırılması sunulmaktadır. Şekil 5(a) etiket yüksekliğinin 1.5 m olduğu sonuçlara göre, LS algoritmasının tüm test noktaları için konum tahmini gerçekleştirebildiği, 1 ve 7 numaralı test noktaları gibi bazı test noktaları için herhangi bir iyileştirme sağlayamadığı, 2 ve 8 gibi bazı test noktaları içinde performansın kötüleştiği görülmektedir. HDOP tekniği de tüm noktalar için konum tahmini gerçekleştirebilmektedir. Bununla birlikte HDOP tekniği de LS algoritması gibi bazı noktalar için daha iyi performans ortaya koyarken, bazı noktalar için UWB sistem ölçümlerinden daha yüksek hataya sahip tahminler gerçekleştirebilmektedir. NLR algoritmasının genel anlamda LS algoritmasına göre daha iyi gözüken konum tahmin performansı, HDOP tekniğine göre daha kötü olarak gerçekleşmiştir. Etiket yüksekliğinin 0.75 m olduğu Şekil 5(b)'deki sonuçlara göre; tüm teknikler 12 numaralı test noktası için yaklaşık olarak aynı performansta (0.039 m) bir tahmin ortalaması sağlamaktadırlar. Ortalama hatalara bakıldığında en iyi performansı HDOP tekniği sağlamaktadır.



Şekil 4. Test alanındaki tüm noktaların HDOP seviyesi ile filtrelenmesi sonucunda elde edilen ortalama konum hataları (The mean localization errors of all test points obtained by HDOP level)

Elde edilen tüm sonuçlar ve performans değerlendirmeleri sonucunda genel bir konum belirleme değerlendirmesi Tablo 1'de verilmektedir. Buna göre LS algoritması 0.75 m ve 1.5 m etiket yüksekliklerinde sırasıyla yaklaşık olarak %2.52 ve %0.32 oranında performans artırımını, %100'lük konum tahmini gerçekleştirme ile sağlamıştır. NLR algoritması ise tüm test noktaları için 0.75 m ve 1.5 m etiket yüksekliklerinde sırasıyla %9.32 ve %0.36 oranında performans artırımını sağlamıştır. Önerilen HDOP tekniği ise her iki algoritmaya göre daha iyi konum belirleme performansı sağlamaktadır.

Tablo 1. Tüm test noktaları ve etiket yükseklikleri için algoritmaların ortalama hatalarda sağlamış oldukları performans iyileştirmeleri oranları (The performance rate of the algorithms provide at mean position error)

	Konum tahmini iyileşme oranı	
	0.75 m	1.5 m
LS	%2.52	%0.32
NLR	%9.32	%0.36
HDOP	%14.97	%4.4

5. TARTIŞMA VE SONUÇLAR (CONCLUSIONS)

Bu çalışmada, UWB konum belirleme sisteminin deneysel olarak bina içi konum belirleme performansının elde edilmesinin yanında; bu performansının HDOP yöntemi kullanılmasına dayanan bir iyileştirme tekniği sunulmaktadır. Ayrıca bu yöntem ile iki farklı etiket yüksekliğinde elde edilen konum tahminlerinin literatürde mevcut LS ve NLR tekniklerinin performans iyileştirmeleri karşılaştırılması olarak değerlendirilmektedir. UWB konum belirleme sistemi, kalibrasyon ve alıcıların oryantasyon problemlerinden dolayı; bazı test noktaları için konum belirleme sonuçları açısından sistemin genel performansına göre daha kötü sonuçlar verebilmektedir. HDOP ölçeğinin kullanılmasının önerildiği ve bunun için bir eşik seviye tespiti yapılabildiği konum belirleme tekniği ile UWB konum belirleme sisteminin elde ettiği konum tahminleri iyileştirildiğinde, tüm performans değerlendirilmesine göre yaklaşık olarak %7'lik bir performans iyileştirilmesi elde edilmektedir. Bunun dışında önerilen eşik seviye tespiti ile tüm test noktaları için, bu tekniğin konum tahmin performansı ortaya koyması garanti edildiği gösterilmiştir. Bununla birlikte gerçekleştirilen performans karşılaştırılmasında HDOP ölçeğinin kullanıldığı teknik literatürdeki mevcut LS ve NLR algoritmalarına göre her iki etiket yüksekliğinde de daha iyi sonuçlar vermektedir.

KAYNAKÇA (REFERENCES)

- [1] Y. Chen, S. Zhang, S Xu, G. Y. Li, "Fundamental Trade-offs on Green Wireless Networks," IEEE

- Communications Magazine, vol. 49, no. 6, pp. 30-37, 2011.
- [2] P. H. Tseng and K. T. Feng, "Hybrid Network/Satellite-Based Location Estimation and Tracking Systems for Wireless Networks," *IEEE Transactions on Vehicular Technology*, vol. 58, no. 9, pp. 5174-5189, 2009.
- [3] S. C. Yeh, W. H. Hsu, M. Y. Su, C. H. Chen, K. H. Liu, "A Study on Outdoor Positioning Technology using GPS and WiFi Networks," *Int. Conference on Networking, Sensing and Control, Japan*, 26-29 March 2009, pp. 597-601.
- [4] L. Huan and R. Bo, "Wireless location for indoor based on UWB," *34th Chinese Control Conference, Hangzhou*, 2015, pp. 6430-6433.
- [5] S. Gezici and H. Poor H, "Position Estimation via Ultra-Wide-Band Signals", *Proceedings of the IEEE*, vol. 97, no. 2, pp. 386-403, 2009.
- [6] H. Liu, H. Darabi, P. Banerjee, J. Liu, "Survey of Wireless Indoor Positioning Techniques and Systems," *IEEE Tran. on Systems, Man, and Cybernetics, Part C: Applications and Reviews*, vol. 37, no. 6, pp. 1067-1080, 2007.
- [7] D. Niculescu and B. Nath, "Ad Hoc Positioning System (APS) using AOA," *The IEEE 22nd Annual Joint Conference of the IEEE Computer and Communications Societies, USA*, April 2003, vol. 3, pp. 1734-1743.
- [8] A. Catovic and Z. Sahinoglu, "The Cramer-Rao bounds of hybrid TOA/RSS and TDOA/RSS location estimation schemes," *IEEE Communacitions Letter*, vol. 8, pp. 626-628, 2004.
- [9] T. Wang, X. Chen, N. Ge, Y. Pei, "Error analysis and experimental study on indoor UWB TDoA localization with reference tag," *2013 19th Asia-Pacific Conference on Communications, Denpasar*, 2013, pp. 505-508.
- [10] S. Monica and G. Ferrari, "UWB-based localization in large indoor scenarios: optimized placement of anchor nodes," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 51, no. 2, pp. 987-999, 2015.
- [11] X. Li and S. Yang, "The indoor real-time 3D localization algorithm using UWB," *International Conference on Advanced Mechatronic Systems, Beijing*, 2015, pp. 337-342.
- [12] H. Cai, G. Wu, Y. Chen, L. Jiang, "Indoor collaborative localization method based on ultra-wideband ranging," *9th European Conference on Antennas and Propagation, Lisbon*, 2015, pp. 1-2.
- [13] G. Zhao and H. Niu, "Research and application of indoor positioning based on UWB," *Third International Conference on Cyberspace Technology, Beijing*, 2015, pp. 1-5.
- [14] D. Briese, H. Kunze, G. O. Rose, "High Precision UWB-Based 3D Localization for Medical Environment," *IEEE International Conference on Ubiquitous Wireless Broadband, Montreal, QC*, 2015, pp. 1-5.
- [15] G. Feng, C. Shen, C. Long, F. Dong, "GDOP index in UWB indoor location system experiment", *IEEE Sensors, Busan*, 2015, pp. 1-4.
- [16] K. Kucuk, "Horizontal dilution of precision based ultra wide band positioning technique for indoor environments," *Turkish Journal of Electrical Engineering & Computer Sciences*, vol. 22, pp. 1307-1322, 2014.
- [17] M. Baum, "RTL in Longueuil selects bus yard management solution provided by Solotech", *ISR Transit and Ubisense*, 2011.
- [18] W. Chantaweesomboon et al., "On performance study of UWB real time locating system," *2016 7th International Conference of Information and Communication Technology for Embedded Systems (IC-ICTES), Bangkok*, 2016, pp. 19-24.
- [19] J. Youn and Y. Cho, "Performance Study of an Ultra-Wideband Indoor Localization and Asset Tracking System," in title of *Ultra Wideband, Rijeka, Croatia, InTech Europe*, 2010.
- [20] R. Yarlagadda, I. Ali, N. Al-Dhahir, J. Hershey, "GPS GDOP Metric," *Radar, Sonar and Navigation IEE Proceedings*, vol. 147, no. 5, pp. 259-264, 2000.
- [21] J. Yan, C. C. J. M. Tiberius, P. J. G. Teunissen, G. Bellusci and G. J. M. Janssen, "A Framework for Low Complexity Least-Squares Localization With High Accuracy," in *IEEE Transactions on Signal Processing*, vol. 58, no. 9, pp. 4836-4847, Sept. 2010.
- [22] S. Marano, W. M. Gifford, H. Wymeersch and M. Z. Win, "NLOS identification and mitigation for localization based on UWB experimental data," in *IEEE Journal on Selected Areas in Communications*, vol. 28, no. 7, pp. 1026-1035, September 2010.

PID Denetleyici Uygulamaları İçin Yeni Bir Online Deney Seti Tasarımı

Sezgin Kaçar^{1*}, Ali Fuat Boz², Burak Arıcıoğlu³, Harun Tekin⁴

09.09.2016 Geliş/Received, 31.10.2016 Kabul/Accepted

doi: 10.16984/saufenbilder.12551

ÖZ

Bu çalışmada mühendislik eğitiminde destek olarak kullanılabilen, DC motorun PID denetleyici ile hız denetimi için internet üzerinden uzaktan erişimli deney düzeneği geliştirilmiştir. Deney düzeneği kablosuz olup, ortamda bulunan bir kablosuz ADSL modem aracılığıyla TCP/IP üzerinden uzak sunucu ile haberleşmektedir. Kullanıcılar internet bağlantısı olan herhangi bir bilgisayardan uzak sunucudaki web sayfalarına erişerek gerçek zamanlı olarak deneyleri gerçekleştirebilmektedir. Tasarlanan etkileşimli web sayfaları üzerinden, kullanıcılar motorun referans hız değerini ve PID denetleyici parametrelerini değiştirerek hızın değişimini görebilmekte ve ölçüm değerlerini bilgisayarına kaydedebilmektedir. Ayrıca bir web kamerası yardımıyla deney setinin çalışması da web sayfası üzerinden izlenebilmektedir. Ayrıca deney düzeneği için çevre birimleri hazırlanarak kullanıcı-deney seti etkileşimi artırılmıştır. Bu kapsamda kullanıcı web sayfası üzerinden deney setinin bulunduğu ortamın sıcaklığını görebilmekte, deney seti üzerindeki LCD'ye mesajını yazdırabilmekte ve istediği takdirde deney setini aydınlatabilmektedir.

Anahtar Kelimeler: Uzaktan erişim, Mikrodenetleyici, PID, TCP/IP, uzak sunucu

Design Of A Novel Online Experiment Setup For PID Controller Applications

ABSTRACT

In this study, an internet based remote access experiment setup was developed for induction direct current motor speed control with PID controller which can be used as a support material in engineering education. The experiment setup is wireless and communicates with the remote server using transfer control protocol/internet protocol through a wireless ADSL modem. Users can perform the experiments as real time accessing the web pages in the remote server by using any computer which has internet connection. By means of interactively-designed web pages, users can monitor the speed change executing alterations of the PID controller parameter and motor reference speed. Also users can save the measured values on their own computers. In addition to this, with the support of a webcam, the running of the experimental set can be monitored on the web page. Additionally, for the experimental set, preparing the peripheral units card, the interaction was expanded between the user and the experimental set. Relatively to this, the user can monitor the ambient temperature of the experimental set's current place on the web page and can make his/her own message write on LCD of the experimental set and can enlighten it if he/she wants.

Keywords: Remote Access, Microcontroller, PID, TCP/IP, Remote Server.

* Sorumlu Yazar / Corresponding Author

1 Sakarya Üniversitesi, Teknoloji Fakültesi, Elektrik-Elektronik Mühendisliği, Sakarya - skacar@sakarya.edu.tr

2 Sakarya Üniversitesi, Teknoloji Fakültesi, Elektrik-Elektronik Mühendisliği, Sakarya - afb@sakarya.edu.tr

3 Sakarya Üniversitesi, Teknoloji Fakültesi, Elektrik-Elektronik Mühendisliği, Sakarya - baricioğlu@sakarya.edu.tr

4 Milli Eğitim Bakanlığı - haruntekin06@gmail.com

1. GİRİŞ (INTRODUCTION)

Üniversitelerde bulunan laboratuvar alt yapıları, kısıtlı laboratuvar saatleri, yardımcı eleman azlığı ve öğrenci sayısı ile birlikte geleneksel deneylerin değişik kısıtlamaları göz önüne alındığında laboratuvar ortamındaki gerçek uygulamaların zaman, maliyet, uygulama zorluğu açısından sıkıntılı durumlar oluşturduğu görülmektedir. Bahsedilen problemlerin giderilmesi için tüm dünyada geleneksel laboratuvarlara uygun bir alternatif olarak uzaktan laboratuvarların oluşturulması yaygınlaşmıştır [1-3].

Günümüz bilişim teknolojisi sayesinde, internet üzerinden laboratuvar düzenek ve cihazlarına uzaktan erişim, ya da bunların benzetimi mümkün olmaktadır [1]. Bilişim teknolojilerinin bu şekilde kullanılmasıyla internet tabanlı laboratuvar ortamları geliştirilmiştir. Bu laboratuvar ortamlarına günün her saati uzaktan erişilerek çalışmalar istenildiği kadar tekrarlanabilmektedir. Literatürde, iki çeşit internet tabanlı laboratuvar ortamı göze çarpmaktadır. Birincisi benzetim veya gösterilerle desteklenmiş sanal laboratuvarlar, ikincisi uzaktan erişimli deney veya ölçüm düzenekli gerçek laboratuvarlardır [1-3].

İnternet tabanlı laboratuvar ortamları öğrencilerin öğrenmesine katkıda bulunan, eldeki tüm kaynakları en iyi bir şekilde kullanmayı sağlayan, etkin ve etkileşimli ortamlardır. İnternet tabanlı laboratuvar ortamları, zaman ve yer esnekliği, grup ile öğrenme kolaylığı ve kaynaklara anında ulaşma gibi nedenlerden dolayı çekici hale gelmiştir [2]. Uluslararası alanda bu konuda yapılan çalışmalar incelendiğinde, özellikle benzetim tabanlı çalışmalara ağırlık verildiği gözlenmiştir [4-14].

Benzetim tabanlı laboratuvar uygulamaları her ne kadar önemli olsa da, hiçbir zaman gerçek laboratuvar çalışmasının yerini alamaz. Bunun alternatifi ise internet üzerinden kontrol edilebilen, gerçek zamanlı ve uzaktan erişimli deney düzenekleri geliştirmektir. Literatürde bu konu üzerinde birçok çalışma yapılmıştır. Örneğin; çok girişli, çok çıkışlı eş iki tank düzeneği üzerine bir laboratuvar deneyi, robot kontrolü için bir gerçek zamanlı web platformu çalışması, telerobotik, online örüntü tanıma sistemi, kontrol sistemleri deneyleri, güç elektroniği deneyleri, ağlar arası iletişim çalışmaları, elektrik makinelerinin kontrolü ve benzer birçok konuda, mühendislik eğitimine ve teknik eğitime katkı sağlayabilecek gerçek zamanlı uygulamalara literatürde rastlanılmaktadır [2, 3, 15-19].

Bu çalışmada mühendislik eğitimi ve teknik eğitime yönelik geleneksel veya uzaktan eğitimde destek olarak kullanılacak, internet üzerinden erişimli ve kablosuz bir deney seti geliştirilmiştir. Geliştirilen deney seti ile

DC motorun hız denetimi PID (Proportional, Integral, Derivative) denetleyici ile gerçekleştirilmiştir. Kullanıcılar, internet bağlantısı olan herhangi bir bilgisayardan deney setine bağlanarak, PID denetleyici parametrelerini, arzu ettikleri motor hız değerini ve denetim süresini belirleyebilmekte ve deneyden elde edilen sonuçları web sayfası üzerinde grafik ve tablo verileri şeklinde görebilmektedirler. İstedikleri takdirde bu bilgileri kendi bilgisayarlarına kaydederek verilerin analizini de yapabilmektedirler. Ayrıca, bir web kamerası yardımıyla deney setinin çalışması da web sayfasından izlenebilmektedir.

Deney seti ile kullanıcı arasındaki etkileşimin artırılması için deney seti çevre birimleri kartı hazırlanmıştır. Hazırlanan bu kart deney seti ile birleştirilerek deney setinin sunduğu işlevler artırılmıştır. Kullanıcıya bu kart ile deney setinin bulunduğu ortamın sıcaklığını görebilme, deney seti üzerindeki LCD'ye mesajını yazdırabilme ve deney ortamının karanlık olma ihtimaline karşılık istediği takdirde deney setini aydınlatabilme imkanları verilmiştir.

Yapılan çalışmanın en önemli avantajları, kullanıcıların internet bağlantısı olan herhangi bir yerden, herhangi bir zamanda sisteme bağlanarak deneylerini gerçekleştirebilmeleri, deney setinin kablosuz olması ve deney ortamında veya yakınında sunucu olarak çalışacak bir bilgisayara ihtiyaç duyulmamasıdır. Bunlara ek olarak, geleneksel laboratuvar çalışmalarında deney esnasında yüksek gerilime maruz kalma, keskin veya döner parçalardan yaralanma gibi öğrencilerin karşılaştıkları riskler de ortadan kaldırılmıştır.

Çalışmada, ikinci bölümde PID denetim ile alakalı teorik bilgi verilmiştir. Üçüncü bölümde deney seti tasarımı anlatılmıştır. Dördüncü bölümde sistemin haberleşme yapısı anlatılarak geliştirilen web ara yüzünden bahsedilmiştir. Son bölümde ise sonuç ve değerlendirmelerden bahsedilmiştir.

2. TEORİK ALTYAPI (THEORITICAL BACKGROUND)

Bu kısımda oransal (P), integral (I) ve türevsel (D) denetleyicilerin karakteristikleri ile istenen cevabın elde edilmesi için bu denetleyicilerin nasıl kullanılacağı gösterilmektedir. Şekil 1' de birim geri beslemeli bir sistem görülmektedir. Burada $u(t)$; denetleyici çıkışı, $e(t)$; hata sinyali, K_p ; oransal kazanç, $c(t)$; sistem çıkışı, $r(t)$; referans değeridir. Hata sinyali denetleyici bloğundan geçerek uygun şekle gelir. Buradaki denetleyici P, PI, PD veya PID gibi herhangi bir denetim algoritmasını içerebilecek bir bloktur. Burada önce P, I ve D denetleyicilerin karakteristiklerinden, sonra

sırasıyla P, PI, PD ve PID denetleyicilerden bahsedilecektir.



Şekil 1. Geri beslemeli basit bir sistemin blok şeması (Block diagram of a simple feedback system)

2.1. Oransal (P) Denetim (Proportional Control)

Oransal denetimde Şekil 1'deki denetleyici yerine sabit bir K_p kazancı gelir. Oransal denetimde denetleyici, sistemin talep ettiği enerjiyi sürekli olarak denetleme değişkenini ayarlayarak verir. Gereksinim duyulan enerji ile verilen enerji arasında bir dengeye ihtiyaç vardır. Denge durumuna gelene kadar sistem çıkışında değişim olması, hatta sistem çıkışı ile referans değeri arasında belli bir fark kalması oransal denetimin en belirgin özelliğidir. Denetim çıkışının zaman tanım bölgesindeki ifadesi denklem 1' deki gibidir.

$$u(t) = K_p e(t) \quad (1)$$

Referans değeri ile sistemin oturduğu ve sabit kaldığı değer arasındaki farka kararlı durum hatası (off-set) denir. Kararlı durum hatasını azaltmak için oransal bant küçültülebilir. Ancak oransal bant küçüldükçe aç-kapa denetime yaklaştığı için referans değeri etrafında salınımlar artabilir. Geniş oransal bantta kararlı durum hatasının büyük olacağı düşünülerek sisteme en uygun oransal bantın seçilmesi gerekir.

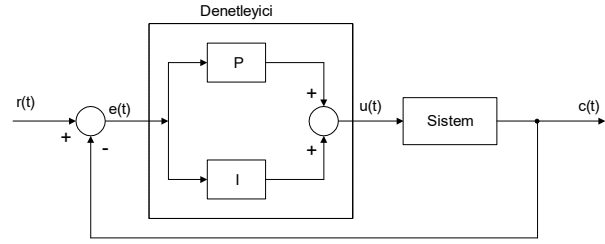
2.2. PI Denetim (PI Control)

Oransal denetimde oluşan kararlı durum hatası, el ile (manuel) veya otomatik olarak kaldırılabilir. Otomatik sıfırlama (resetleme) için denetleyicide, elektronik integratör devresi kullanılır. Ölçülen değer ile referans değeri arasındaki fark sinyalinin, yani hata sinyalinin zamana göre integrali alınır. İntegratör devresiyle sisteme verilen enerji otomatik olarak artırılır veya azaltılır ve sistem referans değerine oturtulur.

İntegratör devresi gerekli enerji değişikliğine, referans değeri ile ölçülen değer arasındaki fark kalmayınca kadar devam eder. Fark sinyali sıfır olduğu anda artık integratör devresinin integralini alacağı bir sinyal söz konusu değildir.

Herhangi bir şekilde bazı değişiklikler olup, çıkış değeri referans değerinden uzaklaşacak olursa tekrar hata sinyali oluşur ve integratör devresi düzeltici etkiyi

gösterir. Şekil 2' de PI denetim sistemi blok şema halinde verilmektedir.



Şekil 2. PI denetim sistemi blok şeması (Block diagram of PI control)

PI denetimin en belirgin özelliği, ilk başlatmada sistem çıkışının referans değerini geçip, önemli bir yükselme yapmasıdır (overshoot). Çıkış, referans değeri etrafında bir-iki salınım yaptıktan sonra referans değerine oturur. Blok şemada görüldüğü gibi hata sinyali PI denetleyiciden geçerek sistemi referans değerine oturtmaya çalışır. Denetim çıkışının zaman tanım bölgesindeki ifadesi denklem 2' deki gibidir.

$$u(t) = K_p e(t) + K_I \int_0^t e(t) dt \quad (2)$$

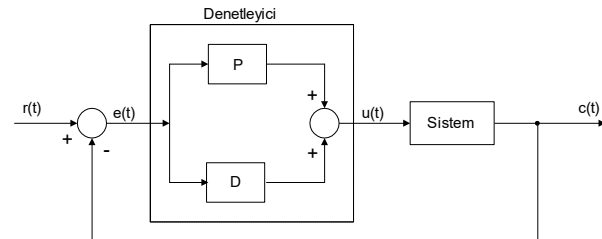
Burada K_p ; oransal kazancı, K_I ; integral kazancı ifade etmektedir.

2.3. PD Denetim (PD Control)

Oransal denetimde oluşan kararlı durum hatası PD denetim ile de kaldırılmaya çalışılabilir. Ancak türevsel etkinin asıl fonksiyonu, sistem çıkışının referans değeri üzerindeki aşımı (overshoot) ve referans değeri altında kalması (undershoot) durumunu azaltmaktır. Bu iki olumsuz durum azalırken bir miktar kararlı durum hatası kalabilir. Şekil 3'te blok şema halinde PD denetimi görülmektedir. PD denetimde denetleyici çıkışı, zaman tanım bölgesinde denklem 3' teki gibi ifade edilebilir:

$$u(t) = K_p e(t) + K_D \frac{d}{dt} e(t) \quad (3)$$

Burada K_p ; oransal kazancı, K_D ; türev kazancı olarak ifade edilir.



Şekil 3. PD denetim sistemi blok şeması (Block diagram of PD control)

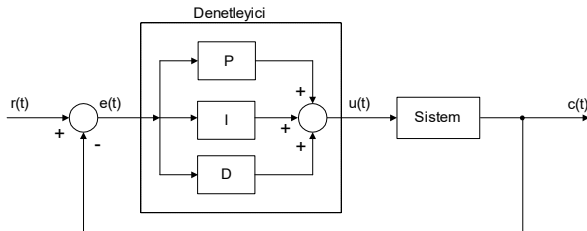
2.4. PID Denetim (PID Control)

Denetlenmesi güç, karmaşık ve P, PI, PD denetleyicilerin yeterli olmadığı sistemlerde PID denetim tercih edilmelidir. Oransal denetimde oluşan kararlı durum hatası PI denetim ile giderilir. Ancak, meydana gelen aşımalar, bu denetime türevsel etkinin de eklenmesiyle en aza indirilir veya tamamen kaldırılır.

Şekil 4'te PID denetimin blok şeması görülmektedir. PID denetleyicinin kapalı çevrimde nasıl çalıştığına bakılırsa; hata sinyali $e(t)$, PID denetleyiciye gönderilir ve denetleyici bu sinyalin hem türevini hem de integralini alır. Denetim çıkışının zaman tanım bölgesindeki ifadesi denklem 4' teki gibidir.

$$u(t) = K_p e(t) + K_i \int_0^t e(t) dt + K_d \frac{d}{dt} e(t) \quad (4)$$

Burada K_p ; oransal kazancı, K_i ; integral kazancı, K_d ; türev kazancı ifade etmektedir. Denetim sinyali $u(t)$ sisteme gönderilir ve yeni çıkış $c(t)$ elde edilir. Bu yeni çıkış $c(t)$ yeni hata sinyali $e(t)$ 'nin bulunması için toplayıcı bloğa geri gönderilir. Denetleyici yeni hata sinyalini alır ve tekrar integral ve türevini alır. Bu işlem sistem çıkışı referans değerine eşitleninceye kadar, yani hata sinyali $e(t)$ sıfır oluncaya kadar devam eder.



Şekil 4. PID denetim sistemi blok şeması (Block diagram of PID control)

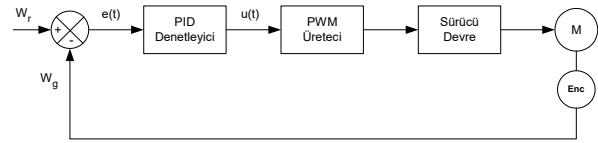
Son olarak, gerekmediği takdirde bu üç denetleyicinin (P, I, D) hepsinin aynı sistemde kullanılmasının şart olmadığı bilinmelidir. Örneğin, eğer PI denetleyici sistemin yeterince iyi cevap vermesini sağlıyorsa, sisteme türevsel denetleyicinin eklenmesinin gereği yoktur. Sistemi mümkün olduğunca basit tutmak gerekir [3].

2.5. PID Yöntemi İle DC Motorun Hız Kontrolü (Speed Control of DC Motor with PID)

PID denetim klasik denetleme yöntemlerinden birisi olup endüstriyel denetim süreçleri içerisinde yaygın olarak kullanılmaktadır. Bunun nedeni PID denetleyicinin hemen hemen tüm denetim sistemlerine uygulanabilmesidir. Sistemin istenilen set değerine

gelmesi ve set değerinin değişimlerinde onu takip etmesi istenir. İyi bir denetim performansı için de bu işlemin en kısa zamanda ve az hata ile yapılması gerekir. Hatanın anlamı set değeri ile gerçek değer arasındaki farktır. PID denetleyici, bu hataya ve hatanın değişim hızına bağlı olarak çıkış verir.

Şekil 5'te kapalı döngü DC motor hız kontrol sisteminin blok diyagramı verilmiştir. Kapalı döngü sistemle motor hız kontrolünün amacı, çıkış hızını istenilen referans hız değerine oturtmaktır [4].



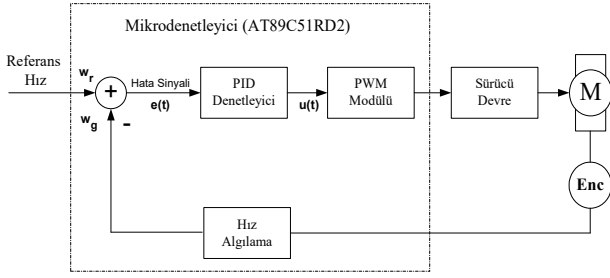
Şekil 5. PID denetleyicili DC motor hız kontrol blok diyagramı (Block diagram of DC motor's speed control with PID)

Burada referans hız (W_r), enkoder yardımıyla ölçülen motor hızı (W_g) ile karşılaştırılarak hata sinyali $e(t)$ üretilmektedir. PID denetleyicinin oluşturduğu $u(t)$ değeri motor sürücü devreye uygulanacak PWM sinyalinin değerini belirler. PWM sinyallerine bağlı olarak sürücü devredeki güç transistörleri yardımıyla motora gerilim uygulanmaktadır.

Şekil 5'te PID denetleyicisinin görevi referans hız ile gerçek hız arasındaki hatayı en aza indirmektir. Hata sinyali PID denetleyiciye girdiğinde oransal kazanç (K_p), integrali alınarak integral etki kazancı (K_i) ve türevi alınarak türevsel kazanç (K_d) ile çarpılmaktadır. Oransal-integral-türevsel (PID) tipi denetim, oransal, integral ve türevsel denetim etkilerinin birleştirilmesinden meydana gelir.

PID denetleyicisi kapalı döngü bir sistemle birleştirildiğinde, hata sıfır olana kadar kontrol etkisinin değişimine neden olacaktır. PID denetiminin en büyük üstünlüğü sistemde kalıcı durum hatasının olmaması, kolay uygulanabilirliği ve katsayılar doğru seçildiğinde hızlı tepki verebilmesidir [4].

Şekil 6'da AT89C51RD2 mikrodenetleyicisinin PID denetleyici olarak kullanıldığı kapalı döngü DC motor hız kontrol sisteminin blok diyagramı verilmiştir. Kapalı döngü sistemle motor hız kontrolünün amacı, çıkış büyüklüğünü istenilen referans hız değerine oturtmaktır [4].



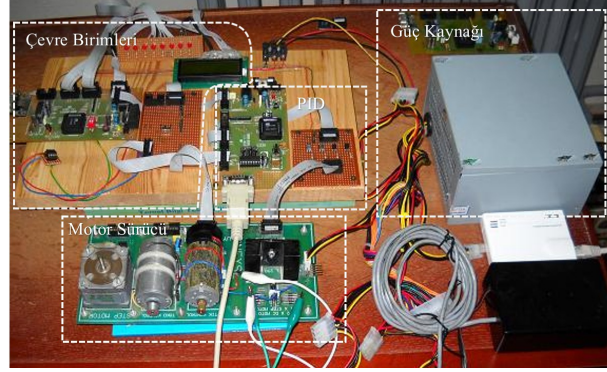
Şekil 6. AT89C51RD2 mikrodenetleyicisi tarafından yürütülen PID denetim algoritması ile motor hız kontrolünün blok diyagramı (Block diagram of DC motor's speed control via AT89C51RD2 microcontroller with PID algorithm)

Motor kontrol kartındaki AT89C51RD2 mikrodenetleyicisi DC motorun hız kontrolünde PID denetleyicisi olarak kullanılmıştır. Bu işlem mikrodenetleyicilerde PID komut ya da program parçasıyla yapılabilir. Hazırlanan PID denetim algoritması AT89C51RD2 mikrodenetleyicisi tarafından yürütülmüş ve motorun hız kontrolü gerçekleştirilmiştir. Uygulamada denetleyici parametrelerindeki değişikliklerin DC motorun hız denetimi üzerindeki etkileri görülmek istendiğinden bu amaçla hazırlanan PID denetimli programın ihtiyaç duyduğu denetleyici parametreleri (K_p , K_i , K_d), referans hız bilgisi ve PID çevrim süresi seri porttan alınmaktadır.

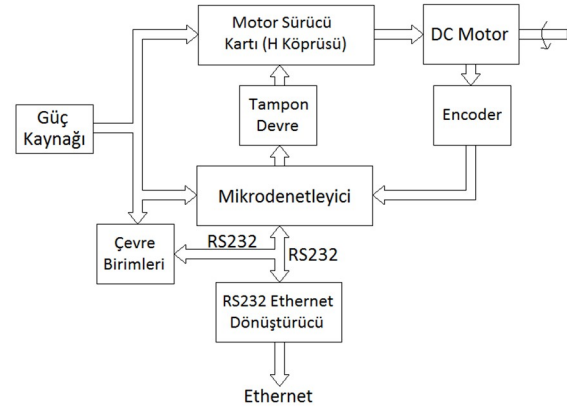
3. DENEY SETİ TASARIMI (THE DESIGN OF EXPERIMENT SETUP)

Tasarlanan deney seti Şekil 7'de de görülebileceği gibi 4 ana kısımdan oluşmaktadır: Mikrodenetleyici tabanlı PID kartı, motor sürücü kartı, çevre birimleri kartı ve güç kaynağı.

Gerçekleştirilen deney setinin genel mimarisi Şekil 8'de verilmiştir. Mikrodenetleyici tabanlı PID kartı, seri porttan alınan referans parametrelerin kullanılarak motor hız denetiminin yapıldığı ve sonuçlarının yine seri porttan gönderildiği deney setinin asıl bölümüdür. Motor kontrol kartı donanımı, kontrolü yapılacak bir DC motor ve ona bağlanan optik enkoder ile bir motor sürücü devresi, tampon devre ve denetleyici olarak kullanılan bir mikrodenetleyiciden oluşmaktadır. Burada motor kontrol kartını oluşturan donanım birimlerinden bahsedilerek mikrodenetleyicinin PID denetleyicisi olarak kullanılması ve mikrodenetleyiciye yüklenen programın yapısı anlatılacak ve son olarak motor kontrol devresinin bağlantı şeması verilecektir.



Şekil 7. Geliştirilen deney setinin fotoğrafı (The picture of designed experiment setup)



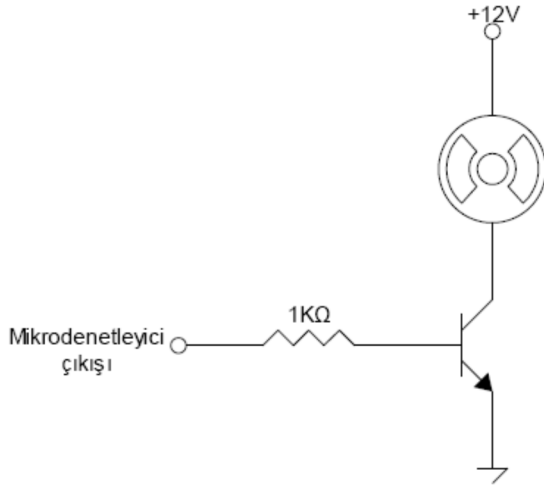
Şekil 8. Deneysel setinin mimarisi (The architecture of the designed experiment setup)

Bu çalışmada geniş bir yelpazeye sahip olması, hızlı ve güçlü olması, geniş bir donanım ve yazılım geliştirme araçları desteğinin sunulmuş olması, Dünyada endüstri standardı tanımlanmasına sahip 8-bitlik mikrodenetleyici olması, fiyatının düşük olması vb. sebeplerden dolayı 8051 mikrodenetleyicisi tercih edilmiştir [21].

DC motorlar, küçük boyutlara sahip olması ve maliyetlerinin düşük olması sebebiyle yaygın kullanım alanına sahiptirler [5]. Sıklıkla endüstriyel alanda ve robotik uygulamalarında kullanılmaktadırlar [6]. Geniş uygulama alanı bulmasının diğer bir sebebi de alternatif akım (AC) motorlarına göre kontrolünün daha kolay olmasıdır [7].

Motor doğrudan mikrodenetleyicinin çıkışına bağlandığı zaman mikrodenetleyicinin çıkışından motoru istenilen hızda çevirecek kadar akım alınamaz. Çıkışı +5V olarak kabul eden mikrodenetleyici gerekli akımı sağlamaya çalışacak ancak başarılı olamayınca kullanılamaz hale gelecektir. Aynı zamanda V_{maks} olarak +5V kullanmak yerine daha küçük ya da daha büyük voltaj ile motor

sürülmek istenebilir. Bu yüzden mikrodenetleyici ile motor arasında bağlantı kuracak bir elemana ihtiyaç olacaktır. Ara eleman olarak motor sürmek için hazırlanmış olan motor sürücü entegreleri kullanılabileceği gibi transistörlerin anahtarlama özelliğinden Şekil 9'da gösterildiği gibi de faydalanılabilir.

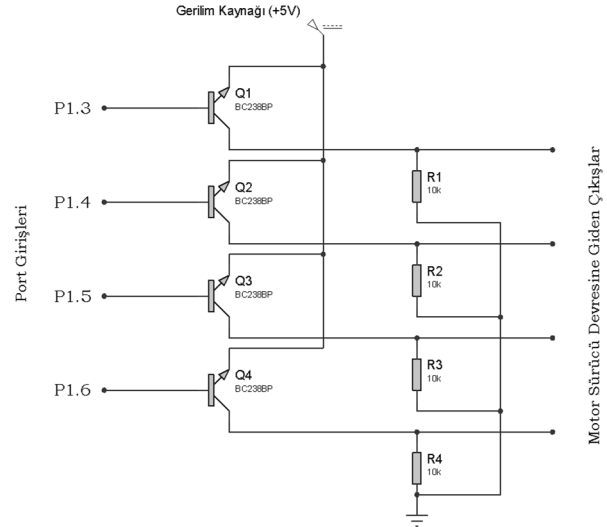


Şekil 9. Transistörlerin anahtarlama özelliğinden faydalanılarak motorun sürülmesi (Driving the motor by using the switching property of transistors)

DC motorun uçları bir gerilim kaynağına bağlandığında, motor bir yönde dönmeye başlar, DC motorun uçları gerilim kaynağına ters olarak bağlandığında ise motorun ters yönde hareket ettiği görülür. DC Motorun yön kontrolünü sağlayabilmek için H-Bridge (H-Köprüsü) denilen bir yöntem geliştirilmiştir. Yapısı gereği H harfine benzediğinden dolayı bu şekilde adlandırılır. H-Bridge genel olarak 4 adet transistör, diyot ya da mosfet ile gerçekleştirilen motorun iki yönlü dönebilmesini sağlayan bir yöntemdir. H-Bridge yöntemi için en çok kullanılan entegreler L293 ve L298 dir. Bu çalışmada L298 entegresi motorları daha yüksek akımla sürebildiği için bu entegre kullanılmıştır.

Mikrodenetleyici çıkışları motor sürücü entegresi için yeterli akımı sağlamadığından bu devrenin girişine transistörlü emiter izleyici tampon devresi BC238 transistörleri kullanılarak tasarlanmıştır.

Şekil 10' da tasarlanan tampon devrenin bağlantı şeması görülmektedir.



Şekil 10. Tampon Devresi (the buffer circuit)

4. HABERLEŞME VE ARAYÜZ (COMMUNICATION AND INTERFACE)

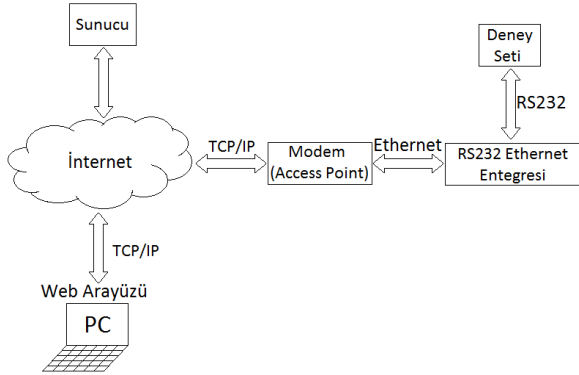
4.1. Kullanıcı – Deney Seti Haberleşme Yapısı (User-Experiment Setup Communication Structure)

Deney düzeneklerinin internet üzerinden uzaktan erişime açılmasında kullanılan yöntemlerden birisi, deney düzeneklerinin web sunucusu olarak çalışacak bir bilgisayarın seri ve/veya paralel portuna bağlanmasıdır. Deney düzeneklerinin yanında düzenekle birlikte sürekli çalışacak bir bilgisayar ihtiyacının olması, deney düzeneklerinin taşınmasının neredeyse imkansız olması ve deney düzeneklerinin kablolar yardımıyla bilgisayara bağlı olması bu yöntemin olumsuz yönlerinden bir kaçıdır. Bu olumsuzluklar birlikte düşünüldüğünde, farklı bir yöntemin kullanılması ihtiyacı ortaya çıkmaktadır.

Bu çalışmada kullanıcı-deney seti haberleşmesi için hazırlanacak sistemden beklenen, kullanıcıların internet bağlantısı olan herhangi bir bilgisayardan, herhangi bir zamanda deney setine bağlanarak deneylerini gerçekleştirebilmelerine ve deney düzeneklerinin kablosuz olarak taşınabilmesine imkan tanınması, ayrıca deney setinin yanında ya da yakınında set ile birlikte sürekli çalışacak bir bilgisayara ihtiyaç bırakmamasıdır.

Kullanılan haberleşme sistemi sayesinde deney seti uzak sunucudaki web sayfaları aracılığıyla kullanıcı erişimine açılabilmiştir. Kullanıcılar internet bağlantısı olan herhangi bir bilgisayardan uzak sunucudaki web sayfalarına erişerek, bu sayfalar üzerinden deney için gerekli parametre ve referans değerler (K_p , K_i , K_d , motor hızı, denetim süresi) ile deney seti çevre birimlerinin kontrolü için gereken komut ve/veya verileri (sıcaklığı

ölç, mesaj yazdır, ışıkları yak, mesaj bilgisi) deney setine gönderebilmekte ve setin gönderdiği sonuçları yine web sayfası üzerinden görebilmektedir.



Şekil 11. Haberleşme mimarisi (The communication architecture)

Tasarlanan deney düzeneğinin haberleşme mimarisi Şekil 11’de verildiği gibidir. Gerçekleştirilen haberleşmede deney düzeneği, ortamda bulunan bir kablosuz ADSL modem aracılığıyla TCP/IP üzerinden web sayfalarını barındıran uzak sunucuya haberleşmektedir. Bu sayede deney setinin yanında veya yakınında, deney seti ile birlikte çalışacak bir bilgisayar ihtiyacı ortadan kaldırılmıştır. Deney setinin kablosuz modemle haberleşebilmesi için deney setinin seri port çıkışı seri-ethernet çevirici entegresine bağlanmış daha sonra entegrenin ethernet tarafı bir erişim noktasına (Access point) bağlanarak kablosuz olarak modemle iletişime geçilmiştir. Bu şekilde, kullanıcıların internet bağlantısı olan herhangi bir bilgisayar yardımı ile uzak sunucudaki web sayfaları üzerinden sadece seri port girişinden haberleşebilen deney seti ile iletişime geçmeleri mümkün olmaktadır.

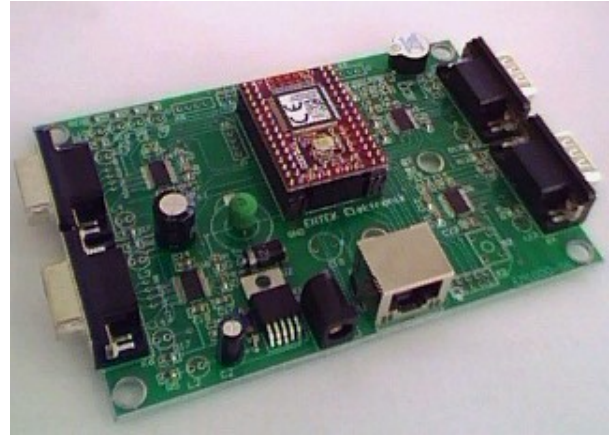
Deney seti tarafında kullanılan ADSL modem ile uzak sunucu arasındaki haberleşme ise sunucu bilgisayarda çalışan web yazılımı aracılığıyla TCP/IP protokolü kullanılarak, modem ve sunucu arasında internet bağlantısının kurulması ile gerçekleşir. Bu haberleşme sırasında sunucuda çalışan web yazılımı ile veriler modeme kadar iletilirken deney setinden modeme kadar ulaşan sonuç verileri de yine web yazılımı aracılığıyla TCP/IP üzerinden sunucuya çekilmekte ve sonuçlar web sayfası üzerinde gösterilmektedir.

4.2. Seri – Ethernet Çevirici Entegresi ve Konfigürasyonu (Serial-Ethernet Converter IC and Its Configuration)

Seri – Ethernet çeviriciler her hangi bir yerden, internet aracılığıyla seri haberleşmeli cihazları izlemek ve kontrol etmek için kolay ve ekonomik bir yol sağlar. Seri-Ethernet çevirici, herhangi bir seri haberleşmeli cihazı, kullanıcı yazılımında hiç bir değişiklik yapılmasına

gerek kalmadan dünya çapındaki ağa (internet) kolayca bağlanabilir [8]. Cihazların dönüştürücü sayesinde ağa bağlanabiliyor olması mesafe sorununu ortadan kaldırmaktadır. Ayrıca, RS232'nin kablo mesafe kısıtlaması da ethernet ağına erişim sağlanarak ortadan kaldırılmaktadır. Bu şekilde seri haberleşmeli cihazların tüm parametreleri cihazın yanına gidilmeden, internet yoluyla denetlenebilir.

Bu çalışmada, Entek Elektronik & Yazılım firmasının, temsilcisi olduğu Tayvan uyruklu Tibbo firmasının ürünlerini kullanarak otomasyon sektörü için tasarladığı, EDS serisinin 1000-GB (4 Port) modeli bir seri-ethernet (TCP/IP) çevirici entegresi kullanılmıştır.



Şekil 12. Seri – Ethernet çevirici entegresi (serial-ethernet converter IC)

Çalışmada kullanılan seri-ethernet çeviricisine ait temel özellikler şu şekilde verilebilir.

- RS232 protokolü ile haberleşebilen her türlü cihazın ethernet ağına bağlanmasını mümkün kılar.
- Seri port ile ethernet arasında tamamen şeffaf olarak veri akışına izin verir.
- Cihaz (seri-ethernet çevirici) sadece bir PC ile değil, aynı zamanda başka bir cihaz (seri-ethernet çevirici) ile de ethernet ağı üzerinden haberleşebilir.
- Seri ve Ethernet ayarları temin edilen bir program (DS Manager) vasıtası ile yapılabilir.
- IP adresi ve diğer pek çok parametre ağ üzerinde değiştirilebilir.
- Bütün değişiklikler cihaz çalışırken ve bağlantı kesilmeden yapılabilir.

Seri ethernet çevirici - modem ve seri ethernet çevirici – deney seti arasında haberleşmenin gerçekleşebilmesi için seri-çevirici cihazı önce konfigüre edilmiştir. Cihazın konfigürasyonunu yapmak için Tibbo Device Server Toolkit yazılımı cihazın çalışacağı yerel ağdaki bir bilgisayara kurulmuştur.

Aşağıda Tibbo DS Manager programıyla çalışmada kullanılan seri-ethernet çevirici cihazın genel olarak konfigürasyonunun yapıldığı pencere görülmektedir.

Çalışmada cihazın sahip olduğu dört adet seri (RS232) kanaldan iki tanesi kullanılmıştır. Bu kanallardan ilki (Channel 1) mikrodenetleyici tabanlı PID kartı ile haberleşmek için, ikincisi ise (Channel 2) çevre birimleri kartı ile haberleşmek için kullanılmıştır. Seri-Ethernet çevirici entegresi ile deney setini oluşturan kartlar arasında sağlıklı bir haberleşme gerçekleşmesi için haberleşmede kullanılan iki seri kanalında konfigürasyonunun yapılması gerekmektedir.

Lorem ipsum dolor sit amet, consetetur sadipscing elitr, sed diam nonumy eirmod tempor invidunt ut labore et dolore magna aliquyam erat, sed diam voluptua. At vero eos et accusam et justo duo dolores et ea rebum. Stet clita kasd gubergren, no sea takimata sanctus est Lorem ipsum dolor sit amet, consetetur sadipscing elitr, sed diam nonumy eirmod tempor invidunt ut labore et dolore magna.

4.3. Web arayüzü (Web Interface)

Geliştirilen çalışmada asıl hedef, kullanıcıların web sayfaları üzerinden hazırlanan deney setine ve onun çevre birimlerine erişerek deneylerini gerçekleştirebilmesidir. Amaca yönelik web sayfaları hazırlanarak, kullanıcılara internet bağlantısı olan herhangi bir bilgisayar ile bu sayfalar üzerinden deneylerini gerçekleştirme ve deneyden elde edilen sonuçları bilgisayar ekranında görme imkanı sunulmuştur. Bu sayede kullanıcılar günün her saatinde hazırlanan web sayfalarını ziyaret edilebilmekte ve deneylerini gerçekleştirebilmektedir.

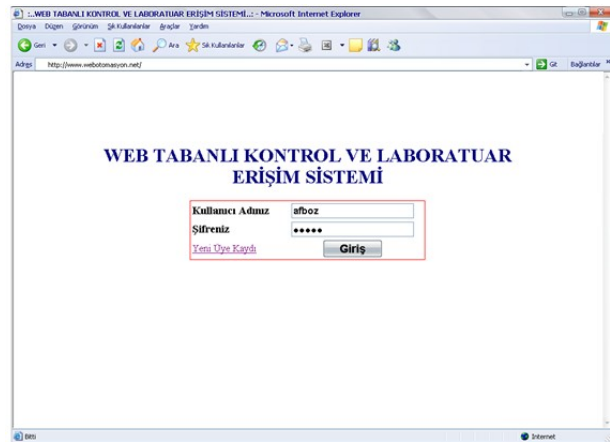
Deney seti ile haberleşme ve çevre birimlerin kontrolünde bütün işlemler sunucudaki web yazılımı aracılığıyla gerçekleşeceğinden kullanıcı bilgisayarında internet erişiminin olması ve internet sayfalarında gezinti yapmaya olanak sağlayan bir tarayıcı programın yüklü olması yeterlidir.

Kullanıcılar uzak sunucudaki web sayfalarına eriştiklerinde, bu sayfalar üzerinden deney için gerekli parametre ve referans değerler (K_p , K_i , K_d , motor hızı, denetim süresi) ile deney seti çevre birimlerinin kontrolü için gereken komut ve/veya verileri (sıcaklığı ölç, mesaj yazdır, ışıkları yak, mesaj bilgisi) deney setine gönderebilecek ve setin gönderdiği sonuçları bilgisayar ekranında görebilecektir.

Deney seti ile haberleşmede ve çevre birimlerin kontrolünde kullanılan etkileşimli web sayfaları "Microsoft Visual Studio" ortamında geliştirilmiştir.

Web arayüzünün ve ve internet tarafındaki diğer sayfaların hazırlanmasında HTML ile birlikte ASP.NET teknolojisi ve Ajax tekniği kullanılmıştır. Burada, web uygulamaları geliştirme kavramına getirdiği yeni yaklaşım, yöntem, programlama teknikleri ve sunucu ile deney seti tarafındaki modem arasında bağlantı kurulabilmesini sağlayan gelişmiş bileşen sınıflarına sahip olması nedeniyle ASP.NET teknolojisi tercih edilmiştir. Programlama dili olarak ise güçlü, modern ve nesne tabanlı bir dil olan "C#" tercih edilmiştir. Sunucu bilgisayar tarafındaki veritabanı işlemleri için ise Microsoft Office Access kullanılmıştır.

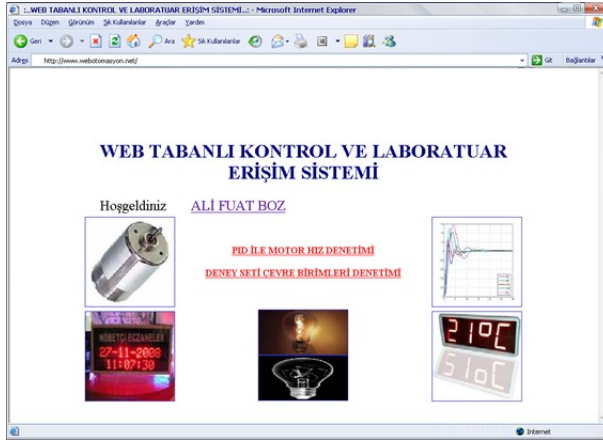
İlk arayüz giriş sayfası olarak Şekil 13'teki gibi tasarlanmıştır.



Şekil 13. Giriş sayfası (The home page)

Burada sadece yetkili kişilerin girişine olanak sağlamak için ilk kullanıcı adı ve şifre bilgileri istenmektedir. Doğru kullanıcı adı ve şifresiyle giriş yapan kullanıcıya, PID ile motor hız denetimini ve deney seti çevre birimlerinin kontrolünü gerçekleştirebilmesi için hazırlanan iki farklı arayüz seçeneği sunulmaktadır. Şekil 14'te giriş sayfasının doğru kullanıcı adı ve şifresi ile giriş yapıldıktan sonra alacağı görünüm verilmiştir. Yine giriş sayfası üzerinden siteye üye olmayan kullanıcılar üye kaydı yapabilmektedir.

Doğru kullanıcı adı ve şifresiyle giriş yapan kullanıcı bu sayfa üzerinden istediği arayüze geçiş yapabilecektir. Buradan "PID İLE MOTOR HIZ DENETİMİ" seçeneğini seçen kullanıcının karşısına Şekil 15'teki, "DENEY SETİ ÇEVRE BİRİMLERİ DENETİMİ" seçeneğini seçen kullanıcının karşısına ise Şekil 17'deki web sayfası gelmektedir.

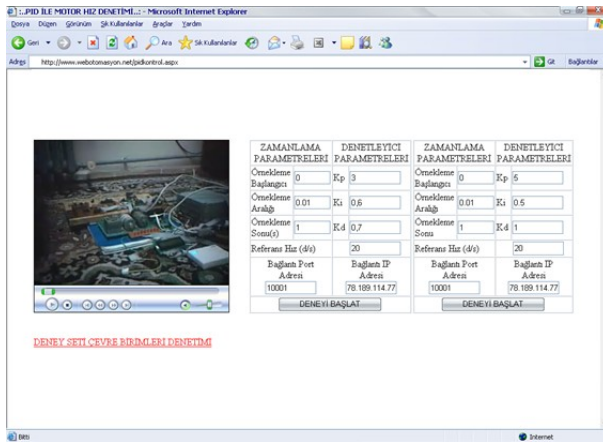


Şekil 14. Giriş sayfasının kullanıcı girişinden sonra alacağı görünüm (The screenshot of home page after sign-in)

Bu çalışmada sadece deney seti tasarımı konu olarak ele alınmıştır. Laboratuvar oluşturulmak istenildiğinde deney setlerinin donanımsal maliyeti düşük olduğundan öncelikle mevcut deney seti sayısı artırılarak çok sayıda kişinin aynı anda çalışabilmesi sağlanacaktır. Eğer tüm deney setleri kullanılıyor ise yeni bir kullanıcının sisteme girişi engellenecek daha sonra giriş yapması için bir uyarı mesajı verilecektir.

4.3.1. Motor hız denetimi arayüzü (The interface of motor speed control)

DC motorun PID ile hız denetiminin yapıldığı çalışmada, hazırlanan arayüz ile kullanıcılara değişik parametreler altında denetim performanslarını gerçek ortamda inceleme imkanı sunulmuştur.



Şekil 15. Motor hız denetimi parametre giriş sayfası (Web page for entering motor speed control parameters)

Kullanıcı hazırlanan arayüz üzerinden, tabloda "DENETLEYİCİ PARAMETRELERİ" bölümündeki ilgili metin kutularını kullanarak hız denetleyici

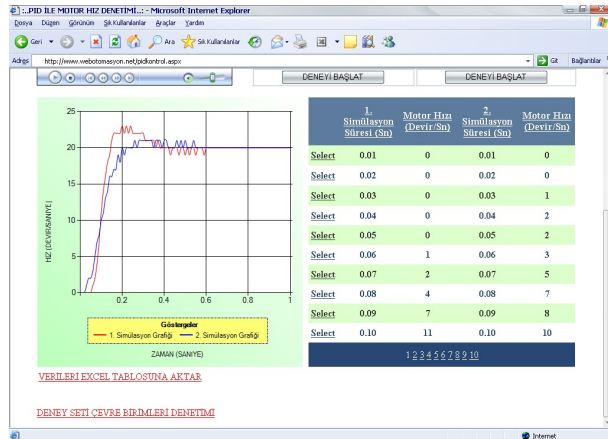
parametreleri K_p , K_i , ve K_d 'yi değiştirebilmekte ve referans hız bölümünden saniyedeki devir sayısı cinsinden referans hız değerini belirleyebilmektedir. "ZAMANLAMA PARAMETRELERİ" bölümünde ise denetleyici için örnekleme süresi başlangıcı sıfır (0) olarak, örnekleme adımı ise saniye türünden "0.01" olarak sabitlenmiştir. Bu sabitlemenin sebebi PID denetleyici olarak kullanılacak mikrodenetleyicideki yazılımda daha önce denetim tarama frekansının 100/s olarak belirlenmesi ve her deney başlangıcında zamanlama parametrelerinin sıfırlanmasıdır. Yine "Örnekleme Sonu" alanındaki metin kutusu aracılığıyla saniye türünden deney süresi belirlenebilmektedir. Kullanıcının farklı denetleyici parametreleri (K_p , K_i , K_d) ile gerçekleştirilecek denetimlerin performanslarını karşılaştırmalı analiz yapabilmesi için web sayfasında iki farklı deney için denetleyici ve zamanlama parametre girişleri bir arada bulunmaktadır.

Parametre girişlerinin yapıldığı tabloların en altındaki "Bağlantı Port Adresi" ve "Bağlantı IP Adresi" bölümü, deney setinin internet üzerindeki yerinin yazılıp bağlantı kurulmasını sağlamaktadır. "Port" etiketinin yanındaki ilgili metin kutusuna, deney seti tarafındaki modemle bağlantı kurulup, veri akışının sağlanacağı port adresi yazılmalıdır. Bu portun aynı zamanda sunucudan gelecek veri paketlerinin deney setindeki motor kontrol kartına ulaşması için yönlendirilmesi yapılan "10001" adresli ağ portu olması gerekmektedir; aksi takdirde deney parametreleri motor kontrol kartına ulaşmayacaktır. "Bağlantı IP Adresi" etiketinin yanındaki ilgili metin kutusuna ise, deney seti tarafındaki modemden geniş ağ (WAN) IP adresi yazılmalıdır. Bu çalışmada deney seti tarafındaki modemden geniş ağ IP adresi "78.189.114.77" olduğundan "Bağlantı IP Adresi" alanına bu IP adresi yazılmıştır. Eğer yerel bir ağda kontrol yapılacaksa bu kısma deney setinin bağlı olduğu seri-ethernet çevirici entegresinin yerel IP adresi yazılmalıdır.

Değerler doğru bir şekilde yazılıp "Deneyi Başlat" butonuna basıldığında sunucuda çalışacak program aracılığıyla, TCP/IP protokolü kullanılarak, internet üzerinden "Bağlantı IP Adresi:Bağlantı Port Adresi" bilgisi ile deney seti tarafındaki modemle bağlantı kurulacak ve deney için girilen parametre değerlerini içeren veri paketleri motor kontrol kartına ulaştırılmak üzere modeme gönderilecektir. Modeme gönderilen veri paketine önce mikrodenetleyicide çalışan yazılımın kullandığı başlangıç karakteri sonra parametre değerleri ve son olarak da bitiş karakteri eklenmiştir. Modemle bağlantının kurulması ile veri paketlerinin modeme gönderilmesi ve modemden alınması işlemlerinde ASP.NET'in gelişmiş bileşen (component) sınıflarından yararlanılmıştır. Sunucu ve modem arasında bağlantı kurulabilmesini sağlayan ASP.NET componentleri, TCP/IP protokol kümesi özelliklerini taşıyan

“TcpClient” ve “NetworkStream” bileşenleridir. Bu sınıflar “System.Net.Sockets” isim alanı (namespace) altında yer almaktadır.

Deneysel parametrelerinin seri-ethernet çevirici entegresi üzerinden motor kontrol kartına ulaşması ile birlikte deneysel başlayacaktır. Deneysel süresinin sonunda ise motor hız denetimi sona erecek ve motor duracaktır. Deneysel bitmesinin ardından Deneysel süresince saniyede yüz defa hesaplanan anlık hız değerleri motor kontrol kartından seri-ethernet çevirici entegresi aracılığıyla modeme “10001” portundan ulaşacaktır. Sunucuda çalışan program internet üzerinden modemle kurulan bağlantı kapatılmadan deneysel süresince uykuda beklemektedir. Deneysel süresinin sonunda ise başlangıçta kurulan aynı bağlantı ve akış üzerinden modeme ulaşan veriler okunarak hız dizisine değişken olarak kaydedilmektedir. Verilerin okunması işlemi tamamlandıktan sonra akış sonlandırılıp kurulan bağlantı kapatılmaktadır. “Deneysel Başlat” isimli buton ile çalışan program sonlandığında oluşan çıkış sayfasının parametre giriş alanlarının altındaki bölümü Şekil 16’da verilmiştir.



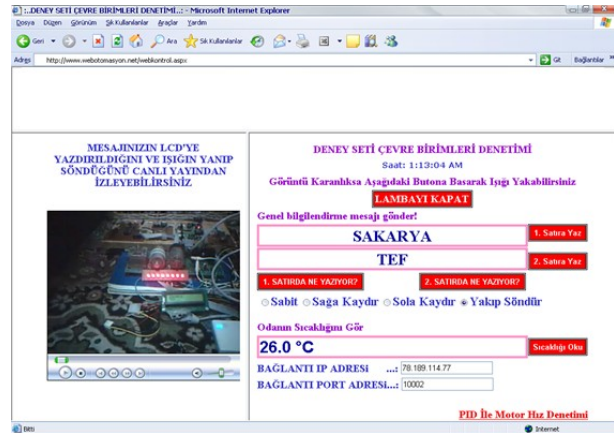
Şekil 16. PID ile motor hız denetimi çıkış sayfası (Web page for result of the motor speed control with PID)

Deneysel seti tarafındaki modemle internet üzerinden kurulan bağlantının kapatılmasının ardından okunan anlık hız verileri sunucudaki Access veri tabanı dosyasına kaydedilmekte ya da daha önceki deneysel sonuçları kayıtlı ise dosya yeni hız verileri ile güncellenmektedir. Bu işlemin ardından kaynağını veri tabanındaki anlık hız verilerinden alan grafik alanı, Ajax tekniği kullanılarak sayfanın tamamının yenilenmesine gerek kalmadan güncellenmektedir. Web arayüzündeki grafik için .Net Framework 3.5 SP1 ile birlikte gelen Chart nesnesi kullanılmıştır. İki farklı deneysel için hazırlanan parametre giriş alanları kullanılarak gerçekleştirilen deneylerin sonucundaki motor hız değişimleri aynı grafik alanı üzerine çizdirilerek kullanıcının sonuçları yorumlaması kolaylaştırılmıştır.

Kullanıcının anlık sayısal hız değerlerini arayüz üzerinden görebilmesi için sonuçlar kullanıcıya bir de tablo halinde sunulmuştur. “VERİLERİ EXCEL TABLOSUNA AKTAR” seçeneği ile sunucu bilgisayarda deneysel çalışmanın sonucunda elde edilen anlık hız verilerinden oluşan bir Excel tablosu hazırlanır. Kullanıcı isterse bu dosyayı kendi bilgisayarına kaydedebilmektedir. Ayrıca kullanıcılar deneysel setini görüntüleyen bir web kamera yardımıyla deneysel setinin çalışmasını arayüzden izleyebilmektedir. Böylece, geliştirilen uygulamanın görsel olarak da etkileşimli bir yapıya sahip olması sağlanmıştır.

4.3.2. Çevre birimleri denetimi (Interface for controlling peripheral units)

Deneysel seti çevre birimlerinin kontrolü için hazırlanan web arayüzü Şekil 17’de görülmektedir. Kullanıcı arayüz üzerinden çevre birimlerin kontrolü için gereken komut (sıcaklığı ölç, mesaj yazdır, ışıkları yak) ve/veya verileri (mesaj bilgisi) deneysel setine gönderebilmekte ve setin gönderdiği sonuçları yine web sayfası üzerinden görebilmektedir.



Şekil 17. Çevre birimleri denetimi sayfası (Web page for peripheral units control)

Hazırlanan arayüz ile kullanıcıya deneysel setinin bulunduğu ortamın sıcaklığını görebilme, deneysel seti üzerindeki LCD’ye mesajını yazdırabilme ve deneysel ortamının karanlık olma ihtimaline karşılık deneysel setini aydınlatabilme imkanı verilmiştir.

Kontrol işlemlerinin yapıldığı bölümün altındaki “Bağlantı Port Adresi” ve “Bağlantı IP Adresi” bölümü, deneysel setinin internet üzerindeki yerinin yazılıp bağlantı kurulmasını sağlamaktadır. “Port” etiketinin yanındaki ilgili metin kutusuna, deneysel seti tarafındaki modemle bağlantı kurulup, veri akışının sağlanacağı port adresi yazılmalıdır. Bu portun aynı zamanda sunucudan gelecek veri paketlerinin deneysel setindeki çevre birimler kartına ulaşması için yönlendirilmesi yapılan “10002” adresli ağ

portu olması gerekmektedir; aksi takdirde kontrol verileri çevre birimler kartına ulaşmayacaktır. “Bağlantı IP Adresi” etiketinin yanındaki ilgili metin kutusuna ise, deney seti tarafındaki modem geniş ağ IP adresi olan “78.189.114.77” IP adresi yazılmıştır. Eğer yerel ağda kontrol yapılacaksa bu kısma deney setinin bağlı olduğu seri-ethernet çevirici entegresinin yerel IP adresi yazılmalıdır.

Arayüz ile çevre birimler kartında yapılabilecek bütün kontrol işlemlerinde öncelikle deney seti tarafındaki modemle TCP/IP protokolü kullanılarak internet üzerinden bir bağlantı sağlanacak ardından kontrol verilerini içeren veri paketleri modeme iletilecektir.

Arayüzdeki “LAMBAYI KAPAT” ya da ledler yakılmamış iken “LAMBAYI YAK” şeklinde gözükecek butona basıldığında çevre birimler kartına (sunucudan modeme), karttaki mikrodenetleyicide çalışan yazılımın kullandığı bir baytlık başlangıç karakteri ile lambayı yak ya da kapat anlamına gelen komut ve bir baytlık bitiş karakteri gönderilir. LCD’ye yazdırılan mesaja verilebilecek efekt seçeneklerinin işlenmesinde de benzer işlem adımları gerçekleştirilerek verilmek istenen efekt yazıya uygulanmaktadır.

“Sıcaklığı Oku” butonuna basıldığında ise yine bir baytlık başlangıç karakteri, ardından sıcaklığı ölç komut karşılığı ve son olarak bir baytlık bitiş karakteri çevre birimler kartına ulaştırılır ve hemen ardından internet üzerinden modemle kurulan aynı bağlantı kapatılmadan çevre birimler kartının göndermiş olduğu ve seri-ethernet çevirici entegresinden modeme ulaşan sıcaklık bilgisi okunarak metin kutusunda gösterilmektedir. “1. SATIRDA NE YAZIYOR” VE “2. SATIRDA NE YAZIYOR” butonları da benzer işlem adımlarıyla o anda LCD’de yazan mesaj bilgisini kullanıcıya göstermektedir.

Mesajını Deney setindeki LCD’ye yazdırmak isteyen kullanıcı, “1. Satıra Yaz” ya da “2. Satıra Yaz” butonuna bastığında bir baytlık başlangıç ve bitiş karakterlerinin dışında 1. satıra yaz veya 2. satıra yaz komut karşılığı ile birlikte diğer butonlardan farklı olarak mesaj bilgisini de içeren veri paketleri yine çevre birimleri kartına iletilmek üzere kurulan internet bağlantısı üzerinden modeme gönderilmektedir.

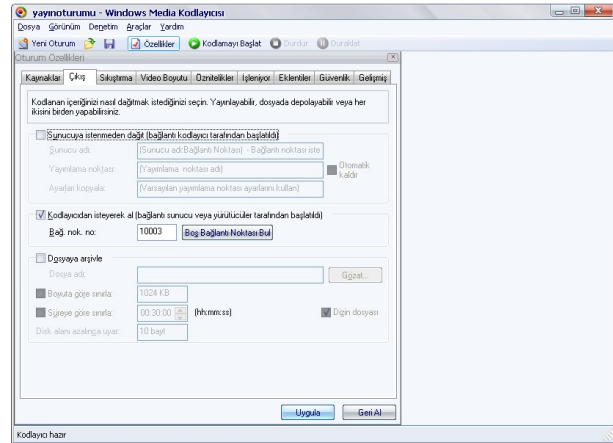
4.3.3. Görüntü aktarma yazılımı (Image transfer software)

Bir olayı canlı olarak yayınlamak için görüntü kodlayıcı yazılımlar kullanılır. Bu çalışmada, Windows media kodlayıcısı ortam görüntüsünün web sayfasında yayınlanması için kullanılmıştır (Şekil 18). Kodlayıcı

yazılım görüntü verilerini bir media sunucusu üzerinden web sayfasına aktarır.

Windows media kodlayıcısı basit bir arayüze sahiptir. Bir yayını gerçekleştirmek için yapılacak işlemler oldukça kolay bir şekilde gerçekleştirilir. Öncelikle yeni bir yayın oturumu oluşturulur. Bundan sonraki işlemler, yayın özellikleri ile kaynak ve çıkış noktalarının programa bildirilmesinden ibarettir.

Yayın ile ilgili ayarlar program penceresindeki özellikler sekmesine tıklanarak gerçekleştirilir. Açılan pencerede kaynaklar sekmesinde video veya sesin hangi kaynaktan alınacağı belirtilir. Çıkış sekmesinde görüntünün iletileceği sunucu ve yayımlama noktası belirtilir. Sıkıştırma sekmesinden, görüntünün kodlanma hızı ya da saniyedeki kare sayısı seçilir. Ayrıca bu pencerede video boyutu ve güvenlik ayarları ile bazı özel ayarlar gerçekleştirilir. Yayın oturumu ile ilgili ayarlar gerçekleştirildikten sonra kodlamayı başlat sekmesine tıklanarak görüntünün belirtilen sunucuya kodlanmasına başlanır. Gerçekleştirilen yayının istenildiğinde alınması, istenmeden sunucuya dağıtılması ve bazı istemcilerin yayına ulaşmasının engellenmesi, programın sağladığı seçenekler arasındadır.



Şekil 18. Windows media kodlayıcısı (Windows media coder)

Pencerede bağlantı noktası “10003” olarak belirlenmiştir. Belirlenen port bilgisiyle modeme gelen bağlantı istekleri, modemde yapılan port yönlendirme işlemi sayesinde kameranın bağlı olduğu bilgisayara yani “192.168.1.50” numaralı IP’nin “10003” portuna yönlendirilmiştir. Yapılan bu işlemle, “192.168.1.50” IP numaralı bilgisayarda çalışan ve “10003” portundan yayın yapan kodlayıcı yazılım ile iletişim sağlanmakta ve deney seti tarafındaki kamera görüntüsü web sayfası üzerinden izlenebilmektedir.

5. SONUÇLAR (CONCLUSION)

Bu çalışmada, hazırlanan bir deney düzeneğine, internet üzerinden erişime olanak sağlayan bir uygulama sunulmuştur. Deney düzeneği PID denetim ile DC motorun hız kontrolünü gerçekleştirmektedir. Burada PID denetleyici olarak piyasadan 8-10 \$'a satın alınabilen 8051 tabanlı AT89C51RD2 mikrodenetleyicisi kullanılarak maliyet düşürülmüştür.

Deney düzeneklerinin internet üzerinden uzaktan erişime açılmasında alışlagelmiş bir yöntem deney düzeneklerinin web sunucusu olarak da çalışacak bir bilgisayarın, seri ve/veya paralel portuna bağlanmasıdır. Bu çalışmada deney düzeneği ortamda bulunan bir kablosuz ADSL modeme bağlanarak TCP/IP üzerinden uzak sunucuyla haberleşmektedir. Bu sayede deney setinin yanında veya yakınında, deney seti ile birlikte çalışacak bilgisayar ihtiyacı ortadan kaldırılmıştır.

Deney setinin kablosuz modemle haberleşebilmesi için deney setinin seri port çıkışı seri-ethernet çevirici entegresine bağlanmış daha sonra da bir erişim noktası vasıtasıyla kablosuz olarak modemle iletişime geçilmiştir.

Çalışmanın önemli özelliklerinden birisi de, uygulama gerçekleştirme süresinin oldukça kısa olmasıdır. Deneysel çalışmada web sayfasından referans değerlerin deney setine gönderilmesini takiben, deney setinde; gelen değerlerin okunması, PID denetleyicisi ile motor hız denetiminin yapılması, sonuçların sunucu bilgisayara gönderilmesi ve sunucu bilgisayarda da değerlerin okunarak grafik ve tabloların çizdirilmesiyle sonuç ekranının görüntülenmesi işlemleri gerçekleşmektedir. Bu gerçekleştirilen işlemlerin süresi, motor hız denetiminin yapıldığı süre düşüldüğünde yaklaşık olarak 2-3 saniye aralığında olmaktadır. Geleneksel laboratuvar deneylerinde böyle bir deney düzeneğinin kurulması, deneylerin gerçekleştirilmesi ve sonuçlarının alınmasının yaklaşık olarak 2 saat gibi bir zaman aldığı düşünüldüğünde, 2-3 saniyelik bir gecikme kullanıcı açısından hiçbir problem oluşturmayacaktır.

Deney setinin öğrenciler açısından yararı düşünüldüğünde, bilgisayar destekli bu deney seti sayesinde öğrenciler simülasyon sonuçları ile yetinmeyip gerçek bir sistem üzerinden elde ettikleri sonuçlar ile daha iyi bir tecrübe edinmiş olacaklardır. Aynı zamanda gerçek ortam deneyleri ile daha kalıcı öğrenme sağlanacaktır.

KAYNAKÇA (REFERENCES)

[1] A. Tekin ve F. Ata, "PI Denetleyici İle Asenkron Motorların Hız Denetimi İçin İnternet Tabanlı Sanal Bir Laboratuvar Geliştirme", Fırat Univ.

Mühendislik Bilimleri Dergisi, cilt 21, no. 2, pp. 161-172, 2009.

[2] Ş. Demirbaş, "İnternet Tabanlı PI Kontrollü Bir Doğru Akım Motoru Deney Seti", Gazi Üniv. Müh. Mim. Fak. Der., cilt 22, no. 2, pp. 401-410, 2007.

[3] A. B. Dumanay, A. İstanbullu, M. Demirtaş, "Dc Motorun PID ile Hız Denetimi İçin Uzaktan Laboratuvar Uygulaması", Elektrik Elektronik Bilgisayar Biyomedikal Mühendislikleri Eğitimi IV. Ulusal Sempozyumu Bildiri Kitabı, 241-244, Osmangazi Üniversitesi, Eskişehir, 2009.

[4] S. Kaçar ve C. Bayılmış, "A Web-Based Educational Interface for an Analog Communication Course Based on MATLAB Builder NE With WebFigures", IEEE Transactions on Education, cilt 56, no. 3, pp. 346-354, 2013.

[5] C. Bayılmış, "Development of a Web-based educational interface using MATLAB builder NE with WebFigures for digital modulation techniques", Comput. Appl. Eng. Educ, cilt 20, pp. 604-610, 2012.

[6] B. Balamuralithara ve P. C. Woods, "Virtual laboratories in engineering education: The simulation lab and remote lab.", Comput. Appl. Eng. Educ, cilt 17, no. 1, p. 108-118, 2008.

[7] I. Colak, E. Irmak, E. Kabalci ve F. Issi, "Design and implementation of a remote laboratory platform using MATLAB builder for NE", Comput. Appl. Eng. Educ, 2011.

[8] J. Djordjevic, B. Nikolic ve A. Milenkovic, "Flexible Web-based educational system for teaching computer architecture and organization", IEEE Trans. Educ., cilt 48, no. 2, p. 264-273, 2008.

[9] S. Kacar, C. Bayılmış, I. Cankaya ve M. Cakiroglu, "Design of ASP.NET based WEB interface with MATLAB builder NE and MATLAB Web figure for wireless sensor networks", e-J. New World Sci. Acad., cilt 4, no. 4, pp. 360-370, 2009.

[10] V. Nerguizian, R. Mhiri, M. Saad, H. Kane, J. S. Deschênes ve H. Saliyah-Hassane, "Lab@home for analog electronic circuit laboratory", e-Learning in Industrial Electronics (ICELIE), Montreal, 2012.

[11] W. Xie, X. Yang ve F. Li, "A virtual laboratory platform and simulation software based on web", Control, Automation, Robotics and Vision ICARV, Hanoi, 2008.

[12] A. Maiti, A. Maiti ve C. K. Maiti, "Online technology CAD laboratory for microelectronics education", EUROCON - International Conference on Computer as a Tool (EUROCON), Lisbon, 2011.

- [13] A. Špernjak, “omputer-supported laboratory as an effective educational tool”, MIPRO, 2011 Proceedings of the 34th International Convention, Opatija, 2011.
- [14] I. Sanogo, S. Ouya, A. Dahirou ve C. Lishou, “Proposal of cloud-based online laboratory model for practical training in the telecoms and networking fields”, IEEE Global Engineering Education Conference (EDUCON), Abu Dhabi, 2016.
- [15] Y. Linn, “An ultra-low cost wireless communications systems laboratory for education and research”, IEEE Trans. Educ., cilt 55, no. 2, pp. 169-179, 2012.
- [16] M. Stefanovic, M. Matijevic, V. Cvijetkovic ve V. Simic, “Web-based laboratory for engineering education”, Comput. Appl. Eng. Educ., cilt 18, pp. 526-536, 2010.
- [17] S. C. Sivakumar, W. Robertson, M. Artimy ve N. Aslam, “A web-based remote interactive laboratory for Internetworking education”, IEEE Transactions on Education, cilt 48, no. 4, pp. 586-598, 2005.
- [18] R. Baharudin, “Delivering a frequency response laboratory in a web-based learning environment”, Computational Science and Technology (ICCST), Kota Kinabalu, 2014.
- [19] D. Bonatti, G. Pasini, L. Peretto, E. Pivello ve R. Tinarelli, “A Novel Approach for Laboratory Activities in E-Learning Courses”, IEEE Instrumentation & Measurement Technology Conference IMTC, Warsaw, 2007.
- [20] Y. Sari, “Standart Formlar İle Optimal Kontrolör Tasarımı”, Doktora Tezi, Sakarya Üniversitesi, Fen Bilimleri Enstitüsü, Sakarya, 2005.
- [21] A. Özcerit, M. Çakıroğlu ve C. Bayılmış, 8051 Mikrodenetleyici Uygulamaları, İstanbul: Papatya Yayıncılık Eğitim, 2005.
- [22] İstanbul Teknik Üniversitesi IEEE Öğrenci Kolu., “DC Motor”.
- [23] İ. COŞKUN ve H. TERZİOĞLU, “Gerçek Zamanda Değişken Parametrelili PID Hız Kontrolü”, 5. Uluslararası İleri Teknolojiler Sempozyumu (IATS’09), Karabük, 2009.
- [24] “
http://www.dghcorp.com/datasheets/dgh_a3k.pdf
“ Son Erişim 16.04.2016.

Multi-repairmen problem for disaster recovery of optical networks

Ferhat Dikbıyık¹

24.06.2016 Geliş/Received, 30.11.2016 Kabul/Accepted

doi: 10.16984/saufenbilder.82562

ABSTRACT

Survivability of optical networks is a growing concern because of strong reliance on the internet to accomplish our daily activities. This high reliance makes telecommunication infrastructure vital to our daily life. Accordingly, when large network failures occur due to a disaster, the whole community (network operator and end users) incur grave consequences. Hence, the quest of telecommunication infrastructures recovery after a disaster is indispensable. Even though some research focus on how to avoid such large scale failures, sometimes it is inevitable and fast recovery is required. In this study, we investigate the problem of multi-repairmen scheduling and assignment for disaster recovery of optical networks. Given a set of repairmen to repair a set of failures in the network in the aftermath of a disaster, the goal is to allocate each repairman to a set of failures in an intelligent manner such that we maximize recovered capacity as soon as possible for each recovery schedule thereby recovering more capacity as early as possible. We address the problem by proposing a Multi-Repairmen Disaster Recovery Algorithm (MRDRA) that provides intelligent recovery schedule for a given set of failures and repairmen. Finally, we present numerical results that show the potential merits of our study by considering a 24-node US nation-wide topology and an 11-node COST239 European topology. Numerical results show that our approach can recover more capacity compared to classical scheduling significantly.

Keywords: Multi-Repairmen, Disaster Recovery, Optical Network

Optik ağların afet sonrası onarımı için çoklu tamirci problemi

ÖZ

Günlük aktivitelerimizi sağlamak için internete olan bağımlılığımız her geçen gün artmakta ve internetin omurgasını oluşturan optik ağların dayanıklılığı da bu nedenle büyüyen bir endişeye dönüşmektedir. Bu nedenle, afet kaynaklı geniş çaplı ağ arızaları oluştuğunda, toplum (ağ operatörleri ve son kullanıcılar) ağır sonuçlarla karşı karşıya kalmaktadırlar. Bundan dolayı afet sonrası haberleşme altyapısının ivedilikle onarılması için arayışlar kaçınılmazdır. Her ne kadar bazı araştırmalar geniş çaplı afetlerden optik bağlantıları sakınma üzerine odaklanmış olsa da bazen arızaların oluşması kaçınılmazdır ve servislerin çalışabilir hale gelmesi için hızlı bir onarma gereklidir. Bu çalışmada afet sonrası onarım için çoklu tamirci zamanlaması ve ataması problemi incelenmiştir. Afet sonrası oluşan bir arızalar kümesi ve ağ tamiri üzerine uzmanlaşmış bir tamirci kümesi verildiği düşünüldüğünde, bu çalışmanın amacı her bir tamirciyi arızalara uygun bir şekilde atamaktır. Öyle ki onarım sürecinde mümkün olan en kısa sürede mümkün olan en büyük miktardaki kapasite onarılmış olsun. Bu problemin çözümü için, verilen bir arızalar ve tamirciler kümesi için akıllı bir şekilde kurtarma planı oluşturan bir Çoklu Tamirci Afet Onarma Algoritması önerdik. Çalışmanın sonunda önerilen algoritmanın potansiyel faydaları, 24 düğümlü ABD ulusal ağı ve 11 düğümlü COST239 Avrupa ağı topolojileri için elde edilen sayısal sonuçlarla gösterildi. Sayısal sonuçlar, geliştirilen algoritmanın klasik onarım yaklaşımları ile kıyaslandığında daha kısa sürede daha çok kapasite onardığını göstermiştir.

Anahtar Kelimeler: Çoklu tamirci problemi, afet sonrası onarım, optik ağlar

¹ Sakarya University, Department of Computer Engineering, Sakarya, Turkey – dikbiyik@sakarya.edu.tr

1. INTRODUCTION

Network survivability is a growing concern in telecommunication industry nowadays due to strong reliability on communication services to accomplish our daily activities. When there is telecommunication infrastructure failure the whole community suffers a grave consequence in terms of monetary and social aspects [1] [2] [3] [4]. In principal, survivability of a network is an augmentation of fault tolerance. Authors in [5] define survivability as the ability of a network system to accomplish its mission on time in the presence of attacks, such as natural disasters or man-made attacks. Thus, the problem of network survivability can be considered as provisioning network services regardless of the constraints facing network infrastructures. Generally, it involves mechanisms that ensure fault tolerance and fault recovery of a network.

Terrestrial optical networks are susceptible to failures associated with human activities or natural disasters such as earthquakes, hurricanes, etc. Failures caused by natural disasters usually prompt monumental impact to network operators and network end users due to huge bandwidth loss and long recovery time [2]. Terrestrial optical networks are affected frequently by link failures prompted by human activities thereby causing cable cuts [6]. Large-scale natural disasters also affect terrestrial optical networks [6] - [7]. The problem of multi-link failure (as in a post-disaster scenario) recovery as well as relocation has been investigated in [8] [9] [10]. In order to mitigate network failures prompted by disasters some telecommunication companies have established network emergency management unit that plans and responds to network failures. For example, in 2001, AT&T Company established Network Disaster Recovery (NDR) program aiming at responding to network failures prompted by disasters [11].

Recovery after a disaster should consider not only network restoration through re-provisioning, but also repair activities executed by the repair crew. Even though re-provisioning may restore some of the connections between nodes that are not in disaster zones, the survived network resources may not be sufficient to carry all the connections that need to be restored. Besides, if the source or destination node of a connection is in a disaster area, this connection cannot be restored by re-provisioning. Repair of network resources is a complicated process when the available repair resources, e.g., number of repairmen, is limited. The problem involves quantitative limitations such as, at a given time, number of available repairmen, number of available repair equipments, number of available vehicles for transportation, the availability of fuel to operate such vehicles, the availability of roads, the reliability of

structures to conduct repair activities, etc. It also involves qualitative limitations such as physical and mental conditions of repair crew, the management and coordination of repair activities, interdependency to other physical and/or management systems, etc. In this work, we only focus on number of available repairmen (that dynamically changes in time) and type of failure (e.g., the number of repairmen to recover that failure, time to repair, etc.). Surely, the other aspects should be also taken into consideration; however, this work can serve as a preliminary work to conduct further research in that direction. Note that even considering only a few aspects of the problem, the solution is not straightforward and requires an extensive work.

Some studies focus on stage-by-stage progressive network recovery on optical networks [12] [13]. The approaches proposed in [12] [13] aims to determine the network equipments to repair at each stage to maximize the recovered traffic at early stages. Another work [14] also focuses on network recovery where they propose a scheme that maximizes traffic during the recovery. However, these work assume that the required repair resources for a failure are available when needed, an assumption that is not practical in real life scenarios.

C. Ma et al. focus on multi-repairmen problem in [15] and [16] for restoration of virtual networks. They consider that the problem is static such that number of failures and number of repairmen do not change during recovery phases. However, after a disaster, there are usually correlated failures (e.g., aftershocks) and new failures may arrive. The size of repair crew may also change. It may decrease due to injuries, sickness, or stress (due to relatives and friends lost because of disaster) or it may increase due to support from other branches. Thus, the problem of assigning available repairmen to current failures is a dynamic problem. In this study, we investigated this problem and provide a heuristic, Multi-Repairmen Disaster Recovery Algorithm (MRDRA), which maximizes the recovered traffic as early as possible. The proposed solution is designed to be applied whenever recovery state changes, i.e., a failure is repaired or a new failure occurs.

2. PROBLEM DESCRIPTION

The general flowchart of our proposed solution is shown in Fig. 1. After a disaster, if there is sufficient number of repairmen to cover all the failures, then recovery is completed after all the failures are repaired. However, in practice, the number of repairmen may not be sufficient and a subset of the failures should be recovered first. In this case, the determination of failures that will be repaired depends on the benefit from the repair (how

much capacity, i.e., bandwidth, will be recovered after the repair of each failure), the travel time to the failure location, and the repair time. The goal should be recovered as much bandwidth as possible and as early as possible.

Our approach is subject to following assumptions:

- We consider that the number of repairmen may change in time. Some have to take breaks, some repairmen’s shifts may over, some may not be available due to sickness, injury, loss of a relative or a friend. Thus, every time that a recovery decision is made, the number of available repairmen is re-evaluated.
- There are no flaws and setbacks on coordination and management of recovery activities.
- The network recovery activities are independent from structural recovery and search and rescue operations.
- There are always vehicles ready to transport repair crew to relevant failure locations.
- The repair resources required are always available at the hands of repair crew.

The repairmen assignment problem can be reduced to Binary (0/1) Knapsack Problem (BKP) which is an NP-Hard problem. BKP is applicable in solving dynamic programming problems [17] [18] [19]. BKP defines which items to put in a backpack among an item set to increase the value inside the backpack. Thus given a set of items, each having weight and value; and the capacity of knapsack, the problem can be formulated as:

$$\max \sum_{i=1}^n v_i c_i \quad (1)$$

Subject to:

$$\sum_{i=1}^n w_i c_i \leq \omega \quad (2)$$

$$c_i \in \{0, 1\} \quad (3)$$

Whereby, c_i indicates whether item i is included in the knapsack or not ($c_i = 1$ if it is included and equal to zero otherwise). We modify BKP to address our problem by considering a set of network failures after a disaster and each failure requires a certain amount of repairmen for repair. At a given time, we have set of failures (the items that will be put into the knapsack) with the benefit to be gained after their repair (the value of each item), the repairmen required for each failure (the weight of each

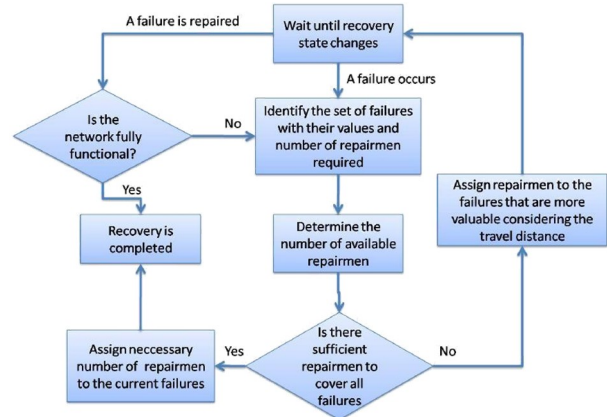


Figure 1. Flowchart of Multi-Repairmen Disaster Recovery Algorithm (MRDRA).

item), and the total number of available repairmen at that time (the capacity of knapsack).

Note that failures are heterogeneous. Some failures (e.g., transponder failures) can be fixed in minutes with a few (or even one) repairmen, while some others (e.g., fiber cuts) may take hours and a significant number of repairmen. The repair of a failure may depend on other failures as well. For instance, to recover the bandwidth of a fiber link all related failures, such as amplifier failures and/or fiber cuts on that link, should be repaired. If we consider sole benefit from the repair of an amplifier on a fiber where there is also a fiber cut on that fiber, then the benefit would be zero, because the repair of amplifier alone cannot provide connectivity on that cable. Thus, we also consider the dependencies between failures and consider failure sets to be repaired instead of single failures.

The gain from a repair depends not only the amount of bandwidth that can be recovered, but also how quickly it can be recovered. Hence, we define the benefit from a repair as a division of bandwidth recovered to time of recovery (sum of travel time and repair time).

We propose a Multi-Repairmen Disaster Recovery Algorithm (MRDRA) where the flow chart is shown in Fig. 1 that solves the problem considering a disaster, multiple links, amplifiers and transponders failures, as well as number of available repairmen. In this context, a network is thought of a system of nodes connected by links of fiber optics. Amplifiers are included in the network for amplification of signal after a certain propagation distance (e.g., 100 km). Moreover, there are transponders for transmission and reception of optical signal at each node.

3. FAILURE IDENTIFICATION AND REPAIRMEN ASSIGNMENT

Given:

- $G(N, E, A)$: network topology where N is a set of nodes, E is a set of edges, and A is as set of amplifiers in a network.
- ϑ : set of dependent-failure sets in a network in the aftermath of a disaster at a given time. Each element $f \in \vartheta$ includes the set of failures with their coordinates, the capacity recovered after the repair of failure set, and number of repairmen required to repair all the failures in f .
- C_f : capacity recovered after the repair of failure set $f \in \vartheta$.
- R_f : repair time of failure set $f \in \vartheta$.
- B_f : benefit from repair of failure set $f \in \vartheta$.
- $m[i; \eta]$: maximum benefit that can be attained with number of repairmen required less than or equal to η using failure sets up to i .
- μ : set of available repairmen at a given time.
- η_f : minimum number repairmen required for failure set $f \in \vartheta$.

In principle, the problem is divided into two sub-problems viz: (i) Identification of failures to be recovered at a given time and (ii) repairmen assignment to each failure such that the benefit is maximized. The solution for the first sub-problem is given in Alg. 1. Here, we look at the failures from the repair of each link. Because, some bandwidth is recovered if and only if the fiber link is repaired. The repair of fiber links consists of repairing a set of dependent failures. First, we need to determine capacity recovered if we repair each link's failure set and number of repairmen required for this failure set. Calculation of C_f is straightforward and depends on how many transponders are in the failure set. In current networks, one transponder at each end node will give a bandwidth of a wavelength capacity. The determination of η_f depends on the repair type. For instance, repair of a transponder can be done by one repairman, while repair of fiber cable break requires more repairmen. Depending on the skills of the repairmen and repair policy of network operator, this number can be considered fixed for each type of failure.

After the identification of failure sets, we need to determine the benefit that can be achieved through repair of each failure set. After this point, the problem becomes similar to BKP (the number of repairmen required is the weight and benefit is the value for each failure set, while

number of available repairmen is the capacity of knapsack). There is a pseudo-polynomial time algorithm using dynamic programming [19] that we modify it for our problem as shown in Alg. 2.

Algorithm 1: Identification of failures to be repaired²

```

for each link in network  $G(N, E, A)$  do
    Check if link, amplifiers and transponders are
    functional.
    if any equipment on the link is not functional
    then
        Determine the failure set  $f$ .
        Get coordinate of each failure in  $f$ .
        Calculate  $C_f$  and  $\eta_f$ .
        Add  $f$  to  $\vartheta$ .
    end if
end for

```

Since the number of repairmen required for each failure set (η_f) and the total number of available repairmen at a given time (μ) are positive integer values, we can define $m[i; \eta]$ (the maximum benefit that can be attained with number of repairmen less than or equal to η using failure sets up to i (first i failure sets)) recursively as follows:

- $m[0; \eta] = 0$
- $m[1; \eta] = m[0; \eta]$ if $\eta_1 > \eta$ (the new failure set is more than the current repairmen limit), else
 $m[1; \eta] = \max(m[0; \eta], m[0; \eta - \eta_1] + B_1)$
-
- $m[i; \eta] = m[i - 1; \eta]$ if $\eta_i > \eta$, else
 $m[i; \eta] = \max(m[i - 1; \eta], m[i - 1; \eta - \eta_i] + B_i)$

Then the solution is obtained by computing $m[|\vartheta|; \mu]$ from the outputs of Alg. 2. The calculations of R_f and B_f in Alg. 2 are straightforward. R_f (repair time of failure set) is equal to sum of time to required of repairmen to reach the failure locations and time to required to repair. Estimated times to repair values of different equipments are usually known and are publicly available. The calculation of B_f is division of capacity recovered to repair time (C_f / B_f).

Note that we apply Alg. 1 and Alg. 2 consecutively whenever the recovery state changes, i.e., a new failure occurs or a failure is repaired, as shown in Fig. 1.

² Our focus on this work is failures of fiber optical cables, amplifiers, and transponders.

Algorithm 2: Repairmen Assignment

```

for each  $f$  in  $\vartheta$  do
    Calculate the repair time of failure set  $f$  ( $R_f$ )
    based on current locations of available repairmen.
    Calculate the benefit (capacity that can be
    recovered times over the repair time) of failure set  $f$ 
    ( $B_f$ ).
end for
for  $j$  from 0 to  $|\mu|$  do
     $m[0; j] := 0$ .
end for
for  $i$  from 1 to  $|\vartheta|$  do
    for  $j$  from 0 to  $|\mu|$  do
        if  $w[i - 1] > j$  then
             $m[i; j] := m[i - 1, j]$ 
        else
             $m[i; j] := \max(m[i - 1, j - \eta_i] + B_i)$ 
        end if
    end for
end for

```

4. ILLUSTRATIVE NUMERICAL EXAMPLES

We conducted numerical examples on two physical network topologies namely 24-node US topology and 11-node COST239 Europe topology shown in Fig. 2 and 3, respectively. We assume that a link is operational only if all amplifiers on it are operational and there is at least one transponder working on each end point of the link. Moreover, if all transponders are not functional, then the link is not fully functional. The speed of each repairman when moving to the repair point is 50 km/hr. The number of transponders is 16, number of repairmen is 30, capacity of each transponder is 10 Gbps and the interval between amplifiers is 100 km. We compare our approach with a classical approach where repairmen are assigned to the closest failure points to minimize recovery time. We report the amount of recovered capacity for each hour.

We run simulation for 50 scenarios and for each physical network topology, whereby in each scenario we generate a random disaster that cause some equipment to fail with some probability that follows Gaussian probability depending on distance from the disaster’s epicenter [20]. Examples are conducted on a computer with an Intel i3 2.4 GHZ CPU, 4 GB DDR3 RAM, and 64 bit Microsoft Windows 10 operating system. Below, we present numerical results for each physical network that shows potential benefits of our study.

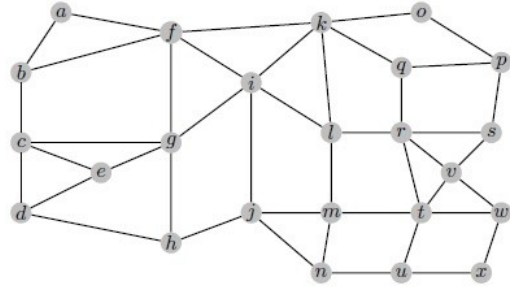


Figure 2. A 24-node US nation-wide topology

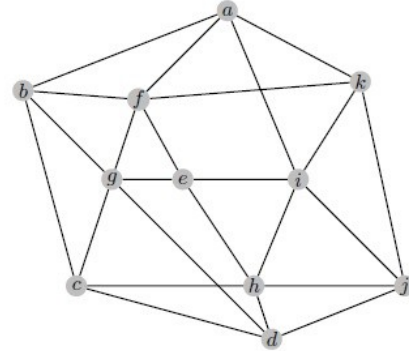


Figure 3. 11-node COST239 European topology

Figure 4 presents numerical results for 24-node US mesh network topology. Since our approach aims to maximize recovered capacity as early as possible, the recovered capacity reaches 2 TB in 10 hours, while the same amount of capacity can only be recovered after 14 hours with classical approach. Similarly, 4 TB capacity is recovered in 18 hours with our approach that is 5 hours earlier than classical approach. After 25 hours, our approach recovers 6 TB capacity, while classical recovers 5TB.

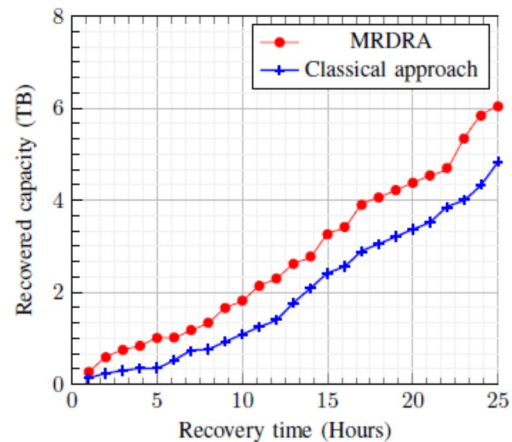


Figure 4. Cumulative recovered capacity per hour for 24-node US mesh network.

Figure 5 reports the cumulative recovered capacity for COST239 topology. We see a similar trend to Fig. 4. In early hours (the first 3 hours), it seems that both approaches show very close results to each other, but then our approach starts to recover faster than the classical approach. For instance, our approach recovers 2 TB in 10 hours and 4 TB in 20 hours, while classical approaches recovers same amounts of capacities in 14 and 23 hours respectively. At the end of 25 hours, while our approach recovers 5.8 TB capacity, classical approach recovers only 4.8 TB.

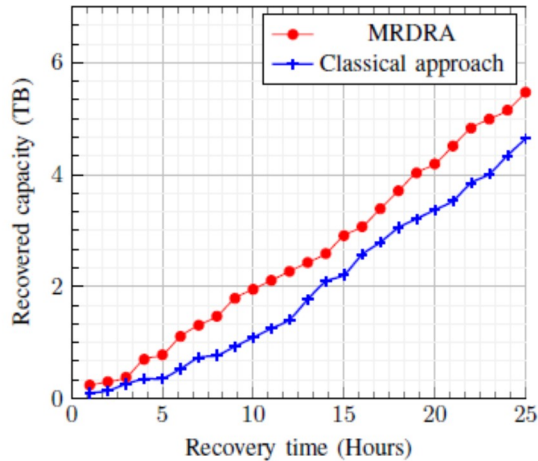


Figure 5. Cumulative recovered capacity per hour for 11-node COST239 network.

5. DISCUSSION

In this study, we focus on the network recovery problem by considering the changes in number of repairmen in time, new failure arrivals after a disaster, interdependency between failures. However, the nature of the problem is more complicated and requires interdisciplinary work with civil engineers, management information system experts, behavioral scientists, etc. Taking this work as a baseline approach, a further research can be conducted with researchers from different disciplines.

6. CONCLUSION

In this study, we investigated the problem of optical network recovery wherein the focus is increasing recovered capacity while minimizing recovery time following network failures associated with disasters. We considered repairing multiple failures in a network such as amplifier, link, and transponder failures given a limited set of repairmen to be involved in reparation activity. We proposed Multi-Repairmen Disaster Recovery Algorithm (MRDRA) that provides recovery schedule such that we can maximize recovered capacity

at a given time during recovery. The numerical examples showed for 24-node US and 11-node COST239 Europe topologies that, with an intelligent recovery plan that considers dependency between failures, capacity recovered, and time of recovery (travel time plus repair time), would provide larger capacity at a given time during the recovery compared to a blind approach that only minimizes recovery time.

REFERENCES

- [1] M. Kobayashi, «Experience of Infrastructure Damage Caused by the Great East Japan Earthquake and Countermeasures against Future Disasters,» *IEEE Commun. Mag.*, cilt 52, no. 3, pp. 23-29, 2014.
- [2] D. L. Msongaleli, F. Dikbiyik, M. Zukerman ve B. Mukherjee, «Disaster-Aware Submarine Fiber-Optic Cable Deployment for Mesh Networks,» *IEEE/OSA J. Lightwave Tech.*, cilt 34, no. 18, pp. 4293-4303, 2016.
- [3] mi2g, [Çevrimiçi]. Available: <http://www.mi2g.com/cgi/mi2g/frameset.php?pageid=http%3Awww.mi2g.com/cgi/mi2g/press/220705.php>. [Erişildi: 2016].
- [4] Information Week,, [Çevrimiçi]. Available: <http://www.informationweek.com/it-downtime-costs-265-billion-inlost-revenue/d/d-id/1097919?>. [Erişildi: 2016].
- [5] J. P. G. Sterbenz ve R. J. P., «Predicting topology survivability using path diversity,» %1 içinde *IEEE RNDM*, Budapest/Hungary, 2011.
- [6] M. F. Habib, M. Tornatore, F. Dikbiyik ve B. Mukherjee, «Disaster Survivability in Optical Communication Networks,» *Computer Communications*, cilt 36, no. 6, pp. 630-644, 2013.
- [7] D. P. Juniora ve M. C. Penna, «A new algorithm for dimensioning resilient optical networks for shared-mesh protection against multiple link failures,» *Optical Switching and Netw.*, cilt 13, pp. 158-172, 2014.
- [8] Y. Xuan, Y. Shen, N. P. Nguyen ve M. T. Thai, «Efficient Multi-Link Failure Localization Schemes in All-Optical Networks,» *IEEE Trans. Commun.*, cilt 61, no. 3, pp. 1144-1151, 2013.
- [9] M. Khair, J. Zheng ve H. T. Mouftah, «Distributed Multi-Failure Localization Protocol for All-Optical Networks,» %1 içinde *IEEE ONDM*, Braunschweig, Germany, 2009.
- [10] H. Yan, R. Wang, Q. Mao ve D. Wu, «A fast multi-fault localization mechanism for multi-domain all-optical networks,» %1 içinde *Adv. Comp.Theory and Eng. (ICACTE)*, Chengdu, China, 2010.
- [11] K. T. Morrison, «Rapidly recovering from the catastrophic loss of a major telecommunications office,» *IEEE Commun. Mag.*, cilt 49, no. 1, pp. 28-35, 2011.

- [12] J. Wang, C. Qiao ve H. Yu, «On progressive network recovery after a major disruption,» %1 içinde *IEEE INFOCOM*, Shanghai, China, 2011.
- [13] K. A. Sabeih, M. Tornatore ve F. Dikbiyik, «Progressive network recovery in optical core networks,» %1 içinde *IEEE RNDM*, Munich, Germany, 2015.
- [14] H. Yu ve C. Yang, «Partial network recovery to maximize traffic demand,» *IEEE Comm. Lett.*, cilt 15, no. 12, pp. 1388-1390, 2011.
- [15] C. Ma, J. Zhang, Y. Zhao, M. Habib, S. S. Savas ve B. Mukherjee, «Traveling repairman problem for optical network recovery to restore virtual networks after a disaster [invited],» *IEEE J. Opt. Commun. Netw.*, cilt 7, no. 11, pp. 81-92, 2015.
- [16] C. Ma, J. Zhang, Y. Zhao ve M. F. Habib, «Scheme for optical network recovery schedule to restore virtual networks after a disaster,» %1 içinde *IEEE/OSA OFC*, Los Angeles, CA, USA, 2015.
- [17] R. Andonov ve S. Rajopadhye, «Knapsack on VLSI: From algorithm to optimal circuit,» *IEEE Trans. on Parallel and Distributed Systems*, cilt 8, no. 6, pp. 545-561, 1997.
- [18] L. Ken-Li, D. Guang-Ming ve L. Qing-Hua, «A genetic algorithm for the unbounded knapsack problem,» %1 içinde *Int. Conf. in Machine Learning and Cybernetics*, Xi'an, China, 2003.
- [19] M. Yanjun, L. Jiandong, L. Qin ve C. Rui, «Group Based Interference Alignment,» *IEEE. Commun. Lett.*, cilt 15, no. 4, pp. 383-385, 2011.
- [20] X.Wang, X. Jiang ve A. Pattavina, «Assessing Network Vulnerability Under Probabilistic Region Failure Model,» %1 içinde *IEEE HPSR*, Cartagena, Spain, 2011.
- [21] K. Kumar ve A. K. Garg, «A survey on protection and restoration methods in Optical Networks,» *Int. J. of Enhanced Research in Science Tech. and Eng.*, cilt 3, no. 5, pp. 84-89, 2014.

Karaciğer mikrodizi kanser verisinin sınıflandırılması için genetik algoritma kullanarak ANFIS'in eğitilmesi

Bülent Haznedar^{1*}, Mustafa Turan Arslan², Adem Kalınlı³

21.06.2016 Geliş/Received, 30.11.2016 Kabul/Accepted

doi: 10.16984/saufenbilder.41925

ÖZ

Sınıflandırma, verilerin analiz edilmesi için önemli bir veri madenciliği tekniği olup tıp, genetik ve biyomedikal mühendisliği başta olmak üzere birçok alanda kullanılmaktadır. Özellikle tıp alanında DNA mikrodizi gen ekspresyon verilerini sınıflandırmaya yönelik yapılan çalışmalarda artış görülmektedir. Ancak, mikrodizi gen ekspresyon (ifade) verilerinde bulunan gen sayılarının çokluğu ve bu veriler arasında doğrusal olmayan bağıntılar bulunması gibi problemlerden dolayı geleneksel sınıflandırma algoritmalarının başarımları sınırlı kalabilmektedir. Bu sebeplerden dolayı son yıllarda sınıflandırma probleminin çözümü için yapay zekâ tekniklerine dayalı sınıflandırma yöntemlerine olan ilgi giderek artmaya başlamıştır. Bu çalışmada, karaciğer mikrodizi kanser veri setinin sınıflandırılması için Uyarlamalı Ağ Tabanlı Bulanık Mantık Çıkarım Sistemi (ANFIS) ve Genetik Algoritmaya (GA) dayalı hibrid bir yaklaşım önerilmiştir. Simülasyon sonuçları, diğer bazı yöntemlere ait sonuçlarla karşılaştırılmıştır. Elde edilen sonuçlardan, önerilen yöntemin diğer yöntemlere göre daha başarılı olduğu görülmüştür.

Anahtar Kelimeler: Neuro-fuzzy, ANFIS, genetik algoritma, sınıflandırma, mikrodizi gen ifade

Training ANFIS structure using genetic algorithm for liver cancer classification based on microarray gene expression data

ABSTRACT

Classification is an important data mining technique, which is used in many fields mostly exemplified as medicine, genetics and biomedical engineering. The number of studies about classification of the datum on DNA microarray gene expression is specifically increased in recent years. However, because of the reasons as the abundance of gene numbers in the datum as microarray gene expressions and the nonlinear relations mostly across those datum, the success of conventional classification algorithms can be limited. Because of these reasons, the interest on classification methods which are based on artificial intelligence to solve the problem on classification has been gradually increased in recent times. In this study, a hybrid approach which is based on Adaptive Neuro-Fuzzy Inference System (ANFIS) and Genetic Algorithm (GA) are suggested in order to classify liver microarray cancer data set. Simulation results are compared with the results of other methods. According to the results obtained, it is seen that the recommended method is better than the other methods.

Keywords: Neuro-fuzzy, ANFIS, genetic algorithm, classification, microarray gene expression

* Sorumlu Yazar / Corresponding Author

1 Hasan Kalyoncu Üniversitesi, Mühendislik Fakültesi, Bilgisayar Mühendisliği, Gaziantep – bulent.haznedar@hku.edu.tr

2 Mustafa Kemal Üniversitesi, Kırıkhan Meslek Yüksekokulu, Bilgisayar Teknolojileri Bölümü, Hatay - mtarslan@mku.edu.tr

3 Erciyes Üniversitesi, Mühendislik Fakültesi, Bilgisayar Mühendisliği, Kayseri - kalinlia@erciyes.edu.tr

1. GİRİŞ (INTRODUCTION)

Günümüzde teknolojik gelişmeler ve gelişmiş donanımlar sayesinde artık yüksek boyutlu verileri depolamak mümkün hale gelmiştir. Bu yüzden yüksek boyutlu verilerden anlamlı bilgiler çıkarmak için veri madenciliği fonksiyonlarının kullanımında son yıllarda artış görülmüştür. Yüksek boyutlu verilere genel olarak genetik çalışmalar sonucu elde edilen biyolojik veriler gösterilebilir.

İnsan yapısındaki dokuların hepsi aynı genetik materyali içermektedir. Fakat her hücrede aynı genler aktif değildir. Hangi genlerin aktif olup hangilerinin aktif olmadığı bilgisi, hücrelerin nasıl bir fonksiyona sahip olduğu ve bazı genler normal şekilde çalışmadığında hücrenin bu olaydan nasıl etkileneceği bilgisini vermektedir. Hücrenin gen ifadesi işte bu aktiflik bilgisine eşittir [1]. Bütün organlarımız aynı genetik materyali içermesine rağmen genlerin farklı hücrelerde farklı ifade edilmelerinden dolayı böbrek, beyin, karaciğer gibi hücreler birbirinden farklı fonksiyonlara sahiptir [2].

Bilgisayar teknolojisinin moleküler biyoloji ile birlikte gelişmesi sonucunda biyoteknolojinin kavramsal olarak ulaşabileceği noktalardan biri olan mikrodizi çipler (gen çip) ortaya çıkmıştır [3]. Mikrodizi çip teknolojisi sayesinde aynı anda birden fazla genin incelenmesi ve genlerin birlikte araştırılması fırsatı doğmuştur [4].

DNA mikrodizi teknolojisi sayesinde yüzlerce genin ifade düzeyleri eş zamanlı olarak incelenebilmektedir [5]. Bu teknoloji sayesinde hasta ve sağlıklı hücrelerin gen ifadelerinin karşılaştırılması neticesinde hastalığa neden olan genlerin belirlenmesi mümkün olabilmektedir [6]. Mikrodizi yöntemi kullanarak gen ifade verilerinin analizi sayesinde psikolojik, kalp ve bulaşıcı hastalıklar gibi birçok hastalık hakkında bilgi vermektedir. Elde edilen bu bilgiler sayesinde hastalıkların önlenmesi yönünde tedbirler alınabilmektedir. Bu teknik sayesinde ayrıca gen üzerinde taşınan kalıtsal hastalıklar, alt türlerine kadar incelenebilmektedir [7].

DNA mikrodizi teknolojisi ile bir çalışmada çok fazla veri analizi yapıldığından sonuçlar için çok fazla zaman gerekmektedir. Elde edilen sonuçlar ise yüzlerce genin ifadesini barındırdığından dolayı sonuçları yorumlamanın bir hayli zor olduğu ve çok pahalı bir yöntem olduğundan dolayı çok fazla deneylerin tekrarlanmaması ise dezavantajlarını oluşturur [7].

Mikrodizi gen ifade verilerini sınıflandırmak için literatürde Bayes Ağları, Destek Vektör Makineleri ve Karar Ağaçları gibi istatistiksel yöntemler sıklıkla kullanılmaktadır. Diaz-Uriarte ve Andres [8], rastgele

orman algoritmasını kullanarak 9 farklı mikrodizi veri seti üzerinde hem gen seçimi hem de sınıflandırma yapmışlardır. Dudoit ve arkadaşları [9], gen ifade verisi kullanarak tümör sınıflandırması için fisher lineer diskriminant analizi, k-en yakın komşu algoritması ve torbalama yöntemlerini kullanmışlardır. Furey ve arkadaşları [10], mikrodizi ifade verilerini kullanarak kanser dokusu örneklerini sınıflandırmak için destek vektör makinelerini kullanmışlardır. Lee ve arkadaşı [11], kan kanseri gen ifade verisini kullanarak destek vektör makineleri yardımıyla çok sınıflı sınıflandırma yapmışlardır. Liu ve arkadaşları [12], gen ifade profillerini kullanarak k-en yakın komşu, naive bayes, destek vektör makineleri ve C4.5 karar ağacı gibi sınıflandırma yöntemlerinin karşılaştırmalı bir performans analizini yapmışlardır. Statnikov ve arkadaşları [13] ise mikrodizi gen ifade kanser teşhisinde destek vektör makineleri, k-en yakın komşu ve yapay sinir ağları gibi çok sınıflı yöntemler kullanılarak bu sınıflandırma yöntemlerinin karşılaştırmalı bir değerlendirmesini yapmışlardır. Mikrodizi verilerinde yüzlerce gen bulunması ve bu genler arasında doğrusal olmayan ilişkiler bulunması sebebiyle geleneksel yöntemlerin başarımları sınırlı kalabilmektedir. Bu zorluklar bilim adamlarını daha güçlü ve modern yöntemler keşfetmeye yöneltmiştir. Bu amaçla, araştırmacıların ilgisi zor problemlerin çözümünde yapay zekâ tabanlı yöntemlerin kullanılmasına yönelik olarak artmaya başlamıştır. Khan ve arkadaşları [14], yapay sinir ağları ve gen ifade profillerini kullanarak kanser hastalıkların tahmin edilmesi ve sınıflandırılması üzerine bir çalışma yapmışlardır. Ringner ve Peterson [15], yapay sinir ağları yardımıyla mikrodizi tabanlı kanser teşhisi üzerine bir çalışma gerçekleştirmişlerdir.

Son yıllarda yüksek boyutlu problemlerin çözümünde yapay zekâ tabanlı yöntemlerin kullanılması yaygınlaşmış ve öznelik seçme ve sınıflandırma problemlerine karşılık algoritmalar geliştirilmiştir. Pirooznia ve arkadaşları [16], mikrodizi gen ifade (ekspresyon) verilerini kullanarak sınıflandırma çalışması yapmışlardır. Bu çalışmada istatistiksel yöntemler ile yapay zekâ yöntemleri karşılaştırılmış ve yapay zekâ yöntemlerinin daha iyi sonuçlar verdiği görülmüştür. Loganathan ve Girijia [17], ANFIS ağırlık parametrelerini Runge Kutta Learning algoritması ile eğitmişler ve oluşturulan modelin performansını, mikrodizi gen ekspresyon kanser verilerini sınıflandırarak ölçmüşlerdir. Kumar ve Punithavalli [18], ANFIS ile kanser verilerini sınıflandırmış ve elde edilen sonuçları istatistiksel yöntemler ile karşılaştırmışlardır.

Bu çalışmada karaciğer kanseri mikrodizi gen ekspresyon verilerini doğru sınıflara atayabilmek için popülasyon tabanlı bir algoritma olan genetik algoritma kullanılarak ANFIS modelinin parametrelerinin

eğitilmesine yönelik bir yaklaşım önerilmiştir. Önerilen yaklaşımın performansı k-katlı çapraz doğrulama yöntemi ile test edilmiş ve k parametresi 5 seçilmiştir. Önerilen yöntem ile bulunan sonuçların performansı diğer yöntemlerin performansları ile karşılaştırılmıştır. Bir sonraki bölümde ise kullanılan yöntemler hakkında bilgi verilmiş, 3. bölümde algoritmanın eğitildiği karaciğer kanser veri seti tanımlanmıştır. 4. Bölümde ise önerilen yöntem hakkında bilgi verilmiş ve elde edilen sonuçlar ise bölüm 5'te sunulmuştur.

2. METOTLAR (METHODS)

Bir sınıflandırma fonksiyonu temelde iki bölümden oluşur. Bunlar, öznitelik seçimi ve verilerin doğru sınıflara atanmasıdır. Bu çalışmada kullanılan sistem bileşenleri aşağıda açıklanmaktadır.

2.1. Öznitelik Seçme (Feature Selection)

Öznitelik seçme olayı, öznitelik uzayı içerisinde optimal bir öznitelik alt kümesinin bulunmasıdır. Ayrıca optimal öznitelik alt kümesinin bulunması ile problemin boyutunun indirgenmesi ve öznitelik kümesini en uygun alt kümenin temsil etmesidir. Mikrodizi verileri yüksek boyutlu veriler olduklarından dolayı sınıflandırma yöntemlerinin performanslarını arttırmak için öznitelik seçme yönteminin kullanılmasının bir gereklilik olduğundan söz edilebilir.

2.1.1. Korelasyon Tabanlı Öznitelik Seçimi (KTÖS-Correlation Based Feature Selection)

KTÖS, Sınıf ile yüksek korelasyona sahip ve birbirleriyle düşük korelasyona sahip özniteliklerin bir arada bulunduğu alt kümeleri oluşturur.

$$G_s = \frac{k\bar{r}_{ci}}{\sqrt{k + k(k-1)\bar{r}_{ii}}} \quad (1)$$

Eşitlik (1)'de yer alan k parametresi veri alt kümesinin öznitelik sayısı, r_{ci} parametresi ortalama öznitelik korelasyonu ve r_{ii} parametresi ise ortalama öznitelik iç korelasyonudur [19].

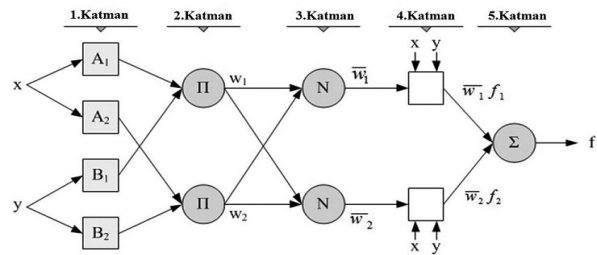
2.2. ANFIS ile Sınıflandırma (Classification with ANFIS)

Sınıflandırma işlemi, ağ tabanlı bulanık mantık çıkarım sisteminde (ANFIS) bulunan başlangıç ve sonuç parametrelerinin genetik algoritması (GA) ile optimize edilmesi ile oluşan model kullanılarak yapılmıştır.

2.2.1. Adaptif Ağ Tabanlı Bulanık Mantık Çıkarım Sistemi (Adaptive Neuro-Fuzzy Inference System, ANFIS)

Adaptif ağ tabanlı bulanık mantık çıkarım sistemi (ANFIS), Takagi Sugeno bulanık modeli dikkate alınarak geliştirilmiş bir yapay sistemdir [20]. Sinir ağlarının geriye yayımlı öğrenme yeteneği ile bulanık mantığın sonuç çıkarma özellikleri ANFIS yapısında birleştirilmiştir. Üyelik fonksiyonları ile bulanıklaştırıldığı giriş verilerini bulanık kurallar ile ağ üzerinde dağıtarak çıkış hesaplamaktadır. Bu süreç ANFIS modeline çıkarım yeteneği sağladığı için tahmin problemlerinde başarıyı oldukça yüksektir.

ANFIS başlangıç ve sonuç parametreleri olmak üzere iki tür parametreye sahiptir. Bu iki parametre türü bulanık kuralları birbirine bağlar. Modelin eğitimi ise bu parametrelerin optimizasyonu ile sağlanır. ANFIS temel olarak beş katmandan oluşmaktadır. Şekil 1'de iki giriş ve bir çıkıştan oluşan temel bir ANFIS yapısı verilmiştir [21].



Şekil 1. Temel bir ANFIS yapısı [22] (A basic structure of ANFIS)

1. Katman

Bulanıklaştırma katmanı olarak adlandırılan bu katmandaki her düğümden alınan sinyal diğer katmana aktarılır. Her düğümden alınan sinyal giriş değerlerine ve kullanılan üyelik fonksiyonunun türüne bağlı olarak oluşmaktadır. Bu katmandaki düğümlerin çıkışları (O_{li}) için Eşitlik (2) ve Eşitlik (3) aşağıda verilmiştir [23].

$$O_{li} = \mu A_i(x) \quad i = 1,2 \quad (2)$$

$$O_{li} = \mu B_{i-2}(x) \quad i = 3,4 \quad (3)$$

Burada A_i ve B_i girişlere ait herhangi bir üyelik fonksiyonu olup μA_i ve μB_i ise bu fonksiyon için hesaplanmış üyelik dereceleridir. Çan eğrisi türünde üyelik fonksiyonu için μA_i aşağıdaki eşitlik ile hesaplanmaktadır

$$\mu A_i = e^{-\frac{1}{2}\left(\frac{x-c}{a}\right)^2} \quad i=1,2 \quad (4)$$

Burada a_i ve c_i sırasıyla, üyelik fonksiyonunun sigma ve merkez parametreleridir.

2.Katman

Kural katmanı olarak adlandırılan bu katmanda bir önceki katmandan gelen üyelik dereceleri kullanılarak her kuralın ateşleme seviyesi hesaplanmaktadır.

$$O_{2i} = w_i = \mu A_i(x) \cdot \mu B_i(y) \quad i=1,2 \quad (5)$$

3.Katman

Normalizasyon katmanı olarak adlandırılmaktadır. Bu katmanda her kuralın normalleştirilmiş ateşleme seviyesi hesaplanmaktadır.

$$O_{3i} = \bar{w}_i = \frac{w_i}{w_1 + w_2} \quad i=1,2 \quad (6)$$

4.Katman

Durulaştırma katmanı olarak adlandırılan bu katmanda her bir kural için çıkış değeri hesaplanmaktadır. Kuralların çıkışı, bir önceki katmandan gelen normalize edilmiş ateşleme seviyesi değerinin, birinci dereceden polinom ile çarpılmasıyla elde edilmektedir.

$$O_{4i} = \bar{w}_i \cdot f_i = \bar{w}_i \cdot (p_i x + q_i y + r_i) \quad i=1,2 \quad (7)$$

5.Katman

Çıkış katmanıdır. Durulaştırma katmanında elde edilen her kurala ait çıkış değerleri toplanarak ANFIS'in çıkışı elde edilir.

$$O_{5i} = f = \sum \bar{w}_i \cdot f_i = \frac{\sum w_i \cdot f_i}{\sum w_i} \quad i=1,2 \quad (8)$$

Bulanık sistemlerin, öğrenme ve adaptasyon sürecini kolaylaştırması, sinirsel ağların ise kontrol parametrelerini ağ üzerinde dağıtarak doğrusal olmayan problemlerde başarılı olması, sinirsel bulanık bir ağ modeli olan ANFIS mimarisine büyük üstünlük kazandırmaktadır.

ANFIS modelinin eğitilmesindeki amaç giriş ve çıkış değerlerine bağlı olarak ağırlık değerleri için optimal değerlerin üretilmesidir. ANFIS'in ağırlık parametrelerinin eğitiminde türeve dayalı algoritmalar

yaygın olarak kullanılmaktadır. Ancak türev tabanlı algoritmalarda eğitim hesaplaması gibi zorluklar bulunmasının yanında yerel minimuma takılma gibi problemlere sebep olmaktadır. Bu sebeptendir ki türev tabanlı algoritmalar ile ANFIS in eğitilmesi ve parametrelerin güncellenmesi temel sorunlardan biridir. Araştırmalar ANFIS'in parametrelerini eğitmek için son zamanlarda farklı algoritmalar önermiştir. Bu algoritmalar bazılarını ise türeve dayalı olmayan Genetik algoritma, PSO ve Diferansiyel Gelişim Algoritması gibi sezgisel algoritmalar [24].

2.2.2. Genetik Algoritma (Genetic Algorithm)

1970'li yıllarda temel ilkeleri John Holland tarafından ortaya konulan Genetik Algoritma (GA) pek çok problem türünde başarı ile uygulanmaktadır [25]. GA, optimizasyon veya arama probleminde tam yada yaklaşık sonuçlar bulabilmek için kullanılan sezgisel bir algoritmadır. Bu algoritma kalıtım, mutasyon, seçim ve çaprazlama gibi evrimsel biyolojideki tekniklerden esinlenerek geliştirilmiştir. GA arama uzayı büyük olan ve ayrıca değişken sayısı çok fazla olan, çok boyutlu problemlere bile oldukça rahat uygulanabilmektedir. Problemlerin arama uzaylarında, uzayın tamamını aramak yerine daha iyi olabilecek değerleri deneme eğilimi sonucu makul sayılabilecek süreler içerisinde optimal sonuçlar üretebilme kabiliyetine sahiptir. GA'nın temel adımları Şekil 2'de verilmiştir.

- | |
|--|
| <p>Adım 1. Rastgele değerlerden başlangıç popülasyonunu üret</p> <p>Adım 2. Kromozomların uygunluk değerlerini hesapla</p> <p>Adım 3. Şartlar sağlanıncaya kadar devam et (Uygunluk değeri, işlem süresi vb)</p> <ul style="list-style-type: none"> (i) Popülasyonda kötü uygunluk değerine sahip kromozomları belirle (ii) Çocuk kromozomlar için ebeveyn kromozomları tayin et (iii) Çaprazlama işlemi ile ebeveyn kromozomlardan çocuk kromozomlar üret (iv) Çocuk kromozomları mutasyona uğrat. (v) Popülasyona yeni dahil olan kromozomların uygunluk değerlerini hesapla |
|--|

Şekil 2. Genetik Algoritmanın Temel Adımları (Main steps of the basic GA algorithm)

GA, popülasyon temelli bir optimizasyon algoritmasıdır. Popülasyonu oluşturan aday çözümlerin algoritmadaki karşılığı kromozomlardır. Bu kromozomlar çeşitli evrim işlemleri sayesinde daha iyi sonuçları temsil eden çözüm

adaylarına dönüşürler. Bu işlem kabul edilebilir bir uygunluk değerine ulaşmaya kadar veya önceden belirlenen bir işlem süresi veya nesil sayısı gibi kriterler karşılanıncaya kadar sürdürülür.

GA'da kromozomlar (aday çözümler) temsil ettiği çözüme ait ayrık veya sürekli değerlere sahip değişkenleri tutan katarlardır. Uygunluk fonksiyonu ise kromozomların kalitesini ölçen amaç fonksiyonudur.

a) Başlangıç Popülasyonunun Oluşturulması (Creating Initial Populations)

Başlangıç popülasyonu rastgele aday çözümlerden oluşur. Popülasyonun büyüklüğü için belirli bir kriter yoktur. Ancak popülasyondaki birey sayısının çok fazla olmasının problemin çözüm zamanına veya elde edilecek çözümün iyiliğine etkisi olmadığı değerlendirilmektedir. Bu yüzden genellikle popülasyon 20-50 arasında alınır [26].

b) Uygunluk Değerinin Hesaplanması (Determination of Eligibility Value)

Uygunluk fonksiyonu yardımıyla hesaplanabilen bir değerdir. Uygunluk değeri, her bir kromozomun uyguluk fonksiyonuna tabi tutulması ile elde edilir. Uygunluk fonksiyonu değeri bireyin çözüm kalitesini gösterir. Bir sonraki nesle aktarılacak bireyler, uygunluk fonksiyonu değerlerine göre belirlenir.

c) Seçim (Selection)

Seçim stratejisi için sıklıkla Rulet Tekerlek yöntemi ve Turnuva yöntemi kullanılmaktadır.

Rulet Tekerleği Yöntemi

Bu yöntemin temel mantığı uygunluk değeri yüksek olan kromozomların ebeveyn kromozom olarak seçilebilme olasılığının, uygunluk değeri nispeten düşük olan kromozomlardan daha fazla olmasıdır [26].

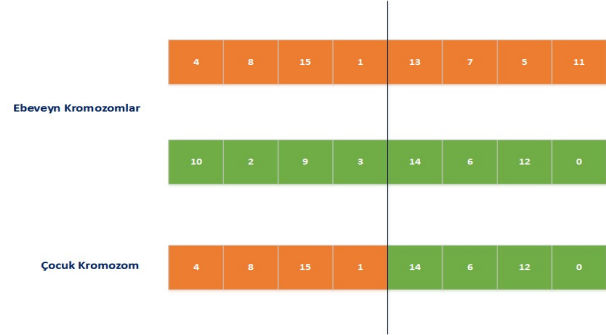
Turnuva Yöntemi

Bu yöntemde popülasyondan rastgele seçilecek n adet kromozom içinden en iyi kromozomlar ebeveyn olarak seçilir [26].

d) Çaprazlama (Crossover)

Çaprazlama, genellikle iki ebeveyn kromozomdan alınan genlerle oluşturulur. Kromozomların başlangıç noktasından belirlenecek bir noktaya kadar olan genler

ilk ebeveyn kromozomdan alınırken söz konusu noktadan sonraki genler ise ikinci ebeveyninden alınarak yapılır. Farklı şekillerde çaprazlama yapılabilmektedir. Tek noktadan çaprazlama Şekil 3'de görülmektedir.



Şekil 3. Tek Noktadan Çaprazlama (One-Point Crossover)

e) Mutasyon (Mutation)

Mutasyon işlemi, popülasyondaki kromozomların çeşitliliğini arttırarak yeni çözüm adaylarının oluşmasını sağlar. Bir bireyin bir veya birkaç geninin değişerek farklı bir birey haline gelmesidir. Mutasyon popülasyonda çeşitliliğin oluşmasını sağlayan operatörlerden biridir. Mutasyon oranı genellikle %0,5 - %5 arasında bir sayıdır.

3. DENEYSEL VERİ SETİ (EXPERIMENTAL DATA SET)

Bu çalışmada veri kaynağı olarak Karaciğer kanseri mikrodizi gen ifade veri seti kullanılmıştır. Bu veri seti Rutgers Üniversitesinin Biyoinformatik Laboratuvarında bulunan veri tabanından elde edilmiştir [27].

Karaciğer Kanseri (Liver Cancer Chen-2002)

Chen ve arkadaşları [28] tarafından 2002 yılında çalışılan karaciğer kanser veri seti, hastalıklı ve normal kişilerin karaciğer dokularından alınmış 179 adet örnekten oluşmaktadır. Bu veri setindeki her örnek 85 mikrodizi gen dizilimi ile temsil edilmektedir. Veri seti HCC adında tümörlü doku ve normal doku olmak üzere 2 farklı sınıf ile temsil edilmektedir. Toplamda 179 örnek bulunan veri setinde 104 adet tümörlü doku bulunurken 75 adet de sağlıklı karaciğer dokusu bulunmaktadır.

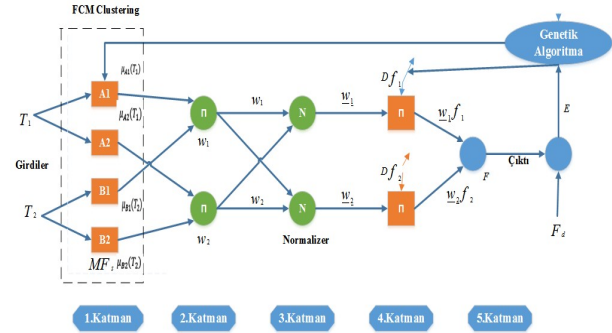
4. DENEYSEL ÇALIŞMA (EXPERIMENTAL STUDY)

Bu çalışmada karaciğer kanseri mikrodizi gen ifade verisi kullanılarak sınıflandırma yapılmıştır. Veri seti sınıflandırma işleminden önce [0-1] aralığına normalize

edilmiş daha sonra veri setlerine öznitelik seçme yöntemi uygulanmıştır. Bu çalışmada birçok öznitelik seçme yöntemi incelenmiş ve başarısından dolayı veri setlerinin bir alt setini oluşturmak için Korelasyon tabanlı öznitelik seçimi kullanılmıştır. Böylece yüksek boyutlu veri setleri yerine bu veri setlerini en iyi şekilde ifade eden alt veri setleri ile sınıflandırma yöntemlerinin performanslarını arttırmak ve sınıflandırmada daha başarılı sonuçlar almak mümkün hale gelmiştir.

Bu çalışmada, ANFIS modelinin GA kullanılarak eğitilmesine yönelik karaciğer kanseri mikrodizi gen ifade verisi kullanılarak uygulama geliştirilmiştir. Toplamda 179 örnek bulunan karaciğer verisini sınıflandırmada örneklerin az olması sebebiyle, modelin genelleştirebilme yeteneğini doğru bir şekilde değerlendirebilmek için *çapraz doğrulama (cross validation)* yöntemlerinden *K parçalı* yöntemi kullanılmıştır. *K parçalı çapraz geçerlilik* yönteminde amaç bir deneyi bağımsız koşullarda yineleyerek sonuçlarının geçerliliğini sınamaktır. Örneğin, sınıflandırma problemlerinde, bir veri kümesini aşağı yukarı eşit *k* tane kümeye bölüp, her bir *k* için, sınıflandırıcıyı oluşturmak için, *k-1* kümeyi kullanıp ve arta kalan *k*'inci küme üzerinde test işlemini yapmaktır. Bu çalışmada, 5-kat çapraz doğrulama yöntemi kullanılmıştır.

Kullanılan veri setlerindeki parametre sayılarının çokluğundan dolayı, bulanık kural ve üyelik fonksiyonu sayılarının az sayıda olduğu ANFIS modelleri oluşturabilmek için, ANFIS ile FCM kümelemenin entegre olduğu bir bulanık kural oluşturma tekniği kullanılmıştır. FCM uygulanarak oluşan ANFIS modellerindeki giriş sayısı her bir veri setindeki gen sayısına eşit olup her bir giriş için üyelik fonksiyonu türü *gauss*, üyelik fonksiyonu sayısı 10 ve bulanık kural sayıları 10 olarak belirlenmiştir. Yapılan simülasyon çalışmalarında kullanılan modelin oluşturulmasını ve oluşturulan modelin parametrelerinin optimize edilmesini gösteren blok diyagram Şekil 4'de verilmiştir. ANFIS'in güncellenmesi gereken iki parametre türü vardır. Bunlar başlangıç parametreleri ve sonuç parametreleridir. Başlangıç parametreleri Eşitlik (4)'de $\{a_i, c_i\}$ olarak verilen *gauss* üyelik fonksiyonlarına aittir. Buradaki a_i üyelik fonksiyonlarının varyansı, c_i üyelik fonksiyonlarının merkezidir. Başlangıç parametrelerinin toplam sayısı tüm üyelik fonksiyonlarındaki parametrelerinin toplamına eşittir. Sonuç parametreleri ise durulaştırma katmanında kullanılan Eşitlik (7)'de $\{p_i, q_i, r_i\}$ olarak gösterilen parametrelerdir. Tasarlanan ANFIS modelinin tüm parametrelerinin optimize edilmesi için GA algoritması kullanılmıştır.



Şekil 4. Önerilen yöntemle oluşturulan ANFIS'in temel yapısı (Basic Structure of ANFIS with FCM Clustering)

Bu çalışmada önerilen yöntem karaciğer kanseri problemine uygulanmıştır. Kanser verisinin doğru bir şekilde sınıflandırılabilmesi Şekil 4'de verilen ANFIS modelinin 1. ve 4. katmanındaki başlangıç ve sonuç parametrelerinin başlangıç değerleri için FCM kümeleme yöntemi ile bir çözüm uzayı oluşturulmuş ve sonrasında parametreler her bir iterasyonda GA kullanılarak güncellenmiştir. Eşitlik (9) ile tanımlanan RMSE hata fonksiyonu çözümün hata değerinin hesaplanmasında kullanılmıştır. Bu hata fonksiyonunda kullanılan *F* ve *F_d* parametreleri sırasıyla ANFIS tarafından elde edilen çıkış ve veri setinin gerçek çıkışıdır. GA ile ANFIS parametrelerinin eğitimi sürecinde, MATLAB ortamında oluşturulan özgün kodlar kullanılmıştır.

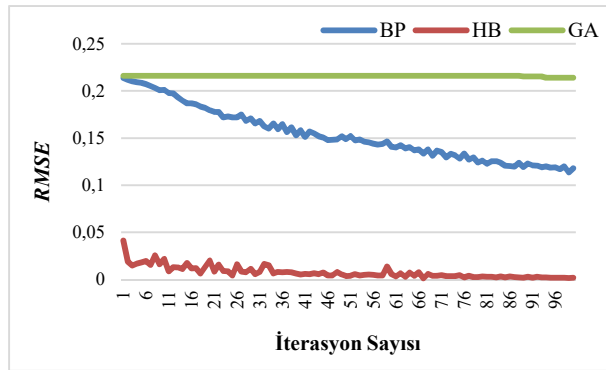
$$RMSE = \sqrt{\frac{\sum_{i=1}^N (F(i) - F_d(i))^2}{N}} \quad (9)$$

Yapay zeka optimizasyon algoritmalarının performansı algoritmaların kontrol parametrelerine fazlaca bağlıdır ve genellikle kontrol parametreleri için hangi değerlerin kullanılması gerektiğine yönelik kesin kural ve yöntemler bulunmamaktadır. Bu kapsamda, parametreler için araştırmacıların yaygın olarak önerdiği aralıklarda değerler tayin edilmesi veya çok sayıda deneme yapılarak optimal parametre değerlerinin belirlenmesi yaklaşımı kabul görmektedir. Çalışmada, GA algoritmasının parametre değerlerine karar vermek için birçok deneme gerçekleştirilmiş ve denemeler sonucunda GA için popülasyon sayısı 25, çaprazlama oranı 0,7 ve mutasyon oranı 0,15 olarak alınmıştır.

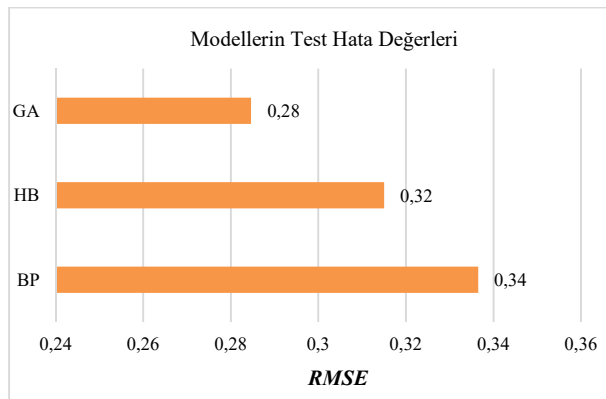
Önerilen yaklaşımın performansının farklı yöntemlerle karşılaştırılması amacıyla ANFIS ağı karaciğer mikrodizi gen ekspresyon kanser verisinin sınıflandırılmak için BP ve Hibrid algoritmalarla da eğitilmiştir. Bu amaçla, kullanılan BP algoritması için öğrenme katsayısı 0.2 ve momentum oranı 0.4 olarak seçilmiştir. Hibrid algoritma olarak da en küçük kareler tahmini ve BP algoritmasının birleştirilmesin oluşan bir yöntem kullanılmıştır. Ayrıca

bu iki algoritma için iterasyon sayısı 100 olarak belirlenmiştir.

Oluşturulan ANFIS modelleri BP, Hibrid ve GA algoritmaları ile eğitilerek sınıflandırmadaki performansları karşılaştırılmıştır. Her algoritma için ANFIS 15 kez eğitilmiştir. Ayrıca her model için elde edilen *RMSE* hata değerlerinin ortalaması alınarak, ortalama *RMSE* ($RMSE_{AVG}$) değeri bulunmuştur. Şekil 5'te yöntemlere ait 100 iterasyon için eğitim hata değerlerinin (*RMSE*) değişim grafiği verilmiştir. Şekil 6'da ise karaciğer verisini sınıflandırmak için farklı algoritmalar ile eğitilen ANFIS modellerine giriş olarak verilen veri setinin gerçek çıkışı ile ANFIS tarafından elde edilen çıkış arasındaki farkı gösteren test hata değerleri (*RMSE*) verilmiştir.



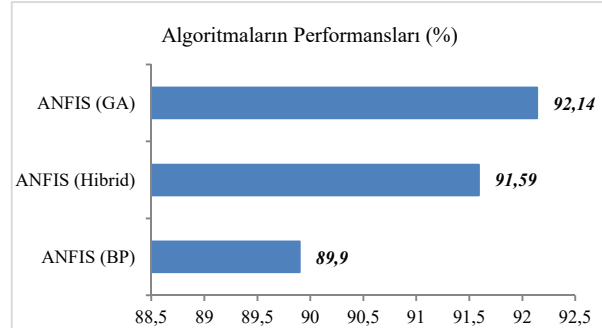
Şekil 5. ANFIS modellerine ait *RMSE* eğitim hata değerleri (Training *RMSE* error values of the ANFIS models)



Şekil 6. ANFIS modellerine ait *RMSE* test hata değerleri (Testing *RMSE* error values of the ANFIS models)

Şekil 5'te görüldüğü üzere en az eğitim hata değeri Hibrid algoritma ile bulunmuş olmasına rağmen, Şekil 6'da en az test hata değeri GA ile elde edilmiştir. Eğitim ve test hata değerleri arasındaki farkın çok olması aşırı öğrenme olduğunu ve eğitilen modelin uygulanabilir olmadığını ortaya koymaktadır. Eğitim ve test hata değerlerinin ise birbirine yakın olması daha güçlü ve başarıyı yüksek modeller ortaya koymaktadır. Yapılan

simülasyon çalışmaları neticesinde optimal parametre değerleri ile elde edilen sınıflandırma performanslarına ait bilgiler Şekil 7'de verilmiştir. Görüldüğü üzere en başarılı sınıflandırma performansı ANFIS-GA modeline aittir.



Şekil 7. Algoritmaların Karaciğer Kanseri Sınıflandırma Performansları (Classification performance of the algorithms for Liver Cancer)

Şekil 7' de görüldüğü gibi ANFIS - GA yönteminin ortalama % 92.14 oranıyla karaciğer kanser veri setini sınıflandırmada en yüksek performansı sergilemektedir. Ayrıca sınıflandırma yöntemlerinin ortalama başarımları ve birbirlerine göre gelişim yüzdeleri de aşamalı olarak Tablo 1'de verilmiştir. ANFIS ağınnın Hibrid algoritma ile eğitilmesi durumunda geleneksel BP algoritmasına göre sınıflandırma başarımının % 1.88 iyileştiği, GA ile eğitilmesi durumunda ise Hibrid algoritmasına göre % 0.60 daha başarılı olduğu görülmektedir. Gelişim yüzdeleri genetik algoritmanın sınıflandırma probleminde diğer yöntemlere göre daha başarılı olduğunu göstermektedir.

Tablo 1. Metotların ortalama başarımları ve gelişim yüzdeleri (The average performances and the percentages of development of the methods)

Metot	Başarımlar (%)	Gelişim Yüzdeleri (%)
ANFIS - BP	89.90	-
ANFIS - Hibrid	91.59	1.88
ANFIS -GA	92.14	0,60

5. SONUÇLAR VE TARTIŞMA (CONCLUSIONS AND DISCUSSION)

Bu makalede, karaciğer mikrodizi gen ekspresyon kanser verisinin sınıflandırılması amacıyla ANFIS modelinin BP, Hibrid algoritmaları ve GA optimizasyon algoritmasıyla eğitilerek performansları karşılaştırılmıştır. Şekil 7'de verilen sonuçlardan, GA algoritması kullanılarak ANFIS'in eğitilmesine dayalı

yaklaşımın daha başarılı olduğu görülmüştür. Ayrıca, GA algoritmasının türeve dayalı algoritmalar gibi kısıtlamalar içermemesi ve problemlere kolaylıkla uygulanabilir olması nedeniyle, ANFIS ağıının farklı problemlere yönelik uygulamalarında da kullanılabilceği değerlendirilmiştir.

TEŞEKKÜR (ACKNOWLEDGEMENT)

Bu çalışma Erciyes Üniversitesi Bilimsel Araştırma Projeleri Koordinasyon birimi tarafından desteklenmiştir. (Proje Numarası: FDK-2016-6371) Ayrıca, bu çalışmanın sorumlu yazarını Yurt İçi Lisansüstü Burs Programı kapsamında yürütülen 2211-A Genel Yurt İçi Doktora Bursu ile destekleyen TÜBİTAK Bilim İnsanı Destekleme Daire Başkanlığı birimine teşekkür ederiz.

KAYNAKLAR (REFERENCES)

- [1] G. S. Özcan, "Bütünleştirici Modül Ağlarıyla Gen Düzenleme Analizi," Başkent Üniversitesi, 2014.
- [2] H. Ü. Lüleyap, *Moleküler Genetiğin Esasları*. İzmir: Nobel Kitabevi, 2008.
- [3] H. S. BAL and F. Budak, "Mikroarray Teknolojisi," *Uludağ Üniversitesi Tıp Fakültesi Derg.*, vol. 38, no. 3, pp. 227–233, 2012.
- [4] Ö. Şimşek, "Mikroarray Teknolojisi ve Diş Hekimliğinde Kullanımı," *Atatürk Üniversitesi Diş Hekim Fakültesi Derg.*, vol. 7, pp. 55–62, 2013.
- [5] K. Shakya, H. J. Ruskin, G. Kerr, M. Crane, and J. Becker, "Comparison of Microarray Preprocessing Methods," Springer New York, 2010, pp. 139–147.
- [6] K. Ipekdal, "Microarray Technology," 2011. [Online]. Available: http://yunus.hacettepe.edu.tr/~mergen/sunu/s_mikroarrayandecology.pdf. [Accessed: 05-Jul-2016].
- [7] H. Liu, I. Bebu, and X. Li, "Microarray probes and probe sets.," *Front Biosci (Elite Ed)*, vol. 2, pp. 325–38, 2010.
- [8] R. Díaz-Uriarte and S. Alvarez de Andrés, "Gene selection and classification of microarray data using random forest.," *BMC Bioinformatics*, vol. 7, p. 3, 2006.
- [9] S. Dudoit, J. Fridlyand, and T. P. Speed, "Comparison of Discrimination Methods for the Classification of Tumors Using Gene Expression Data Comparison of Discrimination Methods for the Classification of Tumors Using Gene Expression Data," *J Am Stat Assoc*, vol. 97, no. July, pp. 77–87, 2002.
- [10] T. S. Furey, N. Cristianini, N. Duffy, D. W. Bednarski, M. Schummer, and D. Haussler, "Support vector machine classification and validation of cancer tissue samples using microarray expression data," *Bioinformatics*, vol. 16, no. 10, pp. 906–914, Oct. 2000.
- [11] Y. Lee and C.-K. Lee, "Classification of multiple cancer types by multicategory support vector machines using gene expression data.," *Bioinformatics*, vol. 19, no. 9, pp. 1132–1139, 2003.
- [12] H. Liu, J. Li, and L. Wong, "Classification and Study on Feature Gene Patterns Selection Expression and Profiles Methods Using Proteomic," *Genome Informatics*, vol. 13, pp. 51–60, 2002.
- [13] A. Statnikov, C. F. Aliferis, I. Tsamardinos, D. Hardin, and S. Levy, "A comprehensive evaluation of multicategory classification methods for microarray gene expression cancer diagnosis," *Bioinformatics*, vol. 21, no. 5, pp. 631–643, 2005.
- [14] J. Khan, J. S. Wei, M. Ringnér, L. H. Saal, M. Ladanyi, F. Westermann, F. Berthold, M. Schwab, C. R. Antonescu, C. Peterson, and P. S. Meltzer, "Classification and diagnostic prediction of cancers using gene expression profiling and artificial neural networks.," *Nat Med*, vol. 7, no. 6, pp. 673–9, 2001.
- [15] M. Ringnér and C. Peterson, "Microarray-based cancer diagnosis with artificial neural networks," *Biotechniques*, vol. 34, no. 3 SUPPL., 2003.
- [16] M. Pirooznia, J. Y. Yang, M. Q. M. Yang, and Y. Deng, "A comparative study of different machine learning methods on microarray gene expression data.," *BMC Genomics*, vol. 9 Suppl 1, p. S13, 2008.
- [17] C. Loganathan and K. V. Girija, "Cancer Classification using Adaptive Neuro Fuzzy Inference System with Runge Kutta Learning," *Int J Comput Appl*, vol. 79, no. 4, 2013.
- [18] K. Anandakumar and M. Punithavalli, "Efficient Cancer Classification using Fast Adaptive Neuro-Fuzzy Inference System (FANFIS) based on Statistical Techniques," *IJACSA) Int J Adv Comput Sci Appl Spec Issue Artif Intell*, pp. 132–137, 2011.
- [19] M. a. Hall and L. a. Smith, "Practical feature subset selection for machine learning," *Comput Sci*, vol. 98, pp. 181–191, 1998.
- [20] J. S. R. Jang, "ANFIS: Adaptive-Network-Based Fuzzy Inference System," *IEEE Trans Syst Man Cybern*, vol. 23, no. 3, pp. 665–685, 1993.
- [21] D. Karaboga and E. Kaya, "Training ANFIS using artificial bee colony algorithm for nonlinear dynamic systems identification," in *2014 22nd Signal Processing and*

- Communications Applications Conference (SIU)*, 2014, pp. 493–496.
- [22] S. Uzundurukan, “Determining and modelling of principal parameters affecting swelling properties of soils,” Suleyman Demirel , 2006.
- [23] J. S. R. Jang and C. T. Sun, “Neuro-Fuzzy Modeling and Control,” *Proc IEEE*, vol. 83, no. 3, pp. 378–406, 1995.
- [24] D. Simon, “Training fuzzy systems with the extended Kalman filter,” *Fuzzy Sets Syst*, vol. 132, no. 2, pp. 189–199, 2002.
- [25] M. Mitchell, *An introduction to genetic algorithms*. MIT press, 1998.
- [26] M. M. Kervan, “Çoklu Sensör Konumlandırma Probleminin Genetik Algoritmalar Ve Gen Havuzu Tabanlı Genetik Agoritmalar İle Çözülmesi,” Hava Harp Okulu Komutanlığı, 2009.
- [27] “Karaciğer Kanseri Mikroarray Gen İfade Profili Veri Seti (Chen-2002).” [Online]. Available: http://bioinformatics.rutgers.edu/Static/Supplements/CompCancer/CDNA/chen-2002/chen-2002_database.txt.
- [28] X. Chen, S. T. Cheung, S. So, S. T. Fan, C. Barry, J. Higgins, K.-M. Lai, J. Ji, S. Dudoit, I. O. L. Ng, M. van de Rijn, D. Botstein, and P. O. Brown, “Gene Expression Patterns in Human Liver Cancers,” *Mol Biol Cell*, vol. 13, no. 6, pp. 1929–1939, 2002.

A novel learning algorithm to estimate the optimum fuselage drag coefficient

Tuğrul Oktay^{1*}, Harun Çelik², Metin Uzun³

27.10.2016 Geliş/Received, 12.01.2017 Kabul/Accepted

doi: 10.16984/saufenbilder.59146

ABSTRACT

In this study, a novel algorithm to estimate the optimum value of the fuselage drag coefficient is designed by integrating the artificial neural network (ANN) which is an artificial intelligent method into the algorithm of simultaneous perturbation stochastic approximation (SPSA) which is a fast method. SPSA converges to the optimum value for solution very fast. However using SPSA alone requires a function of problem to estimate the optimum solution. On the other hand, ANN is able to estimate the solutions for the problem without need of its any objective function. However ANN needs a certain data set to be effectively trained. Also, the best ANN architecture which accomplish with different data sets of problem may alter. Thus, ANN architecture alone is not adequate for estimating the best result for each different data set. The main target of this study is making SPSA able to be applicable for the problem that has not any objective function by using training capability of ANN. For this purpose, initially, ANN is trained by the data of fuselage drag coefficient obtained by previous experimental results conducted in wind tunnel and varies depending on the geometry of fuselage. Thus, ANN becomes capable to estimate the fuselage drag coefficient for each parameter values of the fuselage shape. Therefore, ANN estimates the fuselage drag coefficient with respect to inputs without the requirement of any experimental computations. Note that ANN does not estimate the optimum value as output but estimates the output regarding to the inputs. The ANN is integrated into the SPSA to fulfill the need of cost function for SPSA. More clearly, the new algorithm evaluates ANN to estimate the fuselage drag coefficient with respect to inputs while evaluates SPSA to estimate the optimum inputs for the optimum fuselage drag coefficient. Through integrating the trained ANN into the SPSA, an effective and novel algorithm estimates the fuselage drag coefficient fast and accurately without defining an objective function is improved.

Keywords: Aerial Vehicle, Fuselage Drag Coefficient, Artificial Neural Network, Simultaneous Perturbation Stochastic Approximation

En uygun gövde sürüklenme katsayısı hesabı için yeni bir öğrenme algoritması

ÖZ

Bu çalışmada gövde sürüklenme katsayısının en uygun değerini hesaplamak için yapay zeki bir yöntem olan Yapay Sinir Ağları (YSA), hızlı bir yöntem olan Eşzamanlı Dağılım Rassal Yaklaşım (EDRY) algoritması içerisine yerleştirilerek yeni bir algoritma tasarlanmıştır. EDRY en iyi çözüme oldukça hızlı bir şekilde yakınsayabilmektedir. Ancak tek başına EDRY çözümü bulabilmek için problemin bir fonksiyonuna ihtiyaç duymaktadır. YSA ise problemin bir fonksiyonu olmaksızın da çözümü bulabilmektedir. Ancak YSA'nın iyi eğitilebilmesi için belirli bir veri kümesine ihtiyaç vardır. Ayrıca problemin değişen her veri kümesi için en iyi sonucu veren uygun ağ yapısı ve parametreleri de değişebilmektedir. Bu nedenle YSA yalnız başına kullanılarak tek bir ağ yapısı ile her farklı veri kümesi için en iyi sonuca ulaşmak mümkün değildir. Bu çalışmanın temel amacı YSA'nın öğrenme kabiliyetinden faydalanarak

* Corresponding Author

1 Erciyes University, Faculty of Aeronautics and Astronautics, Aeronautical Engineering, Kayseri – oktay@erciyes.edu.tr

2 Erciyes University, Faculty of Aeronautics and Astronautics, Aircraft Electric-Electronic, Kayseri – haruncelik@erciyes.edu.tr

3 Erciyes University, Faculty of Aeronautics and Astronautics, Aircraft Airframe-Powerplant, Kayseri– metinuzun@erciyes.edu.tr

EDRY'yi bir amaç fonksiyonu olmayan problemler için de kullanılabilir hale getirmektir. Bu amaçla, öncelikle daha önce rüzgâr tüneline yapılmış deneysel çalışmalar neticesinde elde edilen gövde şekillerine göre gövde sürüklenme katsayısının değişimi verileri ile YSA eğitilmiştir. Böylece YSA şekil değerlerine göre gövde sürüklenme katsayısını kendisi tahmin edebilecek yeteneğe gelmiştir. YSA artık herhangi bir deneysel hesaplamaya ihtiyaç duymadan giriş değerlerine göre gövde sürüklenme katsayısını tahmin edebilmektedir. Fakat burada YSA ile en uygun değer değil, her giriş değeri için bir çıkış değeri bulunmaktadır. EDRY'nin her adımda hesaplamaya ihtiyaç duyduğu maliyet fonksiyonu bu şekilde YSA, EDRY'ye gömülerek giderilmiştir. Yani burada tasarlanan yeni algoritma YSA'yı hangi durumlarda gövde sürüklenme katsayısının ne olacağını bulmak için kullanırken EDRY'yi de en iyi katsayının oluşması için en uygun durumların ne olduğunu bulmak için kullanmaktadır. Bu şekilde eğitilen YSA, EDRY'ye gömülerek, gövde sürüklenme katsayısını bir bağıntıya ihtiyaç olmaksızın hızlı ve doğru bir şekilde hesaplayan, başarılı ve yeni bir algoritma geliştirilmiştir.

Anahtar kelimeler: Hava Aracı, Gövde Sürüklenme Katsayısı, Yapay Sinir Ağları, Eşzamanlı Dağılım Rassal Yaklaşım

1. INTRODUCTION

Drag force is very important for aircraft due to not only directly affects fuel consumption but also indicates the payload of aircraft that includes weather instruments. In order to minimize drag force, various researches are conducted. In most of these researches, the optimum value of the aspects of the fuselage such as the shape of tail and nose are estimated, convenient materials described to reduce the aerial vehicle surface roughness, and aerodynamic shape is optimized [1]. Additionally, in order to reduce the drag forces, placing vortex generators on the surface of aircraft that accelerate the transition into turbulent flow is studied [2]. When designing an aircraft, it is crucial to decide the most appropriate shape of aircraft due to its significant effect of the drag force.

Experimentally, by placing the fuselage in the wind tunnel and using a force measuring system, it is possible to calculate the aerodynamic forces that effect on the fuselage. However it is very expensive to calculate the drag coefficient of each designed fuselage separately by examining them in the wind tunnels. Moreover it is not possible to compute the drag coefficient analytically due to its non-linear complex components. Therefore, stochastic approximation based estimation methods are needed to be implemented to estimate fuselage drag coefficient.

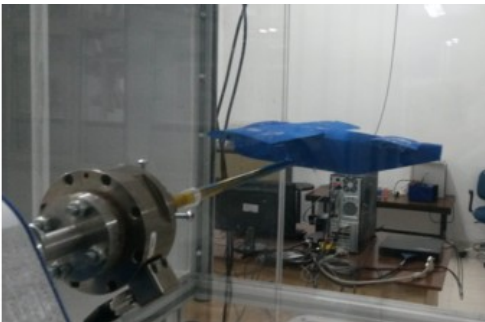


Figure 1. A wind tunnel that provides to calculate the aerodynamic forces

Simultaneous Perturbation Stochastic Approximation (SPSA), one of stochastic approximation algorithm, uses only two observations at each iteration to estimate the solution among random directions for the computation of the gradient. That makes SPSA very attractive and fast optimizer [3]. When compared to computationally expensive algorithms such as fast simulated annealing and genetic algorithms, numerical studies showed that SPSA is more efficient in solving various optimization problems [4]. Also, SPSA is successful in solving constrained optimization problems [5] as seen in this study. In addition, as a stochastic method, SPSA includes an inherent randomness that protects it to stick around a local minimum and provides to converge to the global minimum of the best solution just in a few iterations.

Artificial neural networks [6] are using in solving various engineering problems because of their many advantages such as capability of learning, easily applicable to different problems, capacity for generalization, requirement of less data compared to the traditional estimation techniques, ability to work fast because of parallel structures and flexibility in design. In recent years, ANN is also utilized in solving problems in aeronautics [7], [8].

In previous studies, a few researchers [9], [10] are combined ANN with SPSA, and they used SPSA as a training method of ANN. However, in this paper, SPSA is not applied to train the ANN. On the contrary, ANN is mounted in SPSA algorithm to make SPSA able to estimate the optimum value of fuselage drag coefficient without any equation of objective function. The data of fuselage drag coefficient obtained from experimentally studies are used to train ANN to be capable of estimating the fuselage drag coefficient with respect to new parameter values of fuselage shape. Trained ANN is

integrated into the SPSA, and so a novel stochastic algorithm is improved to estimate the optimum fuselage drag coefficient value without any equation of objective function. Also, while it is possible to evaluate only the ANN to estimate the optimum fuselage drag coefficient value, with the novel algorithm in this study, it is aimed to make SPSA able to estimate the optimum fuselage drag coefficient value without using any numerical computation of cost function. Additionally, using a single ANN architecture may not deal with any data set of the problem. On the other hand, valuating only ANN instead of SPSA requires designing new ANNs for our each different experimental data set which is not logical.

2. FUSELAGE DRAG COEFFICIENT OF AN AERIAL VEHICLE

An object located in air flow is influenced by the aerodynamic forces. The forces acting at the vertical axis are defined as weight and lift, and thrust and drag acting at the horizontal axis. Drag force is defined as the resistance against air flow produced by the object in the air. This force is expressed by:

$$D = C_D \frac{\rho}{2} V^2 S \quad (1)$$

where in C_D is the dimensionless drag coefficient, ρ is air density, V is airspeed of the freestream, and S represents straight wing surface. The air flow speed, angle of attack, wing shape, density and compressibility of the air affect the aerodynamic drag force D directly. The components that affecting dimensionless drag coefficient, C_D , are fuselage, wings, and vertical and horizontal control surfaces [11].

It is possible to measure the aerodynamic forces acting on aircraft numerically and experimentally. Numerically, the aerodynamic forces can be calculated as 2D and 3D for different angles of attack and speed values by a numerical analysis program. Experimentally, the aerodynamic forces acting on an air vehicle or wing profile placed in a wind tunnel can be calculated by a force measuring system. In experimental studies [12], [13] carried out to estimate the fuselage drag coefficient C_{Df} was realized at high Reynolds numbers approximately as:

$$C_{Df} = 0.0003 \left[3 \frac{l_f}{d_f} + 4.5 \sqrt{\frac{d_f}{l_f}} + 21 \left(\frac{d_f}{l_f} \right)^2 \right] \quad (2)$$

where l_f is length of fuselage and d_f is frontal average diameter of fuselage.

3. ARTIFICIAL NEURAL NETWORK INTEGRATED SIMULTANEOUS PERTURBATION STOCHASTIC APPROXIMATION

3.1. Model of Simultaneous Perturbation Stochastic Approximation

In classical SPSA, let $a = 500$ denote the vector includes optimization variables (l_f and d_f data for this paper) and $x_{[k]}$ is the estimate of x at the k th iteration, then

$$x_{[k+1]} = x_{[k]} - a_k g_{[k]} \quad (3)$$

where a_k is a decreasing sequence of positive numbers, and $g_{[k]}$ is the estimate of the objective's gradient at

$x_{[k]}$, calculated using a simultaneous perturbation as introduced above. Let $\Delta_{[k]} \in R^p$ be a vector of p mutually independent mean-zero random variables $\{\Delta_{[k]1}, \Delta_{[k]2}, \dots, \Delta_{[k]p}\}$ satisfying given conditions [14], [15]. Then,

$$g_{[k]} = \left[\frac{\Gamma_+ - \Gamma_-}{2d_k \Delta_{[k]1}} \dots \frac{\Gamma_+ - \Gamma_-}{2d_k \Delta_{[k]p}} \right]^T \quad (4)$$

where Γ_+ and Γ_- are the estimations of the objective evaluated at $x_{[k]} + d_k \Delta_{[k]}$ and $x_{[k]} - d_k \Delta_{[k]}$, respectively. In this study, a novel adaptive SPSA that deal with the constraints which means optimization variables are required to be between lower and upper limits (i.e., $x_{\min} \leq x_i \leq x_{\max}$, ∓ 5 percent for this article) is developed to solve related problems. All the perturbed vector elements, $x_{[k]} + d_k \Delta_{[k]}$ and $x_{[k]} - d_k \Delta_{[k]}$, must also be between the certain lower and upper limits. With these requirements and the guidelines of [3], [16] for the selection of sequences d_k and a_k , d_k is chosen as

$$d_k = \min \left\{ d / k^\alpha, 0.95 \min \left\{ \min \{ \eta_l \}, \min \{ \eta_u \} \right\} \right\} \quad (5)$$

where η_l and η_u are vectors whose components are $(x_{[k]i} - x_{\min_i}) / \Delta_{[k]i}$ for each positive, $\Delta_{[k]i}$ and $(x_{\max_i} - x_{[k]i}) / \Delta_{[k]i}$ for each negative $\Delta_{[k]i}$, respectively. Similarly, a_k is selected as

$$a_k = \min \left\{ a / (S + k)^2, 0.95 \min \left\{ \min \{ \mu_l \}, \min \{ \mu_u \} \right\} \right\} \quad (6)$$

Where μ_l and μ_u are vectors whose components are $(x_{[k]i} - x_{\min_i}) / g_{[k]i}$ for each positive $g_{[k]i}$ and

$(x_{\max_i} - x_{[k]i}) / g_{[k]i}$ for each negative $g_{[k]i}$, respectively.

3.2. Artificial Neural Network Architecture

In this paper, the multi layered perceptron (MLP) is utilized as ANN architectures because of their simple structure [17]. After several trials, the most appropriate network architecture was found as two hidden layers with six and five for first and second hidden layers, respectively. Two inputs are defined. The first one is the length of aerial vehicle fuselage, (l_f), and the second one is frontal average diameter of fuselage, (d_f). ANN estimates the output that is the fuselage drag coefficient while obeying constraints (variance of ∓ 5 percent) on the inputs. By the evaluation of (2), input and output data sets are conducted for 100 different l_f , d_f and C_{Df}

values. Totally, 100 training and testing data are obtained by using (2) which is an approximation of experimental results. Then, the data set is divided as the training: testing ratio that is 80: 20. Because of its fast learning and good convergence capabilities, Levenberg-Marquardt [18], [19] algorithm is underlined to train the MLP. The transfer function is experimentally selected as the linear transfer functions for the input and output layers and the hyperbolic tangent sigmoid functions for the hidden layers to obtain better testing performance. In order to design the most suitable network architecture the number of the neurons for the hidden layers is iterated with combinations of 2–10 neurons in each layer. The iterations are performed to minimize the mean square error function. The best results are the combination of double hidden layers including 6 and 5 neurons, respectively. The epoch number is selected as 300 since any improvement is not seen above this value. After training and testing the ANN, observed results are proposed in Table 1.

Table 1. ANN results after training and testing

Training Algorithm	Neurons in Hidden Layers	Transfer Function for Hidden Layers	Epoch	Training Percentage	Testing Percentage
LM	6-5	Sigmoid	300	92,1	90,2

Although (2) is evaluated to obtain data sets to train the ANN, note that using this equation is not necessary or compulsory. The ANN can be trained by experimental results. Actually, our purpose for using (2) is that it is already defined after experimental measurements. (2) is just used as a preliminary study. Here it is important to note that (2) is just an approximation defined from experiments. The fuselage drag coefficient data is extracted from (2) for the initial design of our novel algorithm. However, after using (2) for this initial design, the algorithm will be capable of and applicable for our experimental data which is conducting in our wind tunnel laboratory. Much bigger data sets conducting at our wind tunnel laboratory will be used in future studies of this algorithm. Also note that, the decided ANN architecture and parameters are compatible for different sized and parameterized fuselage drag coefficient. More clearly, parameters such as number of hidden layers and epoch are selected to be capable of more complex fuselage drag coefficient data sets. Additionally, the values of ANN parameters that not given in this section are selected as their default values.

3.3. Proposed Algorithm

Normally, a classical SPSA algorithm estimates the optimum value of cost function by using the computation of two neighbors of cost function. However, our algorithm estimates the two neighbors via ANN regardless of computation. Thus, our algorithm does not need any equation of cost function. All part of this algorithm is established by operating MATLAB.

The following novel algorithm is generated by using the ANN integrated SPSA that is improved by handling with inequality constraints on the design parameters.

The main purpose of this algorithm is to estimate the optimum fuselage drag coefficient with respect to the length of aerial vehicle fuselage and frontal average diameter of fuselage. The other parameters of SPSA such as d , a , λ , Θ , S are chosen using guidelines provided in [3], [16].

Step 1. Train ANN.

Step 2. Set initial values d , a , λ , Θ , S and $k = 1$ and $x = x_{[k]}$.

Step 3. Estimate the objective value by using trained ANN within the constrains for $x_{[k]}$ input parameter

Step 4. Perturb $x_{[k]}$ to $x_{[k]} + d_k \Delta_{[k]}$ and $x_{[k]} - d_k \Delta_{[k]}$, and estimate new objective values for these perturbed values by using ANN to obtain Γ_+ and Γ_- , respectively.

Then, compute the approximate gradient, $g_{[k]}$, using (4) with d_k given by (5).

Step 5. Compute a_k given by (6). If $\|a_k g_{[k]}\| < \delta x$ (where δx is the minimum allowed variation of x) or $k+1$ is greater than the maximum number of iterations allowed, exit, else calculate the next estimate of $x_{[k+1]}$ using (3), set $k = k + 1$, and return to step 2.

Step 6. Show estimation results.

Although the SPSA is a fast converging optimization method, it has not the ability of converging to the optimum fuselage drag coefficient value without the neighbor values of the optimum. Therefore, the cost functions are supposed to be computed for SPSA. However, in this article, regardless of computing the cost functions for each evaluation, ANN is applied to estimate them for SPSA. This means that ANN estimates the new fuselage drag coefficients from given length and diameter of fuselage for SPSA estimating the optimum fuselage drag coefficient with respect to the optimum length and diameter of fuselage.

4. RESULTS

Using SPSA parameters, $S = 5$, $\lambda = 0.602$, $a = 500$, $d = 20$ and $\Theta = 0.101$, as depicted in Figure 2, the optimum value of fuselage drag coefficient, C_{Df} , is achieved by very fast convergence and accuracy of the algorithm which means the proposed ANN integrated SPSA algorithm is capable for estimating the fuselage drag coefficient effectively and properly.

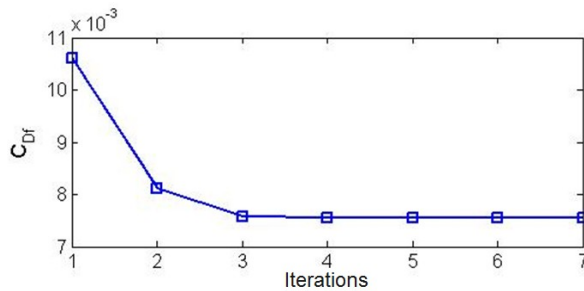


Figure 2. Optimization of fuselage drag coefficient by proposed algorithm.

Although the optimum value of fuselage drag coefficient is estimated using only two variables (l_f and d_f) in this paper, it is seen that the proposed algorithm has the

capacity of solving optimization problems such as drag estimation using various variables at the same time due to its two-stepped evolutionary structure. The algorithm is very effective in decreasing the value of the objective function in the first several iterations. Also it is very fast as it is not computing objective function at the each step. After training ANN one time, the cost function can be estimated by ANN for each step.

5. CONCLUSIONS

In this article, a novel ANN integrated SPSA algorithm is proposed. The advantage of ANN which is an artificial intelligent technique and SPSA that has fast convergence capability are combined to estimate fuselage drag coefficient which takes a significant part in aerial vehicle design. Initially, the data set for training and testing ANN is obtained using the results of previous experimental studies in wind tunnels. ANN is trained using the data includes experimental results of fuselage drag coefficient with respect to fuselage shape in order to learn to estimate the new fuselage drag coefficients from given length and diameter of fuselage. Finally, trained ANN is integrated into classical SPSA to generate adaptive SPSA which is able to estimate the objective by two observations for approximations of each component. With ANN, it is possible to use experimental results instead of defining any equations by them.

Although the proposed algorithm is evaluated to estimate the optimum values of fuselage drag coefficient using only two design parameter, trials are proven that our algorithm is capable of estimating the objective using various design parameters at same time as well as solving optimization problems that include various variables, accurately and fast. Using ANN provides the algorithm to work without an equation of cost function. Thus, it is decided to implement this algorithm to solve more complex optimization problems in further studies.

REFERENCES

- [1] Z. Tang and J. Périaux, "Uncertainty based robust optimization method for drag minimization problems in aerodynamics," *Computer Methods in Applied Mechanics and Eng.*, vol. 217, pp. 12-24, 2012.
- [2] S. Sarada, M. Shivashankar, and G. Rudresh, "Numerical Simulation of Viscous, Incompressible Flow around NACA 64618 Subsonic Airfoil Using Computational Fluid Dynamics," *Advances in Mechanical Engineering*, vol. 256, 2010.
- [3] J. C. Spall, "Multivariate stochastic approximation using a simultaneous perturbation

- gradient approximation,” IEEE Trans. Autom. Control, vol. 37, no. 3, pp. 332–341, 1992.
- [4] J. L. Maryak and D. C. Chin, “Global random optimization by simultaneous perturbation stochastic approximation,” American Control Conference, IEEE Proceedings of the 2001, vol. 2, 2001.
- [5] I. J. Wang and J. C. Spall, “Stochastic optimization with inequality constraints using simultaneous perturbations and penalty functions,” 42nd IEEE Conf. on Decision and Control, Proceedings of the 2003, vol. 4, pp. 3808–3813, 2003.
- [6] S. Haykin, “Neural networks: A Comprehensive Foundation,” New York: Macmillan College Publishing Company, 1994.
- [7] T. Rajkumar and J. Bardina, “Prediction of Aerodynamic Coefficients Using Neural Network for Sparse Data,” Proc. of FLAIRS, Florida, USA, 2002.
- [8] M. C. dos Santos, B. S. de Mattos, and R. da Mota Girardi, “Aerodynamic Coefficient Prediction of Aircraft Using Neural Network,” 19th International Congress of Mechanical Engineering, November 5-9, Brasília, DF, 2007.
- [9] Q. Song, J. C. Spall, Y. C. Soh, and J. Ni, “Robust neural network tracking controller using simultaneous perturbation stochastic approximation,” Neural Networks, IEEE Trans. on, vol. 19, no. 5, pp. 817–835, 2008.
- [10] Y. Y. Hong, H. L. Chang, and C. S. Chiu, “Hour-ahead wind power and speed forecasting using simultaneous perturbation stochastic approximation (SPSA) algorithm and neural network with fuzzy inputs,” Energy, vol. 35, no. 9, pp. 3870–3876, 2010.
- [11] P. J. Boschetti, E. M. Cárdenas, and A. Amerio, “Aerodynamic Optimization of an UAV Design,” AIAA Paper, 7399, 2005.
- [12] S. F. Hoerner, “Résistance á L’avancement dans les Fluides,” edited by Gauthier Villars Editeurs, Paris, France, Chapter XIV, 1965.
- [13] F. Zeidan, “Estudio Teórico-Practico de la Resistencia al Avance de una Aeronave,” Aerodinámica y Práctica Avanzada, edited by Consejo de Publicaciones de la Universidad de los Andes, Mérida, Venezuela, pp. 89–96, 1995.
- [14] Y. He, M. C. Fu, and S. I. Marcus, “Convergence of Simultaneous Perturbation Stochastic Approximation for Non-Differentiable Optimization,” IEEE Trans. on Aerospace and Electronic Systems, vol. 48, no. 8, pp. 1459–1463, 2003.
- [15] P. Sadegh and J. C. Spall, “Optimal Random Perturbations for Multivariable Stochastic Approximation Using a Simultaneous Perturbation Gradient Approximation,” IEEE Trans. on Automatic Control, vol. 43, no. 10, pp. 1480–1484, 1998.
- [16] T. Oktay and C. Sultan, “Constrained predictive control of helicopters,” Aircraft Eng. and Aerospace Technology, vol. 85, no. 1, pp. 32–47, 2013.
- [17] I. Turkmen and H. Celik, “Incorporation of Neural Network to HPMHT for Tracking Multiple Targets,” Elektronika ir Elektrotechnika, vol. 21, no. 4, pp. 3–6, 2015.
- [18] K. Levenberg, “A method for the solution of certain nonlinear problems in least squares,” Quart Appl Math, vol. 2, pp. 164–168, 1944.
- [19] D. W. Marquardt, “An algorithm for least-squares estimation of nonlinear parameters,” J Soc Ind Appl Math, vol. 11, pp. 431–441, 1963.

Dış ortam kablosuz networklerde uygun sinyal dağılımının görünürlük analizi yöntemiyle belirlenmesi

Mustafa Coşar^{1*}, Fazlı Engin Tombuş², İbrahim Murat Ozulu³, Veli İlçi⁴

14.07.2016 Geliş/Received, 15.01.2017 Kabul/Accepted

doi: 10.16984/saufenbilder.24070

ÖZ

Üniversite yerleşkelerinde kablosuz ağlar son yıllarda büyük önem kazanmıştır. Bu ağların üniversite yerleşkeleri gibi büyük bölgelerde planlanması, tasarlanması ve kurulması sırasında birçok problemle karşı karşıya kalınmaktadır. Bu problemlerden bazıları, yerleşkelerin fiziki özellikleri, uygun cihaz konfigürasyonu ve gerekli olan cihazların adedinin belirlenmesidir. Özel firmalar proje için kendi yöntemleri ile marka, model, maliyet ve adet belirlemektedirler. Ancak çıkan sonuç her zaman kurum açısından pekte uygun olmamaktadır. Bu nedenle yerleşkelere uygun sinyal dağılımına sahip planlama ve tasarımın yapılabilmesi birçok sorunu çözmekle birlikte kullanımı ve memnuniyeti artırıcı etkiye sahip olacağı düşünülmektedir. Bu çalışmada, Hitit Üniversitesi Kuzey Yerleşkesinde kurulu olan bir adet dış ortam erişim noktasından (Access Point-AP) elde edilen sinyal dağılımına göre coğrafi bilgi sistemleri görünürlük analizi yardımıyla tüm alan için kestirim yapılarak uygun sinyal dağılımı belirlenmeye çalışılmıştır. Çalışma sonunda 56 hektarlık bir bölgeye toplam 9 adet dış ortam AP ile uygun sinyal dağılımı elde edilmiştir.

Anahtar Kelimeler: kablosuz ağ, Wi-Fi, sinyal dağılımı, arazi modellemesi, görünürlük analizi

Determination of outdoor signal propagation via visibility analysis in outdoor wireless networks

ABSTRACT

Wireless networks on university campuses has gained importance in recent years. These networks in major areas such as university campuses, are faced with many problems during the planning, design and establishment. These problems are among the first that comes to mind, the physical properties of the campus and is selected according to the characteristics of network equipment. There is no doubt at all points of a wireless network set up in order to provide uninterrupted service and quality of the signal is expected to be good. However, it should be understood literally cannot meet these expectations. Therefore, to solve many problems to campus planning and design can be made to have acceptable signal distribution will have the appropriate use of and satisfaction with increasing effect. In this study, due to the start of construction on the North Campus of Hitit University, wireless signal spread using the current spread has been determined with the help of geographic information systems visibility analysis. An area of 56 hectares, with the total of 9 AP the acceptable signal distribution was obtained.

Keywords: wireless networks, Wi-Fi, signal distribution, terrain modelling, visibility analysis

* Sorumlu Yazar / Corresponding Author

¹ Hitit Üniversitesi, Mühendislik Fakültesi, Bilgisayar Mühendisliği Bölümü, Çorum – mustafacosar@hitit.edu.tr

² Hitit Üniversitesi, Teknik Bilimler Meslek Yüksekokulu, Harita ve Kadastro Programı, Çorum – fengintombus@hitit.edu.tr

³ Hitit Üniversitesi, Teknik Bilimler Meslek Yüksekokulu, Harita ve Kadastro Programı, Çorum – imuratozulu@hitit.edu.tr

⁴ Hitit Üniversitesi, Teknik Bilimler Meslek Yüksekokulu, Harita ve Kadastro Programı, Çorum – veliilci@hitit.edu.tr

1. GİRİŞ (INTRODUCTION)

Kablosuz ağ sistemlerinin kurulumunda, uygun yatırım maliyetleri ile kaliteli bir sinyal dağılımı yaparak daha geniş alanlara ve daha çok kullanıcıya hizmet vermek arzulan bir durum haline gelmiştir. Bu nedenle planlama ve tasarım aşamalarında mümkün olduğu kadar farklı bilim dallarından destek alarak karar vermek ve uygulama yapmak gerekmektedir.

Ağ projelerinin planlama aşamasında bazı özel bilişim firmalarından bilgi ve destek alınmaktadır. Bu firmalar kendilerinin geliştirdiği bazı yöntemler ile cihaz konfigürasyonu, adet, yerleşim planı ve proje bütçesi çıkarabilmektedirler. Ancak bu tür çalışmalar kimi zaman gerçeği yansıtamamaktadır. Kurumlar proje sonunda gereksiz mali harcamalar yapabildiği gibi işlerine yaramayacak birçok donanım ve yazılım sistemini de almak zorunda kalabilmektedirler. Bunu en aza indirmenin yolu ise yerleşmeye uygun projenin ihtiyaç analizinin doğru bir şekilde yapılarak ardından alım ve kurulum işlemlerinin yapılmasından geçmektedir.

Bu ihtiyaç analizinin ilk aşamasında yerleşkenin büyüklüğü, sınırları, arazi yapısı, yapılan ve planlanan bina yerleşimleri gibi coğrafi faktörleri göz önünde tutmak gerekmektedir. Bu ve buna benzer faktörlerin oluşturduğu problemleri bilimsel bir yaklaşımla çözebilmenin bir yolu da Coğrafi Bilgi Sistemlerinden geçmektedir. Son yirmi yıldır birçok projede kullanılan Coğrafi Bilgi Sistemleri (CBS) destek alınabilecek alanlar arasında yerini almış durumdadır [1]. CBS, disiplinler arası yaygın bir şekilde kullanılan, sayısal coğrafi verilerin bilgisayar ortamında değerlendirilmesine imkân sağlayan bir bilim dalıdır. Sayısal ya da tablosal verilerle amaca uygun alternatifli görsel sonuçlar elde edilir [1,6].

Görünürlük Analizi, CBS’de önemli bir mekânsal analiz yöntemidir. Bu yöntemin kullanım amacı bir veya birkaç görüş noktası oluşturarak arazinin görüş alanının belirlenmesidir. Bu yöntem, kablosuz (wireless) ağların ve televizyon sistemlerinin kulelerinin yer seçiminde yüksek doğruluğu garanti ederek yaygın bir şekilde kullanılmaktadır [2].

Chamberlaina ve Meitner’e [3] göre, Görünürlük Analizi, CBS’nin en yaygın olarak kullanılan fonksiyonlarından biridir. Bu fonksiyon, rüzgâr türbinlerinin yerleşimi, iletişim sistemlerinin planlanması ve arazinin görsel özelliklerini araştırarak kentsel tasarım yapmak gibi çeşitli uygulamalarda kullanılmaktadır.

Uzaktan algılama (UA) cisimler ile fiziksel bir temas olmaksızın yeryüzüne ilişkin verilerin toplanması ve kayıt altına alınmasıdır. Çeşitli algılama, tarama ve kayıt cihazları ile gerçekleştirilen UA işlemi için temel gereksinim enerjidir. Enerji kaynağı olarak doğal kaynak olan güneş veya yapay bir güç kaynağı kullanılmaktadır. Cisimlerin gelen enerji ile etkileşimi sonucunda bilgi alma işlemidir. Bu yöntemle cisimlere ait bilgiler hızlı ve kolaylıkla elde edilebilmektedir [4,5].

Son yıllarda birçok planlama çalışmasında toplanan konumsal verilerin değerlendirilmesinde yaygın bir şekilde CBS ve UA teknikleri kullanılmaktadır [1]. CBS ve UA konumsal veri yönetimi, analizi ve tematik haritalama özellikleri ile öne çıkmaktadır. Planlama çalışmaları için gerekli olan haritalama ihtiyaçları yukarıda bahsedilen yöntemler sayesinde daha hızlı ve verimli bir şekilde elde edilebilmektedir.

Özellikle son 10 yıl içerisinde kablosuz ağların ve cihazların dağılımı ve konumlarının planlanması üzerine yapılan çalışmalarda CBS analiz araçları etkin şekilde kullanılmaktadır. Yapılan çalışmalarda büyük yerleşkeler veya şehir içerisindeki sinyal şiddet dağılımları ölçülmüş, elde edilen sonuçlar doğrultusunda kablosuz ağların tasarımı ve kapsama alanlarının geliştirilmesi konuları ön plana çıkarılmıştır [7,8,9].

Görünürlük analizi (GA), kentsel ve kentsel olmayan bölgelerde gözlemsel sensör sistemlerinin modellenmesinde 3D ve Sayısal Yüzey Modelleme (Digital Surface Model) olarak iki yöntem kullanılmaktadır. Bu modelleme sırasında binalar, inşaat sistemleri ve ağaçlar gibi mekânsal özellikler beraber düşünülmektedir. Analiz sonucunda sensör ağları için farklı gözlem noktalarından bölgenin görünen ve görünmeyen parçaları ortaya koyan raster haritası oluşturulur [10].

Bu ve buna benzer örnek çalışmalar ışığında, görünürlük analizi yardımıyla üniversite kampüsleri gibi büyük arazilere sahip yerleşkelerde dış ortam Wi-Fi ağlarının tasarımını yapmak amacıyla görünürlük analizinden yararlanılabileceği fikri ön plana çıkmaktadır.

Bu çalışmada, çalışma alanı olarak, Hitit Üniversitesi’nin yaklaşık 56 hektar olan kuzey yerleşkesi seçilmiştir. Kurulu olan erişim noktasının (Access Point-AP) sinyal kapasitesinin ölçümünde hata oranını en aza indirmek amacıyla yaygın olarak kullanılan 2 adet yazılımın yanında, araştırmacılar tarafından mobil cihazlar üzerinden çalışabilen Android tabanlı bir bilgisayar yazılımı daha geliştirilmiştir. Bu yazılımlar sayesinde 870 noktadan yapılan sinyal ölçümü ile AP’nin kapsama alanı belirlenmiştir. Ardından diğer noktaların uygun sinyal dağılımına sahip olması için harita

üzerinden işaretlenen yerlerde görünürlük analizleri yapılarak AP'lerin kurulabilecek yerleri ve sinyal dağılımları belirlenmeye çalışılmıştır.

Çalışmanın sonunda 9 adet dış ortam AP yardımı ile bu yerleşkenin uygun sinyal dağılımına sahip olarak planlanabileceği belirlenmiştir. Çalışmanın bu kapsamda kablosuz ağlara yatırım yapacak kurumlara rehberlik edebileceği düşünülmektedir.

2. METOT (METHOD)

2.1. Materyal (Materials)

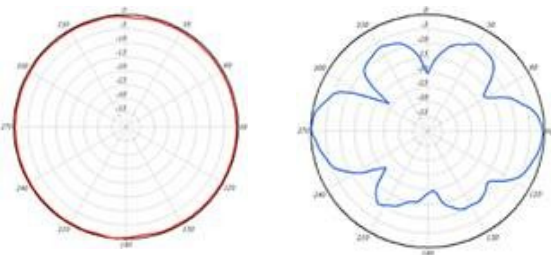
Çalışmada veri toplama aracı olarak, bir adet kurulu olan ve çalışan dış ortam AP ve 3 farklı sinyal ölçüm yazılımı kullanılmıştır. Toplanan veriler öncelikle MS Excel ortamına aktarılmıştır. Ardından veriler ArcGIS yazılımı ile değerlendirilmeye çalışılmıştır.

2.1.1. Kablosuz Erişim Noktası (Wireless Access Point)

Kuzey yerleşkesinde kullanılan dış ortam AP Cisco AIR-CAP1552E-E-K9 modelidir. Bu cihazın üretici firma kataloğundan alınan bazı teknik özellikleri ve sinyal dağılım grafikleri aşağıdaki tabloda ve Şekil 1'de verilmiştir.

Tablo 1. Cisco AIR-CAP1552E-E-K9 model AP'nin bazı özellikleri (Cisco AIR-CAP1552E-E-K9 model AP's same properties) [11]

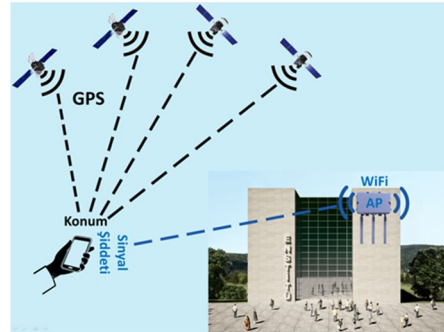
Özellik	Açıklama
Frekans	2.4 GHz - 5 GHz
IEEE Standardı	IEEE 802.11a/b/g/n,
Arayüz	10/100/1000BASE-T
Kullanıcı sayısı	32 ve üzeri
Sinyal aralığı	-69 db / -93 db (horizontal beamwidth 360°)
Data rate	300 Mbit/s
Kanal	Toplam 8 kanaldan 300 Mbit/s
Çalışma ortamı	-40 to 55°C
Mesafe	500-1000 ft. (152.4-304.8 m)
Kurulum yüksekliği	Max. 35 ft (10.67m)



Şekil 1. Cisco AIR-CAP1552E dış ortam AP sinyal dağılımı (Cisco AIR-CAP1552E-E-K9 model AP's signal propagation) [11]

2.1.2. Sinyal Ölçüm Programı (Signal Measurement Program)

Mobil ortamlardan Wi-Fi ağlarının bulunmasını ve özelliklerinin listelenmesini sağlayan birçok bilgisayar programı bulunmaktadır. Bunlardan en yaygın kullanılanları arasında WigleWifi ve Vistumbler örnek gösterilebilir. Bu uygulamada hem bu yazılımlar hem de araştırmacılar tarafından geliştirilen Hwfinder adında bir program kullanılmıştır. Bu program java kodlama dili yardımıyla android ortamında geliştirilmiştir.



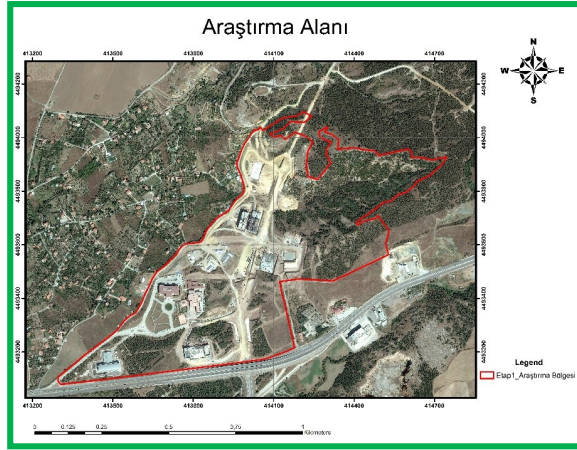
Şekil 2. Sinyal ölçüm programı işlem süreci (Process of signal measurement program)

Şekil 2'de gösterildiği gibi söz konusu programla dış ortamdaki mobil cihazın konum bilgisi GPS sisteminden alınabilmekte ve eşzamanlı olarak dış ortam AP'sinden ortama yayılan sinyal şiddeti bilgisi de okunabilmektedir. Böylece mobil cihazın ve AP'nin konumu bilindiğinden ikisi arasındaki Öklid mesafesi konum farkından hesaplanabilmekte ve mesafe ile sinyal şiddeti arasındaki bağıntı belirlenebilmektedir.

2.1.3. Arazi Modellemesi (Terrain Modelling)

Kuzey yerleşke arazisinin haritalarının hazırlanması, sayısallaştırılması ve modellemesi için bilgisayar ortamında yapılan işlemler arazi modellemesi olarak adlandırılmıştır.

Bu çalışmada, ilk olarak uzaktan algılama tekniği ile elde edilmiş olan 2015 yılına ait WorldView-3 uydu görüntüsü üzerinde Hitit Üniversitesi Kuzey yerleşkesi gelişim planında 1.etap olarak isimlendirilen alan çalışma bölgesi olarak belirlenmiştir. Şekil 3'te gösterilen bu bölge yaklaşık olarak 56 hektarlık (556191 m²) bir alana sahiptir.



Şekil 3. Uygulama bölgesinin sınırları (The boundaries of the application area)

Çalışma alanı Çorum iline ait 1/1000 ölçekli G33C20C1A, G33C20C1B, G33C20C1C, G33C20C1D dört pafta sınırı içerisinde kalmaktadır. Çalışma bölgesine ait uydu görüntüsü ve sayısal verilerin değerlendirilmesinde ArcGIS v.10 paket programı kullanılmıştır.

2.2. Çalışma Planı (Work Plan)

Üniversitenin daha önce hazırlamış olduğu yerleşke planı çerçevesinde 1.etap olarak adlandırılan arazinin üzerinde aşağıdaki işlemler yapılmıştır.

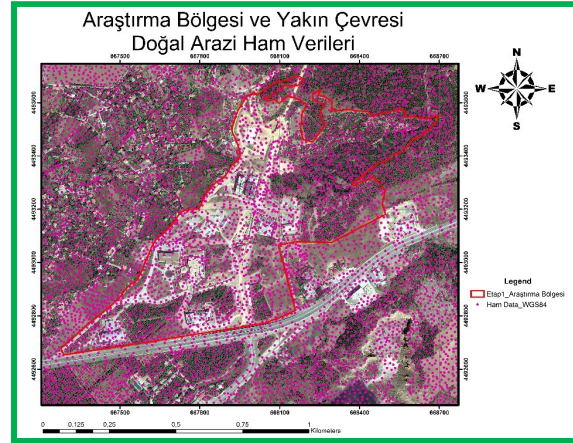
- Sınırlarının belirlenmesi ve çalışma bölgesinin 2 boyutlu haritasının hazırlanması,
- 3 boyutlu modelleme ile yapımı düşünülen binaların arazi üzerinde sayısallaştırılarak gösterimi ve özelliklerinin tanımlanması,
- Sinyal ölçme yazılımı yardımıyla referans dış ortam AP'den alınan sinyal dağılımının belirlenmesi.
- Belirlenen sinyal dağılımına göre diğer bina ve alanlara CBS-GA yardımıyla tahmini uygun sinyal dağılımının gösterilmesi.

Bu aşamalar sonucunda referans AP'den belirlenen mevcut durum ve gelecekte olması planlanan durumun bir ön uygulaması yapılmıştır.

Farklı referans sistemlerinde mekânsal bilgiye sahip sayısal veriler için ortak bir referans sistemi belirlenmiştir. Çalışmada kullanılan tüm veriler için projeksiyon sistemi olarak WGS_1984_UTM_Zone_36N seçilmiş [12], farklı referansa sahip sayısal bilgiler ise bu sisteme dönüştürülmüştür.

Çalışma bölgesinin topografik durumunun ortaya konulması ve bölgenin 3D modelinin oluşturulabilmesi için çalışma bölgesi ve yakın çevresinin doğal arazi

durumu ham bilgileri ArcGIS ortamında oluşturulan veri tabanına alınmış ve Şekil 4'te gösterilmiştir.



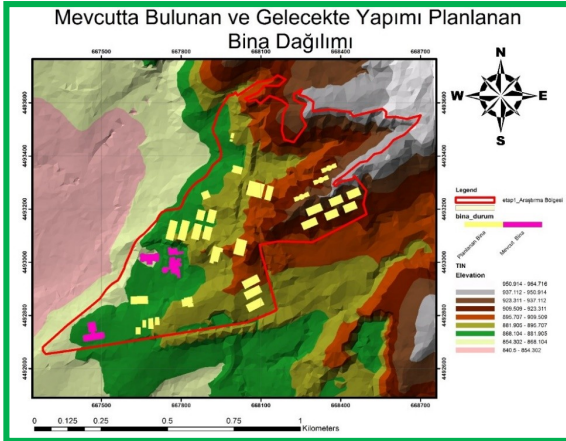
Şekil 4. Uygulama bölgesinin ve yakın çevresinin doğal arazi ham bilgileri. (Unprocessed information of the surrounding and the application area)

Kablosuz sinyal dağılımlarının doğal arazideki karakteristiğinin belirlenmesi ve CBS ortamında görünürlük analizlerinin yapılabilmesi için sayısal yükseklik modellerinin oluşturulması gerekmektedir. Sayısal yükseklik modelleri harita bilimi başta olmak üzere çok farklı disiplinlerde kullanılmaktadır [13].

Sayısal yükseklik modelleri sayısal anlamda nokta ve çizgi elemanları ile arazi yüzeyinin geometrik gösterimini sağlanmaktadır. Arazi yüzeyinin, sert veya yumuşak kırıklı çizgilerle ve noktalarla gösterilerek topografik detaylarla tamamlanan, farklı özelliklerdeki noktalarla temsil edilmesidir [14,15].

Sayısal yükseklik modelleri farklı tiplerde ve özelliklerde oluşturulabilmektedir. Literatürde yaygın olarak kullanılan şekillerine Sayısal Yükseklik Modeli (DEM - Digital Elevation Model) ve Düzensiz Üçgen Ağları (TIN-Triangulated Irregular Network) örnek verilebilir [14,15].

Bu amaçla Şekil 4'te görülmekte olan ham veriler kullanılarak çalışma bölgesinin doğal yapısına ait yükseklik değerleri Şekil 5'deki TIN yapıdaki sayısal yükseklik modeline dönüştürülmüştür.



Şekil 5. Uygulama bölgesinin TIN yapıdaki modeli (TIN model of the application area)

Şekil 5’de TIN üzerinde araştırma bölgesi olarak belirlenen Hitit Üniversitesi Kuzey Kampüsü mevcut sınırları, hali hazır durumda bulunan binalar pembe renkle, yerleşke master planı kapsamında yapımı planlanan binalar ise sarı renkle gösterilmiştir.

Çalışmanın bir sonraki bölümünde sayısal yükseklik modeli üzerine işlenmiş uydu görüntüsü, araştırma alanı sınırı ve mevcut ve yapımı planlanan binaların üç boyutlu görüntüsü ArcScene ortamında hazırlanarak Şekil 6’da gösterilmiştir.



Şekil 6. Uygulama bölgesinin ArcScene ortamındaki 3D görünümü (3D view of the ArcScene environment in the application area)

Mevcut binalardan olan Mühendislik Fakültesi binası dış yüzeyinde bulunan dış ortam AP’den yapılan örnek bir ölçümden alınan koordinat bilgileri, sinyal ölçüm değerleri Şekil 7’de gösterilmiştir.

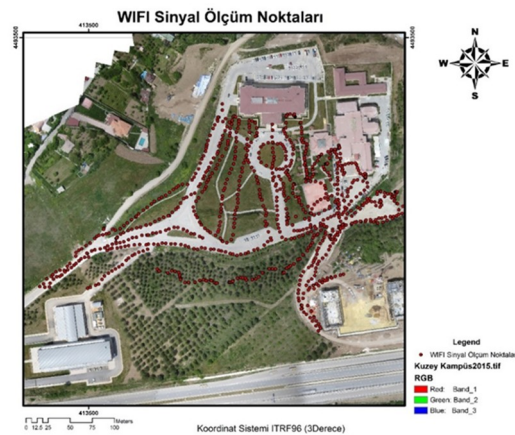


Şekil 7. Dış ortam AP’den alınan örnek koordinat ve sinyal bilgileri (Sample coordinate and signal information received from the outdoor AP)

Ölçümlerin yapıldığı AP’nin konumu mühendislik fakültesi kantininin dış yüzeyine çevre şartlarından etkilenmemesi amacıyla ve üretici tarafından tavsiye edilen yerden yükseklik değeri göz önüne alınarak yaklaşık 7 metre olarak monte edilmiştir. Bu yükseklik master planda yapımı devam eden binalarda da referans olarak alınarak görünürlük analizinde kullanılan AP yerleşim yüksekliği olarak kabul edilmiştir. Böylece AP’nin yerden olan yüksekliğinin sinyal dağılıma olan etkisi en aza indirilmeye çalışılmıştır.

3. BULGULAR (RESULTS)

Aşağıdaki şekilde referans AP’nin sinyal dağılımını belirlemek amacıyla yapılan ölçüm değerlerinin bulunduğu noktalar görülmektedir.



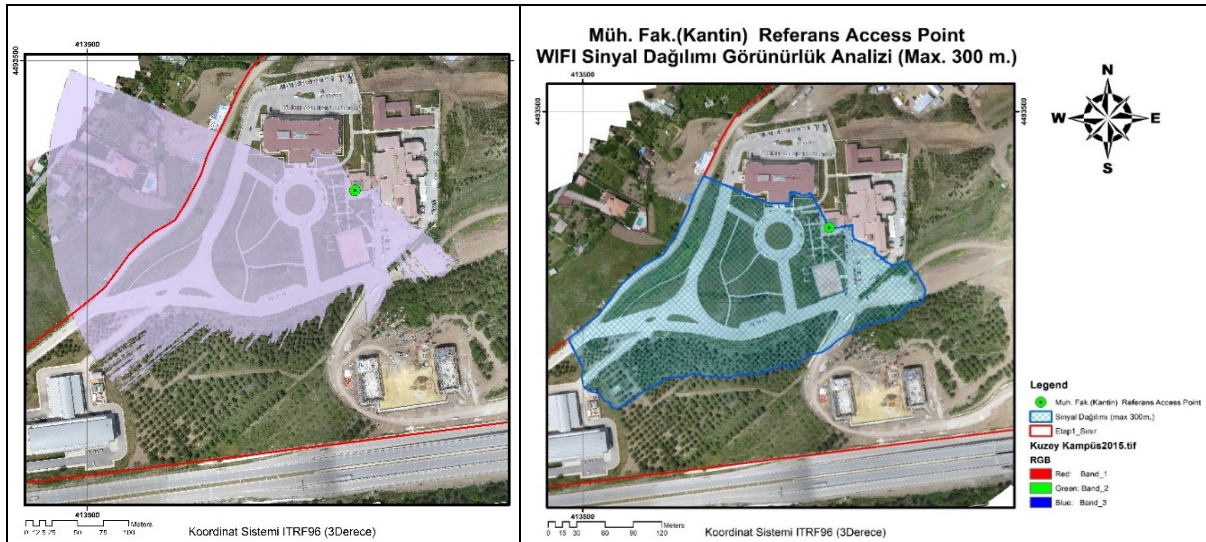
Şekil 8. Wi-Fi sinyal ölçüm noktaları (Wi-Fi signal measurement points)

Bu uygulama için toplam 870 noktadan referans AP’den yayılan sinyalin ölçümü yapılarak değerler sayısallaştırılmış ve nokta içerisine kaydedilmiştir. Bu bilgiler içinde çalışmada kullanılmak üzere,

- Noktanın sayısal koordinatları
- Noktanın AP’ye olan uzaklığı

- Noktadan alınan sinyalin desibel (dB) cinsinden gücü gibi bilgiler toplanmıştır.

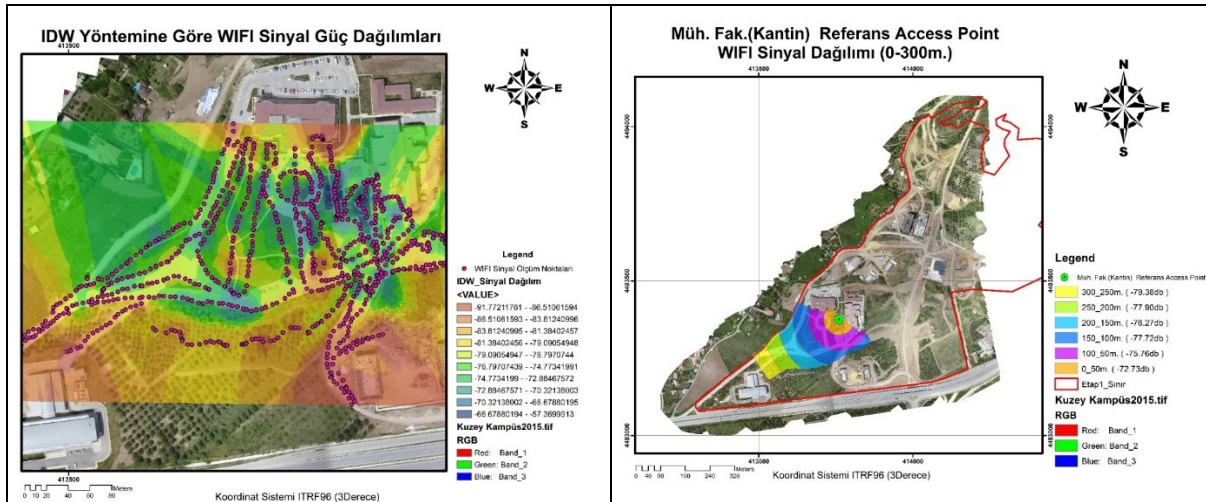
Ölçüm sonucunda elde edilen veriler değerlendirilmek üzere Ms excel ve ArcGIS ortamına taşınarak yorumlanmaya çalışılmıştır.



Şekil 9. Referans AP'nin sinyal kapsamı alanı (The signal coverage of the reference AP)

Bu işlemin ardından referans alınan AP'nin sinyal dağılımından yapılan görünürlük analizi ile elde edilen ve 0-300 metre mesafedeki sinyal yoğunluk haritası Şekil 10'da gösterildiği gibi oluşturulmaya çalışılmıştır. Ardından 56 hektarlık 1.etap yerleşke alanının master

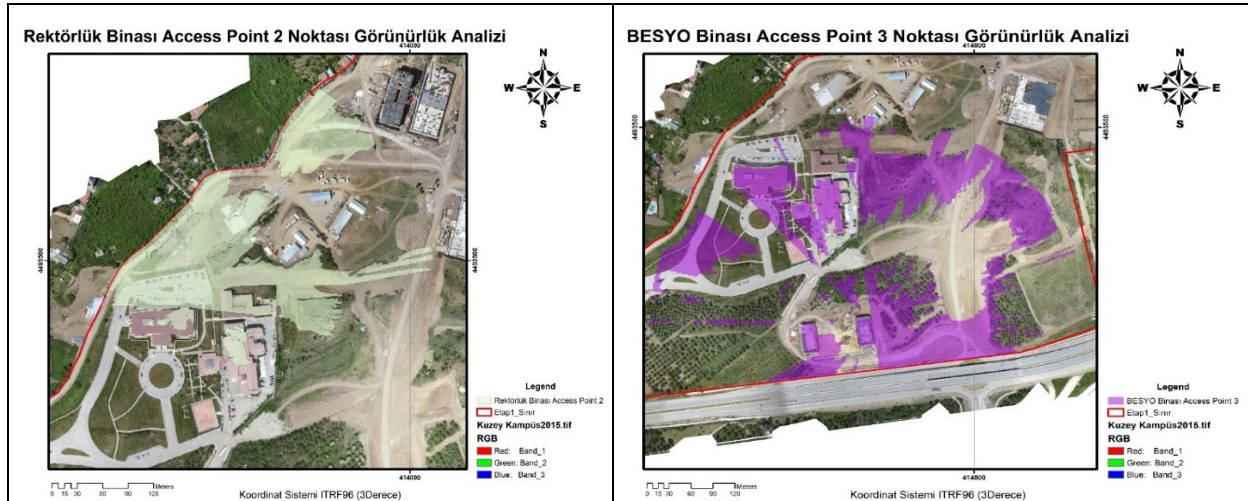
plan çerçevesinde yapımı devam eden binalar düşünülerek bu binaların üzerine veya yakınına takılması ihtimal AP'lerin sinyal yoğunluğunun görünürlük analizleri yapılmaya çalışılmıştır.



Şekil 10. Ölçüm noktaları ve sinyal dağılımı (Measurement points and signal distribution)

Bu amaçla ilk olarak Şekil 10'da gösterilen rektörlük binasının arkasına ve BESYO binasının yan kısmına yayın yapması için düşünülen AP'lerin görünürlük

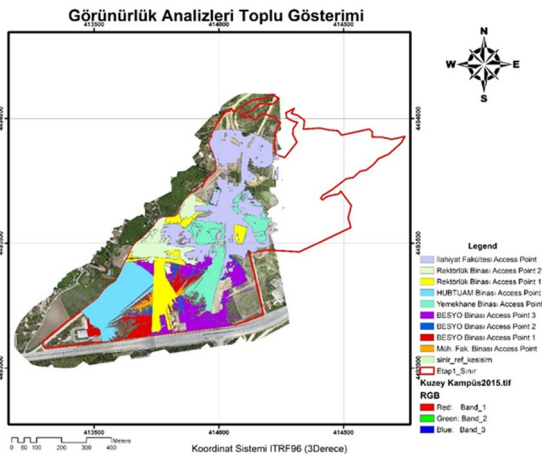
analizi yapılmış ve aşağıdaki Şekil 11'de görülen dağılım elde edilmiştir.



Şekil 11. Rektörlük ve BESYO binalarından AP görünürlük analizi sinyal dağılımı (Visibility analysis signal distribution from BESYO and Rektorate building's AP)

4. SONUÇLAR (CONCLUSION)

Yerleşke üzerinde dış ortam AP'lerin görünürlük analizleri sonucunda, yaklaşık 56 hektarlık 1. etap yerleşke alanının büyük bir kısmı, 9 dış ortam AP ile Wi-Fi kapsamı alanına sahip olacak şekilde sinyal dağılımları ile belirlenmiş ve planlanmıştır. Şekil 12'de kapsama alanlarının farklı renklerde gösterildiği gibi planlama ile uygun sinyal dağılımı belirlenerek olası gelişmelere açık bir öngörü yapılmıştır. AP'ler erişim kolaylığı ve çevre şartlarına karşı koruma amacıyla bina yüzeylerine yerleştirilmiş olsa da açık alanlara direkler tesis edilerek kapsama alanları artırılabilir ve görünürlük açıları genişletilebilir durumdadır.



Şekil 12. Hesaplanan dış ortam AP sinyal dağılımı (Calculated the outdoor AP signal distribution)

Üniversitenin 1. etap yerleşkesinde planlanan 9 dış ortam AP, kapsama alanları olarak birbirlerine geçiş yapsa da aslında kablosuz kontrol cihazının yönetimi yardımıyla

sinyal sönmeye veya değişmesine neden olmamaktadır. Aksine kullanıcı yoğunluğu ve diğer faktörlere göre aynı noktada eş zamanlı çalışmaları mümkün olabilmektedir.

Bu dış ortam AP'lerin görüş alanları içerisinde, gelecekte yapımı planlanan sosyal alanlar belirlendiği takdirde anten yönleri bu alanlara çevrilerek sinyal gücü ve dağılımı artırılabilir.

Bu çalışmaya ek olarak, dış ortam AP'lerden farklı anten modelleri ile ölçümler yapılarak değişik kapsama alanları belirlenebilir. Ayrıca, farklı zamanlarda farklı cihazlarla aynı ölçümler tekrarlanarak sinyal gücündeki değişim ve kapsama alanlarındaki değişimler ortaya konularak karşılaştırılmalı veriler elde edilebilir.

TEŞEKKÜR (ACKNOWLEDGEMENT)

Bu makale, Hitit Üniversitesi Bilimsel Araştırma Projeler Birimi tarafından desteklenen MUH19002.14.003 numaralı projeden türetilmiştir. Teşekkür ederiz.

KAYNAKÇA (REFERENCES)

- [1] M. F. Goodchild, Twenty Years of Progress: GIScience in 2010, Journal of Spatial Information, 2010.
- [2] L. Liu, L. Zhang, C. Chen, H. Chen, An Improved LOS Method for Implementing Visibility Analysis of 3D Complex Landscapes, International Conference on Computer Science and Software Engineering, 2008.

- [3] B.C. Chamberlaina, M.J. Meitner, A Route-based Visibility Analysis for Landscape Management, Land and Urban Planning 111, p13-24, 2013.
- [4] S. Chan, The Development of Planning Support Systems By Integrating Urban Models and Geographic Information Systems, PhD. Thesis, The University of Pennsylvania, Pennsylvania, USA, 1997.
- [5] T. Lillesand, R.W. Kiefer, J. Chipman, Remote Sensing and Image Interpretation, John Wiley & Sons. Science, doi:10.5311/JOSIS.2010.1.2, 2014.
- [6] T. Yomralıoğlu, Coğrafi Bilgi Sistemleri: Temel Kavramlar ve Uygulamalar, Akademik Kitabevi, Trabzon, 2000.
- [7] N. Nayan, M. Hashim, Y. Saleh, H. Mahat, Mapping internet coverage in Malaysia's university campuses: A case study of the Sultan Idris Education University, Perak. GEOGRAFIA Online TM Malaysian Journal of Society and Space 12(6),118-125, 2016.
- [8] Y. Kawamura, A.M. Dewan, B. Veenendaal, M. Hayashi, T. Shibuya, I. Kitahara, N. Nobuhara, K. Ishiet, Using GIS to develop a mobile communications network for disaster-damaged areas. International Journal of Digital Earth, 7(4): 279-293, 2014.
- [9] A.A. Iyad, A.S. Walid, Using GIS in Designing and Deploying Wireless Network in City Plans, International Journal of Computer Networks (IJCN), 6(4) : 2014.
- [10] J. Lubczonek, W. Kazimierski, M. Palczynski, Planning of Combined System of Radars and CCTV Cameras for Inland Waterways Surveillance by Using Various Methods of Visibility Analyses, 2011.
- [11] Cisco, http://www.cisco.com/c/en/us/products/collateral/wireless/aironet-1550-series/data_sheet_c78-641373.html, Erişim Tarihi: Ekim 2015.
- [12] J. Iliffe, Datums and Map Projections for Remote Sensing, GIS, and Surveying. CRC Press, 2000.
- [13] Kennedy, M. D., Introducing Geographic Information Systems with ARCGIS: A Workbook Approach to Learning GIS, John Wiley & Sons, 2013.
- [14] A. Jarvis, J. Rubiano, A. Nelson, A. Farrow, M. Mulligan, Practical Use of SRTM Data in the Tropics—comparisons with Digital Elevation Models Generated from Cartographic Data, Working Document, 198, 32, 2004.
- [15] I.D. Moore, R.B. Grayson, A.R. Ladson, Digital Terrain Modelling: A Review of Hydrological, Geomorphological, and Biological Applications. Hydrological Processes, 5(1), 3-30, 1991.