INTELLIGENT SYSTEMS AND OPTIMIZATION IN ENGINEERING

Editors
Sabri KOÇER
Özgür DÜNDAR

www.isres.org

ISRES Publishing

INTELLIGENT SYSTEMS AND OPTIMIZATION IN ENGINEERING

Editors
Sabri KOÇER
Özgür DÜNDAR

www.isres.org

ISRES Publishing

Intelligent Systems and Optimization in Engineering

Edited by

Prof. Dr. Sabri KOÇER
Necmettin Erbakan University,
Faculty of Engineering,
Department of Computer Engineering, Konya, Türkiye
E-mail: skocer@erbakan.edu.tr

Asst. Prof. Özgür DÜNDAR
Necmettin Erbakan University,
Faculty of Aviation and Space Sciences
Department of Astronautical Engineering, Konya, Türkiye
E-mail: ozdundar@erbakan.edu.tr

Cover Design & Layout
Engineer, MSc. Resul BÜTÜNER
Ministry of National Education,
Directorate General for Innovation and Educational Technologies, Ankara, Türkiye
Email : resul.butuner@eba.gov.tr

Language Editor
Lecturer Ceren DOĞAN
School of Foreign Languages, Necmettin Erbakan University, Konya, Türkiye
Email : cerendogan@erbakan.edu.tr

ISRES Publishing

Intelligent Systems and Optimization in Engineering

Editors

Sabri KOÇER

Özgür DÜNDAR

This book was typeset in 10/12 pt. Times New Roman, Italic, Bold and Bold Italic.

Intelligent Systems and Optimization in Engineering

Published by ISRES Publishing, International Society for Research in Education and Science (ISRES). Includes bibliographical references and index.

ISBN

978-625-6959-45-3

Date of Issue

December, 2024

Contact

Askan Mah. Akinbey Sok. No: 5/A Meram/Konya/Türkiye
isresoffice@gmail.com
www.isres.org

About the Book

The annual publication Intelligent Systems and Optimization in Engineering 2024 brings together a selection of distinguished articles curated by the editorial team. This volume includes 13 rigorously reviewed contributions spanning Technology, Engineering, Basic Sciences, Artificial Intelligence, and Intelligent Systems, each evaluated by at least two international experts to uphold high academic standards.

This edition focuses on fostering a deeper understanding of intelligent systems and optimization methods in engineering. By presenting innovative solutions across diverse domains such as Electronics, Communications, Mechatronics, Software, Artificial Intelligence, and Biomedical Engineering, the book underscores the growing importance of AI-driven methodologies in tackling complex engineering problems.

Central themes of this volume include the optimization and modeling of real-world systems using advanced algorithms and the implementation of intelligent systems for critical sectors like healthcare, manufacturing, and services. Embedded systems and AI-based approaches are highlighted as transformative tools for addressing challenges in these industries.

We believe this book will ignite a passion for research in intelligent systems and optimization, serving as a valuable reference for scholars, practitioners, and anyone with a keen interest in cutting-edge engineering advancements.

Published by ISRES Publishing

December, 2024

Prof. Dr. Sabri KOÇER
Necmettin Erbakan University

Asistant Professor Özgür DÜNDAR
Necmettin Erbakan University

Table of Contents

Citation

Koçer, S., & Dündar, Ö. (Eds.). (2024). *Intelligent Systems and Optimization in Engineering*. ISRES Publishing.

Managing Editors

Sabri KOÇER, PhD, He graduated from the Electrical Engineering Department of Selcuk University. He completed his graduate and his doctorate in Gazi University. Currently, Necmettin Erbakan University, Faculty of Engineering, Computer Engineering is working. Electronics, Computer, Artificial Intelligence, Image Processing, Telecommunication, Signal Processing and Biomedical studies in the area.

E-mail: skocer@erbakan.edu.tr, **ORCID:** 0000-0002-4849-747X

Özgür DÜNDAR, PhD, works at Necmettin Erbakan University, Department of Astronautical Engineering. He graduated from the Electrical and Electronics Engineering Department of Selcuk University. He worked as an Automation Engineer for a while. His master's and doctorate degrees are from Selçuk University, Institute of Science and Technology, Department of Electrical and Electronics Engineering. Special fields of study are Automation, Artificial Intelligence, Robotic, Communication, Electromagnetic and Micro Strip Patch Antenna designs.

E-mail: ozdundar@erbakan.edu.tr , **ORCID:** 0000-0002-4142-4446

In This Book

Chapter 1,

In this study, the challenges of flight route prediction and flight delay prediction are addressed as critical problems in the aviation industry. Flight route prediction plays a vital role in ensuring airspace safety and achieving fuel efficiency, while flight delay prediction is essential for improving operational efficiency and passenger satisfaction. This study highlights that analyzing ADS-B data with deep learning methods provides effective solutions to operational challenges in the aviation industry. By leveraging the advanced capabilities of CNN and LSTM architectures, this research showcases how modern predictive models outperform traditional methods in terms of scalability, accuracy, and adaptability.

Chapter 2,

The rapid development of the Industrial Internet of Things (IIoT) has highlighted the critical need for efficient data transmission, security, and management. Recent advancements focus on integrating data compression, encryption, and error correction to optimize system performance while addressing resource constraints in IIoT devices. This paper reviews various innovative approaches to improve data transmission efficiency in IIoT systems, emphasizing the integration of machine learning, deep learning, and edge-cloud computing. Techniques like projection-based coding, deep anomaly detection, and blockchain integration have demonstrated notable improvements in data compression, security, and system stability. Studies show that combining data compression with federated learning methods enhances data privacy, reduces communication overhead, and improves model accuracy. Additionally, novel compression methods, such as those based on Kronecker multiplication and attention mechanisms, have been shown to effectively manage large datasets, reduce latency, and conserve energy. Security remains a key concern, with studies exploring complex encryption algorithms and the role of image compression in enhancing data security without compromising system performance. Overall, this body of work underscores the importance of optimizing data flow, managing resources effectively, and developing secure, scalable solutions to address the challenges of modern IIoT applications.

Chapter 3,

This section explains AI applications in aerospace industry. Also there are four key areas about aerospace systems. These are remote sensing systems, spacecraft health monitoring system, satellite communication system and autonomous robotic systems. Remote sensing systems and AI applications are vital in air, space and ground observations. Spacecraft health monitoring system and AI-supported applications of it analyse different situations, evaluate performance data, and make informed decisions based on historical data. Satellite communication system and its AI applications provides optimum data transmission by filtering out unwanted harmonics and making the signal stronger. Autonomous robotic systems and AI are used to perform many critical tasks such as space exploration, data collection, maintenance/repair, vehicle motion control and ensuring life safety.

Chapter 4,

Artificial Neural Networks (ANN) are a powerful technique inspired by the human brain's nervous system and are widely employed as a crucial data processing tool today. The most important and difficult phase of an ANN is the training process, where the network's weights are optimized. As the number of connections in the neural network increases, so does the complexity of the weight optimization problem. Numerous algorithms and methods have been suggested over time to address this challenge. In recent years, one of the prominent techniques used for ANN training is meta-heuristic algorithms. This study evaluates the performance of several meta-heuristic algorithms for the solution of this problem. Specifically, six different algorithms, including the Grasshopper Optimization Algorithm, Artificial Hummingbird Algorithm, Arithmetic Optimization Algorithm, Crayfish Optimization Algorithm, Artificial Bee Colony and Tree-seed Algorithm were tested on 21 distinct datasets for ANN training. The performances of the algorithms were measured by using four popular metrics: precision, specificity, F1-score and sensitivity. The experimental findings reveal that the tested algorithms, particularly GOA, demonstrated high effectiveness in ANN training compared to the others. GOA produced the best results in 14 out of 21 datasets, achieving the top position in terms of average ranking success. These outcomes indicate that meta-heuristic algorithms offer a robust solution for handling the complex weight update processes in ANN.

Chapter 5

This section aims to enhance energy efficiency in IoT-based smart home applications, addressing a significant gap in the literature and demonstrating the feasibility of various energy optimization-based solutions. Using the Kaggle dataset Appliances Energy Prediction, algorithms such as Gray Wolf Optimization, Genetic Algorithm, and Particle Swarm Optimization—methods not previously applied to this dataset to the best of our knowledge—contribute significantly to reducing energy consumption while maintaining user comfort. The findings reveal that managing IoT devices efficiently benefits both environmental and economic aspects. For future studies, more comprehensive analyses of energy management in smart homes can be conducted by utilizing different datasets and optimization algorithms, particularly multi-objective optimization techniques. Additionally, predictive analyses can be developed through refined and enhanced machine learning and deep learning methods, offering broader perspectives on energy management solutions.

Chapter 6,

The chapter explores the optimization of controllers for multi-degree-of-freedom (multi-DOF) systems, which are critical for stabilizing complex dynamic systems such as UAVs. It highlights the use of MATLAB/Simulink Response Optimizer Toolbox, a powerful tool for fine-tuning control parameters under specific constraints like stability, response time, and energy efficiency. A case study on the Quanser 3DOF Hover system illustrates the application of Linear Quadratic Regulator (LQR) optimization. The process involved adjusting weighting matrices to meet strict criteria, such as a 5-second rise time and a 10% overshoot limit. Simulations demonstrated that the system effectively transitioned to hover and tracked predefined trajectories, underscoring the toolbox's efficiency in enhancing system performance and its educational value in control engineering.

Chapter 7,

This study provides a comprehensive examination of advanced optimization techniques for the aerodynamic design of turbomachinery. Optimization methods, including adjoint and gradient-based approaches, metaheuristic-based methods, surrogate-based models, neural networks, deep learning-based techniques, and hybrid approaches are evaluated. Special attention is given to emerging machine learning applications, particularly deep learning and artificial neural networks, which are reshaping the optimization landscape. The study also highlights the challenges and limitations of these methods, addressing computational efficiency and industrial applicability. Through the analysis of recent case studies and applications, insights into future research directions and potential developments in turbomachinery optimization are provided.

Chapter 8,

This study examines coil gun technology, focusing on its principles, components, advancements, and potential applications. Coil guns, or Gauss guns, use electromagnetic fields generated by sequentially activated coils to propel projectiles, offering advantages such as reduced wear and higher efficiency compared to traditional firearms. The system relies on multi-stage coils, capacitors, switching circuits, and sensors to synchronize acceleration, maximize energy efficiency, and prevent deceleration. Recent advancements include optimized coil designs, energy recovery mechanisms, and improved control systems, enhancing performance and reducing energy losses. Currently used in research and educational contexts, coil guns hold promise in defense, space exploration, and medical technologies. Applications include non-explosive weapons, cost-efficient payload launches, and innovative drug delivery systems. Despite challenges like energy consumption and scalability, advancements in materials, energy storage, and electronics are paving the way for broader adoption. This research underscores the transformative potential of coil guns in various fields.

Chapter 9,

In recent years, unmanned aerial vehicles have been widely used in civilian areas as well as for military purposes. Its accessibility, especially in terms of cost, diversifies its uses in the civil field day by day, and its benefit-cost ratios increase usage preferences. The use of unmanned aerial vehicles in the agricultural sector provides significant advantages in terms of reducing production costs and increasing product yield. It also offers the opportunity to manage production processes in a shorter time with less labor force.

Chapter 10,

Since practically all of our demands are met via the phone, mobile phone compression is the most crucial area. In graphics, 2D and 3D images are employed, and they can be referred to as textures. They are compressed using different techniques than regular images. Not just the color in mobile phone texture compression, but most techniques are improved for the brightness of human vision. Texture analysis for mobile phones in data compression is a fascinating topic! It involves techniques to reduce the size of texture data, which is crucial for mobile devices with limited memory and bandwidth.

Chapter 11,

Discusses the application of artificial intelligence (AI) techniques in machining processes, focusing on Artificial Neural Networks (ANN), Adaptive Neuro-Fuzzy Inference System (ANFIS), and metaheuristic algorithms for prediction and optimization. ANNs, modeled after the human nervous system, are used to predict surface roughness, optimize machining parameters, and monitor tool wear by learning complex, nonlinear relationships in data. ANFIS combines fuzzy logic with neural networks to handle complex parameter relationships, providing a more effective analysis and optimization tool. Metaheuristic algorithms, such as Genetic Algorithms (GA), Ant Colony Optimization (ACO), and Particle Swarm Optimization (PSO), are used to enhance ANFIS performance in predicting machining outcomes like tool vibration and surface roughness. Studies demonstrate that combining these AI techniques significantly improves accuracy and optimization in machining processes, outperforming traditional methods.

Chapter 12,

In this study, rectangular 1x4 microstrip array antennas operating at 2500 MHz frequency for wireless high speed internet access WiMax were designed, simulated and fabricated. Artificial neural networks (ANN) were used to improve the physical and electrical performances of the simulated and fabricated microstrip structures and the results were evaluated. HFSS (High Frequency Structural Simulator) software simulation program, which solves electromagnetic structures with finite element method, was used in the studies. The measurements were carried out using a Network Analyzer in a laboratory environment where experimental setups were prepared. In terms of electrical results and physical dimensions, a successful product that can be used in the industrial field has been created.

Chapter 13,

Intricacies of computer vision are explored, focusing on object recognition and classification. This chapter explores the fundamental principles, methodologies, and applications of these critical aspects in computer vision. Starting with an introduction to the field's transformative impact across various domains, we discuss the convergence of computer science, artificial intelligence, and signal processing that underpins computer vision. The chapter covers the essential stages of object recognition and classification, including image acquisition and filtering, feature extraction, feature selection, model training, and evaluation. Traditional techniques and modern deep learning approaches, particularly convolutional neural networks (CNNs), are examined in detail. The advantages of deep learning, such as automated feature extraction and superior accuracy, are highlighted alongside traditional methods. The importance of privacy and security in the proliferation of computer vision technologies are addressed, emphasizing the need for robust measures and ethical considerations. The application scope of these technologies is vast, ranging from autonomous systems and medical imaging to security and entertainment. Future directions in the field are discussed, focusing on the potential for more real-time applications, advancements in autonomous systems, and the increasing importance of personal data privacy. Overall, this chapter provides a comprehensive overview of the current state and future potential of object recognition and classification in computer vision, underscoring its significance in enhancing human capabilities and transforming industries.

Contributors

Prof. Dr. Sabri KOÇER
Necmettin Erbakan University,
Faculty of Engineering,
Department of Computer Engineering, Konya, Türkiye
E-mail: skocer@erbakan.edu.tr **ORCID:** 0000-0002-4849-747X

Prof. Dr. Selçuk MISTIKOĞLU
Iskenderun Technical University
Faculty of Engineering and Natural Sciences
Department of Mechanical Engineering, Hatay, Türkiye
E-mail: selcuk.mistikoglu@iste.edu.tr **ORCID:** 0000-0003-2985-8310

Assoc. Professor Mehmet Emin ÇETİN
Necmettin Erbakan University,
Faculty of Aviation and Space Sciences
Department of Aeronautical Engineering, Konya, Türkiye
E-mail:cetin_memin@hotmail.com, **ORCID:** 0000-0002-6314-5261

Asistant Professor Özgür DÜNDAR
Necmettin Erbakan University,
Faculty of Aviation and Space Sciences
Department of Astronautical Engineering, Konya, Türkiye
E-mail:ozdundar@erbakan.edu.tr, **ORCID:** 0000-0002-4142-4446

Assistant Professor Murat KARAKOYUN
Necmettin Erbakan University,
Faculty of Engineering,
Department of Computer Engineering, Konya, Türkiye
E-mail: mkarakoyun@erbakan.edu.tr, **ORCID:** 0000-0002-0677-9313

Assistant Professor Alperen EROĞLU
Necmettin Erbakan University,
Faculty of Engineering,
Department of Computer Engineering, Konya, Türkiye
E-mail: aeroglu@erbakan.edu.tr, **ORCID:** 0000-0002-1780-7025

Assistant Professor Mehmet Ali GÜVENÇ
Iskenderun Technical University
Faculty of Aeronautics and Astronautics
Department of Aeronautics and Astronautics, Hatay, Turkiye
E-mail: mali.guvenc@iste.edu.tr, **ORCID:** 0000-0002-4652-3048

Assistant Professor Mehmet Numan KAYA
Necmettin Erbakan University,
Faculty of Engineering,
Department of Mechanical Engineering, Konya, Türkiye
E-mail:mnkaya@erbakan.edu.tr, **ORCID:** 0000-0003-3116-9442

Assistant Professor Bilal ERVURAL
Necmettin Erbakan University,
Faculty of Engineering,
Department of Industrial Engineering, Konya, Türkiye
E-mail: bervural@erbakan.edu.tr, **ORCID:** 0000-0002-5206-7632

Assistant Professor Engin Hasan ÇOPUR
Necmettin Erbakan University,
Faculty of Aviation and Space Sciences
Department of Aerospace Engineering, Konya, Türkiye
E-mail: engincopur@gmail.com, **ORCID:** 0000-0003-0837-1255

Assistant Professor Hasan Huseyin BILGIC
Necmettin Erbakan University,
Faculty of Aviation and Space Sciences
Department of Aeronautical Engineering, Konya, Türkiye
E-mail: bilgichh@gmail.com, **ORCID:** 0000-0001-6006-8056

Assistant Professor Mehmet DURSUN
Necmettin Erbakan University,
Faculty of Aviation and Space Sciences
Department of Avionics, Konya, Türkiye
E-mail:mehmet.dursun@erbakan.edu.tr, **ORCID:** 0000-0002-0558-6309

Assistant Professor Tarık ÜNLER
Necmettin Erbakan University,
Faculty of Aviation and Space Sciences
Department of Avionics, Konya, Türkiye
E-mail: tarik.unler@gmail.com, **ORCID:** 0000-0002-2658-1902

Assistant Professor Rabia TOPRAK
Karamanoglu Mehmetbey University,
Faculty of Engineering,
Department of Electrical-Electronics Engineering, Kraman, Türkiye
E-mail: rabiatop@kmu.edu.tr, **ORCID:** 0000-0002-3306-1163

Assistant Professor Halil İbrahim AYAZ
Necmettin Erbakan University,
Faculty of Engineering,
Department of Industrial Engineering, Konya, Türkiye
E-mail: hiayaz@erbakan.edu.tr, **ORCID:** 0000-0001-5547-6485

Research Assistant Behice Gülsüm ÇELİK
Necmettin Erbakan University,
Faculty of Engineering,
Department of Computer Engineering, Konya, Türkiye
E-mail: behicegulsum.celik@erbakan.edu.tr, **ORCID:** 0000-0003-4008-1258

Research Assistant Saliha KÖPRÜCÜ
Necmettin Erbakan University,
Faculty of Aviation and Space Sciences
Department of Aeronautical Engineering, Konya, Türkiye
E-mail: salihakoprucu@gmail.com, **ORCID:** 0009-0007-9658-0008

MSc. Gökhan GÜLŞEN
Necmettin Erbakan University,
Faculty of Aviation and Space Sciences
Department of Aeronautical Engineering, Konya, Türkiye
E-mail:gkadirgil@yahoo.com, **ORCID:** 0000-0003-1876-7499

MSc. Münevver BAHAR
Necmettin Erbakan University,
Faculty of Engineering,
Department of Computer Engineering, Konya, Türkiye
E-mail: munevverbahar1@gmail.com, **ORCID:** 0009-0009-0489-0884

BSD. Gamze AKÇAY
Necmettin Erbakan University,
Faculty of Engineering,
Department of Computer Engineering, Konya, Türkiye
E-mail: 20010011039@ogr.erbakan.edu.tr, **ORCID:** 0009-0003-5314-1544

Deep Learning Approaches for Analysis and Prediction of Flight Data

Gökhan GÜLŞEN

Necmettin Erbakan University

Mehmet Emin ÇETİN

Necmettin Erbakan University

To Cite This Chapter

Gülşen, G., & Çetin, M. E. (2024). Deep Learning Approaches for Analysis and Prediction of Flight Data. In S. Koçer & Ö. Dündar (Eds.), *Intelligent Systems and Optimization in Engineering* (pp. 1-8). ISRES Publishing.

Introduction

The aviation industry has grown rapidly in recent years, dramatically increasing the amount of passengers and cargo transported around the world. The increasing number of aircraft in the air necessitates the automation of airspace surveillance. It is well recognized that aircraft with varying purposes operate in distinct flight patterns. This growth has brought with it the need for greater efficiency, safety and cost optimization for companies and airports in the sector. The total cost of all disruptions in commercial aviation in the United States in 2008 was estimated at 33.5 billion dollars (Khaksar, H., et. al.). Only 79% of flights arrived on schedule in 2019, according to airline delay statistics, which resulted in tens of billions of dollars in damages, including lost demand, passenger charges, and other indirect costs (Kim, Y., J., Choi, S., Briceno, S., Mavris, D., 2016). In European airspace, a record was set on June 28, 2019 with 37288 flights(Nats, 2023). Improving factors such as flight safety, fuel consumption, operational costs and passenger satisfaction depends on effective analysis of flight data and accurate prediction of future flight conditions. In this context, issues such as flight route and flight delay prediction are of strategic importance for the aviation industry.

Today, flight route prediction is essential in terms of both safety and fuel consumption. Especially on routes with heavy air traffic, accurate route prediction can improve flight safety and reduce fuel consumption through optimal route selection. At the same time, accurate flight route prediction provides a significant advantage for airport congestion management and air traffic control.

On the other hand, flight delay forecasting is of great importance in terms of passenger satisfaction and operational costs. Delays cause serious financial losses for airlines and cause customer dissatisfaction by negatively affecting passengers' travel plans. Therefore, predicting potential delays using historical flight data allows airlines to plan more effectively and provide accurate information to customers.

The purpose of this study is to make flight route and flight delay predictions using deep learning methods. In particular, models that can process time series data such as CNN (LeCun, L., Bengio, Y., 1995) and LSTM (Hochreiter, S., Schmidhuber, J., 1997) will be used to analyze and predict flight data. The studies to be conducted in this context will contribute to increase the efficiency of flight operations and provide useful information for various stakeholders in the aviation industry.

Problem

This research study seeks to address two questions:

- Flight Path and Safety ADS-B (Automatic Dependent Surveillance-Broadcast) data(Faa, 2023) is very important for accurately tracking and predicting flight paths, as it contains instantaneous position, altitude, speed and direction information of aircraft. Especially in heavily trafficked airspaces, accurate data is needed for aircraft to navigate at a safe distance and determine their routes. This data allows the route to be predicted and optimized according to the air traffic density. By analyzing ADS-B data, air traffic management and airspace safety can be improved.
- It allows the observation of momentary changes in the flight process and the analysis of historical flight data to detect flight delays and ensure passenger satisfaction. Analyzing the causes of delays can improve passenger satisfaction and help regulate airport operations by predicting these problems in advance. Using flight data, delays due to weather, traffic density or technical reasons can be predicted. These predictions will increase the effectiveness of airlines in informing passengers and offering alternative solutions.

Machine Learning

A wide range of algorithms that may “learn” from data on their own are included in machine learning (ML) (Lindholm, A., Wahlstrom, N., Lindsten, F., Schon, T., B., 2022). Machine learning (ML) is defined as a type of data analysis where an algorithm learns on its own to extract information and generate predictions from a dataset. The training set is the name given to this collection of data. A separate data set, known as the test set, that does not intersect with the training set is then used to evaluate the algorithm. The data, the mathematical model, and the learning algorithm are the three main parts of a machine learning algorithm. Using the data, a machine learning system optimizes the parameters in the mathematical mode.

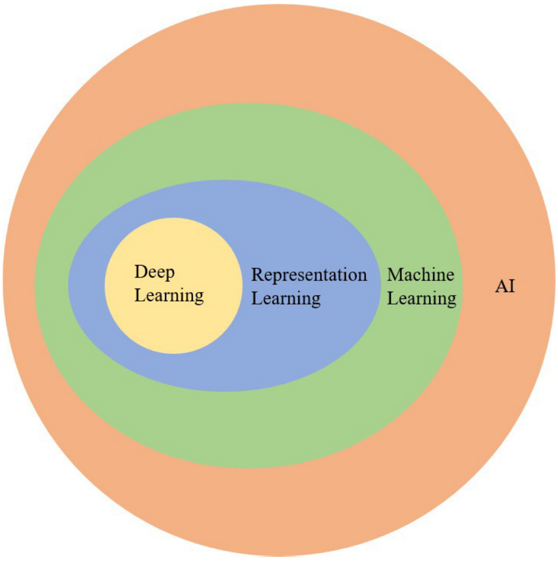
Artificial Neural Networks

Using the human brain as an example, mathematical modeling of the learning process has led to the development of artificial neural networks (ANN). It replicates the architecture of the brain’s biological neural networks as well as their capacity for memory, learning, and generalization. Examples are used to carry out the learning process in artificial neural networks. Rules are established and input and output data are provided during the learning process.

Deep Learning

Deep learning models have demonstrated significant effectiveness across various domains. Examples include applications like speech recognition, language translation, and computer vision. Recent technological advancements, including self-driving cars and intelligent personal assistants, can be largely attributed to developments in deep learning techniques. Improved processing hardware and easier access to data are responsible for the rise in popularity of deep learning techniques in recent years. Goodfellow et al. (Goodfellow, I., Bengio, Y., Courville, A., 2016) illustrate deep learning using the Venn diagram presented in Figure 2. Deep learning models typically necessitate substantial datasets to exceed the performance of traditional machine learning approaches and are more computationally intensive.

Figure 1
Venn diagram illustrating that deep learning represents only a subset of artificial intelligence. (Goodfellow, I., Bengio, Y., Courville, A., 2016)



Whether to ML or DL depends mainly on the type of problem, the volume of the data set and the complexity requirements of the model. When deciding which is more suitable for developing prediction models for the problems in our thesis, we need to consider the following factors:

- Machine Learning (ML) methods generally give good results with smaller data sets and a limited number of features. ML methods are usually sufficient, especially when the data has numeric and categorical/class features, but does not contain a large number of samples.
- Deep Learning (DL) methods need large data sets. Large volumes of data, more processing power, and more time are typically needed for deep learning models. If your data set is large and contains more complex features (e.g. text data, images, sounds), then deep learning models can perform better.

Because of the large and our dataset's complexity (extreme location information, various delay reasons), Deep Learning (DL) methods were used.

Deep Learning Methods

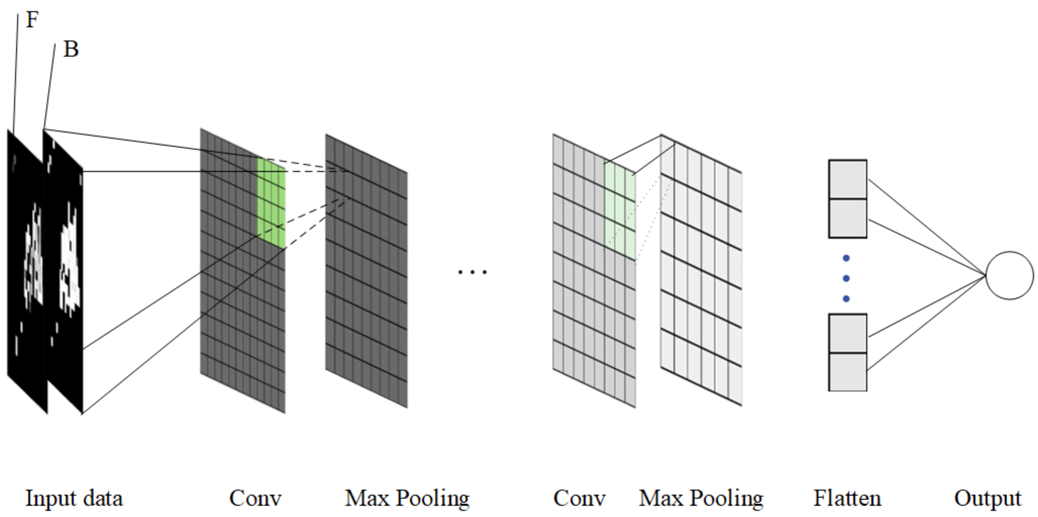
A subset of the larger machine learning family, deep learning focuses on learning data representations as opposed to task-specific methods. Unlike traditional machine learning methods, deep learning has the advantage of constructing deeper architectures that allow for learning more abstract information. Deep learning's capacity to automatically learn feature representations, doing away with the need for laborious manual feature engineering, is one of its primary characteristics. Deep learning has found practical applications in various fields(Coşkun, M., Yıldırım, Ö., Uçar, A., Demır, Y., 2017) through algorithms such as CNN, RNN and LSTM, each of which will be discussed in detail in the following sections.

What is CNN (Convolutional Neural Network)?

CNN (LeCun, Y., Bengio, Y., 1995) is a deep learning algorithm which can recognize local features in two- or three-dimensional data such as images, audio and time series. CNNs are widely used, especially in areas like natural language processing and image processing. Basically, it extracts certain patterns or convolution layers are used to extract features from the input data. and preserves important information by reducing the dimensions through pooling in Figure 2.

Figure 2

Structure of the CNN Model (Qiang L., Xinjia G. & Liu J)



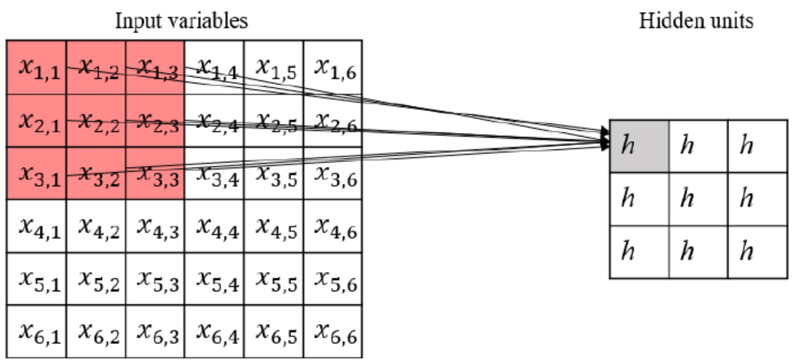
The main advantage of CNN is that it provides high accuracy by examining regionally significant parts of the data, not the whole data. Especially in-flight forecasting, CNN can detect short-term correlations and thus quickly learn important features in time series data.

*Convolution layers: The basic component of the CNN model. It processes the data by dividing it into small pieces through filters, see Figure 3 for an example. Each filter can be used to estimate 2D location data (e.g. latitude-longitude pairs), and convolution layers can be used to learn the environmental relationships and local dependencies of the data. This structure is particularly useful for predicting future position from past movements of position data.

*Pooling: A method used by CNN to decrease the size of the data being processed and escalate the model's processing speed. Pooling preserves important information by averaging or maximizing small regions of the data. In this way, the model operates more quickly and has less complexity.

Figure 3

Diagram of sparse interacting convolutional layers (Goodfellow, I., Bengio, Y., Courville, A., 2016)



CNN's Superiority and Advantages

Ability to Learn Local Dependencies: CNNs are successful in capturing local patterns in data thanks to the filters in their layers. In flight route and delay data, local and short-term changes in features such as speed, position, altitude are important for future predictions. CNN's ability to capture these short-term correlations improves its prediction performance.

Learning Long-Term Patterns in Time Series: Since flight data consists of time series data, CNN layers can provide accuracy in predictions by recognizing long-term patterns in these series. The convolution process of the filters is good at capturing changes in direction or changes in speed of the flight over time, thus learning important information when making flight route and delay predictions.

Faster Processing by Reducing Complexity: CNN models reduce the data size by filtering out unnecessary information thanks to the "pooling" process used to reduce the data size. This feature both increases processing speed and enables analysis with a smaller data size. This speed advantage provides a significant advantage in time-sensitive processes such as flight prediction.

Potential for Visual Data: Flight prediction can use not only textual or numerical data, but also maps or radar images that are 2D/3D visualizations of the flight path. CNN is ideal for working with this type of visual data. Especially in cases where the flight route needs to be followed visually, CNN models can give more successful results.

No Need for Data Processing to Reduce User Errors: CNN has the ability to learn without the need for many data pre-processing steps, especially thanks to its ability to work with raw data. This allows flight route and delay prediction data to be used in the model quickly and effectively.

Advantages that Differentiate CNN from Other Models

While algorithms such as RNN (Recurrent Neural Network) and LSTM are successful in learning long-term dependencies, they do not have the advantage of quickly learning local features and short-term patterns that CNNs provide. CNNs are good at identifying sudden changes in flight routing and short-term anomalies that cause delays.

Traditional methods such as Decision Trees or Regression Models may fail to capture long-term and short-term dependencies in the data at the same time. For this reason, CNN is observed to offer higher accuracy in flight prediction than traditional methods.

What is RNN (Recurrent Neural Network)?

One kind of neural network intended for processing sequential data is called a recurrent neural network (RNN) (Rumelhart, D., E., Hinton, G., E., Williams, R., J., 1986). Text, audio, and time series are examples of temporally oriented data that they can examine. By passing a secret state from one time step to the next, RNNs are able to accomplish this. Based on the input and the previous hidden state, the hidden state is updated at each time step. In sequential data, RNNs are good at identifying short-term dependencies, but they have trouble identifying long-term ones.

Inadequacy of RNNs

Neural networks called RNNs are made to handle sequential data and are good at recognizing short-term dependencies. However, they encounter an issue referred to as the vanishing gradient problem. Because of this issue, the gradients in the network get smaller and eventually vanish as the number of layers rises. RNNs are therefore unable to capture long-term relationships, thus losing important information from the past. In flight route and delay prediction, the long-term effects of historical data such as the previous position, speed, delay times and weather for a given flight need to be taken into account. Since RNNs cannot preserve this information, the accuracy of flight predictions decreases.

What is LSTM (Long Short Term Memory)?

One kind of RNN (recurrent neural network) that may store long-term dependencies in sequential input is called an LSTM (long short-term memory). Sequential data, including time series, text, and voice, may be processed and analyzed using LSTMs. They circumvent the issue of vanishing slope that plagues conventional RNNs by controlling the information flow via a memory cell and gates, allowing them to selectively store or discard information as needed. LSTMs are extensively employed in many different applications, including time series prediction, speech recognition, and natural language processing.

LSTM's Superiority

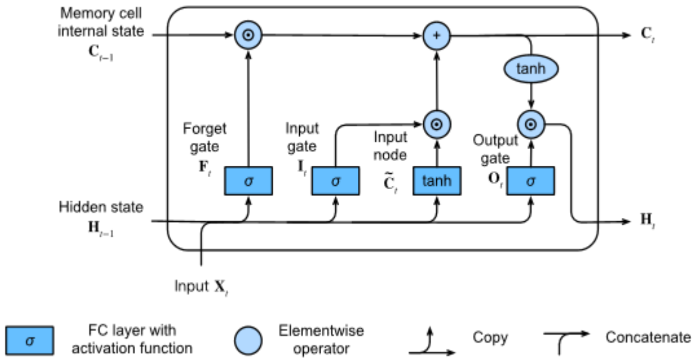
LSTM (Long Short-Term Memory) networks (Hochreiter, S., Schmidhuber, J., 1997) have the capacity to acquire long-term relationships and store important information in data. This is achieved through cell state and gate mechanisms:

- Input Gate: Determines how and to what extent new information is added to the cell.
- Forget Gate: Unnecessary information is removed from the cell, freeing up memory space.
- Output Gate: Controls which part of the information stored in the cell is added to the output.

Thanks to these structures, shown in Figure 4, LSTM is able to recall information from past time periods and make more accurate predictions by preserving long-term relationships.

Figure 4

Structure of the LSTM Model (Zhang A., Lipton Z., Li M. & Smola A.)



Conclusion

In this study, the challenges of flight route prediction and flight delay prediction are addressed as critical problems in the aviation industry. Flight route prediction plays a vital role in ensuring airspace safety and achieving fuel efficiency, while flight delay prediction is essential for improving operational efficiency and passenger satisfaction. To tackle these issues, real-time flight data is used to develop prediction algorithms with deep learning models.

The models can be employed in this study include CNN and LSTM architectures:

The LSTM model can be selected for its ability to process ttime-series data, which records the temporal relationships between past flight metrics such position, velocity, and route to make accurate predictions. The CNN model, on the other hand, is utilized for its capacity to extract spatial relationships and feature patterns within the dataset, providing strong results in predicting both flight routes and delay types. Deep learning methods are preferred over traditional machine learning algoritms because of their superior ability to handle large and complex datasets. Deep learning models, in contrast to conventional methods, do not require considerable human feature engineering since they automatically discover features from raw data. Moreover, deep learning excels at identifying intricate relationships and uncovering hidden patterns in high-dimensional data, making it particularly effective for datasets like ADS-B, which contain diverse and high-volume information.

Experimental evaluations demonstrate that the proposed prediction models successfully address the problems of flight route and delay prediction. The models consistently achieve high levels of accuracy and show their applicability in the aviation industry. These findings emphasize the potential of deep learning to enhance flight safety, reduce operational costs, and improve passenger satisfaction.

In conclusion, this study highlights that analyzing ADS-B data with deep learning methods provides effective solutions to operational challenges in the aviation industry. By leveraging the advanced capabilities of CNN and LSTM architectures, this research showcases how modern predictive models outperform traditional methods in terms of scalability, accuracy, and adaptability. Future work focuses on expanding the diversity of data sources and employing more advanced architectures to further improve model accuracy and address a broader range of aviation-related challenges.

References

Khaksar, H., & Sheikholeslami, A. (2019). Airline delay prediction by machine learning algorithms. *Scientia Iranica*, *26*(5), 2689-2702.

Akenine-Möller, T., & Ström, J. (2003). Graphics for the masses: a hardware rasterization architecture for mobile phones. In *ACM SIGGRAPH 2003 Papers* (pp. 801-808).

Kim, Y. J., Choi, S., Briceno, S., & Mavris, D. (2016, September). A deep learning approach to flight delay prediction. In *2016 IEEE/AIAA 35th Digital Avionics Systems Conference (DASC)* (pp. 1-6). IEEE.

Nats, The european airspace challenge. https://nats.aero/blog/2019/09/ the-european-airspace-challenge/. Accessed: 2023-02-28.

LeCun, Y., & Bengio, Y. (1995). Convolutional networks for images, speech, and time series. The handbook of brain theory and neural networks, 3361(10), 1995.

Li, Q., Guan, X., & Liu, J. (2023). A CNN-LSTM framework for flight delay prediction. *Expert Systems with Applications*, *227*, 120287.

Faa, Automatic dependent surveillance-broadcast (ADS-B). https://www.faa.gov/air_traffic/technology/adsb. Accessed: 2023-05-02.

Bts, https://www.bts.gov/

Lindholm, A., Wahlström, N., Lindsten, F., & Schön, T. B. (2022). *Machine learning: a first course for engineers and scientists*. Cambridge University Press.

Bengio, Y., Goodfellow, I., & Courville, A. (2017). *Deep learning* (Vol. 1). Cambridge, MA, USA: MIT press.

Coşkun, M., Yıldırım, Ö., Uçar, A., & Demır, Y. (2017). An overview of popular deep learning methods. *European Journal of Technique (EJT)*, *7*(2), 165-176.

LeCun, Y., & Bengio, Y. (1995). Convolutional networks for images, speech, and time series. *The handbook of brain theory and neural networks*, *3361*(10), 1995.

Pavlo, A. Lecture# 03: Data Formats & Encoding II.

Rumelhart, D. E., Hinton, G. E., & Williams, R. J. (1986). Learning representations by back-propagating errors. Nature, 323(6088), 533-536.

Hochreiter, S. (1997). Long Short-term Memory. *Neural Computation MIT-Press*.

Zhang, A., Lipton, Z. C., Li, M., & Smola, A. J. (2021). Dive into deep learning. *arXiv preprint arXiv:2106.11342*.

About the Authors

Gökhan GÜLŞEN, MSc, is a professional with a background in software development. He earned his undergraduate degree from the National Defense University in Istanbul in 2015. He holds two master's degrees, one from Aksaray University (2021) and another from Necmettin Erbakan University (2024). He has work experience in various roles, starting as an Aircraft Maintenance Officer in the Turkish Air Force (2015-2020), followed by positions as a Software Development.
E-mail: gokhangulsen@ogr.erbakan.edu.tr, **ORCID:** 0000-0003-1876-7499

Mehmet Emin ÇETİN, Assoc. Prof., completed his BSc degree at Kocaeli University, Kocaeli in 2010 and MSc degree at Karadeniz Technical University, Trabzon in 2012. He received his PhD degree in Mechanical Engineering from Karadeniz Technical University, Trabzon, Turkey in 2017. His research interests are artificial intelligence, additive manufacturing, adhesives, composite materials and finite element analysis.
E-mail: mecetin@erbakan.edu.tr , **ORCID:** 0000-0002-6314-5261

Similarity Index
The similarity index obtained from the plagiarism software for this book chapter is 14%.

Enhancing Efficiency in Industrial IoT through Data Compression: A Review

Halil İbrahim AYAZ

Necmettin Erbakan University

To Cite This Chapter

Ayaz, H. İ. (2024). Enhancing Efficiency in Industrial IoT through Data Compression: A Review. In S. Koçer & Ö. Dündar (Eds.), *Intelligent Systems and Optimization in Engineering* (pp. 9-22). ISRES Publishing.

Introduction

In recent years, the Industrial Internet of Things (IIoT) has emerged as a transformative force in industrial sectors worldwide. Defined as the integration of interconnected sensors, instruments, and computing devices within industrial machinery and processes, IIoT enables unprecedented levels of data collection, analysis, and automation. This technological evolution promises to revolutionize traditional industries by enhancing operational efficiency, optimizing resource utilization, and unlocking new avenues for innovation and growth. The advent of the Industrial Internet of Things has led to the generation of massive amounts of data. To effectively utilize this data and overcome challenges such as storage, processing, and transmission, data compression techniques play a crucial role (Wen et al., 2018).

The significance of IIoT lies not only in its ability to connect previously isolated systems but also in its capacity to generate actionable insights in real time. By leveraging advanced analytics and machine learning algorithms, IIoT empowers businesses to make informed decisions, predict maintenance needs, and improve overall productivity. Moreover, the seamless integration of physical and digital environments through IIoT fosters a more agile and responsive industrial ecosystem, capable of adapting to dynamic market demands and evolving customer expectations. The use of ML techniques in the physical layer of IoT communication systems can significantly improve communication and acquire signal intelligence (Jagannath et al., 2019) .

As industries continue to embrace IIoT technologies, the potential for enhanced efficiency, reduced downtime, and increased profitability becomes increasingly evident. However, alongside these benefits come challenges, such as cybersecurity risks and the need for robust infrastructure and a skilled workforce. Nonetheless, the promise of IIoT in driving operational excellence and fostering innovation underscores its role as a cornerstone of the fourth industrial revolution. (M. V. Silva et al., 2022)

In this introduction, we delve into the fundamentals of IIoT, explore its transformative impact on industrial applications, and highlight the opportunities and challenges that lie ahead as businesses navigate the complexities of this ground-breaking technology.

The sheer volume of data generated by IIoT devices is staggering, driven by factors such as increased sensor deployment, enhanced connectivity, and the adoption of advanced analytics. This influx of data offers valuable insights into operational performance, machine health, and process efficiency. However, effectively managing and extracting actionable intelligence from this data deluge requires sophisticated data management techniques.

Efficiency in data management is paramount to realizing the full benefits of IIoT. Traditional methods often struggle to cope with the scale, variety, and velocity of data produced by IIoT devices. Therefore, businesses are increasingly turning to advanced data storage solutions, such as cloud computing and edge computing, to handle the influx of real-time data streams efficiently. Cloud platforms provide scalable storage and computing capabilities, enabling organizations to process and analyze data rapidly and cost-effectively. Meanwhile, edge computing empowers businesses to perform data processing closer to the source, reducing latency and enhancing real-time decision-making capabilities.

Moreover, efficient data management encompasses data governance, security, and compliance considerations. Protecting sensitive industrial data from cyber threats and ensuring regulatory compliance are critical priorities for organizations operating in IIoT environments. Implementing robust data governance frameworks and leveraging encryption and authentication protocols are essential steps in safeguarding data integrity and confidentiality.

In conclusion, while the exponential growth of data generated by IIoT devices presents significant challenges, it also offers unparalleled opportunities for innovation and operational excellence. By adopting efficient data management techniques and leveraging advanced technologies, businesses can unlock the transformative potential of IIoT, driving sustainable growth and competitive advantage in the digital age.

Fundamentals of Data Compression

Data compression is a fundamental technique used to reduce the size of data for efficient storage, transmission, and processing, without compromising its integrity or usability. This process is particularly crucial in the context of Industrial Internet of Things (IIoT) where large volumes of data are generated continuously. Here's an overview of the basic concepts of data compression, including lossless and lossy compression techniques:

Lossless Compression

Lossless compression is a method where the original data can be perfectly reconstructed from the compressed data. This technique ensures that no information is lost during the compression and decompression process. Key methods include:

Run-Length Encoding (RLE): This method replaces sequences of the same data values (runs) with a single value and count. It is effective for compressing data with repetitive patterns.

Huffman Coding: Huffman coding assigns variable-length codes to input characters based on their frequencies in the data. Characters that appear more frequently are assigned shorter codes, resulting in efficient compression for text and similar data.

Lempel-Ziv (LZ) Compression: LZ compression algorithms, such as LZ77 and LZ78, identify repeated sequences of data and replace them with references to a dictionary or previously encoded data. These algorithms are widely used in file compression formats like ZIP.

Lossy Compression

Lossy compression sacrifices some data accuracy to achieve higher compression ratios, suitable for applications where minor loss of quality is acceptable. It is commonly used for multimedia data like images, audio, and video. Techniques include:

Discrete Cosine Transform (DCT): Used in JPEG compression, DCT converts image data into frequency components, discarding high-frequency components that human eyes are less sensitive to.

Wavelet Transform: Wavelet transforms analyze and transform data into different frequency components, enabling efficient compression of both high and low-frequency data.

Quantization: Quantization reduces the precision of data values to achieve compression. In image and audio compression, quantization reduces the number of distinct colors or audio levels, respectively.

Application in IIoT

In IIoT applications, efficient data compression techniques are essential for reducing bandwidth requirements, minimizing storage costs, and optimizing data processing capabilities. Lossless compression ensures that sensor data and operational logs are accurately preserved for analysis and compliance purposes. Meanwhile, lossy compression techniques are suitable for compressing multimedia data streams from surveillance cameras or sensor arrays, where slight degradation in quality is permissible.

In summary, data compression techniques play a pivotal role in optimizing data handling within IIoT environments, balancing the trade-offs between storage efficiency, data integrity, and processing speed to support the scalable and sustainable deployment of interconnected industrial systems.

Data Compression in IIoT

In current Internet of Things (IoT) applications, data compression, data encryption and error/corruption correction are often implemented separately (Kuldeep & Zhang, 2021). To ensure reliable communication, especially in harsh wireless environments, error/distortion correction codes or Automatic Repeat Request (ARQ) schemes with high correction capacity have been proposed. However, these solutions increase complexity and energy consumption. For resource-constrained IoT devices, implementing all of these processes together is a challenging task. In this context, we propose a lightweight, efficient, and secure fault-tolerant scheme called ENCRUST, which performs these three functions using simple matrix multiplication. ENCRUST is built on a projection-based coding theory, exploiting the inherent sparsity of the signal. The theoretical analysis and experimental study of the proposed scheme is done in comparison with conventional schemes.

Another approach to improve data transmission efficiency in industrial internet of things (IIoT) applications proposes secure and sustainable intelligent supply chain systems (Singh et al., 2023). In this study, we aim to increase the stability of the system and network lifetime by using sensor networks. Compared to the methods of T. Senthil and S. Singh (Senthil & Kannapiran, 2017; Singh, 2020), the proposed method provides a stability increase of 35.19% and 7.23% and a network lifetime extension of 119.33% and 71.72%, respectively. These findings once again emphasize the importance of developing secure and sustainable systems in IIoT applications.

On the other hand, research on data transmission in industrial IoT applications shows the effectiveness of deep learning-based image compression techniques (Sujitha et al., 2021). In these studies, the proposed methods achieved an average peak signal-to-noise ratio (PSNR) of 49.90 dB and a compression ratio (CR) of 89.38%. These results demonstrate the potential of deep learning techniques to improve data transmission efficiency in industrial applications. However, the work on compression methods in industrial embedded systems to ensure secure data transmission is also noteworthy (Kumar & Srinivasan, 2023). In this paper, the AHBO-LBGCCE method combines the LBG model with the AHBO algorithm to generate vector quantization (VQ). This method uses the Burrows-Wheeler Transform (BWT) model for code library compression and the Blowfish algorithm for security. Thus, significant advantages are achieved in terms of both compression and security in data transmission.

Data management and anomaly detection are also of great importance in industrial IoT applications. In this context, it is proposed to optimize data processing processes using edge computing (Kong et al., 2020). By analyzing compressed test data sets with the K-means clustering algorithm, abnormal sensor values and labels are obtained and potential problems in the system are detected. This method demonstrates the positive effects of edge computing on data processing and how it improves data management processes in industrial applications.

In parallel, another study on the use of virtual events to reduce data flow in distributed broadcast/subscriber systems aims to optimize the management of data flow with edge computing (Zehnder et al., 2019). Virtual events have the potential to improve system performance by reducing bandwidth in data transmission. This method makes data flow more efficient in industrial IoT applications, while enabling more effective use of system resources.

Another solution to the high latency and bandwidth issues encountered during the processing of large data sets in industrial IoT systems is the Kronecker-assisted compression design optimized with the integration of cloud computing and edge computing ((S. Chen et al., 2020). This method optimizes the data compression process while significantly improving data management and transmission efficiency in industrial IoT systems. Moreover, the data compression process using Kronecker multipliers is tested under various scenarios to make it more efficient.

Proposals for edge-cloud collaborative IoT networks to improve communication efficiency have also attracted attention (Zhang et al., 2023). The use of deep compression techniques aims to save bandwidth in data transmission while maintaining data quality. This work makes significant contributions towards improving data management and communication efficiency in industrial IoT systems.

A study that aims to optimize deep learning services in three-layer edge systems by combining data compression and load balancing decisions improves data transmission efficiency in industrial IoT applications (Hosseinzadeh et al., 2021). This innovative approach aims to reduce delays and improve system performance by optimizing data flow.

Finally, an image compression algorithm that utilizes attention mechanisms for sensor assembly in industrial IoT applications has been developed (Meier et al., 2022). This algorithm aims to reduce bandwidth and increase energy efficiency in the transmission of image data. Attention mechanisms help to identify important features, filter out redundant data and improve the efficiency of the compression process. This approach offers an innovative solution to address the challenges of data transmission in industrial applications.

The design of data collection systems in industrial IoT environments for small-scale manufacturers aims to improve data transmission efficiency through the integration of data compression methods (Tsai et al., 2019). This approach has the potential to improve cost-effectiveness and efficiency, given the limited resources of small-scale enterprises. The study aims to make data management more efficient by examining various compression techniques and NoSQL database solutions.

Data management and processing methods in the Industrial Internet of Things (IIoT) domain include various strategies to improve system efficiency, reduce communication costs, and protect privacy. For example, (J. He & Li, 2022) used constrained least square restoration and Lucy-Richardson restoration to remove image blur and blind deconvolution restoration to correct motion blur in industrial IoT systems. These methods use an adaptive histogram equalization algorithm to enhance the contrast of digital images collected from industrial IoT and preserve details as much as possible. Moreover, model optimization based on the U-net convolution network enables more efficient analysis of images in industrial applications. Another study that shows that data compression methods play an important role in IIoT systems is by (D. Liu et al., 2021). In this study, they show that the combination of isolated forest algorithm and data compression methods provides an effective solution for anomaly detection and reducing delays in data transmission. In particular, it is emphasized that the isolated forest algorithm outperforms other methods such as K-means clustering. The integration of Ethereum blockchain technology into data management in the context of IIoT is studied by (Toyoda et al., 2019). This study demonstrated the potential of blockchain technology in industrial applications, emphasizing advantages such as data security, transparency, and data integrity. This solution increases security while optimizing data transmission in IIoT systems. On the other hand, (Y. Liu et al., 2021) presented deep anomaly detection methods for time series data in IIoT systems using federated learning approach. This model improves system efficiency by reducing data transmission and at the same time ensures data security and privacy. The authors emphasize that the proposed model provides an effective solution for real-time monitoring and anomaly detection in industrial IoT systems. The integration of data compression methods and federated learning is addressed in (Yang et al., 2024), where it is shown that these technologies are critical for the digital transformation of IIoT systems. In particular, FL optimizes data transmission while minimizing the risks of privacy breaches by enabling local processing of data. In this process, adaptive FL algorithms and resource allocation methods are reported to improve performance. Furthermore, another review by (Barbieri et al., 2024) shows that the integration of data compression methods with federated learning accelerates data transmission in IIoT systems, reducing communication costs and increasing system efficiency. These strategies are a critical element to support the success of IIoT applications. (Tan et al., 2024) In the field of Industrial Internet of Things (IIoT), data compression methods are critical to overcome the challenges faced in data transmission, such as high costs and privacy concerns. In IIoT systems, the large data volumes generated by sensors and devices require effective data management and analysis. In this context, data compression techniques have been used to both reduce data transmission costs and strengthen privacy protection mechanisms. In particular, distributed learning methods such as Federated Learning (FL) allow model updates without sharing local data, but the integration of data compression methods in this process is necessary. Data compression methods play an important role in FL processes. Traditional FL methods carry high communication costs and privacy risks, as local model updates need to be transmitted directly to the server. Therefore, compression techniques can alleviate these problems by reducing the size of the transmitted data. For example, transmitting compressed data increases data security and reduces communication overhead. Moreover, compression methods can also be used to manage different data distributions and heterogeneous data sets, which increases the effectiveness of FL. Privacy protection mechanisms, when integrated with data compression methods, significantly improve the security of FL processes. For example, Differential Privacy (DP) applications can be added on top of compressed data to protect the privacy of user data. In this context, the combination of DP and compression methods not only ensures data security but also reduces communication costs. Such an approach improves the accuracy of the model while protecting the confidentiality of users' data. The effectiveness of data compression methods can be increased with different strategies in FL applications.

Another important work on data compression and security is by (P. Li et al., 2024)This work describes how data compression methods can be used in IIoT systems to achieve energy efficiency and reduce communication costs. It also highlights that compression with quantization and protocols used in data transmission improve the overall reliability of the system.

The effects of data compression methods on the processing and transmission of visual data are discussed in (Jia et al., 2024)Techniques such as saliency detection during visual data processing optimize data transmission and allow for more efficient resource use. These methods offer significant advantages in managing the large data volumes and complex structures encountered in IIoT systems.

Finally, in (Sabbagh et al., 2020), physically inspired data compression and management methods are proposed for industrial data analytics. This study examined the use of distance-based unsupervised clustering methods to organize compressed data and improve its searchability, and showed how efficient these techniques are for industrial data.

As can be seen, the integration of data compression and federated learning methods plays a critical role in improving the efficiency of IIoT systems, protecting privacy and reducing communication costs. Research in this area constitutes an important resource to support the success of industrial IoT applications and offers areas for further development in the future.

Efficiency and security of data transmission are of paramount importance in Industrial Internet of Things (IIoT) applications. In this context, the secure and scalable blockchain system proposed by (Wang et al., 2023) aims to reduce the data load on the network and improve system performance through the integration of data compression methods. The authors state that this approach increases system efficiency while ensuring data security in IIoT applications. (Guo et al., 2022) aims to improve data transmission efficiency with a residual number system (RNS)-based adaptive compression scheme applied on block data and emphasizes that bandwidth savings can be achieved with this method. In addition, the advantages of this approach in terms of data security and integrity are also highlighted. (Qi et al., 2021) presented a system that aims to increase security and efficiency in industrial data sharing. While emphasizing the potential of blockchain technology to increase data security, they aim to reduce data size and lighten the load on the network through the integration of data compression techniques. This approach plays an important role in enabling more effective data management in IIoT applications.

On the other hand, (Eliasson et al., 2015) propose an efficient and interoperable communication framework for the Industrial Internet of Things, focusing on different structures such as multi-agent systems and distributed systems. These systems allow tasks to be performed interactively and offer an effective solution for the design of complex industrial systems. In addressing the security of IIoT, (Serror et al., 2021) noted that the integration of industrial components can lead to security issues. Despite the successes in interconnecting consumer devices, security in the industrial environment presents unique challenges and opportunities. In this context, the need to develop security measures in IIoT systems is emphasized.

(Yu et al., 2021) They aim to improve the monitoring of discrete event systems using time-side information. They demonstrate the applicability of these techniques to optimize bandwidth in data transmission, thus improving system performance in IIoT applications.

(Shamieh & Wang, 2023) presented an innovative system for optimizing data transmission in cloud-hybrid multimedia pipelines in the context of M-IIoT. By efficiently managing data flow, this system aims to both save energy and shorten data transmission times. (Zhang et al., 2022) focused on the integration of machine learning techniques in IIoT data transmission with 6G technology. These methods significantly improve the performance of IIoT systems while increasing data transmission efficiency. (Lei et al., 2023) addressed adaptive compression methods for video services in 5G-U industrial IoT environments, aiming to achieve bandwidth savings in the transmission of video data.

Other studies examine various methods to make data transmission more efficient in IIoT networks. (M. Chen et al., 2022) aims to optimize data transmission with deep packet compression technique, while (Du et al., 2023) aims to increase the efficiency of data flow and learning processes with decentralized federated learning methods. (Du et al., 2023) discusses decentralized federated learning methods in Industrial Internet of Things (IIoT) networks. The study proposes a Markov chain-based consensus mechanism to optimize data flow and reduce network bottlenecks. This approach aims to increase the efficiency of learning processes while improving data privacy. Moreover, the proposed method provides important findings on how federated learning applications can be developed to provide better performance in distributed computing environments.

Hu & Chen (Hu & Chen, 2021) consider the combination of lossy compression and power allocation in low-latency wireless communication and develop a model to improve efficiency. (M. D. V.D. Da Silva et al., 2021) discussed lightweight data compression methods to improve energy efficiency. (Rosenberger et al., 2021) propose various strategies to optimize the processing of streaming data in industrial IoT networks. Finally, (Ahn et al., 2024; Hua et al., 2024) emphasize that data compression methods in IIoT systems are critical for managing and storing large data streams. Both studies indicate that the integration of deep learning-based methods into these processes plays an important role in improving the efficiency of IIoT systems. These methods improve system performance by optimizing data compression processes.

(Yuan & Cai, 2022) present a new method for compressing industrial images, improving data management while preserving image quality. ICHV, introduced by (Yuan & Cai, 2022), is a new compression method for industrial images. It aims to improve data compression efficiency in industrial applications and provides more efficient data management while preserving the quality of images. In this paper, the mathematical foundations and application areas of ICHV are discussed in detail, emphasizing its potential for bandwidth savings in data transmission, especially in the context ofIindustrial Internet of Things (IIoT). ICHV offers an innovative solution to overcome the challenges of processing and transmitting industrial images. Ethernet header compression methods are emerging as an important strategy to improve data transmission efficiency in the industrial internet of things (IIoT). (W.-E. Chen et al., 2019) have carried out an important study on this topic. The authors aim to lighten the load on the network and improve system performance by reducing the size of Ethernet headers. The effectiveness of compression techniques offers potential solutions to improve data transmission processes in IIoT applications. Reducing image compression artifacts is also an important issue in IIoT systems. (J. Li et al., 2023) investigate a proposed network called BARRN to minimize the loss of image quality caused by traditional compression methods. This blind approach is effective in detecting and correcting artifacts introduced in the compression process using deep learning techniques. This model provides better image quality and improves data transmission efficiency in industrial IoT applications.

Data security in Industrial Internet of Things (IIoT) systems should be addressed in conjunction with data compression. (Rani et al., 2024) emphasize the importance of complex algorithms and encryption methods to improve data security in IIoT. As IIoT systems process large amounts of data, securing this data is also critical. In this context, it is possible to improve data security through the use of complex encryption algorithms, especially chaotic systems. The integration of encryption and compression methods increases data transmission efficiency as well as security in IIoT systems. The development of image compression techniques plays a critical role in saving bandwidth and improving network performance in IIoT systems.

The work of (Hajizadeh et al., n.d.) provides important findings in this context. In IIoT applications, transmitting large volumes of images leads to bandwidth issues, and adaptive lossy compression techniques help to overcome this problem. (Hajizadeh et al., n.d.) presents significant innovations in compressing images generated from sensors using K-Means++ and Intelligent Embedded Coding (IEC). These techniques enable more efficient compression of images and improve energy efficiency in IIoT applications.

In manufacturing environments, IIoT devices such as sensors, actuators, and control systems generate vast amounts of data related to production processes, equipment performance, and quality control (Ge et al., 2004). Data compression techniques are essential for: monitoring control and quality assurance. In the energy sector, IIoT applications involve monitoring and managing power generation, distribution, and consumption (Barr et al., n.d.) . IIoT applications in transportation encompass smart logistics, fleet management, and vehicle monitoring systems (Asif et al., 2013). Data compression techniques are utilized for telematics and traffic management systems in transportation. In healthcare, IIoT devices and systems support remote patient monitoring, medical diagnostics, and personalized treatment plans. Data compression techniques are applied to medical imaging and internet of medical things.

Compression Algorithms for IIoT

In the realm of Industrial Internet of Things (IIoT), where efficient data handling is crucial for real-time monitoring, predictive maintenance, and overall system optimization, specific compression algorithms have been developed to address the unique requirements of these environments. Here's an examination of two types of compression algorithms tailored for IIoT scenarios:

Lightweight Compression Algorithms (LCA)

Lightweight Compression Algorithms (LCA) are designed to minimize computational overhead and energy consumption, making them ideal for resource-constrained IIoT devices such as sensors and actuators. These algorithms focus on achieving reasonable compression ratios with minimal processing power and memory usage. Some notable LCAs include:

LZW (Lempel-Ziv-Welch): A variant of the classic LZ77 algorithm, LZW is lightweight and efficient for compressing text and structured data in IIoT applications. It builds a dictionary dynamically as it processes data, enabling effective compression of repetitive patterns.

FLAC (Free Lossless Audio Codec): While originally designed for audio compression, FLAC's lightweight nature and efficient compression ratios make it suitable for IIoT applications where high-fidelity audio monitoring or transmission is required without significant computational overhead (X. He & Cai, 2024).

Simple8b: This algorithm is tailored for compressing time-series data commonly generated by sensors in IIoT environments(Blalock et al., 2018). It uses a simple bit-packing technique to achieve compression while maintaining compatibility with 64-bit systems, making it suitable for data storage and transmission efficiency (Anh & Moffat, 2010).

Predictive Compression Techniques

Predictive Compression Techniques leverage predictive modeling and algorithms to achieve higher compression ratios by exploiting data correlations and trends over time. These techniques are particularly effective in scenarios where data exhibits predictable patterns or where historical data can be used to forecast future values. Examples include:

Delta Encoding: Also known as delta compression, this technique stores the difference between successive data points instead of the absolute values. It is effective for compressing time-series data with incremental changes, such as temperature readings or stock market data(Al-Qurabat et al., 2022).

Linear Prediction Coding: This method predicts future data points based on previous values and encodes the prediction error. It is commonly used in speech and audio compression, where temporal dependencies can be exploited to achieve higher compression ratios.

DPCM (Differential Pulse Code Modulation): DPCM predicts the next sample in a signal and encodes the difference between the predicted value and the actual sample value. It is suitable for compressing data streams with predictable variations, such as video or sensor data.

Application in IIoT Environments

In IIoT environments, where data transmission bandwidth and energy efficiency are critical considerations, LCAs and predictive compression techniques play vital roles:

Energy Efficiency: LCAs reduce the computational load on IIoT devices, extending battery life and reducing power consumption, which is essential for remote and battery-operated sensors in industrial settings.

Real-Time Data Processing: Predictive compression techniques enable efficient real-time data analysis and decision-making by reducing the amount of data transmitted and processed without sacrificing accuracy or reliability.

Data Integrity: Both LCAs and predictive techniques ensure that compressed data retains its integrity, making it suitable for critical applications such as predictive maintenance, process monitoring, and quality control in manufacturing and energy sectors.

Challenges and Considerations

One of the fundamental challenges is balancing the compression ratio with the quality of compressed data. In lossy compression techniques, reducing data size often involves sacrificing some level of data fidelity. Finding the right balance is crucial, especially in applications where accurate data representation is critical, such as in medical diagnostics or precision manufacturing. Ob the other hand, compression and decompression processes require computational resources, including CPU cycles and memory. For resource-constrained IIoT devices like sensors and edge devices, excessive computational overhead can lead to increased power consumption, reduced battery life, and slower response times. Lightweight compression algorithms (LCAs) are developed to address these constraints, but they must strike a balance between compression efficiency and computational cost.

IIoT systems often operate in real-time environments where timely data processing and response are essential. The latency introduced by compression and decompression processes can impact the responsiveness of critical applications such as predictive maintenance and emergency alerts. Optimizing compression algorithms to minimize latency while maintaining effective data reduction is a significant challenge. IIoT ecosystems involve diverse hardware platforms, communication protocols, and data formats. Ensuring compatibility and interoperability of compression techniques across different devices and systems is essential for seamless integration and data exchange. Standardization efforts play a crucial role in addressing these compatibility challenges.

While compression can enhance efficiency, it may also introduce vulnerabilities if not implemented securely. Compression algorithms should include robust encryption and authentication mechanisms to protect compressed data from unauthorized access, tampering, or interception. Security protocols must be integrated into compression frameworks to mitigate potential risks in IIoT environments. IIoT deployments often scale up rapidly, involving an increasing number of devices and data sources. Compression techniques must be scalable to accommodate growing data volumes without compromising performance or scalability. Flexibility in adapting compression strategies to evolving IIoT requirements is crucial for long-term viability.

Conclusions

IIoT environments generate vast amounts of data from sensors, actuators, and connected devices. Data compression techniques, including lightweight algorithms (LCAs) and predictive methods, enable efficient storage, transmission, and processing of this data while minimizing bandwidth usage and computational overhead. LCAs are particularly beneficial for resource-constrained devices in IIoT, such as sensors and edge devices. These algorithms strike a balance between compression ratios and computational efficiency, extending battery life and reducing operational costs. Predictive compression techniques leverage data correlations and trends to reduce the size of data streams without compromising real-time analysis and decision-making capabilities. This is crucial for applications like predictive maintenance and process optimization where timely insights are essential. As IIoT deployments scale up, efficient data compression becomes essential to manage the increasing volume and velocity of data generated by interconnected devices. Compression algorithms ensure that IIoT systems can handle large-scale deployments without overwhelming network infrastructure or backend processing resources. Despite compression, maintaining data integrity and security remains paramount in IIoT. Effective compression techniques ensure that compressed data retains accuracy and reliability, supporting critical applications such as remote monitoring, quality control, and compliance with regulatory standards.

By reducing data size, compression techniques optimize network bandwidth and storage requirements, lowering operational costs and improving system performance. IIoT systems equipped with efficient compression algorithms can scale seamlessly, accommodating growing numbers of connected devices and data-intensive applications without sacrificing performance. Compression enables faster transmission and processing of data, facilitating real-time analytics and decision-making that are essential for proactive maintenance, operational efficiency, and rapid response to dynamic industrial conditions. Reduced data transmission and storage requirements through compression contribute to lower energy consumption and carbon footprint, aligning with sustainability goals in industrial operations. In conclusion, data compression techniques play a pivotal role in enabling efficient, scalable, and responsive IIoT systems. As industries continue to embrace digital transformation and adopt more interconnected technologies, the optimization offered by compression algorithms will be critical in harnessing the full potential of IIoT for driving innovation, competitiveness, and sustainability in industrial sectors worldwide.

References

Ahn, H., Lee, M., Seong, S., Na, G. J., Chun, I. G., Varghese, B., & Hong, C. H. (2024). ScissionLite: Accelerating Distributed Deep Learning With Lightweight Data Compression for IIoT. *IEEE Transactions on Industrial Informatics*. https://doi.org/10.1109/TII.2024.3413340

Al-Qurabat, A. K. M., Idrees, A. K., Makhoul, A., & Jaoude, C. A. (2022). A Bi-Level Data Lowering Method to Minimize Transferring Big Data in the Sensors of IoT Applications. *Karbala International Journal of Modern Science*, *8*(2), 246–261. https://doi.org/10.33640/2405-609X.3228

Anh, V. N., & Moffat, A. (2010). Index compression using 64-bit words. *Software - Practice and Experience*, *40*(2), 131–147. https://doi.org/10.1002/spe.948

Asif, M. T., Kannan, S., Dauwels, J., & Jaillet, P. (2013). Data compression techniques for urban traffic data. *2013 IEEE Symposium on Computational Intelligence in Vehicles and Transportation Systems (CIVTS)*, 44–49. https://doi.org/10.1109/CIVTS.2013.6612288

Barbieri, L., Savazzi, S., & Nicoli, M. (2024). Compressed Bayesian Federated Learning for Reliable Passive Radio Sensing in Industrial IoT. *Proceedings - 2024 IEEE Conference on Artificial Intelligence, CAI 2024*, 344–349. https://doi.org/10.1109/CAI59869.2024.00071

Barr, K. C., Asanovi´c, K., & Asanovi´c, A. (n.d.). *Energy-Aware Lossless Data Compression*.

Blalock, D., Madden, S., & Guttag, J. (2018). Sprintz: Time Series Compression for the Internet of Things. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies*, *2*(3), 1–23. https://doi.org/10.1145/3264903

Chen, M., Lu, H., Xu, X., Tang, X. W., Fan, Q., & Lou, C. (2022). Novel Deep Packet Compression For Industrial Internet of Things. *2022 IEEE GLOBECOM Workshops, GC Wkshps 2022 - Proceedings*, 1628–1633. https://doi.org/10.1109/GCWkshps56602.2022.10008589

Chen, S., Wang, Z., Zhang, H., Yang, G., & Wang, K. (2020). Fog-based Optimized Kronecker-Supported Compression Design for Industrial IoT. *IEEE Transactions on Sustainable Computing*, *5*(1), 95–106. https://doi.org/10.1109/TSUSC.2019.2906729

Chen, W.-E., Chien, W.-C., Lai, C.-F., & Chao, H.-C. (2019). Promising Framework of Ethernet Header Compression in Industrial Internet of Things. *2019 International Conference on Intelligent Computing and Its Emerging Applications (ICEA)*, 134–139. https://doi.org/10.1109/ICEA.2019.8858285

Da Silva, M. D. V.D., Rocha, A., Gomes, R. L., & Nogueira, M. (2021, January 9). Lightweight data compression for low energy consumption in industrial internet of things. *2021 IEEE 18th Annual Consumer Communications and Networking Conference, CCNC 2021*. https://doi.org/10.1109/CCNC49032.2021.9369520

Du, M., Zheng, H., Feng, X., Chen, Y., & Zhao, T. (2023). Decentralized Federated Learning With Markov Chain Based Consensus for Industrial IoT Networks. *IEEE Transactions on Industrial Informatics*, *19*(4), 6006–6015. https://doi.org/10.1109/TII.2022.3192297

Eliasson, J., Delsing, J., Derhamy, H., Salcic, Z., & Wang, K. (2015). Towards industrial Internet of Things: An efficient and interoperable communication framework. *2015 IEEE International Conference on Industrial Technology (ICIT)*, 2198–2204. https://doi.org/10.1109/ICIT.2015.7125421

Ge, M., Xu, Y., & Tarn, T. J. (2004). Kernel-based data compression for internet-based manufacturing monitoring. *The International Journal of Advanced Manufacturing Technology*, *24*(9–10), 693–699. https://doi.org/10.1007/s00170-003-1746-3

Guo, Z., Gao, Z., Liu, Q., Chakraborty, C., Hua, Q., Yu, K., & Wan, S. (2022). RNS-Based Adaptive Compression Scheme for the Block Data in the Blockchain for IIoT. *IEEE Transactions on Industrial Informatics*, *18*(12), 9239–9249. https://doi.org/10.1109/TII.2022.3182766

Hajizadeh, N., Shamsinejad, P., & Javidan, R. (n.d.). *A New Approach to Image Compression in Industrial Internet of Things*. https://doi.org/doi:10.48550/arXiv.2112.05361

He, J., & Li, C. (2022). Research on Digital Image Intelligent Recognition Method for Industrial Internet of Things Production Data Acquisition. *Traitement Du Signal*, *39*(6), 2133–2139. https://doi.org/10.18280/ts.390626

He, X., & Cai, Z. (2024). Complex System-Based Audio Representation for Industrial Information Security. *IEEE Transactions on Industrial Informatics*. https://doi.org/10.1109/TII.2024.3395151

Hosseinzadeh, M., Hudson, N., Zhao, X., Khamfroush, H., & Lucani, D. E. (2021). Joint Compression and Offloading Decisions for Deep Learning Services in 3-Tier Edge Systems. *2021 IEEE International Symposium on Dynamic Spectrum Access Networks, DySPAN 2021*, 254–261. https://doi.org/10.1109/

DySPAN53946.2021.9677398

Hu, S., & Chen, W. (2021). Joint Lossy Compression and Power Allocation in Low Latency Wireless Communications for IIoT: A Cross-Layer Approach. *IEEE Transactions on Communications*, *69*(8), 5106–5120. https://doi.org/10.1109/TCOMM.2021.3077948

Hua, M., Zhang, Y., Sun, J., Adebisi, B., Ohtsuki, T., Gui, G., Wu, H. C., & Sari, H. (2024). Specific Emitter Identification Using Adaptive Signal Feature Embedded Knowledge Graph. *IEEE Internet of Things Journal*, *11*(3), 4722–4734. https://doi.org/10.1109/JIOT.2023.3300689

Jagannath, J., Polosky, N., Jagannath, A., Restuccia, F., & Melodia, T. (2019). Machine learning for wireless communications in the Internet of Things: A comprehensive survey. *Ad Hoc Networks*, *93*. https://doi.org/10.1016/j.adhoc.2019.101913

Jia, N., Liu, X., Sun, Y., & Liu, Z. (2024). Enhancing IIoT Vision Data Transmission and Processing via Spatial-Difference-Attention-Guided Saliency Detection. *IEEE Internet of Things Journal*, *11*(16), 26668–26679. https://doi.org/10.1109/JIOT.2023.3343758

Kong, D., Liu, D., Zhang, L., He, L., Shi, Q., & Ma, X. (2020). Sensor anomaly detection in the industrial internet of things based on edge computing. *Turkish Journal of Electrical Engineering and Computer Sciences*, *28*(1), 331–346. https://doi.org/10.3906/elk-1906-55

Kuldeep, G., & Zhang, Q. (2021). A Novel Efficient Secure and Error-Robust Scheme for Internet of Things Using Compressive Sensing. *IEEE Access*, *9*, 40903–40914. https://doi.org/10.1109/ACCESS.2021.3064700

Kumar, A. S., & Srinivasan, S. (2023). Metaheuristics with Vector Quantization Enabled Codebook Compression Model for Secure Industrial Embedded Environment. *Intelligent Automation and Soft Computing*, *36*(3), 3607–3620. https://doi.org/10.32604/iasc.2023.036647

Lei, J., Yang, P., Kong, L., Ma, Y., Lu, X., Lin, D., Chen, G., & Zhao, E. (2023). ACORN: Adaptive Compression-Reconstruction for Video Services in 5G-U Industrial IoT. *Proceedings - 2023 19th International Conference on Mobility, Sensing and Networking, MSN 2023*, 285–292. https://doi.org/10.1109/MSN60784.2023.00051

Li, J., Liu, X., Gao, Y., Zhuo, L., & Zhang, J. (2023). BARRN: A Blind Image Compression Artifact Reduction Network for Industrial IoT Systems. *IEEE Transactions on Industrial Informatics*, *19*(9), 9479–9490. https://doi.org/10.1109/TII.2022.3228777

Li, P., Ren, C., Chen, C., Zhu, S., Ma, Y., & Guan, X. (2024). Lifetime Reliability Aware Distributed Estimation and Communication Co-Design for IIoT Systems. *IEEE Transactions on Industrial Informatics*, 1–11. https://doi.org/10.1109/tii.2024.3441625

Liu, D., Zhen, H., Kong, D., Chen, X., Zhang, L., Yuan, M., & Wang, H. (2021). Sensors Anomaly Detection of Industrial Internet of Things Based on Isolated Forest Algorithm and Data Compression. *Scientific Programming*, *2021*. https://doi.org/10.1155/2021/6699313

Liu, Y., Garg, S., Nie, J., Zhang, Y., Xiong, Z., Kang, J., & Hossain, M. S. (2021). Deep Anomaly Detection for Time-Series Data in Industrial IoT: A Communication-Efficient On-Device Federated Learning Approach. *IEEE Internet of Things Journal*, *8*(8), 6348–6358. https://doi.org/10.1109/JIOT.2020.3011726

Meier, S., Erkan, A., Thielen, N., Klarmann, S., & Franke, J. (2022). Attention-based Image

Compression in Sensor Assembly. *2022 IEEE 28th International Symposium for Design and Technology in Electronic Packaging, SIITME 2022 - Conference Proceedings*, 136–141. https://doi.org/10.1109/SIITME56728.2022.9987988

Qi, S., Lu, Y., Zheng, Y., Li, Y., & Chen, X. (2021). Cpds: Enabling Compressed and Private Data Sharing for Industrial Internet of Things over Blockchain. *IEEE Transactions on Industrial Informatics*, *17*(4), 2376–2387. https://doi.org/10.1109/TII.2020.2998166

Rani, D., Gill, N. S., & Gulia, P. (2024). A forensic framework to improve digital image evidence administration in IIoT☆. *Journal of Industrial Information Integration*, *38*. https://doi.org/10.1016/j.jii.2024.100568

Rosenberger, J., Buhren, M., & Schramm, D. (2021). Perspective on efficiency enhancements in processing streaming data in industrial IoT networks. *2021 IEEE Global Conference on Artificial Intelligence and Internet of Things, GCAIoT 2021*, 120–121. https://doi.org/10.1109/GCAIoT53516.2021.9693073

Sabbagh, R., Cai, Z., Stothert, A., & Djurdjanovic, D. (2020). Physically Inspired Data Compression and Management for Industrial Data Analytics. *Frontiers in Computer Science*, *2*. https://doi.org/10.3389/fcomp.2020.00041

Senthil, T., & Kannapiran, B. (2017). ECTMRA: Energy Conserving Trustworthy Multipath Routing Algorithm Based on Cuckoo Search Algorithm. *Wireless Personal Communications*, *94*(4), 2239–2258. https://doi.org/10.1007/s11277-016-3378-6

Serror, M., Hack, S., Henze, M., Schuba, M., & Wehrle, K. (2021). Challenges and Opportunities in Securing the Industrial Internet of Things. *IEEE Transactions on Industrial Informatics*, *17*(5), 2985–2996. https://doi.org/10.1109/TII.2020.3023507

Shamieh, F., & Wang, X. (2023). M-IIoT System for Reducing Bandwidth Costs in Cloud-Hybrid Multimedia Pipelines. *2023 IEEE World Forum on Internet of Things: The Blue Planet: A Marriage of Sea and Space, WF-IoT 2023*. https://doi.org/10.1109/WF-IoT58464.2023.10539536

Silva, M. V., Mosca, E. E., & Gomes, R. L. (2022). Green industrial internet of things through data compression. *International Journal of Embedded Systems*, *15*(6), 457. https://doi.org/10.1504/IJES.2022.129802

Singh, S. (2020). An energy aware clustering and data gathering technique based on nature inspired optimization in WSNs. *Procedia Comput Sci*, *94*(4), 2230–2237. https://doi.org/10.1007/s12083-020-00890-w/Published

Singh, S., Kumar, M., Verma, O. P., Kumar, R., & Gill, S. S. (2023). An IIoT based secure and sustainable smart supply chain system using sensor networks. *Transactions on Emerging Telecommunications Technologies*, *34*(2). https://doi.org/10.1002/ett.4681

Sujitha, B., Parvathy, V. S., Lydia, E. L., Rani, P., Polkowski, Z., & Shankar, K. (2021). Optimal deep learning based image compression technique for data transmission on industrial Internet of things applications. *Transactions on Emerging Telecommunications Technologies*, *32*(7). https://doi.org/10.1002/ett.3976

Tan, X., Xiao, D., Huang, H., Wang, M., & Li, M. (2024). Hierarchically Fair and Differentially Private Federated Learning in Industrial IoT Based on Compressed Sensing With Adaptive-Thresholding Sparsification. *IEEE Transactions on Industrial Informatics*, 1–10. https://doi.org/10.1109/tii.2024.3431100

Toyoda, K., Shakeri, M., Chi, X., & Zhang, A. N. (2019). Performance Evaluation of Ethereum-based On-chain Sensor Data Management Platform for Industrial

IoT. *2019 IEEE International Conference on Big Data (Big Data)*, 3939–3946. https://doi.org/10.1109/BigData47090.2019.9005582

Tsai, C., Shih, W.-Y., Lu, Y.-S., Huang, J.-L., & Yeh, L.-Y. (2019). Design of a Data Collection System with Data Compression for Small Manufacturers in Industrial IoT Environments. *2019 20th Asia-Pacific Network Operations and Management Symposium (APNOMS)*, 1–4. https://doi.org/10.23919/APNOMS.2019.8893004

Wang, T., Liu, J., Liu, T., Zhai, D., Wang, J., & Zhang, T. (2023). Secure and Scalable Blockchain for IIoT with Dual Compression Scheme. *2023 IEEE International Conferences on Internet of Things (iThings) and IEEE Green Computing & Communications (GreenCom) and IEEE Cyber, Physical & Social Computing (CPSCom) and IEEE Smart Data (SmartData) and IEEE Congress on Cybermatics (Cybermatics)*, 1–6. https://doi.org/10.1109/iThings-GreenCom-CPSCom-SmartData-Cybermatics60724.2023.00025

Wen, L., Zhou, K., Yang, S., & Li, L. (2018). Compression of smart meter big data: A survey. In *Renewable and Sustainable Energy Reviews* (Vol. 91, pp. 59–69). Elsevier Ltd. https://doi.org/10.1016/j.rser.2018.03.088

Yang, W., Yang, Y., Xiang, W., Yuan, L., Yu, K., Alonso, Á. H., Ureña, J. U., & Pang, Z. (2024). Adaptive optimization federated learning enabled digital twins in industrial IoT. *Journal of Industrial Information Integration*, *41*. https://doi.org/10.1016/j.jii.2024.100645

Yu, S., Chen, W., & Poor, H. V. (2021). Timing Side Information Aided Real-Time Monitoring of Discrete-Event Systems. *Proceedings - IEEE Global Communications Conference, GLOBECOM*. https://doi.org/10.1109/GLOBECOM46510.2021.9685563

Yuan, X., & Cai, Z. (2022). ICHV: A New Compression Approach for Industrial Images. *IEEE Transactions on Industrial Informatics*, *18*(7), 4427–4435. https://doi.org/10.1109/TII.2021.3125375

Zehnder, P., Wiener, P., & Riemer, D. (2019). Using Virtual Events for Edge-based Data Stream Reduction in Distributed Publish/Subscribe Systems. *2019 IEEE 3rd International Conference on Fog and Edge Computing (ICFEC)*, 1–10. https://doi.org/10.1109/CFEC.2019.8733146

Zhang, M., Zhang, H., Fang, Y., & Yuan, D. (2022). Learning-Based Data Transmissions for Future 6G Enabled Industrial IoT: A Data Compression Perspective. *IEEE Network*, *36*(5), 180–187. https://doi.org/10.1109/MNET.109.2100384

Zhang, M., Zhang, H., Zhang, C., & Yuan, D. (2023). Communication-Efficient Quantized Deep Compressed Sensing for Edge-Cloud Collaborative Industrial IoT Networks. *IEEE Transactions on Industrial Informatics*, *19*(5), 6613–6623. https://doi.org/10.1109/TII.2022.3202203

About the Authors

Halil İbrahim AYAZ, PhD, He earned a BS and an MS degree in Industrial Engineering from Hacettepe University. He earned a PhD degree in Industrial Engineering from Eskişehir Technical University. His research interests include machine learning, data mining and analysis, scheduling, and optimization. He has two journal articles and seven studies presented at international meetings. Currently, he works as an assistant professor doctor in the Industrial Engineering department at Necmettin Erbakan University.
E-mail: hiayaz@erbakan.edu.tr, **ORCID:** 0000-0001-5547-6485

Similarity Index

The similarity index obtained from the plagiarism software for this book chapter is 15%.

Aerospace and Intelligent Systems

Rabia TOPRAK

Karamanoglu Mehmetbey University

To Cite This Chapter

Toprak, R. (2024). Aerospace and Intelligent Systems. In S. Koçer & Ö. Dündar (Eds.), *Intelligent Systems and Optimization in Engineering* (pp. 23-32). ISRES Publishing.

Introduction

Aerospace" can be mentioned as "aviation and space" and refers to a broad branch of engineering and science that deals with both aircraft flying in the atmosphere (such as airplanes and helicopters) and spacecraft moving in space (such as satellites, space shuttles, rockets) (Krishnakumar, 2003).

This term covers two main areas (Cohen, 2023):

Aeronautics: It deals with the design, development, and use of vehicles moving in the atmosphere. For example, airplanes, helicopters, and unmanned aerial vehicles fall into this category. Aeronautical engineering includes aerodynamics, flight mechanics, aircraft engines, and aircraft design.

Space (Astronautics): It deals with vehicles and technologies that go outside the atmosphere, that is, into space. This field includes satellites, rockets, space probes, and manned spacecraft. Space engineering covers orbital dynamics, space mechanics, rocket propulsion systems, and space mission planning.

Aviation and space are often considered together because the two fields overlap in many aspects of technology, engineering principles, and research methods. For example, aircraft and spacecraft require similar techniques in aerodynamics, materials science, propulsion systems, and control engineering. For this reason, the branch of engineering that covers these areas is called "aerospace engineering" (Keane & Scanlan, 2007).

The aerospace has various advantages and disadvantages (Chhaya, Khanzode, & Sarode, 2020; Insaurralde, 2018). This field provides great contributions to science, technology and economy in both manned and unmanned projects, but it also carries some important limitations and risks. The advantages of aerospace are like this:

Contribution to scientific progress; space research and aviation projects enable important discoveries to be made in many fields such as astrophysics, earth sciences and atmospheric science. Thanks to satellite technology, significant progress is made in vital areas such as weather forecasts, natural disaster warnings and environmental monitoring (Zhang, Chen, & Hu, 2018).

Economic returns; the aerospace sector creates high employment rates and opens up new business areas. Especially agreements made between large companies and government organizations directly contribute to the economy of countries (Tschan, Kivelevitch, & Melcher, 2013).

Developments in security and defense; aviation and space technologies are of critical importance for military and civil security. National defense systems are strengthened thanks to satellite surveillance, airspace control and other security measures (Ranasinghe et al., 2022a).

Development of new technologies; the aerospace sector is the pioneer of important innovations in areas such as new materials, Artificial Intelligence (AI), robotics and telecommunications due to its high-tech requirements. These developments also contribute to other sectors and are used in daily life (Mieloszyk, 2017).

Exploration and innovation opportunities; exploration of space seeks answers to big questions such as the possibility of life on other planets or learning about worlds outside the solar system. These discoveries open new horizons for humanity and fuel scientific curiosity (Van der Velden, Bil, Yu, & Smith, 2007).

There is some disadvantages of the aerospace (Insaurralde, 2018). The drawbacks of aerospace are like this:

High cost and economic risks; aviation and space projects require large budgets and the return on investment may be uncertain. Such high costs may lead to criticism, especially in projects carried out with public funds (Altavilla & Garbellini, 2002).

Environmental impacts; high-energy-consuming vehicles such as rocket launches and jet engines can harm the environment by releasing greenhouse gases and harmful chemicals into the atmosphere. This increases the need for environmentally friendly technologies in the sector (Mahashabde et al., 2011).

High safety and health risks; astronauts and pilots who take part in both space missions and aviation projects are at high risk. The smallest error or technical failure can pose life-threatening risks. This can lead to both physical and mental health issues for the staff (Oster, Strong, & Zorn, 2013).

Technical and legal complexity; the international and national regulations that must be adhered to in this sector are quite complex. This complexity can lead to project delays and complicate development processes. Additionally, the technology used in each project must undergo legal procedures before it can be shared with other countries (Shelukhin, Kharaberiush, Shelukhin, Tsymbalistyi, & Selevko, 2022).

High risk of failure; space projects and large aviation initiatives require long-term planning, testing, and development to achieve success. However, many of these projects can face cancellation or failure due to technical or financial reasons, leading to a waste of time and resources (Harikumaran et al., 2020).

The aerospace sector significantly contributes to advancements in science and technology due to its many advantages. However, it also presents challenges, including high costs, security risks, and environmental impacts. Therefore, it is essential to develop innovative and sustainable solutions to address these issues and promote the sector's growth (Zhang et al., 2018).

The aerospace sector is highly complex and carries significant risks, continuously pushing the boundaries of technology. The main challenges faced in this field include (Baur & Silverman, 2007; Trélat, 2012): High Costs: Aerospace projects often require substantial financial investment, with research, development, and production processes costing billions of dollars, particularly for projects like rockets and spacecraft (Mahashabde et al., 2011). Technical Complexity: Aerospace vehicles incorporate intricate systems that add to their overall complexity. Overall, these factors make navigating the aerospace industry both challenging and costly. Engineers must integrate various disciplines, such as aerodynamics, materials science, software engineering, and physics, to ensure that each component functions effectively. For instance, adapting a rocket to different atmospheric conditions presents a significant engineering challenge (Oche, Ewa, & Ibekwe, 2024). Safety is one of the most critical aspects of the aerospace industry; even the smallest error can result in millions of dollars in damages or, worse, loss of life. Therefore, systems must undergo multi-layered safety protocols and controls. The aerospace industry is governed by strict regulations and compliance requirements. Compliance with the standards set by various national and international regulatory bodies often increases the cost and duration of projects. The materials and technologies developed in this field must be both durable and lightweight. Material science focuses on creating strong, lightweight materials that can withstand extreme temperatures and pressures. Research into new materials and technologies also requires substantial funding and advanced expertise.

The environmental impact of the aviation and space industries is frequently criticized, particularly concerning their carbon footprints (Mahashabde et al., 2011). Rocket launches, for instance, consume large amounts of fuel and release greenhouse gases, such as carbon dioxide, into the atmosphere. Efforts are underway to develop new, environmentally friendly technologies to mitigate these impacts.

Space exploration presents numerous uncertainties, as it involves journeys into the unknown. Conditions in space cannot be fully predicted, leading to unexpected challenges. The harsh environment of space—characterized by radiation, temperature fluctuations, and microgravity—places extreme stress on vehicles (Brevault, Balesdent, & Morio, 2020).

The implementation of AI and intelligent systems in the aerospace sector offers substantial benefits in terms of safety and efficiency. Applications such as autonomous systems, predictive maintenance, data analysis, and simulations are driving significant advancements in reliability and operational success (Bautista-Montesano, López-Valdés, Jiménez-Ríos, & Gómez-Aladro, 2019; Yang & Zhao, 2004). By utilizing these technologies, risks associated with aerospace projects are minimized, costs are lowered, and human resources are optimized effectively.

Several solutions are available to address these challenges (Léonard, Hallstedt, Nylander, & Isaksson, 2024). One of these solutions is intelligent systems, which include AI and its applications. This section explains AI applications in aerospace across four key areas. These are Remote Sensing Systems, Spacecraft Health Monitoring System, Satellite Communication System and Autonomous Robotic Systems.

Remote Sensing Systems

Remote sensing systems in the aerospace field are technologies used to collect, analyze and interpret data from the air and space. These systems use various sensors to provide information about the earth, atmosphere and space, collecting data without direct physical contact with the areas to be observed (Wang, 2007; Wu, 2024).

Remote sensing systems collect data using electromagnetic waves, which bounce off specific targets and are analyzed by sensors (Sze, Isaacs, Ko, & McElroy, 1981). Active and passive sensors can be used. These sensors are carried on satellites, spacecraft or aircraft. There are various application areas in remote sensing. Some of these are; meteorology and weather forecasting, forest fire monitoring, natural disaster management, military and defense applications, environmental monitoring and climate research.

Nowadays, large amounts of data from remote sensing systems are analyzed with AI algorithms. Machine learning and deep learning techniques provide great benefits, especially in the analysis of satellite images, target detection, change monitoring and classification. With these analyses, faster and more accurate results are achieved in areas such as disaster prediction, plant health assessment or pollution monitoring (Merhav, 2012).

Aerospace remote sensing systems are vital in air, space and ground observations. With the development of sensors and the increase in AI-supported data analysis, remote sensing systems provide more comprehensive and reliable solutions by collecting and analyzing data in various scientific and practical fields. These systems play an important role in both daily life and solving global problems.

Spacecraft Health Monitoring System

Spacecraft health monitoring systems are designed to oversee the condition of various components within a spacecraft, ensuring both safety and operational efficiency. These systems continuously track the status of all vehicle systems, identify potential issues, and facilitate preventive maintenance. By playing a crucial role in mission success and prolonging vehicle lifespan, health monitoring systems are essential for the overall effectiveness of space missions (Ranasinghe et al., 2022b; Tipaldi & Bruenjes, 2014).

Spacecraft have the ability to assess their own status and communicate this information to ground control using onboard AI and machine learning algorithms. These systems analyze different situations, evaluate performance data, and make decisions based on historical information. AI models enhance data analysis and can predict potential failures or maintenance needs by examining the spacecraft's current performance.

Spacecraft have the capability to monitor their own status and communicate with ground control using onboard AI and machine learning algorithms. These systems analyse different situations, evaluate performance data, and make informed decisions based on historical data. AI models enhance data analysis and can predict future failures or maintenance needs by assessing the spacecraft's current performance.

Health monitoring systems are essential components of aerospace technologies, playing a crucial role in both the operational and safety aspects of space missions. When combined with AI and machine learning applications, these systems enhance the efficiency, safety, and long-term operation of spacecraft (Herrera & Chura, 2005).

Software-Based System Health Management (SHM) employs software solutions to monitor the status of spacecraft, predict potential failures, and ensure safe operation throughout the mission. SHM consists of algorithms and software that continuously assess the complex structure of spacecraft and recommend or implement the most suitable interventions when problems arise. These systems optimize vehicle performance using built-in software algorithms and AI techniques, minimizing the need for human intervention (Bao, Bao, & Qiu, 2022).

Sensor Data Collection and Monitoring

Sensors installed throughout the spacecraft continuously monitor essential parameters such as temperature, pressure, vibration, radiation, and voltage. The data collected from these sensors is transmitted to software-based system health management units. This software instantly analyzes the sensor data. For instance, if an abnormality is detected in the engine temperature, the software can identify it and assess the likelihood of a failure (Harris, Bailey, & Dodd, 1998).

Data Analysis and Anomaly Detection

The sensor data collected is meticulously compared to well-defined normal operating conditions. By leveraging advanced machine learning algorithms and statistical analysis methods, the software identifies any anomalies present in the data. For example, it scrutinizes unusual patterns like pressure fluctuations or temperature spikes, enabling it to forecast and alert on the likelihood of failures before they occur. This proactive approach ensures that preventive maintenance can be strategically planned, safeguarding operations and preventing costly disruptions (Basora, Olive, & Dubot, 2019).

Fault Detection and Diagnostics (FDD)

FDD (Fault Detection and Diagnosis) plays a crucial role in managing the health of software-based systems. FDD software analyzes data from sensors to predict potential failures. By using machine learning or deep learning algorithms trained on historical data, these systems can improve their failure predictions. Predictive systems not only anticipate when failures might occur but also pinpoint the locations of the problems. This capability enables the vehicle control center or onboard systems to automatically take corrective actions (Ezzat et al., 2021).

Autonomous Response Systems and Intervention

SHM has the ability to respond immediately in the event of a failure. When a problem is detected, the system can either automatically intervene or notify the control center, depending on the severity of the situation. For instance, if high temperatures are detected in a section of the vehicle, the software system activates cooling units or temporarily reduces power. Autonomous response systems are especially critical during deep space missions, where communication latency can be significant (Bautista-Montesano et al., 2019).

Machine Learning and AI Applications

Structural Health Monitoring (SHM) utilizes AI and machine learning models to analyze data more effectively. These systems can learn from historical datasets to detect anomalies with greater accuracy and predict potential failures. By analyzing thousands of hours of operational data, machine learning can assess failure probabilities. This capability enables SHM to enhance vehicle performance in real time (Ezzat et al., 2021; Sathyan, n.d.).

Preventive Maintenance and Performance Optimization

SHM offers preventive maintenance recommendations that aid in planning maintenance before failures occur. As a result, most issues can be addressed before they happen, ensuring system safety throughout the mission and reducing costs. Additionally, these systems conduct analyses to optimize vehicle performance, promoting long-term and efficient operation (Herrera & Chura, 2005).

Benefits of Software-Based System Health Management

System Health Management (SHM) enhances mission reliability by continuously monitoring the status of systems and preventing unexpected failures that could disrupt mission operations. Early detection of issues allows for intervention before a failure occurs, ultimately reducing maintenance costs (Tipaldi & Bruenjes, 2014).

Software-based SHM is particularly crucial for ensuring astronaut safety, as it identifies potential safety risks in advance and implements necessary precautions. Additionally, SHM can make autonomous decisions without requiring human intervention, which is a vital advantage for deep space missions where communication delays may occur.

In aerospace projects, SHM technology is of great significance. Powered by AI and machine learning, these software systems enable spacecraft to carry out their tasks safely and efficiently while minimizing risks and ensuring cost-effectiveness.

Satellite Communication Systems

Satellite communication systems use satellites to transmit data between various locations around the World (Kodheli et al., 2020). They are essential in numerous fields, including communication, television broadcasting, internet connectivity, military communication, and emergency services. AI (AI) applications within these systems enhance data transfer speed and reliability, improve fault management, optimize network traffic, and enhance the overall user experience.

These algorithms used to optimize network traffic analyze traffic patterns to predict which network paths are busiest and optimize traffic management (Fourati & Alouini, 2021). This increases data transmission speed and minimizes delays. Additionally, AI algorithms increase the efficiency of satellite communication systems by identifying unused or low-density frequencies and prevent signal interference. Also, for healthy communication and maintenance management, AI analyzes data from sensors to predict possible malfunctions in satellites. In this way, satellite maintenance and repair operations can be carried out before malfunctions occur.

In data transmission, image, audio and data packets are compressed by AI algorithms and sent (Abdulwahid & Kurnaz, 2024). This provides faster and higher quality data transfer, especially in video or high data volume transmissions. AI algorithms ensure that the satellite position remains stable and make automatic route adjustments when needed, thus ensuring precise and appropriate orbit management, especially for satellites in low Earth orbit.

In addition, AI provides optimum data transmission by filtering out unwanted harmonics and making the signal stronger (Keane & Scanlan, 2007; Kim, Kennedy, & Gürdal, 2008; Van der Velden et al., 2007). This application is especially useful in very bad weather conditions or in areas with heavy electromagnetic pollution. AI analyzes users' connections and their needs, adjusting frequency, bandwidth, and data flow accordingly. It offers users a faster and uninterrupted connection, especially in internet services.

The benefits of AI in Satellite Communication Systems can be listed as follows: AI increases efficiency in communication systems by performing traffic management, data compression and frequency optimization. In addition, satellites/aircrafts become more effective with predictive maintenance and failure management (Nickels, 2015). With AI, satellite communication systems are more sensitive to user needs and thus service quality increases.

As a result, AI applications play an important role in satellite communication systems to provide faster, more efficient and more reliable communication. These applications improve satellite communication in the next generations by providing advanced communication solutions for both personal and industrial use.

Autonomous Robotic Systems

In the aerospace field, AI applications in autonomous robotic systems are used to perform many critical tasks such as space exploration, data collection, maintenance/repair, vehicle motion control and ensuring life safety. These systems have the advantages of reducing costs, ensuring mission continuity and easily adapting to tiring space situations by reducing the need for manual space missions (Bautista-Montesano et al., 2019).

AI, which uses image processing techniques in space missions, is used for robots to identify and analyze the environment (Valasek, 2018). In this way, they can analyze the surface, detect environmental problems, craters or geological formations. For example, a rover that investigates the surface components of the planet can provide data to researchers by classifying mineral and rock types with machine vision. In this way, research is done more effectively.

In deep aerospace research, a large number of data sets must be analyzed (Badea, Zamfiroiu, & Boncea, 2018). By analyzing this data quickly, AI can find anomalies, provide information about the planet, or predict different variables such as weather. Especially in long-term missions, AI analyzes large amounts of data from outside. It optimizes data transmission by sending only the important findings to the ground control center.

Planning and management in aerospace missions are critical to the completion of tasks. AI is preferred in functions that allow the grading of tasks, determination of the movement route of the robots and optimization of energy consumption (Yang & Zhao, 2004). In this way, tasks are completed faster and safer. For example, algorithms have been developed that select the most useful paths to optimize the battery life used by rovers for their energy needs.

AI enables robots to make decisions on their own in aerospace missions where there is a delay due to communication (Shekhar, 2019). These systems analyze the conditions of the external environment and make decisions accordingly. They contribute to the successful implementation of tasks. Learning algorithms are applied to autonomous robots, allowing them to easily adapt to environmental changes throughout the tasks they perform. In this way, they determine the best route and movements to reach their goals.

AI analyzes data received from sensors to predict failures in robotic systems. In this way, maintenance requirements can be made before the robots fail and mission continuity is ensured. Data that can be determined in advance in robotic systems are energy consumption, motor temperatures or vibration (Monteiro, Carmona-Aparicio, Lei, & Despeisse, 2022). By analyzing this data, failures can be detected before they occur. Moreover, unexpected situations are minimized.

As a result, these systems and AI applications in the aerospace field increase both reliability and enable adaptation to demanding space conditions. In subsequent aerospace exploration and missions involving researchers, these technologies will provide safer, more efficient and lower-cost solutions.

Conclusion

AI applications in the aerospace field provide a major transformation in the industry by exploring space, performing satellite communication, carrying out maintenance and repair activities, and increasing the efficiency of autonomous functions. With AI-supported systems, data is analyzed quickly and accurately, preserving the operational continuity of aerospace vehicles. In this way, mission success is increased. As a simple example, thanks to autonomous robots and intelligent navigation systems, robotic vehicles conducting research on planetary surfaces can detect obstacles and determine their own routes. These systems, which do not require humans, can successfully perform their tasks. It is of critical importance, especially in important and sensitive space missions where communication-based delays are high. With similar applications of AI, it is ensured that even long-range missions progress without any problems and that all collected data is analyzed.

AI-based autonomous systems offer more sustainable solutions for the current and future industry by reducing the costs of aerospace missions. In areas such as remote satellite communication, AI is involved in many stages, starting from spectrum analysis to signal processing processes and frequency management, optimizing communication quality. In addition, AI algorithms used in maintenance/repair and failure management can continuously monitor the health status of aerospace vehicles and detect possible failures in advance. In this way, maintenance operations that need to be performed before failures occur can be planned. In this way, functional continuity is maintained and operational interruptions are minimized. Thanks to applications such as predictive maintenance, the working life of aerospace vehicles is extended and costly failures are prevented.

In the future, AI will become a key component of more complex and long-term missions in the aerospace industry. AI applications will play an important role in tasks such as autonomous exploration and discovery on distant planets or asteroids, defining and even mapping planetary surfaces, and developing suitable habitats for human life. AI-based analyses and algorithms will accelerate the pace of experimental/scientific discoveries by rapidly processing large amounts of data sets. It will make aerospace functions safer and more efficient by making intelligent optimizations for business management plans. Thanks to all these technological developments, the aerospace industry will not only improve life on Earth, but also provide new, safe and stable exploration opportunities for the future of human existence in space.

References

Abdulwahid, M. M., & Kurnaz, S. (2024). The utilization of different AI methods-based satellite communications: A survey. *AIP Conference Proceedings*, *3051*(1). AIP Publishing.

Altavilla, A., & Garbellini, L. (2002). Risk assessment in the aerospace industry. *Safety Science*, *40*(1), 271–298. https://doi.org/https://doi.org/10.1016/S0925-7535(01)00049-2

Badea, V. E., Zamfiroiu, A., & Boncea, R. (2018). Big data in the aerospace industry. *Informatica Economica*, *22*(1), 17–24.

Bao, C., Bao, Q., & Qiu, L. (2022). Design and Implementation of an SHM System Application Software. *ECITech 2022; The 2022 International Conference on Electrical, Control and Information Technology*, 1–7. VDE.

Basora, L., Olive, X., & Dubot, T. (2019). Recent advances in anomaly detection methods applied to aviation. *Aerospace*, *6*(11), 117.

Baur, J., & Silverman, E. (2007). Challenges and opportunities in multifunctional nanocomposite structures for aerospace applications. *MRS Bulletin*, *32*(4), 328–334.

Bautista-Montesano, R., López-Valdés, E., Jiménez-Ríos, D., & Gómez-Aladro, V. (2019). Autonomous Vehicles and Aerospace Engineering as a STEAM development platform. *2019 IEEE Integrated STEM Education Conference (ISEC)*, 384–387. IEEE.

Brevault, L., Balesdent, M., & Morio, J. (2020). *Aerospace system analysis and optimization in uncertainty*. Springer.

Chhaya, K., Khanzode, A., & Sarode, R. D. (2020). Advantages and Disadvantages of Artificial Intelligence and Machine Learning: A Literate Review. *International Journal of Library & Information Science (IJLIS)*, *9*(1), 3. Retrieved from http://iaeme.com

Cohen, K. (2023). Grand challenges in intelligent aerospace systems. *Frontiers in Aerospace Engineering*, *2*(September), 1–4. https://doi.org/10.3389/fpace.2023.1281522

Ezzat, D., Hassanien, A. E., Darwish, A., Yahia, M., Ahmed, A., & Abdelghafar, S. (2021). Multi-Objective Hybrid Artificial Intelligence Approach for Fault Diagnosis of Aerospace Systems. *IEEE Access*, *9*, 41717–41730. https://doi.org/10.1109/ACCESS.2021.3064976

Fourati, F., & Alouini, M.-S. (2021). Artificial intelligence for satellite communication: A review. *Intelligent and Converged Networks*, *2*(3), 213–243.

Harikumaran, J., Buticchi, G., Migliazza, G., Madonna, V., Giangrande, P., Costabeber, A., … Galea, M. (2020). Failure modes and reliability oriented system design for aerospace power electronic converters. *IEEE Open Journal of the Industrial Electronics Society*, *2*, 53–64.

Harris, C. J., Bailey, A., & Dodd, T. J. (1998). Multi-sensor data fusion in defence and aerospace. *The Aeronautical Journal*, *102*(1015), 229–244.

Herrera, L., & Chura, C. (2005). Spacecraft Health & Safety Monitoring Without Limits. In *Infotech@ Aerospace* (p. 7180).

Insaurralde, C. C. (2018). Intelligent autonomy for aerospace engineering systems:

Technologies from different application domains to the aviation sector. *AIAA/IEEE Digital Avionics Systems Conference - Proceedings*, *2018-Septe*, 1–10. https://doi.org/10.1109/DASC.2018.8569358

Keane, A. J., & Scanlan, J. P. (2007). Design search and optimization in aerospace engineering. *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences*, *365*(1859), 2501–2529. https://doi.org/10.1098/rsta.2007.2019

Kim, H. A., Kennedy, D., & Gürdal, Z. (2008). Special issue on optimization of aerospace structures. *Structural and Multidisciplinary Optimization*, *36*(1), 1–2. https://doi.org/10.1007/s00158-008-0256-1

Kodheli, O., Lagunas, E., Maturo, N., Sharma, S. K., Shankar, B., Montoya, J. F. M., … Kisseleff, S. (2020). Satellite communications in the new space era: A survey and future challenges. *IEEE Communications Surveys & Tutorials*, *23*(1), 70–109.

Krishnakumar, K. (2003). Intelligent systems for aerospace engineering - An overview. *National Aeronatics and Space Administration Ames Reseach Centre - Report*, (May), 1–15.

Léonard, P. L. Y., Hallstedt, S. I., Nylander, J. W., & Isaksson, O. (2024). An introductory study of the sustainability transition for the aerospace manufacturing industry. In *Sustainable Production through Advanced Manufacturing, Intelligent Automation and Work Integrated Learning* (pp. 577–588). IOS Press.

Mahashabde, A., Wolfe, P., Ashok, A., Dorbian, C., He, Q., Fan, A., … Waitz, I. A. (2011). Assessing the environmental impacts of aircraft noise and emissions. *Progress in Aerospace Sciences*, *47*(1), 15–52. https://doi.org/https://doi.org/10.1016/j.paerosci.2010.04.003

Merhav, S. (2012). *Aerospace sensor systems and applications*. Springer Science & Business Media.

Mieloszyk, J. (2017). Practical problems of numerical optimization in aerospace sciences. *Aircraft Engineering and Aerospace Technology*, *89*(4), 570–578. https://doi.org/10.1108/AEAT-11-2016-0201

Monteiro, H., Carmona-Aparicio, G., Lei, I., & Despeisse, M. (2022). Energy and material efficiency strategies enabled by metal additive manufacturing–A review for the aeronautic and aerospace sectors. *Energy Reports*, *8*, 298–305.

Nickels, L. (2015). AM and aerospace: an ideal combination. *Metal Powder Report*, *70*(6), 300–303.

Oche, P. A., Ewa, G. A., & Ibekwe, N. (2024). Applications and Challenges of Artificial Intelligence in Space Missions. *IEEE Access*, *12*(October 2021), 44481–44509. https://doi.org/10.1109/ACCESS.2021.3132500

Oster, C. V, Strong, J. S., & Zorn, C. K. (2013). Analyzing aviation safety: Problems, challenges, opportunities. *Research in Transportation Economics*, *43*(1), 148–164. https://doi.org/https://doi.org/10.1016/j.retrec.2012.12.001

Ranasinghe, K., Sabatini, R., Gardi, A., Bijjahalli, S., Kapoor, R., Fahey, T., & Thangavel, K. (2022a). Advances in Integrated System Health Management for mission-essential and safety-critical aerospace applications. *Progress in Aerospace Sciences*, *128*(September 2021), 100758. https://doi.org/10.1016/j.paerosci.2021.100758

Ranasinghe, K., Sabatini, R., Gardi, A., Bijjahalli, S., Kapoor, R., Fahey, T., & Thangavel, K. (2022b). Advances in Integrated System Health Management for mission-essential and safety-critical aerospace applications. *Progress in Aerospace Sciences*, *128*, 100758.

Sathyan, A. (n.d.). *Intelligent Machine Learning Approaches for Aerospace Applications*.

Shekhar, S. S. (2019). Artificial Intelligence in Automation. *Artificial Intelligence*, *3085*(06), 14–17.

Shelukhin, N., Kharaberiush, I., Shelukhin, O., Tsymbalistyi, V., & Selevko, V. (2022). *Organizational and Legal Support of Economic Security of Aerospace Industry of Ukraine BT - Integrated Computer Technologies in Mechanical Engineering - 2021* (M. Nechyporuk, V. Pavlikov, & D. Kritskiy, eds.). Cham: Springer International Publishing.

Sze, N. D., Isaacs, R. G., Ko, M., & McElroy, M. B. (1981). *Atmospheric studies related to aerospace activities and remote sensing technology*. NASA.

Tipaldi, M., & Bruenjes, B. (2014). Spacecraft health monitoring and management systems. *2014 IEEE Metrology for Aerospace (MetroAeroSpace)*, 68–72. IEEE.

Trélat, E. (2012). Optimal control and applications to aerospace: some results and challenges. *Journal of Optimization Theory and Applications*, *154*, 713–758.

Tschan, C., Kivelevitch, E., & Melcher, K. (2013). Advancing intelligent systems in the aerospace domain. *AIAA Infotech at Aerospace (I at A) Conference*. https://doi.org/10.2514/6.2013-5058

Valasek, J. (2018). *Advances in computational intelligence and autonomy for aerospace systems*. American Institute of Aeronautics and Astronautics, Inc.

Van der Velden, C., Bil, C., Yu, X., & Smith, A. (2007). An intelligent system for automatic layout routing in aerospace design. *Innovations in Systems and Software Engineering*, *3*(2), 117–128. https://doi.org/10.1007/s11334-007-0021-4

Wang, W. (2007). Applications of MIMO technique for aerospace remote sensing. *2007 IEEE Aerospace Conference*, 1–10. IEEE.

Wu, X. (2024). *Neural network big data fusion in remote sensing image processing technology*. *33*(1). https://doi.org/doi:10.1515/jisys-2023-0147

Yang, H. I., & Zhao, Y. J. (2004). Trajectory planning for autonomous aerospace vehicles amid known obstacles and conflicts. *Journal of Guidance, Control, and Dynamics*, *27*(6), 997–1008.

Zhang, X., Chen, Y., & Hu, J. (2018). Recent advances in the development of aerospace materials. *Progress in Aerospace Sciences*, *97*, 22–34.

About The Author

Rabia TOPRAK, PhD, is an Assistant Professor of Electrical and Electronics Engineering at Karamanoglu Mehmetbey University in Karaman, Türkiye. She holds a PhD in Electrical and Electronics Engineering from Konya Technical University and a MSc in Electrical and Electronics Engineering from Selcuk University. Her main areas of interest are antenna structures, microstrip antennas, biomedical applications of antenna structures, artificial intelligence and electromagnetic fields/waves.
E-mail: rabiatop@kmu.edu.tr , **ORCID:** 0000-0002-3306-1163

Similarity Index
The similarity index obtained from the plagiarism software for this book chapter is 1%.

Metaheuristic Algorithms in Artificial Neural Network Training

Münevver BAHAR
Necmettin Erbakan University

Halil İbrahim AYAZ
Necmettin Erbakan University

Murat KARAKOYUN
Necmettin Erbakan University

To Cite This Chapter

Bahar, M., Ayaz, H. İ., & Karakoyun, M. (2024). Metaheuristic Algorithms in Artificial Neural Network Training. In S. Koçer & Ö. Dündar (Eds.), *Intelligent Systems and Optimization in Engineering* (pp. 33-52). ISRES Publishing.

Introduction

Various methods are used today for problems in different sectors. Artificial Neural Networks (ANN) technology is one of the newest among these methods and has been successfully applied in many areas. ANN aims to imitate and develop the working principles of the human brain and to perform the basic functions that the human brain performs biologically with a suitable software (Çörekcioğlu et al., 2021). ANN architecture consists of the connections of simple processing units called nodes or neurons. Between the input and output layers, there could be one or several hidden layers, and the connection between each node has a certain weight (Kaytan et al., 2020).

In ANN, operations such as classification, modeling, optimization or prediction are performed by updating the connection settings in the network; this process is called learning. This process, which is carried out in order to increase the performance of the network, is carried out in the form of iterations in computer programs (Çalışkan & Deniz, 2015). In the learning process, the most widely used algorithm for multilayer feedforward networks is the "Back Propagation" (BP) training algorithm, known as backpropagation (Ticknor, 2013). The backpropagation algorithm focuses on minimizing the difference between the target output values and the output values produced by the network as the iterations progress in the training process, and it relies on adjusting the network's connection weights to reduce this difference (Özkan, 2012).

In the ANN model, the weights are usually updated during the backpropagation process with derivative-based techniques (Jiadong et al., 2024). However, derivative-based techniques may not be able to successfully update the weight values in challenging problems, which may cause the weights to get stuck at local minimum points (Emambocus et al., 2023; Karakoyun, 2024; F. N. Özdemir & Özkış, 2024). To overcome this problem, researchers have used metaheuristic algorithms in the backpropagation phase of the network in many studies. Metaheuristic algorithms are strategies designed to solve various problems by mathematically simulating the behaviors of natural entities such as humans, animals, and plants. These algorithms aim to reach the best solution in the solution space faster by using effective search techniques in a high-level working environment (Çelik, 2013).

Given that training artificial neural networks is both a critical and complex task, it has been the focus of extensive research. In recent years, particularly, various metaheuristic algorithms have been applied to address this issue (Karakoyun, 2024). It is virtually impossible to thoroughly review all related studies in the literature due to time constraints. Therefore, this study's literature review section highlights a selection of recent and significant works in the field.

Özdemir (F. N. Özdemir & Özkış, 2024) developed a hybrid model with the Snow Ablation Optimizer (SAO) algorithm to update the weights of the artificial neural network. The developed hybrid model was compared with hybrid models created with gray wolf, reptile search, cuckoo and sine cosine algorithms on five different data sets and achieved the best result with the SAO model in terms of average success order with a value of 1.2 in all metrics. Aksu et al. (Aksu et al., 2022) used two different estimation methods based on multilayer neural network to provide reliable estimation of solar radiation. The network coefficients and bias values of the neural network were trained using Imperialist Competitive Algorithm (ICA) and Particle Swarm Optimization algorithm (PSO). Ateş (Ateş, 2022) created a hybrid approach that combines a multilayer ann model with PSO and the Cultural Algorithm (CA) to achieve minimal error in short-term PV panel output power predictions. Özmen et al. (Özmen et al., 2023) worked on early detection of diabetes by reducing the number of features with metaheuristic methods. They performed feature selection using Salp Swarm Algorithm (SSA), Artificial Bee Colony Algorithm, Whale Optimization Algorithm (WOA) and Ant Colony Algorithm (ACO) with examples from UCI (UCI Machine Learning Repository) data repository. For the evaluation of selected features, K-Nearest Neighbor (KNN), Naive Bayes (NB), Support Vector Machine (SVM) and ANN methods were used to calculate accuracy, sensitivity and specificity parameters. Ayaz (Ayaz & Kamisli Ozturk, 2021) addressed optimization challenges in train seat planning by applying heuristic approaches and parallel machine scheduling to minimize waiting times and maximize resource utilization. Köprü (Köprü, 2020) used artificial neural network to estimate the amount of liquid crude iron produced with monthly raw material information of blast furnace enterprise. Zaimoğlu (Zaimoğlu, 2023) developed a new approach called Binary Chaotic Horse Herd Optimization Algorithm (BCHOAFS) by augmenting the proposed binary version of HOA with five different well-known chaotic maps in order to increase the success and stability of the algorithm. Jama (Jama, 2021) presented a modified version of bio-inspired Ant Lion Optimization Algorithm (ALO) to solve the region growing segmentation problem. Farahani et al (Shahvaroughi Farahani & Razavi Hajiagha, 2021) sought to forecast stock price indices using an ANN and trained it with recent metaheuristic algorithms like Social Spider Optimization (SSO) and the Bat Algorithm (BA). They employed the Genetic Algorithm (GA), a heuristic method, for feature selection and identifying the most relevant indicators. Mu'azu (Abdullahi Mu'azu, 2023) aimed to optimize the hybrid configuration of ANN with Cuttlefish Optimization Algorithm (CFOA), Electrostatic Discharge Algorithm (ESDA) and Henry Gas Solubility Optimization Algorithm (HGSOA) and Sine Cosine Algorithm (SCA) algorithms for soil BC analysis.

Other examples of studies in this area include the enhanced SSA for training multilayer sensors (MLS) (Atlı, 2022), the application of the PSO algorithm in photovoltaic (PV) energy systems (A. Özdemir & Pamuk, 2021), the use of the Vibration Particle System Algorithm for fine-tuning weight matrices (Özkaya et al., 2021), and the development of a hybrid INFO-Simulated Annealing Algorithm to optimize the carrier arm of drones in unmanned aerial vehicles (Yildiz, 2023), among others.

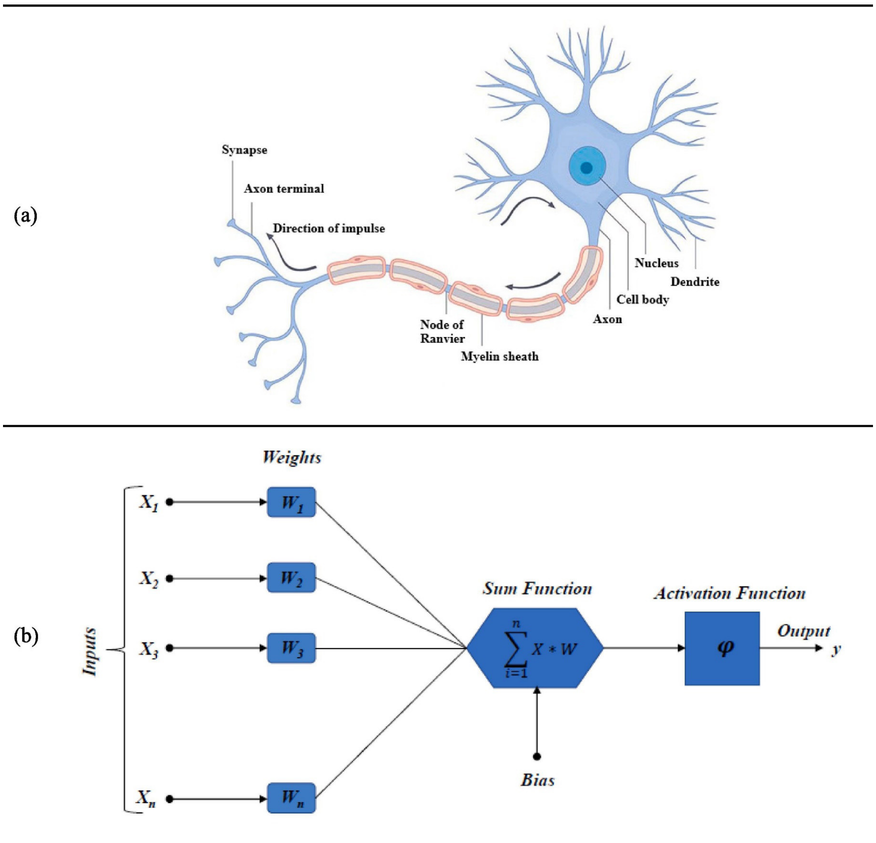
In this study, the performances of the metaheuristic algorithms Grasshopper Optimization Algorithm (GOA) (Saremi et al., 2017), Artificial Hummingbird Algorithm (AHA) (Zhao et al., 2022), Arithmetic Optimization Algorithm (AOA) (Abualigah et al., 2021), Crayfish Optimization Algorithm (COA) (Jia et al., 2023), Artificial Bee Colony (ABC) (Karaboga, 2005) and Tree-seed algorithm (TSA) (Sahman et al., 2019) were evaluated on 21 different datasets. The performance evaluation was performed using various metrics (precision, specificity, F1-score and sensitivity).

Artificial Neural Network

ANNs are computer software that perform basic functions such as learning, remembering, generalizing and producing new information from the obtained data by imitating the learning processes of the human brain. ANNs are used for various purposes such as pattern recognition, classification, modeling, optimization and prediction (Rençber, 2018).

The development of artificial neural networks began with research on the working principles of the human brain, and an important step was taken in 1943 when McCulloch and Pitts developed the first artificial neural network model. In the 1950s, studies in the field of artificial neural networks gained momentum with Hebb's learning theory and Rosentblatt's "Perceptron" model, but these studies entered a period of stagnation in the 1960s due to artificial intelligence research. From the 1980s onwards, artificial neural networks began to attract attention again, and their popularity increased with Hopfield's creation of the mathematical foundations of networks and Rummelhart's parallel programming studies. During this process, developments in computer hardware also contributed to the integration of artificial neural networks into practical applications (Keskenler & Keskenler, 2017). Figure 1.a shows a biological neuron, and Figure 1.b shows an artificial neuron model.

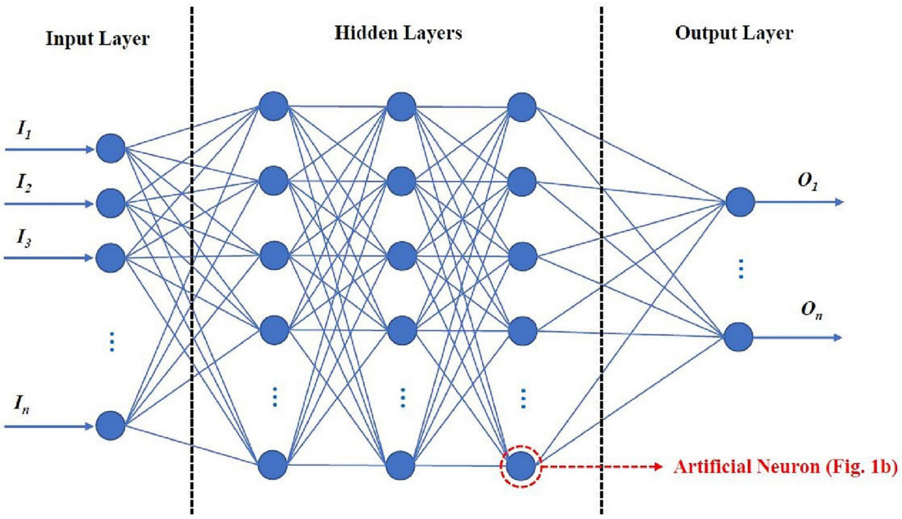
Figure 1 a-b)
Biological Neuron And Artificial Neuron Model (Karakoyun, 2024)



ANN models, which are based on the principle of learning based on experience, aim to produce a single output from many inputs. The basic component of this technique is the processing elements known as neurons (Çınaroğlu & Avcı, 2020). Artificial nerve cells, or neurons, in the network have the ability to make predictions about similar examples that they have not encountered before by comprehending an event based on data, with or without supervision. Neurons are organized in logical groups called layers (Çalışkan & Deniz, 2015).

The structure of ANN is divided into two main groups as single layer perceptron (SLP) and multilayer perceptron (MLP). The first studies on ANN focused on the SLP architecture. However, when it was understood that SLPs could only solve linear problems, the MLP architecture that can also learn nonlinear problems was developed. MLPs have another layer called the intermediate (hidden) layer in addition to the input and output layers (F. N. Özdemir & Özkış, 2024). Hidden layers are responsible for transmitting signals from the input layer to the output layer and do not have direct connections with the external environment. The general structure of the MLP is shown in Figure 2.

Figure 2
General Structure Of The Mlps (Karakoyun, 2024)



Each unit in the network calculates the weighted sum of the input data coming to it. This sum is obtained by multiplying the input data by the connection weights. The data progresses with this process in each layer of the network. First, the weights are randomly assigned. In the first hidden layer, the multiplication results are collected and these results are transferred to the next hidden layer or output layer by being subjected to the activation function. The activation function undertakes the function of processing the input value and converting it into an output value. The most commonly used activation function in MNEs (Multilayer Artificial Neural Networks) is the sigmoid function, which produces outputs ranging from 0 to 1. Thanks to the activation function, the network is transformed into a non-linear structure as a result of the operations performed in the hidden layers, which provides an advantage in solving complex problems (Akel & Karacameydan, 2012).

In the MLP architecture, the learning process is divided into two stages: forward computation and backward computation. Forward computation creates the output of the network, while backward computation deals with updating the weights. This stage of the learning process is performed by the backpropagation algorithm (Ticknor, 2013). The backpropagation algorithm focuses on minimizing the difference between the targeted output and the output produced by the network as iterations progress in the training process and updates the network connection weights to minimize this difference.

Although various methods are used to evaluate error, the most commonly preferred method is the mean square error (MSE). MSE is a technique often used to measure the difference (error rate) between expected and predicted values in machine learning systems (Gölcük et al., 2023). Equation 1. Provides the mathematical notation of MSE.

$$MSE = \frac{1}{n} \sum_{i=1}^{n} \left(p_i - a_i\right)^2 \tag{1}$$

In this equation, p_i is the predicted output value, a_i is the actual output value, and n is the number of samples used in the training process. In summary, the goal of MLPs is to find the biases and weights that minimize the MSE value. In other words, a lower MSE value indicates a more effective training process, while a higher MSE value indicates a more inefficient training process (Karakoyun, 2024).

ANN generalize over the examples presented to the system during the training process. When a change occurs in any dimension of the problem, the network may need to be retrained. The ability of the network to quickly adapt to new situations increases the probability of reaching the desired output faster (Çınaroğlu & Avcı, 2020). The performance of the trained network is evaluated on the test dataset that was not used during training, a process called 'testing the network'. The success in the testing phase is measured by metrics such as precision, specificity, F1-score and sensitivity. These metrics are discussed in detail in Section 5.

Metaheuristic Algorithms

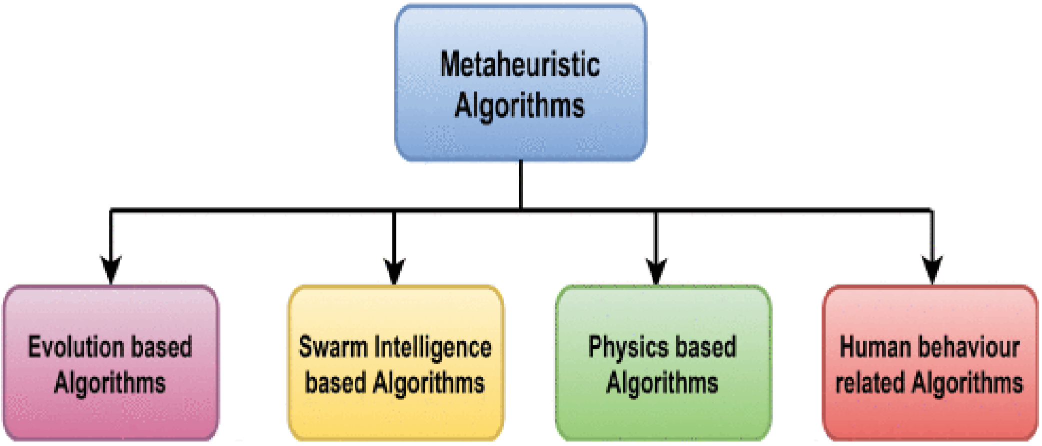
The term metaheuristic refers to high-level heuristic strategies designed to solve a broad range of optimization problems. In recent years, many metaheuristic algorithms have been successfully applied to tackle complex and difficult problems. The attractiveness of these algorithms lies in their ability to find the best or optimal solutions for even very large problem instances within a relatively short time frame (Dokeroglu et al., 2019).

Metaheuristic methods serve three primary purposes: rapidly solving problems, handling large-scale problems, and creating more robust algorithms. These approaches are not only straightforward to design but also flexible and easy to implement. Typically, metaheuristic algorithms utilize a mix of rules and randomization to replicate natural phenomena (Rere et al., 2016).

Metaheuristic algorithms exhibit stochastic behavior, initiating their optimization process by producing random solutions. Unlike gradient-based search methods, they do not require the calculation of search space derivatives. These algorithms are valued for their flexibility and simplicity, owing to their straightforward concepts and ease of implementation. They can be easily adjusted to suit specific problems. A key characteristic of metaheuristic algorithms is their exceptional ability to avoid premature convergence (Agrawal, 2021). Because of their stochastic nature, these techniques function like a hidden mechanism, efficiently avoiding local optima and thoroughly exploring the search space.

Metaheuristic algorithms can be divided into four main categories based on their behavior: evolution-based, swarm intelligence-based, physics-based, and human behavior-based algorithms (Mohamed et al., 2020). These categories are shown in Figure 3.

Figure 3
Classification Of Metaheuristic Algorithms (Agrawal, 2021)



Evolution-based algorithms are inspired by the natural evolution process and start with a randomly generated population of solutions. These algorithms combine the best solutions to create new individuals; in this process, methods such as mutation, crossover and selection of the most suitable solution are used (Mohamed et al., 2020). The most well-known example of this category is the GA based on Darwin's theory of evolution (Holland, 1992).

Swarm intelligence-based algorithms are inspired by the social behavior of living things such as insects, animals, fish and birds (Agrawal, 2021). One of the most popular techniques in this field is PSO, developed by Kennedy and Eberhart (Kennedy & Eberhart, 1995), which is based on the behavior of a flock of birds flying through the search space and finding their best positions.

Physics-based algorithms are inspired by the physical laws that exist in the universe. Algorithms in this category, such as Simulated Annealing (Kirkpatrick et al., 1983) and Harmony Search (Geem et al., 2001), perform optimization by imitating physical processes. Finally, algorithms inspired by human behavior are inspired by the performance and methods that people exhibit when performing different activities. Popular methods include Teaching-Learning Based Optimization (TLBO) (Rao et al., 2012) and League Championship Algorithm (Kashan, 2009).

In this research article, the new generation metaheuristic algorithms such as Grasshopper Optimization Algorithm (Saremi et al., 2017), Artificial Hummingbird Algorithm (Zhao et al., 2022), Arithmetic Optimization Algorithm (Abualigah et al., 2021), Crayfish Optimization Algorithm (Jia et al., 2023), Artificial Bee Colony (Karaboga, 2005) and Tree-Seed Algorithm (Sahman et al., 2019) were used.

Grasshopper Optimization Algorithm

GOA is a new Swarm Intelligence method inspired by the swarm behavior of locusts in nature. This algorithm was proposed by Saremi et al. in 2017 (Saremi et al., 2017). Literature shows that this algorithm is used to solve various optimization problems such as feature selection, scheduling, load frequency control, economic dispatch, engineering, etc. (Meraihi et al., 2021).

The algorithm simulates the repulsion and attraction forces between locusts. Repulsion forces enable the locusts to explore the search space, while attraction forces guide them towards promising areas. GOA includes a factor that progressively decreases the locusts' comfort zone, ensuring a balance between the exploration phase (global search) and exploitation phase (local search) during the optimization. This mechanism assists GOA in accurately approximating the global optimum, reducing the risk of getting trapped in a local optimum. As the best solution found by the swarm so far becomes the target for the swarm to follow, the locusts significantly increase their chances of locating the global optimum by enhancing the target over the optimization process.

Artificial Hummingbird Algorithm

AHA is a meta-heuristic optimization method modeled after the feeding strategies of hummingbirds in the wild. Hummingbirds possess three distinct flight abilities: axial, diagonal, and omnidirectional. Additionally, their memory capacity to choose the optimal food source plays a crucial role in the AHA algorithm. AHA replicates three types of foraging behaviors during the optimization process: directed foraging, territorial foraging, and migratory foraging (Aslanov et al., 2023; Bakır, 2024; Zhao et al., 2022).

In the AHA algorithm, foraging mimics the flight abilities of hummingbirds when searching for food, and this step represents the process of exploring the solution space. Hummingbirds' ability to recall the best food sources through their memories and return to these sources helps the algorithm discover potential solutions. Territorial foraging reflects hummingbirds' behavior in protecting food sources, and represents the stage in which the algorithm focuses on promising solutions and improves these areas. Migratory foraging is the diversification stage, where the algorithm explores new solution spaces, such as when birds migrate to new areas when food sources are scarce, and aims to find better global solutions by avoiding local optima (Khodadadi et al., 2023; Zhao et al., 2022).

Arithmetic Optimization Algorithm

AOA is a metaheuristic optimization algorithm that uses arithmetic operations to solve global optimization problems. Suitable for both discrete and continuous problems, AOA aims to find the best solution in the search space and searches for solutions using arithmetic operations in this process (Abualigah et al., 2021).

The algorithm works in two main stages: In the first stage, a wide search space is scanned to discover potential solutions, and in the second stage, these solutions are optimized in a narrower area. Initially, a random population of solutions is created and new solutions are obtained by applying arithmetic operators to these solutions. In this process, the quality of each candidate solution is evaluated and the best solutions are constantly updated (Abualigah et al., 2021; Gölcük et al., 2023).

The simplicity of AOA both facilitates its implementation and enables it to be used in a wide range of optimization problems. It can be used effectively in many areas such as engineering design, artificial intelligence and energy system optimization. AOA was developed inspired by arithmetic operations in nature and has achieved successful results in different optimization problems (Dhal et al., 2023).

Crayfish Optimization Algorithm

COA is a metaheuristic optimization algorithm developed by modeling the behavior of crayfish in aquatic ecosystems. This algorithm mimics biological and environmental interactions in nature to solve various optimization problems. COA has a population-based structure; a population consisting of a certain number of crayfish individuals represents a solution in the search space (Jia et al., 2023).

The movements of crayfish under water occur for various reasons such as searching for food, avoiding dangers, and mating, and these movements are used to find new solutions in the solution space during the optimization process. The behavior of crayfish scanning their surroundings and finding the best food while searching for food is used as local and global search strategies to find the best solution in COA. While the instinct to avoid dangers aims to avoid bad solutions and find better solutions, social interactions accelerate the sharing of information within the community and the optimization process (Jia et al., 2023, 2024).

Its advantages include the ability to focus on the best solutions while maintaining the diversity among solutions and adapting to various optimization problems with its nature-inspired structure. However, the effectiveness of the algorithm depends on the parameters used, and these parameters need to be adjusted correctly. Also, COA, like other metaheuristic algorithms, has the risk of getting stuck in local optima. COA is an important tool for researchers who are interested in optimization algorithms, especially those inspired by nature and modeling biological processes (Jia et al., 2023, 2024).

Artificial Bee Colony

ABC Algorithm was developed by Karaboğa in 2005 (Karaboga, 2005) and was inspired by the natural food-gathering behavior of honeybees. This algorithm solves optimization problems by simulating the processes of bee colonies in nature to find efficient food sources. ABC is used in various engineering and scientific problems as a metaheuristic optimization algorithm (Karaboga & Basturk, 2007).

The ABC algorithm mimics the behavior of three different types of bees: worker bees, observer bees, and scout bees. Worker bees evaluate existing food sources (solutions) and try to improve these sources. Observer bees focus on the most efficient sources based on the information received from worker bees. If worker bees cannot develop a solution, they turn into scout bees and wander randomly to search for new food sources. This process continues until the specified number of iterations and solutions are developed at each stage (Karaboga, 2005; Karaboga & Basturk, 2007).

The basic operation of the ABC algorithm is similar to the food source-finding behavior of bees in nature. Bees move towards more efficient sources and away from inefficient ones. In this way, the algorithm tries to find the best solutions for optimization problems. Especially in solving nonlinear and complex problems, the ABC algorithm attracts attention with its flexibility and efficiency (Karaboga, 2005; Karaboga & Akay, 2009).

Tree-Seed Algorithm

TSA is an optimization algorithm inspired by the seed propagation and sprouting processes of trees in nature. This algorithm symbolizes the formation of new individuals by spreading the seeds of trees in various ways. TSA is applied to solve complex optimization problems by imitating these natural processes (Sahman et al., 2019).

At the heart of TSA is seed dispersal, where each seed is generated based on either the optimal or a randomly chosen tree position within the population. After creating three potential positions, they are evaluated against the objective function of the problem. For every tree, two different strategies exist for generating seeds, and this selection is guided by the algorithm's primary control parameter, known as Search Bias (ST). Once the seeds are assessed using the objective function, those with superior fitness compared to the current tree positions are selected to form the next generation. This process of seed production and growth is repeated until the algorithm reaches the maximum number of fitness evaluations (Kiran, 2015; Sahman et al., 2019).

Parameter Settings

Some parameters used in the implementation of the algorithms are common. These common parameters and their values are as follows: number of runs 25, population size 50, search space limits [-10, 10] and maximum fitness evaluation (maxFEs) 20,000. In addition, some algorithms have some particular parameters. These params and their associated values utilized in this study are presented in Table 1.

Table 1
Particular Parameters Of The Comparison Algorithms

Algorithms	Params
GOA	cMax = 1, cMin = 0.00004
AHA	No particular parameter
AOA	C1 = 2, C2 = 6, C3 = 1, C4 = 2, u = 0.9, l = 0.1
COA	No particular parameter
ABC	limit = 100
TSA	ST = 0.1, least_seed = 0.1, most_seed = 0.25

The Implementation of the Metaheuristic Algorithms in Training of ANN

The primary goal of training an ANN is to optimize the network's biases and weights. For this reason, the optimization algorithms used in the training process focus on solution sets consisting of biases and weights (Karakoyun, 2024). In this article, the architecture of MLPs was generated dynamically by considering the features and classes in the dataset used. In determining the MLP structure, a set of rules stated below were followed.

- The count of hidden neurons = (2 * number of attribute) + 1
- The count of biases = the count of hidden neurons + the count of classes
- The count of weights = (the count of attributes * the count of hidden neurons) + (the count of classes * the count of hidden neurons)
- Dimension of the problem (solution) = the count of biases + the count of weights

In a MLP, the count of hidden neurons refers to the total count of neurons present in the hidden layer. The bias count represents the total biases associated with neurons in both the hidden and output layers, as each neuron in the MLP requires a bias term. The count of weights represents the total connections between the neurons in the hidden layer and the input/output layers. Lastly, the size of the problem is determined by summing the weight and bias numbers.

A key challenge in training ANNs is designing the MLP architecture and defining the solution vector representation. After this issue is solved, metaheuristic algorithms were applied to ANN training by combining them with other steps. In this process, first, the data to be used for training and testing the model is read from a file. If the data is not separated as training and test, it is separated as training and test data in this step. Then, the MLP model is created according to the characteristics of the data and a solution vector suitable for the MLP structure is designed. Weights and biases are optimized using optimization algorithms. It is checked whether the data is separated as training and test; if not, the results obtained using the training data are returned as output, if separated, the model trained with the training data is applied on the test data and the results are produced. Finally, both training and test results are evaluated with various metrics and presented.

Experimental Results

In this section, the datasets and comparison metrics used in the study are introduced. Subsequently, comprehensive comparative findings using various metrics and approaches are provided.

Data Sets

In this study, a large dataset collected from various sources was used (Karakoyun, 2024; Qaddoura et al., 2020). A total of 21 different datasets were examined. The number of features in these datasets varied between 2 and 34, while the number of classes varied between 2 and 8. While the appendicitis dataset had the fewest samples with 106 samples, the Aniso, Blobs and Varied datasets had the largest samples with 1500 samples. The other datasets were between these two limits (between 106 and 1500). In order for the datasets used in the study to be a reference for future research, care was taken to ensure that they had diversity in terms of the sample size, the count of classes and the count of features.

Based on the working principle of ANN, determining the weights requires a training phase. Additionally, a testing phase is essential to evaluate the effectiveness of the training. Therefore, datasets need to be separated into train and test sets.

In this research, datasets containing over 150 instances were allocated as 75% for training and 25% for testing. On the other side, for datasets with 150 instances or fewer, all available data was fully used in the training phase to guarantee a qualified training process. Table 2 provides comprehensive details about the datasets used.

Table 2
Comprehensive Details Regarding The Datasets Utilized

#	Dataset	Feature	Label	Instances	Training Data	Test Data
1	Aniso	2	3	1500	1125	375
2	Blobs	2	3	1500	1125	375
3	Varied	2	3	1500	1125	375
4	Aggregation	2	7	788	594	194
5	Balance	4	3	625	469	156
6	Smiley	2	4	500	375	125
7	Mouse	2	3	490	368	122
8	Ionosphere	34	2	351	264	87
9	Liver	6	3	345	259	86
10	Ecoli	7	8	336	252	84
11	Vertebral3	6	3	310	233	77
12	Pathbased	2	3	300	225	75
13	Heart	13	2	270	203	67
14	Glass	9	6	214	162	52
15	Seeds	7	3	210	158	52
16	Wine	13	3	178	133	45
17	Iris	4	3	150	150	N/A
18	Iris2D	2	3	150	150	N/A
19	Vary-density	2	3	150	150	N/A
20	Diagnosis_II	6	3	120	120	N/A
21	Appendicitis	7	2	106	106	N/A

Comparison Metrics

To evaluate and compare the performance of the algorithms used for ANN training, four distinct metrics were employed: precision, specificity, F1-score and sensitivity.

These metrics are metrics obtained from the confusion matrix. The confusion matrix is a performance measurement method frequently used in the literature to measure the accuracy of the model and the classification success of data sets. Figure 4 shows the confusion matrix.

Figure 4
Confusion Matrix

		Predicted condition	
	Population (P+N)	**Predicted Positive (PP)**	**Predicted Negative (PN)**
Actual Condition	**Positive (P)**	True positive (TP)	False negative (FN)
	Negative (N)	False positive (FP)	True negative (TN)

Some basic concepts are essential for a clearer understanding of the metric calculation process. Positive (P) represents true positive situations in the dataset, while negative (N) represents true negative situations. True positive (TP) indicates that a condition is correctly detected, while true negative (TN) indicates that lack of a condition is correctly detected. False positive (FP) indicates that a condition is present when it is not present, and false negative (FN) indicates that a condition is present with an incorrect result that it is not present.

From a statistical perspective, specificity and sensitivity quantify the accuracy in detecting the existence or non-existence of a condition. When conditions with the presence of a condition are classified as positive and those without as negative, sensitivity measures how effectively a test identifies true positives, while specificity measures its ability to identify true negatives. Precision, on the other hand, is calculated by dividing the number of true positive cases by the total number of cases classified as positive, whether correctly or incorrectly. This metric reflects how precisely a class is identified. The F1 score combines sensitivity and precision using their harmonic mean, providing a balanced view of the results from the confusion matrix (Karakoyun, 2024; Tharwat, 2021). The equations for calculating these metrics depend on the confusion matrix and the aforementioned definitions are provided below:

$$Sensitivity = \frac{TP}{TP + FN} \tag{2}$$

$$Specificity = \frac{TN}{TN + FP} \tag{3}$$

$$Precision = \frac{TP}{TP + FP} \tag{4}$$

$$F1\ Score = \frac{2TP}{2TP + FP + FN} \tag{5}$$

Results

To facilitate the comparison of the experimental outcomes, we computed the precision, specificity, F1-score and sensitivity for each algorithms depend on the mean worths achieved after conducting 25 runs. Every metric is presented with its mean and standard deviation derived from the algorithms' results over these 25 runs. Additionally, the success ranking was established using the mean worth of the 25 runs and included in the result tables. The performance metrics for precision, specificity, F1-score and sensitivity for the algorithms are displayed in Table 3, Table 4, Table 5, and Table 6, respectively. In the tables, A represents the mean, S represents the standard deviation, and R represents the success ranking.

Table 3

Experimental Results Of The Algorithms For Sensitivity Metric

	GOA			AHA			AOA			COA			ABC			TSA		
Datasets	**A**	**S**	**R**	**A**	**S**	**R**	**A**	**S**	**R**	**A**	**S**	**R**	**A**	**S**	**R**	**A**	**S**	**R**
Aniso	0.86	0.16	3	0.87	0.11	2	0.89	0.14	1	0.86	0.14	4	0.78	0.15	6	0.8	0.13	5
Blobs	0.91	0.15	5	0.95	0.11	2	0.96	0.09	1	0.9	0.17	6	0.92	0.12	4	0.92	0.11	3
Varied	0.87	0.13	1	0.86	0.08	2	0.82	0.11	4	0.84	0.1	3	0.76	0.11	6	0.79	0.09	5
Aggregation	0.23	0.07	2	0.23	0.06	1	0.23	0.07	3	0.21	0.06	4	0.18	0.06	6	0.19	0.06	5
Balance	0.59	0.07	1	0.5	0.08	5	0.57	0.09	2	0.51	0.08	4	0.47	0.1	6	0.55	0.06	3
Smiley	0.26	0.04	1	0.25	0	3	0.25	0	3	0.25	0	3	0.25	0.01	2	0.25	0	4
Mouse	0.73	0.17	1	0.55	0.16	5	0.65	0.16	2	0.48	0.18	6	0.61	0.13	3	0.59	0.12	4
Ionosphere	0.5	0	1	0.5	0	1	0.5	0	1	0.5	0	1	0.5	0	1	0.5	0	1
Liver	0.35	0.04	2	0.35	0.03	4	0.34	0.03	5	0.35	0.03	3	0.34	0.03	6	0.35	0.03	1
Ecoli	0.21	0.06	1	0.16	0.04	3	0.19	0.05	2	0.15	0.05	4	0.12	0.06	6	0.14	0.05	5
Vertebral3	0.55	0.13	3	0.57	0.1	2	0.61	0.05	1	0.5	0.14	5	0.49	0.11	6	0.55	0.12	4
Pathbased	0.47	0.12	1	0.43	0.17	4	0.46	0.13	2	0.37	0.18	5	0.45	0.13	3	0.36	0.17	6
Heart	0.5	0	1	0.5	0	1	0.5	0	1	0.5	0	1	0.5	0	1	0.5	0	1
Glass	0.19	0.06	1	0.18	0.04	3	0.18	0.04	4	0.19	0.04	2	0.17	0.05	6	0.17	0.03	5
Seeds	0.6	0.13	1	0.54	0.08	3	0.59	0.08	2	0.47	0.12	6	0.48	0.13	5	0.51	0.09	4
Wine	0.47	0.15	5	0.56	0.15	1	0.51	0.14	4	0.54	0.14	3	0.38	0.11	6	0.56	0.14	2
Iris	0.72	0.2	2	0.73	0.1	1	0.72	0.11	3	0.62	0.15	6	0.63	0.15	5	0.7	0.1	4
Iris2D	0.79	0.17	1	0.7	0.07	2	0.69	0.11	3	0.62	0.12	6	0.64	0.1	5	0.65	0.03	4
Vary-density	0.83	0.15	1	0.69	0.06	6	0.76	0.14	2	0.7	0.15	5	0.7	0.13	4	0.7	0.1	3
Diagnosis_II	0.65	0.05	1	0.55	0.08	5	0.61	0.08	2	0.58	0.1	3	0.45	0.11	6	0.57	0.08	4
Appendicitis	0.5	0	1	0.5	0	1	0.5	0	1	0.5	0	1	0.5	0	1	0.5	0	1
Avg. Rank	**1.71**			2.71			2.33			3.86			4.48			3.52		

Table 3 shows that GOA achieves the best results in 14 out of 21 datasets in the precision metric and ranks first in average ranking success with a score of 1.71. The algorithm that closely follows GOA in average ranking success is AOA. AOA ranks first in 6 out of 21 datasets with a score of 2.33. However, it shares the first place with GOA in 3 out of these 6 datasets.

Table 4

Experimental Results Of The Algorithms For Specificity Metric

	GOA			AHA			AOA			COA			ABC			TSA		
Datasets	**A**	**S**	**R**	**A**	**S**	**R**	**A**	**S**	**R**	**A**	**S**	**R**	**A**	**S**	**R**	**A**	**S**	**R**
Aniso	0.93	0.08	3	0.94	0.06	2	0.94	0.07	1	0.93	0.07	4	0.89	0.07	6	0.90	0.06	5
Blobs	0.95	0.08	5	0.97	0.05	2	0.98	0.05	1	0.95	0.08	6	0.96	0.06	4	0.96	0.06	3
Varied	0.94	0.06	1	0.93	0.04	2	0.91	0.05	4	0.92	0.05	3	0.88	0.05	6	0.89	0.05	5
Aggregation	0.88	0.02	3	0.89	0.02	1	0.88	0.02	2	0.88	0.02	4	0.87	0.02	6	0.87	0.02	5
Balance	0.88	0.06	1	0.81	0.07	5	0.87	0.07	2	0.82	0.07	4	0.78	0.08	6	0.86	0.05	3
Smiley	0.75	0.02	1	0.75	0	3	0.75	0	3	0.75	0	3	0.75	0.00	2	0.75	0.00	4
Mouse	0.87	0.08	1	0.80	0.08	4	0.83	0.08	2	0.74	0.09	6	0.82	0.07	3	0.80	0.06	5
Ionosphere	0.50	0	1	0.50	0	1	0.50	0	1	0.50	0	1	0.50	0	1	0.50	0	1
Liver	0.68	0.04	2	0.68	0.03	4	0.68	0.03	5	0.68	0.03	3	0.67	0.03	6	0.69	0.03	1
Ecoli	0.91	0.02	1	0.89	0.02	3	0.90	0.02	2	0.88	0.02	5	0.88	0.02	6	0.89	0.02	4
Vertebral3	0.81	0.09	4	0.83	0.07	2	0.85	0.04	1	0.78	0.10	5	0.77	0.08	6	0.82	0.08	3
Pathbased	0.73	0.06	1	0.71	0.08	4	0.72	0.07	2	0.68	0.09	5	0.72	0.06	3	0.67	0.09	6
Heart	0.50	0	1	0.50	0	1	0.50	0	1	0.50	0	1	0.50	0	1	0.50	0	1
Glass	0.84	0.01	1	0.84	0.01	3	0.84	0.01	4	0.84	0.01	2	0.83	0.01	6	0.84	0.01	5
Seeds	0.80	0.06	1	0.77	0.04	3	0.80	0.04	2	0.74	0.06	6	0.74	0.07	5	0.75	0.05	4
Wine	0.75	0.08	5	0.79	0.08	2	0.77	0.08	4	0.78	0.07	3	0.69	0.06	6	0.79	0.08	1
Iris	0.86	0.10	2	0.87	0.05	1	0.86	0.06	3	0.81	0.07	6	0.82	0.07	5	0.85	0.05	4
Iris2D	0.90	0.09	1	0.85	0.04	2	0.85	0.05	3	0.81	0.06	6	0.82	0.05	5	0.82	0.02	4
Vary-density	0.91	0.07	1	0.84	0.03	6	0.88	0.07	2	0.85	0.08	5	0.85	0.06	4	0.85	0.05	3
Diagnosis_II	0.98	0.05	1	0.88	0.08	5	0.94	0.07	2	0.91	0.10	3	0.79	0.11	6	0.90	0.08	4
Appendicitis	0.50	0	1	0.50	0	1	0.50	0	1	0.50	0	1	0.50	0	1	0.50	0	1
Avg. Rank		**1.81**			2.71			2.29			3.90			4.48			3.43	

In Table 4, which details the results of the specificity metric, GOA again stands out and achieved the best results in 14 out of 21 datasets with an average score of 1.81. AOA ranked second again in this metric with an average score of 2.29.

Table 5

Experimental Results Of The Algorithms For Precision Metric

	GOA			AHA			AOA			COA			ABC			TSA		
Datasets	**A**	**S**	**R**	**A**	**S**	**R**	**A**	**S**	**R**	**A**	**S**	**R**	**A**	**S**	**R**	**A**	**S**	**R**
Aniso	0.79	0.25	4	0.88	0.13	1	0.87	0.20	2	0.87	0.16	3	0.75	0.23	6	0.78	0.20	5
Blobs	0.86	0.23	6	0.93	0.16	2	0.95	0.14	1	0.86	0.24	5	0.91	0.16	4	0.91	0.16	3
Varied	0.83	0.21	2	0.90	0.04	1	0.82	0.17	4	0.83	0.16	3	0.75	0.19	6	0.78	0.17	5
Aggregation	0.15	0.08	2	0.14	0.07	3	0.16	0.06	1	0.13	0.08	4	0.08	0.07	6	0.10	0.07	5

Balance	0.57	0.12	1	0.52	0.09	4	0.54	0.09	2	0.49	0.11	5	0.44	0.14	6	0.54	0.06	3
Smiley	0.20	0.03	1	0.19	5.67E-17	3	0.19	5.67E-17	3	0.19	5.67E-17	3	0.20	0.03	2	0.19	9.35E-05	4
Mouse	0.65	0.23	1	0.46	0.17	5	0.56	0.21	2	0.36	0.21	6	0.55	0.18	3	0.55	0.16	4
Ionosphere	0.32	1.70E-16	1	0.32	1.70E-16	1	0.32	1.70E-16	1	0.32	1.70E-16	1	0.32	1.70E-16	1	0.32	1.70E-16	1
Liver	0.32	0.10	2	0.31	0.11	3	0.29	0.10	6	0.29	0.12	5	0.30	0.11	4	0.35	0.08	1
Ecoli	0.13	0.07	1	0.10	0.04	3	0.11	0.06	2	0.10	0.06	5	0.06	0.05	6	0.10	0.08	4
Vertebral3	0.42	0.17	4	0.45	0.14	2	0.53	0.10	1	0.40	0.18	6	0.40	0.16	5	0.45	0.17	3
Pathbased	0.39	0.17	3	0.39	0.23	2	0.41	0.20	1	0.34	0.23	5	0.36	0.13	4	0.25	0.14	6
Heart	0.22	1.42E-16	1	0.22	1.42E-16	1	0.22	1.42E-16	1	0.22	1.42E-16	1	0.22	1.42E-16	1	0.22	1.42E-16	1
Glass	0.10	0.07	2	0.09	0.06	3	0.07	0.04	4	0.11	0.07	1	0.06	0.06	6	0.07	0.04	5
Seeds	0.50	0.23	1	0.44	0.11	3	0.50	0.16	2	0.39	0.16	5	0.34	0.20	6	0.41	0.14	4
Wine	0.31	0.19	5	0.45	0.18	2	0.37	0.19	4	0.43	0.19	3	0.19	0.14	6	0.47	0.15	1
Iris	0.62	0.29	2	0.67	0.19	1	0.59	0.19	3	0.49	0.17	6	0.53	0.20	5	0.56	0.18	4
Iris2D	0.71	0.26	1	0.58	0.16	3	0.60	0.20	2	0.53	0.19	4	0.52	0.17	5	0.50	0.07	6
Vary-density	0.76	0.24	1	0.61	0.17	4	0.67	0.24	2	0.61	0.23	5	0.62	0.20	3	0.59	0.19	6
Diagnosis_II	0.66	0.03	1	0.57	0.10	5	0.62	0.09	2	0.57	0.15	4	0.43	0.17	6	0.60	0.04	3
Appendicitis	0.40	0	1	0.40	0	1	0.40	0	1	0.40	0	1	0.40	0	1	0.40	0	1
Avg. Rank		**2.05**			2.52			2.24			3.86			4.38			3.57	

According to the data presented in Table 5, concerning the sensitivity metric, GOA achieved the highest success in 11 out of the 21 data sets and ranks first in average success ranking with a score of 2.05. AOA, coming in second for average success ranking, has a score of 2.24.

Table 6

Experimental Results Of The Algorithms For F1-Score Metric

	GOA			AHA			AOA			COA			ABC			TSA		
Datasets	**A**	**S**	**R**	**A**	**S**	**R**	**A**	**S**	**R**	**A**	**S**	**R**	**A**	**S**	**R**	**A**	**S**	**R**
Aniso	0.82	0.22	4	0.86	0.14	1	0.86	0.19	2	0.83	0.18	3	0.73	0.20	6	0.76	0.18	5
Blobs	0.88	0.20	5	0.93	0.14	2	0.95	0.13	1	0.87	0.22	6	0.90	0.16	4	0.91	0.15	3
Varied	0.84	0.18	2	0.85	0.10	1	0.79	0.16	4	0.81	0.14	3	0.71	0.15	6	0.75	0.14	5
Aggregation	0.15	0.07	3	0.16	0.07	2	0.16	0.07	1	0.14	0.07	4	0.09	0.06	6	0.11	0.07	5
Balance	0.56	0.10	1	0.47	0.11	5	0.54	0.10	2	0.47	0.11	4	0.41	0.14	6	0.53	0.07	3
Smiley	0.22	0.04	1	0.22	8.50E-17	3	0.22	8.50E-17	3	0.22	8.50E-17	3	0.22	0.01	2	0.22	2.57E-04	4
Mouse	0.68	0.21	1	0.49	0.17	5	0.59	0.19	2	0.40	0.20	6	0.56	0.14	3	0.54	0.13	4
Ionosphere	0.39	5.67E-17	1	0.39	5.67E-17	1	0.39	5.67E-17	1	0.39	5.67E-17	1	0.39	5.67E-17	1	0.39	5.67E-17	1
Liver	0.30	0.07	2	0.29	0.05	3	0.29	0.06	4	0.28	0.06	5	0.28	0.05	6	0.31	0.05	1

Ecoli	0.15	0.06	1	0.12	0.04	3	0.14	0.06	2	0.10	0.05	4	0.07	0.05	6	0.10	0.05	5
Vertebral3	0.47	0.15	4	0.49	0.11	2	0.55	0.06	1	0.42	0.17	5	0.41	0.13	6	0.48	0.15	3
Pathbased	0.38	0.10	1	0.34	0.16	4	0.37	0.12	2	0.30	0.16	5	0.34	0.12	3	0.27	0.14	6
Heart	0.31	1.70E-16	1	0.31	1.70E-16	1	0.31	1.70E-16	1	0.31	1.70E-16	1	0.31	1.70E-16	1	0.31	1.70E-16	1
Glass	0.12	0.06	2	0.11	0.05	3	0.10	0.04	4	0.12	0.05	1	0.07	0.06	6	0.09	0.04	5
Seeds	0.51	0.18	2	0.45	0.10	3	0.51	0.12	1	0.37	0.15	5	0.36	0.16	6	0.41	0.11	4
Wine	0.36	0.18	5	0.47	0.18	1	0.41	0.17	4	0.45	0.17	3	0.24	0.13	6	0.46	0.17	2
Iris	0.65	0.26	2	0.67	0.15	1	0.64	0.16	3	0.52	0.17	6	0.54	0.16	5	0.61	0.14	4
Iris2D	0.74	0.23	1	0.60	0.11	3	0.60	0.14	2	0.53	0.14	6	0.54	0.13	5	0.54	0.02	4
Vary-density	0.78	0.20	1	0.60	0.09	6	0.69	0.19	2	0.60	0.20	5	0.62	0.16	3	0.61	0.14	4
Diagnosis_II	0.65	0.05	1	0.54	0.10	5	0.60	0.09	2	0.57	0.13	3	0.42	0.14	6	0.56	0.08	4
Appendicitis	0.45	0	1	0.45	0	1	0.45	0	1	0.45	0	1	0.45	0	1	0.45	0	1
Avg. Rank		**2.00**			2.67			2.14			3.81			4.48			3.52	

Lastly, Table 6 shows that GOA achieved the best results in 11 out of 21 datasets in the f1-score metric and ranked first in average ranking success with a score of 2.00. The algorithm that closely follows GOA in average ranking success is AOA. AOA ranked first in 7 out of 21 datasets with a score of 2.14.

When we analyze the average results and rankings across precision, specificity, F1-score and sensitivity metrics, GOA consistently emerges as the best performing algorithm. GOA proves its effectiveness by achieving the best results on 14 out of 21 datasets in sensitivity and specificity metrics, with average ranking scores of 1.71 and 1.81, respectively. In precision, GOA leads on 11 datasets, maintaining an average ranking of 2.05. Similarly, in f1-score metric, GOA stands out with the best results on 11 datasets and an average ranking score of 2.00. AOA follows GOA closely. It achieves the best rankings on 6 datasets for sensitivity and specificity, 7 datasets for precision and f1-score, and in some cases, it shares the first place with GOA. These findings highlight the consistent performance superiority of GOA in training ANNs across multiple evaluation metrics.

Discussion and Conclusion

In this article, the performances of six different meta-heuristic algorithms for training artificial neural networks were compared. ANN training was performed on 21 different classification datasets using Grasshopper Optimization Algorithm, Artificial Hummingbird Algorithm, Arithmetic Optimization Algorithm, Crayfish Optimization Algorithm, Artificial Bee Colony and Tree-Seed Algorithm. The performances of the algorithms were evaluated using four basic metrics such as precision, specificity, F1-score and sensitivity. The experimental results obtained show that GOA in particular provides high success compared to other algorithms for ANN training. GOA achieved the best results in 14 out of 21 datasets and ranked first in the average success ranking. Although AOA and other algorithms also achieved good results, GOA was the algorithm with the highest overall success. These results show that metaheuristic algorithms provide an effective solution to solve complex weight update processes in ANN training.

It was observed that as the number of features of the datasets used in the study increases, the problem size also increases significantly. From this perspective, it is seen that metaheuristic algorithms, especially GOA, are successful on large-scale continuous problems. Comparisons between the algorithms were examined in detail in terms of both average metric values and success rankings. As a result of the comparisons made, it is seen that metaheuristic algorithms have significant potential for ANN training and provide effective solutions for such complex problems.

In further studies, it is recommended that these algorithms can be tested on different problems and spread to a wider application area. Additionally, developing hybrid approaches by combining different meta-heuristic algorithms may be an important research topic for future studies.

References

Abdullahi Mu'azu, M. (2023). Hybridized artificial neural network with metaheuristic algorithms for bearing capacity prediction. *Ain Shams Engineering Journal*, *14*(5), 101980. https://doi.org/10.1016/j.asej.2022.101980

Abualigah, L., Diabat, A., Mirjalili, S., Abd Elaziz, M., & Gandomi, A. H. (2021). The Arithmetic Optimization Algorithm. *Computer Methods in Applied Mechanics and Engineering*, *376*, 113609. https://doi.org/10.1016/j.cma.2020.113609

Agrawal, P. (2021, February 2). *Metaheuristic Algorithms on Feature Selection: A Survey of One Decade of Research (2009-2019)*. https://ieeexplore.ieee.org/abstract/document/9344597

Akel, V., & Karacameydan, F. (2012). *Yatırım Fonları Net Varlık Değerlerinin Yapay Sinir Ağları Yöntemiyle Tahmin Edilmesi*. https://earsiv.anadolu.edu.tr/xmlui/handle/11421/161

Aksu, İ. Ö., Esenboğa, B., Yavuzdeğer, A., Nazligül, H., TekiN, P., & DemiRdelen, T. (2022). *YSA Tabanlı Metasezgisel Yöntemlerle Kısa Vadeli Solar Güç Tahmini*.

Aslanov, T., Çavdar, B., Öztürk, Ö., Başlık, Ş., & Akyazi, Ö. (2023, September 22). *İki Alanlı Termal Güç Sisteminde AHA Tabanlı PID, FOPID ve FOPI-FOPD Denetleyicilerin Siber Saldırı Analizi Cyber Attack Analysis of AHA Based PID, FOPID and FOPI- FOPD Controllers in Two Area Thermal Power System*.

Ateş, K. T. (2022). Çok Katmanlı Yapay Sinir Ağı Modeli ve Kültürel Algoritma Modeli Kullanılarak Geliştirilen Melez Yöntem ile Kısa Vadeli Fotovoltaik Enerji Santrali Çıkış Gücü Tahmini. *Osmaniye Korkut Ata Üniversitesi Fen Bilimleri Enstitüsü Dergisi*, *5*(1), Article 1. https://doi.org/10.47495/okufbed.1028813

Atlı, F. D. (2022). *Yapay Sinir Ağlarının Eğitimi Için Salp Sürü Optimizasyonu Algoritmasının İYileştirilmesi* [M.Sc., Necmettin Erbakan University (Turkey)]. https://www.proquest.com/docview/2784407806/abstract/D25DEFE4EE024945PQ/1

Ayaz, H. I., & Kamisli Ozturk, Z. (2021). A mathematical model and a heuristic approach for train seat scheduling to minimize dwell time. *Computers & Industrial Engineering*, *160*, 107590. https://doi.org/10.1016/j.cie.2021.107590

Bakır, H. (2024). A novel artificial hummingbird algorithm improved by natural survivor method. *Neural Computing and Applications*, *36*(27), 16873–16897. https://doi.org/10.1007/s00521-024-09928-z

Çalışkan, M. M. T., & Deniz, D. (2015). Yapay Sinir Ağlarıyla Hisse Senedi Fiyatları ve Yönlerinin Tahmini. *Eskişehir Osmangazi Üniversitesi İktisadi ve İdari Bilimler Dergisi*, *10*(3), Article 3.

Çelik, Y. (2013). *Optimizasyon problemlerinde bal arıları evlilik optımızasyonu algoritmasının (marriage in honey bee optimization-MBO) performansının geliştirilmesi* [Doktora, Selçuk Üniversitesi]. https://tez.yok.gov.tr/UlusalTezMerkezi/tezDetay.jsp?id=oDMHpvC1u3eqlJ8Sf0Fwrg&no=6_0T-

032nAUXAYvxRNfRNQ

Çınaroğlu, E., & Avcı, T. (2020). THY Hisse Senedi Değerinin Yapay Sinir Ağları İle Tahmini. *Atatürk Üniversitesi İktisadi ve İdari Bilimler Dergisi*, *34*(1), Article 1. https://doi.org/10.16951/atauniiibd.530322

Çörekcioğlu, M., Ercan, E., & Elibüyük, S. A. (2021). Yapay Sinir Ağı Yöntemlerinin Tekstil Sektöründe Kullanım Uygulamaları. *Teknik Bilimler Dergisi*, *11*(2), Article 2. https://doi.org/10.35354/tbed.884531

Dhal, K. G., Sasmal, B., Das, A., Ray, S., & Rai, R. (2023). A Comprehensive Survey on Arithmetic Optimization Algorithm. *Archives of Computational Methods in Engineering*, *30*(5), 3379–3404. https://doi.org/10.1007/s11831-023-09902-3

Dokeroglu, T., Sevinc, E., Kucukyilmaz, T., & Cosar, A. (2019). A survey on new generation metaheuristic algorithms. *Computers & Industrial Engineering*, *137*, 106040. https://doi.org/10.1016/j.cie.2019.106040

Emambocus, B. A. S., Jasser, M. B., & Amphawan, A. (2023). A Survey on the Optimization of Artificial Neural Networks Using Swarm Intelligence Algorithms. *IEEE Access*. https://ieeexplore.ieee.org/abstract/document/10004960

Geem, Z. W., Kim, J. H., & Loganathan, G. V. (2001). A New Heuristic Optimization Algorithm: Harmony Search. *SIMULATION*, *76*(2), 60–68. https://doi.org/10.1177/003754970107600201

Gölcük, İ., Ozsoydan, F. B., & Durmaz, E. D. (2023). An improved arithmetic optimization algorithm for training feedforward neural networks under dynamic environments. *Knowledge-Based Systems*, *263*, 110274. https://doi.org/10.1016/j.knosys.2023.110274

Holland, J. H. (1992). Genetic Algorithms. *Scientific American*, *267*(1), 66–73.

Jama, B. S. A. (2021). *Modifiye edilmiş karınca aslanı optimizasyon algoritması kullanılarak bölge büyütme yöntemi ile gri seviye görüntü segmentasyonu* [Master Thesis, Konya Teknik Üniversitesi]. https://gcris.ktun.edu.tr/handle/20.500.13091/2250

Jia, H., Rao, H., Wen, C., & Mirjalili, S. (2023). Crayfish optimization algorithm. *Artificial Intelligence Review*, *56*(2), 1919–1979. https://doi.org/10.1007/s10462-023-10567-4

Jia, H., Zhou, X., Zhang, J., Abualigah, L., Yildiz, A. R., & Hussien, A. G. (2024). Modified crayfish optimization algorithm for solving multiple engineering application problems. *Artificial Intelligence Review*, *57*(5), 127. https://doi.org/10.1007/s10462-024-10738-x

Jiadong, Q., Ohl, J. P., & Tran, T.-T. (2024). Predicting clay compressibility for foundation design with high reliability and safety: A geotechnical engineering perspective using artificial neural network and five metaheuristic algorithms. *Reliability Engineering & System Safety*, *243*, 109827. https://doi.org/10.1016/j.ress.2023.109827

Karaboga, D. (2005). An idea based on honey bee swarm for numerical optimization. *Technical Report-Tr06, Erciyes University, Engineering Faculty, Computer Engineering Department*.

Karaboga, D., & Akay, B. (2009). A comparative study of Artificial Bee Colony algorithm. *Applied Mathematics and Computation*, *214*(1), 108–132. https://doi.org/10.1016/j.amc.2009.03.090

Karaboga, D., & Basturk, B. (2007). A powerful and efficient algorithm for numerical function optimization: Artificial bee colony (ABC) algorithm. *Journal of Global*

Optimization, *39*(3), 459–471. https://doi.org/10.1007/s10898-007-9149-x

Karakoyun, M. (2024). Artificial neural network training using a multi selection artificial algae algorithm. *Engineering Science and Technology, an International Journal*, *53*, 101684. https://doi.org/10.1016/j.jestch.2024.101684

Kashan, A. H. (2009). League Championship Algorithm: A New Algorithm for Numerical Function Optimization. *2009 International Conference of Soft Computing and Pattern Recognition*, 43–48. https://doi.org/10.1109/SoCPaR.2009.21

Kaytan, M., Yeroğlu, C., & Aydilek, İ. B. (2020). Yapay Sinir Ağları Eğitiminde Kullanılan Optimizasyon Yöntemlerinin İncelenmesi ve Kan Nakli Hizmet Merkezi Veri Seti Üzerinden Değerlendirilmesi. *Computer Science*, *5*(2), Article 2.

Kennedy, J., & Eberhart, R. (1995). Particle swarm optimization. *Proceedings of ICNN'95 - International Conference on Neural Networks*, *4*, 1942–1948 vol.4. https://doi.org/10.1109/ICNN.1995.488968

Keskenler, M. F., & Keskenler, E. F. (2017). *Geçmişten Günümüze Yapay Sinir Ağları ve Tarihçesi*.

Khodadadi, N., Mirjalili, S. M., Zhao, W., Zhang, Z., Wang, L., & Mirjalili, S. (2023). Multi-Objective Artificial Hummingbird Algorithm. In A. Biswas, C. B. Kalayci, & S. Mirjalili (Eds.), *Advances in Swarm Intelligence: Variations and Adaptations for Optimization Problems* (pp. 407–419). Springer International Publishing. https://doi.org/10.1007/978-3-031-09835-2_22

Kiran, M. S. (2015). TSA: Tree-seed algorithm for continuous optimization. *Expert Systems with Applications*, *42*(19), 6686–6698. https://doi.org/10.1016/j.eswa.2015.04.055

Kirkpatrick, S., Gelatt, C. D., & Vecchi, M. P. (1983). Optimization by Simulated Annealing. *Science*, *220*(4598), 671–680. https://doi.org/10.1126/science.220.4598.671

Köprü, E. Y. (2020). *Yapay Sinir Ağları İle Sıvı Ham Demir Tamini Ve 5. Yüksek Fırın Uygulaması* [Thesis]. http://acikerisim.karabuk.edu.tr:8080/xmlui/handle/123456789/962

Meraihi, Y., Gabis, A. B., Mirjalili, S., & Ramdane-Cherif, A. (2021). Grasshopper Optimization Algorithm: Theory, Variants, and Applications. *IEEE Access*, *9*, 50001–50024. IEEE Access. https://doi.org/10.1109/ACCESS.2021.3067597

Mohamed, A. W., Hadi, A. A., & Mohamed, A. K. (2020). Gaining-sharing knowledge based algorithm for solving optimization problems: A novel nature-inspired algorithm. *International Journal of Machine Learning and Cybernetics*, *11*(7), 1501–1529. https://doi.org/10.1007/s13042-019-01053-x

Özdemir, A., & Pamuk, N. (2021). Kısmi Gölgelenme Şartları Altındaki Kompleks Yapılı Fotovoltaik Enerji Sistemlerinde Maksimum Güç Noktası Takibinin Metasezgisel Algoritmalar Kullanılarak İncelenmesi. *Avrupa Bilim ve Teknoloji Dergisi*, *31*, Article 31. https://doi.org/10.31590/ejosat.1006248

Özdemir, F. N., & Özkış, A. (2024). Kar Erime Optimizasyonu Algoritması ile Çok Katmanlı Yapay Sinir Ağının Eğitimi. *Çukurova Üniversitesi Mühendislik Fakültesi Dergisi*, *39*(2), Article 2. https://doi.org/10.21605/cukurovaumfd.1514409

Özkan, F. (2012). Döviz Kuru Tahmininde Parasal Model ve Yapay Sinir Ağları Karşılaştırması. *Business and Economics Research Journal*.

Özkaya, S., Conker, Ç., & Bilgiç, H. H. (2021). Esnek Robot Kol Sistemi İçin Lqr Denetleyici Parametrelerinin Metasezgisel Algoritmalar Kullanılarak Belirlenmesi. *Konya Journal of Engineering Sciences*, *9*(3), Article 3. https://doi.org/10.36306/konjes.896087

Özmen, T., Kuzu, Ü., Koçyiğit, Y., & Sarnel, H. (2023). Metasezgisel yöntemlerle öznitelik sayısını azaltarak diyabetin erken dönemde tespiti. *Pamukkale Üniversitesi Mühendislik Bilimleri Dergisi*, *29*(6), Article 6.

Qaddoura, R., Faris, H., Aljarah, I., & Castillo, P. A. (2020). EvoCluster: An Open-Source Nature-Inspired Optimization Clustering Framework in Python. In P. A. Castillo, J. L. Jiménez Laredo, & F. Fernández de Vega (Eds.), *Applications of Evolutionary Computation* (pp. 20–36). Springer International Publishing. https://doi.org/10.1007/978-3-030-43722-0_2

Rao, R. V., Savsani, V. J., & Vakharia, D. P. (2012). Teaching–Learning-Based Optimization: An optimization method for continuous non-linear large scale problems. *Information Sciences*, *183*(1), 1–15. https://doi.org/10.1016/j.ins.2011.08.006

Rençber, Ö. F. (2018). *Basamak Korelasyon, Kohonen ve ANFIS Yapay Sinir Ağ Modellerinin Sınıflandırma Performanslarının Karşılaştırılması: Lojistik Performans Endeksi Üzerine Uygulama*.

Rere, L. M. R., Fanany, M. I., & Arymurthy, A. M. (2016). Metaheuristic Algorithms for Convolution Neural Network. *Computational Intelligence and Neuroscience*, *2016*(1), 1537325. https://doi.org/10.1155/2016/1537325

Sahman, M. A., Cinar, A. C., Saritas, I., & Yasar, A. (2019). Tree-seed algorithm in solving real-life optimization problems. *IOP Conference Series: Materials Science and Engineering*, *675*(1), 012030. https://doi.org/10.1088/1757-899X/675/1/012030

Saremi, S., Mirjalili, S., & Lewis, A. (2017). Grasshopper Optimisation Algorithm: Theory and application. *Advances in Engineering Software*, *105*, 30–47. https://doi.org/10.1016/j.advengsoft.2017.01.004

Shahvaroughi Farahani, M., & Razavi Hajiagha, S. H. (2021). Forecasting stock price using integrated artificial neural network and metaheuristic algorithms compared to time series models. *Soft Computing*, *25*(13), 8483–8513. https://doi.org/10.1007/s00500-021-05775-5

Tharwat, A. (2021). Classification assessment methods. *Applied Computing and Informatics*, *17*(1), 168–192. https://doi.org/10.1016/j.aci.2018.08.003

Ticknor, J. L. (2013). A Bayesian regularized artificial neural network for stock market forecasting. *Expert Systems with Applications*, *40*(14), 5501–5506. https://doi.org/10.1016/j.eswa.2013.04.013

Yildiz, B. (2023). Yeni Bir Hibrid Metasezgisel Algoritma İle Drone Kolunun Yapısal Optimizasyonu. *Makina Tasarım ve İmalat Dergisi*, *21*(2), Article 2. https://doi.org/10.56193/matim.1302774

Zaimoğlu, E. A. (2023). *Büyük boyutlu veriler için metasezgisel yöntemler ile öznitelik indirgemede yeni bir yaklaşım geliştirilmesi = Developing a new approach to feature selection with metaheuristic methods for large scale data* [doctoralThesis, Sakarya Üniversitesi]. https://acikerisim.sakarya.edu.tr/handle/20.500.12619/101468

Zhao, W., Wang, L., & Mirjalili, S. (2022). Artificial hummingbird algorithm: A new bio-inspired optimizer with its engineering applications. *Computer Methods in Applied Mechanics and Engineering*, *388*, 114194. https://doi.org/10.1016/j.cma.2021.114194

About the Authors

Münevver BAHAR has a bachelor's degree in Computer Engineering from Necmettin Erbakan University. She is currently doing her master's degree at Necmettin Erbakan University, Department of Computer Engineering. She works in the fields of Artificial Intelligence, Machine Learning and Optimization.
E-mail: munevverbahar1@gmail.com, **ORCID:** 0009-0009-0489-0884

Halil İbrahim AYAZ, PhD, He earned a BS and an MS degree in Industrial Engineering from Hacettepe University. He earned a PhD degree in Industrial Engineering from Eskişehir Technical University. His research interests include machine learning, data mining and analysis, scheduling, and optimization. He has two journal articles and seven studies presented at international meetings. Currently, he works as an assistant professor doctor in the Industrial Engineering department at Necmettin Erbakan University.
E-mail: hiayaz@erbakan.edu.tr, **ORCID:** 0000-0001-5547-6485

Murat KARAKOYUN, received the master and Ph.D. degrees from the Department of Computer Engineering of Selçuk University and Konya Technical University, respectively. He is working as an Assistant Professor at Computer Engineering Department of Necmettin Erbakan University. Areas of interest: artificial intelligence, image processing, data mining, machine learning.
E-mail: mkarakoyun@erbakan.edu.tr, **ORCID:** 0000-0002-0677-9313

Similarity Index
The similarity index obtained from the plagiarism software for this book chapter is 19%.

Optimizing Energy Efficiency in IoT-Based Smart Home Systems

Gamze AKÇAY
Necmettin Erbakan University

Behice Gülsüm ÇELİK
Necmettin Erbakan University

Alperen EROĞLU
Necmettin Erbakan University

To Cite This Chapter

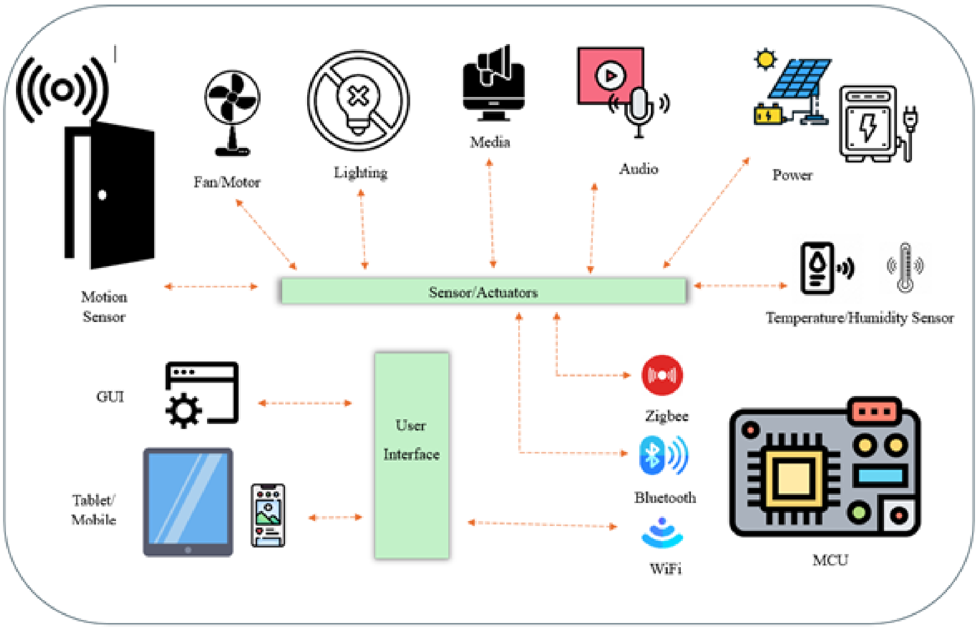
Akçay, G., Çelik, B. G., & Eroğlu, A. (2024). Optimizing Energy Efficiency in IoT-Based Smart Home Systems. In S. Koçer & Ö. Dündar (Eds.), *Intelligent Systems and Optimization in Engineering* (pp. 53-66). ISRES Publishing.

Introduction

The advancement of technology, the equipping of communication infrastructures with new-generation systems, and the need for a completely digital and autonomous world of future communications such as 5G and beyond (5BG) and 6G have required people to enter a different evolution with the adaptation of these technologies not only in their business lives but also in many areas such as health, agriculture, and defense (Eroglu, 2022). New communication and sensing innovations also allow connections to be ubiquitous, allowing communication among devices from anywhere at any time. This connectivity form is called the Internet of Things (Kaya et al., 2019). One of these important areas is that smart homes must be built with new-generation requirements. The smart home automation paradigm has increased significantly in recent years due to the growing demands of people and their desire for comfortable living. However, with the rising use of smart home automation, significant energy consumption poses a problem (Geraldo Filho, et al., 2019).

A smart home can be seen as a living space where devices, systems, and services received by users are automated, optimized, and integrated using sensors, digital systems, and IoT platforms. With smart home technology, users can remotely manage devices in the home and live in a way that provides energy efficiency and increased comfort. As depicted in Figure 1, a smart home uses some off-the-shelf IoT sensors and actuators such as motion sensors, thermal sensors, lighting, power, audio systems, air conditioning, and a fan or motor. A smart home also includes various communication technologies such as Wi-Fi, Zigbee, and Bluetooth. A smart home has a user interface can be controlled via remote access with a web interface or mobile app. All these basic requirements consume energy.

Figure 1
A Basic Smart Home Consisting of IoT- instruments.



In IoT-based smart home applications, many situations such as making life easier, controlling health status, etc. are realized using the data collected. While various artificial intelligence algorithms are analyzed by statistical and mathematical modeling and turned into valuable insight, on the other hand, the production, transmission, storage, and processing of this data, i.e. the use and existence of software, hardware, and data, increase energy consumption (Eroglu and Unlu Eroglu, 2023). At this point, using energy efficiently by applying the optimization method to find the best-desired result is inevitable. Hence, in this chapter, we address this problem.

Goal and Motivation

The Internet of Things (IoT) paradigm has gained great importance. It is a global network of physical objects embedded with sensors, software, and other technologies for data connection and sharing with other devices/systems over the Internet that can communicate with each other via IoT communication protocols. The Internet of Things shows its presence in many areas. It has made progress, especially in smart home automation. In addition to important innovations such as remote electronic device control and system security in home automation, it has also provided comfort and convenience. In addition to its positive effects, it has also brought problems that need to be solved. One of these main problems is that it has seriously increased energy consumption. The problem of reducing and optimizing energy consumption has been the subject of many research studies. The study focuses on the pivotal issue of energy efficiency in IoT-driven smart home applications.

Research Questions

In this study, we want to understand and find answers to the research questions listed below.

- How are IoT devices and the networks created by these appliances used in smart home applications?
- What are the devices commonly used in smart home applications?
- What are the consumption amounts of the instruments?
- Can we turn off the devices, put them in sleep mode, or limit their communication to provide less energy, taking into account the user’s comfort?

- What are the data sets used in the literature?
- Can optimization solutions be applied in smart home applications using these data sets?
- Can these solutions be generalized?

Problem Definition

In recent years, there has been a significant increase in energy consumption associated with smart home systems. Effective management of energy consumption is crucial for both reducing environmental impacts and lowering energy costs for users. This study focuses on preserving user comfort and flexibility while enhancing energy efficiency within smart homes. In this context, the primary aim is to optimize energy consumption levels in smart home environments.

For this research, we applied three different optimization algorithms to our dataset: Grey Wolf Optimization (GWO), Genetic Algorithm (GA), and Particle Swarm Optimization (PSO). Each algorithm approaches energy utilization in the smart home context from diverse perspectives, offering a range of potential improvements. We assume the desired values of temperature and humidity are provided by the home user so that we consider user comfort.

Methodology

In our study, the most up-to-date studies in the literature on energy-efficient smart home applications are investigated and presented with a comparative analysis. In this study, we first find the data set obtained from an IoT-based smart home application. Then, the most commonly applied optimization algorithms are applied to this data set by considering user provided optimal temperature and humidity values.

Contributions

This study focuses on IoT-based smart home applications with real-life implementations and simulations. Studies implemented using embedded systems and shared data sets are both categorically examined and implementing the three most commonly used optimization algorithms is carried out using a data set. Since very few studies generally examine efficient IoT-based home applications using energy optimizations, this gap in the literature is filled in this study.

In this context, we approach the smart home energy optimization problem with a comparative analysis and comprehensive evaluation. We also compare the results of three different optimization algorithms (PSO, GA, and GWA) on the Appliances Energy Prediction dataset published on Kaggle, which to our knowledge has yet to be run before.

The rest of the chapter is like the following.

The subsequent section presents the state of the art for energy-efficient applications in IoT-based smart home systems. The “Energy Optimization in IoT-based Smart Home Systems” section defines the common datasets in the literature and three of the most common preferred optimization algorithms. In the “Experimental Results and Discussion” section, we present the outcomes of The algorithms PSO, GA, and GWA, and a detailed discussion regarding the energy optimization problem in IoT-based smart homes. The section “Future Directions” discusses the most possible open research questions for IoT-based smart home energy management systems. Ultimately, we conclude the chapter by focusing on the most prominent research observations and future studies.

Related Work

This section comprehensively analyzes smart home energy management research. In the literature, studies on smart homes can be divided into different categories. These vary according to how the relevant scenario is realized and at which stages the optimization solution is used. The studies in the first category can be seen as those using a real testbed or experimenting and verifying with simulation. A real testbed can be directly obtained on platforms using embedded systems such as Arduino or Raspberry Pie (Francis et al., 2023) and various sensors such as temperature and humidity, while other studies are IoT-supported smart home applications, where there are many more devices and their designs can be considered as studies where the energy efficiency of smart buildings (Goudarzi et al., 2021) and studies can be evaluated in this context. The studies in the second category are encountered with studies such as minimizing energy consumption of optimization algorithms, user comfort, and bills. In contrast, the other category is used in solutions on parameter optimization and selection of algorithms that will be used in predictive analysis, scheduling systems, and decision support systems (Priyadarshini et al., 2022).

Obtaining energy consumption for energy management using accurate data taken at the correct intervals is emphasized in many studies (Leitao et al., 2020). Home energy management systems (HEMs) are important systems that can guide the user with solutions like timing by using the correct consumption data. A typical HEMs consists of a user interface which can be a mobile application or a remote terminal to make communication with a device and some measurements and state information about appliances; a central management unit to monitor and control energy consumption; measuring and sensing devices to get some physical phenomena such as light, humidity, and temperature; other electronic units including air conditioners, and other smart domestic appliances (Leitao et al., 2020). In a smart home management system smart meters are used to record the energy consumption of each device as well as the consumption resulting from people's activities. In a HEMS, sensors continuously collect data. This data is generated by monitoring the activities in the home. Usually, the energy consumption signals of the devices are recorded, but techniques such as Non-Intrusive Load Monitoring (NILM) can be used to determine the consumption of individual devices. This data is transmitted to a central management unit and processed. Weather information and billing forecasts can also be collected and used in the optimization steps. A HEMS monitors the characteristics and preferences of the home users, takes into account user behavior and profiles, and optimizes the operating schedules of the devices according to physical constraints. The communication protocols between the central platform and the smart devices ensure the implementation of the most appropriate planning. In particular, the communication protocols are also being developed to ensure energy efficiency.

Environmentally friendly applications and preventing waste of resources are made possible with smart homes, smart buildings, and green applications. In particular, there is a need for efficient energy management for the optimal use of electrical energy-requiring devices and systems such as lighting. At this point, IoT-based designs and applications that facilitate accurate and timely data monitoring enable energy management and efficiency to be carried out dynamically. Internet of Things (IoT) home automation systems provide reliable and flexible communication between home devices and the user via the Internet. The increasing use of smart home devices, the widespread use of the Internet, the development of smartphone technology, and the rise of mobile communication standards offer comfort, security, and energy efficiency for users.

Big data (Al-Ali et al., 2017, Machorro-Cano et al., 2020), machine learning, and deep learning algorithms are important for smart home energy management systems. With the help of IoT devices, which are used especially in smart home management systems, it has become an easy application to track device behaviors, create certain patterns, and calculate energy consumption with various measurement techniques. Various analysis studies, predictive analytics studies, and optimization studies can be done on the data to be created here. In particular, the realization of models for making predictions with machine learning and deep learning methods has become an important solution that has taken its place in many studies (Devi et al., 2023).

Most of the studies utilize optimization techniques to make energy-aware smart home management. There are various kinds of optimization algorithms such as PSO, GA, GWA, Butterfly Optimization Algorithm (BOA), artificial neural network (ANN), decision support model (DSM), heuristic system identification (HIS), linear reinforcement learning controller (LRLC), Markov decision problems (MDP), model-based predictive control strategy (MBP), multinomial logistic regression (MLR), ant colony optimization algorithm (ACO), bat algorithm (BAT), and artificial bee colony (ABC), and Simplex Optimization Algorithm (SOA) to find an optimal solution for HEMs. In the literature, most of these algorithms are applied to consider single - or multiple-user comfort, reduction of energy consumption, home user behavior, and learning how to manage (Shah et al., 2019). The multi-objective optimization method has paramount importance for energy optimization. Multi-objective studies take into account important parameters such as temperature, humidity, climate data, the accurate information on energy consumption of different devices and services, as well as user satisfaction parameters, which can include the effect of user behaviors, especially in energy consumption problems (Wang et al., 2021).

All in all, when the studies on energy management in smart homes are examined, it is revealed that there are goals such as reducing power consumption and increasing the comfort index of users with optimization techniques. Especially in smart home environments where annual, weekly, hourly, and even instantaneous data can be collected, it is seen that energy consumption optimization, indoor environment parameter estimation, and energy-aware improvement techniques are important areas of study to achieve the relevant goals of energy management. In this study, we focus on the energy consumption optimization problem and try to show how different optimization techniques, which have not been applied to the relevant data set to the best of our knowledge, can be applied.

Energy Optimization in IoT-based Smart Home Systems

This section explains the dataset from Kaggle obtained from IoT-based smart home systems, and how the three most frequently encountered optimization algorithms in the literature are applied to this data set will be discussed.

Data Set

This dataset, obtained from Kaggle (Appliances Energy Prediction, 2017), is designed to analyze and optimize energy use in smart homes, consisting of 29 columns with 19,735 data entries each. The data headers include the measurement date (date), energy consumption of household appliances (appliances), lighting consumption (lights), and temperature measurements (T1-T9) obtained from various rooms (including the kitchen, living room, and bedroom) indicative of indoor temperature levels, along with relative humidity measurements (RH_1-RH_9) from these rooms, illustrating indoor humidity conditions. Furthermore, the dataset contains external conditions such as outdoor temperature and wind speed that can influence indoor temperatures. Data is collected using two sensors for temperature and humidity measurements, with four actuators (heater, cooler, humidifier, and dehumidifier) to regulate conditions. This dataset is highly suitable for optimizing energy use based on temperature and humidity, managing device energy consumption, and controlling actuators in response to environmental conditions in a smart home context.

Comparison and Implementation of Optimization Algorithms

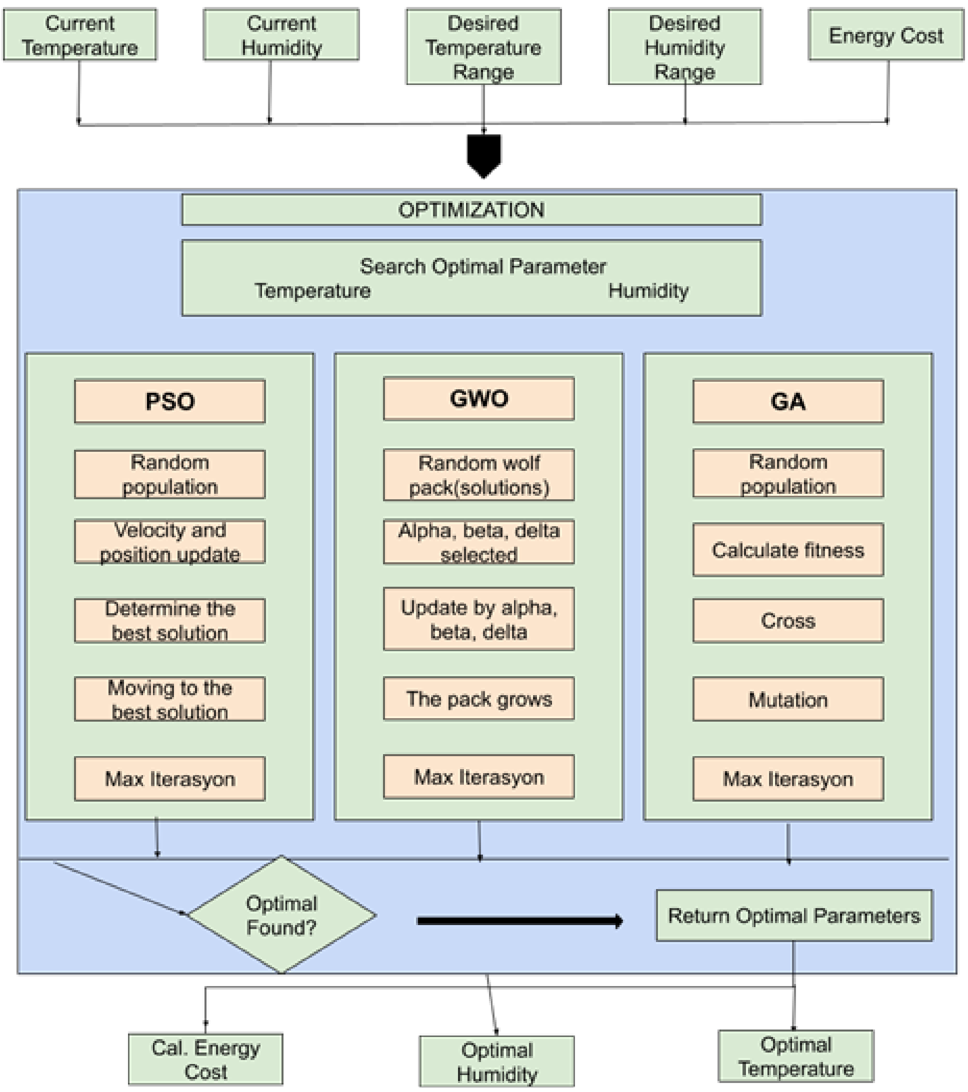
This section explains how our optimization mechanism works, which data set we use, what is our objective function, and which algorithms we implement.

Optimization Approach

Our methodology is demonstrated in Figure 2. We use IoT-based measurements such as temperature, humidity, and the amount of energy cost of each device. We use a multi-objective optimization technique, our methodology uses two important parameters which are temperature and humidity. The objective function is inspired from (Malik et al., 2020). After that, we apply three different well-known optimization techniques to see the differences between each technique and how to apply them to the same problem.

Figure 2

Our Methodology to Apply Optimization Technique by Using IoT-based Measurements.



Objective Function

This section explains the partially modified objective function design we used. Table 1 presents the parameters and their descriptions. In our study, where we have taken the data from the user and thus the user's comfort is seen as optimal values, the aim is to minimize the energy consumption of the function and to achieve this by using temperature and humidity values.

Table 1
The Notations for Parameters

Notation	Description of Parameters
E_{total}	Total energy cost
T_{init}	Initial temperature value
H_{init}	Initial humidity value
T_{max}	Upper temperature target value.
T_{min}	Lower temperature target value
H_{max}	Upper humidity target value
H_{min}	Lower humidity target value.
Tc	Temperature Control
Hc	Humidity control
$Temp\_Error$	Difference between the desired and current temperature.
$Humidity\_Error$	Difference between the desired and current humidity
C_{cost}	Cost of differences
E_T	Energy cost for temperature.
E_H	Energy cost for humidity

$$Temp\_Error = |Tmax - Tc| \tag{1}$$

$$Humidity\_Error = |Hmax - Hc| \tag{2}$$

$$C\text{cost} = (Temp\_Error * mean(E\text{T})) + (Humidity\_Error * mean(E\text{H})) \tag{3}$$

$$E\_balance = [ET + EH] - C\text{cost} \tag{4}$$

$$E\_total = (T\text{init} * mean(E\text{T})) + (H\text{init} * mean(E\text{H})) \tag{5}$$

(1) calculates the absolute difference between the desired maximum temperature and the current temperature. It is used to determine the necessary adjustments for temperature control. (2) calculates the absolute difference between the desired maximum humidity and the current humidity. This difference is used to determine the necessary adjustments for humidity control. (3) calculates the energy costs associated with temperature and humidity errors. Each error is multiplied by the corresponding average energy cost to determine the total cost. (4) computes the energy balance by taking the total current energy costs and subtracting the calculated cost from it. This shows how effective the current energy usage is. (5) calculates the energy costs associated with the initial temperature and humidity values. Each initial value is multiplied by the corresponding

average energy cost to find the total initial energy cost.

The Particle Swarm Optimization Algorithm

Particle Swarm Optimization constitutes an optimization algorithm motivated by the communal conduct of fish schools and avian flocks, predicated on the principle that individuals within a swarm enhance their positions by mutually updating one another (Bahmanyar et al., 2022; Önder, 2011). Algorithm 1 explains the impalmantatiın details of the PSO approach. In the initial phase, a swarm (population) is established, with each individual referred to as a "particle," and a search domain is delineated wherein each particle is allocated a position and velocity. Subsequently, each particle formulates a solution predicated on its fitness value. During each iteration, the particle's optimal value (pbest) and the swarm's optimal value (gbest) are ascertained, and particles revise their positions and velocities before proceeding to the subsequent iteration. This procedure persists until a predetermined cessation criterion, such as a maximum number of iterations (maxIter), is attained, ultimately discerning the optimal solution.

Algorithm 1

The Implementation of Particle Swarm Optimization

- **Start:**
- **Define parameters:**
- *tmax, tmin, hmax, hmin, t0, h0, ect, eht, lb, ub*
- **Define PSO parameters:**
- *maxiter* (maximum number of iterations)
- *pbest* (best particle positions)
- *gbest* (global best position),
- **Define objective function (objective_function):**
- Get *tc, hc* values
- Calculate *t_err: |tmax - tc |*
- Calculate *h_err: |hmax - hc |*
- Calculate *c_cost: (t_err * ecost_t) + (h_err * ecost_h)*
- Calculate energy costs *eb, ec0, et*
- **Run PSO algorithm:**
- Initially determine random particle positions
- Assign *pbest* and *gbest* values
- For iteration in range(*maxiter*):
- For each particle:
- Evaluate objective function and update *pbest*
- If pbest value is better than *gbest*, update *gbest*
- Update particle speed and position according to *pbest* and *gbest*
- Obtain *xopt, fopt* values
- **Print results:**
- Optimized temperature and humidity (*xopt*)
- Minimum energy cost (*fopt*)

The Genetic Algorithm

It is a metaheuristic optimization algorithm inspired by biological evolution processes (Mahmood et al.,2023; Önder, 2011). It is generally used to find the best solutions in large and complex solution spaces. In Algorithm 2, we explain the implementation of the genetic algorithm. The working principle is as follows: an initial population is created from random individuals. A ranking is made among the generated solutions using a fitness function. The most suitable individuals selected from this ranking are designated as parent individuals, and they produce new solutions (offspring) by altering their genetic material. This ensures the diversity of the population and allows for the discovery of new solutions. Small random changes are made. This is used to find potential good solutions that might be overlooked during crossover. After crossover and mutation, new individuals are created, and the process continues with the new population. It uses genetic operators (selection, crossover, and mutation) to produce better solutions from generation to generation.

Algorithm 2

The Implementation of Genetic Algorithm

- **Start:**
- **Define parameters:**
- *tmax, tmin, hmax, hmin, t0, h0, ect, eht*
- *lb, ub, n_generations, cxpb, mutpb, n_runs*
- **Define objective function (objective_function):**
- Get *tc, hc* values
- Calculate *t_err,h_err*
- Calculate *ccost*
- Return *et*
- **Define genetic algorithm structures:**
- Fitness function, individual structure, crossover(*cxpb*), mutation(*mutpb*), selection operations
- **Perform each run (up to n_runs):**
- Create a population (*n=250* individuals) in each run.
- hof (Hall of Fame): Create a structure to store the best individuals.
- stats: Create a structure to calculate statistics.
- **Implement genetic algorithm:**
- algorithms.eaSimple:
- Evolve population (up to *n_generations*).
- Apply crossover, mutation, and selection operations.
- Update the *best individual* and *fitness values* at the end of each generation.
- Get the *best individual* and its *cost* in the hof structure and add it to the list.
- Record the *best individual* and *cost* of each run
- **Find the best result:**
- Find the index of the minimum cost in *all_best_fitnesses* (*min_fitness_index*)
- Set the values of *best_individual* and *best_fitness*
- **Print the results:**
- Print the best individual and cost for each run
- Print the best individual and cost at the end of the *n_runs run*

The Grey Wolf Optimization Algorithm

Grey Wolf Optimization is a meta-heuristic optimization algorithm conceived by deriving inspiration from the hierarchical configuration and predation tactics of grey wolves (Erdoğan, 2023; Makhadmeh et al., 2021). Within this hierarchical configuration, the pack is subdivided into four principal categories: the alpha (α) wolves who govern the pack, the beta (β) wolves occupying the second rank, the delta (δ) wolves positioned third, and ultimately the omega (ω) individuals at the lowest tier. The implementation of GWA is depicted in Algorithm 3. The operational principle of the GWO algorithm is delineated as follows: The initial populace is generated with randomly produced values within specified confines. Subsequently, the fitness value of each individual in the populace is computed, and the locations of the three preeminent individuals are designated as alpha, beta, and delta. Through these steps, both the initial populace and the leading individuals are ascertained, and the algorithm's principal loop commences. Thereafter, the positions of all individuals within the populace are revised utilizing the pertinent objective function equations. Based on the amended positions, the fitness values are recalibrated, and the top three individuals adjust the positions of the alpha, beta, and delta individuals. The principal loop persists until the cessation criterion is satisfied. Upon the conclusion of the loop, the position and fitness value of the alpha individual, who possesses the most favorable fitness value, are recognized as the optimal solution, and the algorithm is finalized (Karakoyun ve Özkış; Mirjalili et al., 2014; Karakoyun, 2021).

Algorithm 3

The Implementation of Grey Wolf Optimization

- **Start:**
- **Define the parameters:**
- *tmax, tmin, hmax, hmin, t0, h0, ect, eht*
- *lb, ub, epoch, pop_size*
- **Define the Grey Wolf Optimization (GWO) problem:**
- *fit_func*: Objective function.
- *lb*: Lower bounds.
- *ub*: Upper bounds.
- *minmax*: "min"
- *verbose*: Detailed output
- **Define objective function (objective_function):**
- Get *tc, hc* values
- Calculate *t_err: |tmax - tc|*
- Calculate *h_errh: |hmax - hc|*
- Calculate *ccost:(t_err * ecost_t) + (h_err* ecost_h)*
- Calculate *ec, eb, et* energy costsa
- **Create a GWO model and find the solution:**
- *model = BaseGWO(problem, epoch=epoch, pop_size=pop_size)*
- *best_position, best_fitness = model.solve()*
- **Find the best result:**
- *best_position*: Best position (*tc, hc*) found as a result of optimization.
- *best_fitness*: Determine the value representing the lowest energy cost.
- **Print results:**
- Print best (*tc, hc*) values found as a result of optimization.
- Print *optimum energy cost*.

Experimental Results and Discussion

We implement three different optimization methods PSO, GWO, and GA are used to reduce energy consumption in a smart home environment. We apply these algorithms to analyze the data set obtained from Kaggle. These methods are tested and analyzed with various objectives focused on temperature and humidity control.

The PSO method is tested to reduce energy consumption by optimizing based on the average temperature and humidity data, providing stable and reliable results in minimizing energy usage. The results show that PSO delivers consistent outcomes with average calculations, focusing on reducing overall energy costs. The GWO method aims to reduce energy consumption by minimizing deviations from target temperature and humidity, effectively achieving comfort conditions. This approach helps us to maintain a low humidity level, though it resulted in higher energy expenditures compared to PSO. The Genetic Algorithm is analyzed through 50 different trials, each using a population of 250 individuals. Crossover and mutation operations are applied over 150 generations, selecting the individuals with the lowest energy consumption. GA effectively controls temperature and humidity with moderate energy usage, and balancing crossover and mutation rates played an important role in reducing energy consumption. According to the light of the results, each optimization method offers unique advantages in reducing energy consumption.

Figure 3

Temperature Humidity and Energy Costs are Obtained as A Result of Applying Our 3 Algorithms to 3 Different Rooms in The Dataset

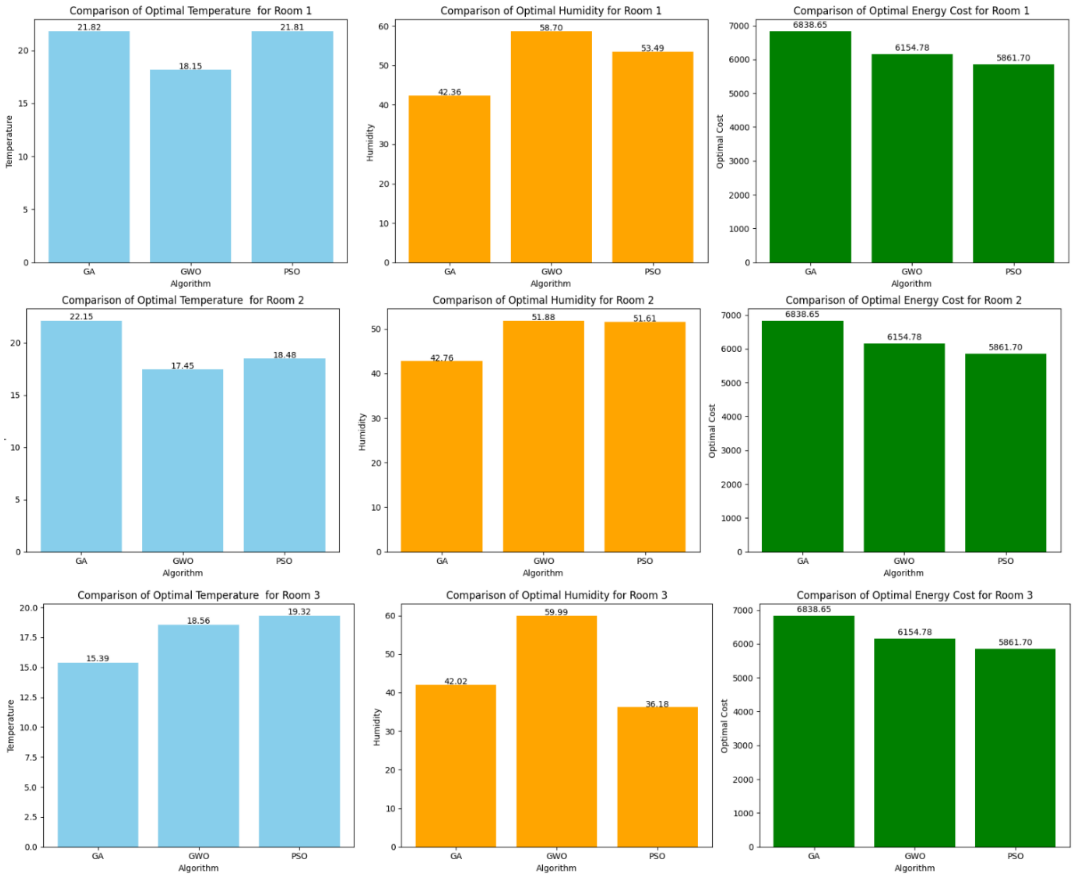


Figure 3 demonstrates that applying our three algorithms to three different rooms in the dataset yields values for temperature, humidity, and energy costs. When we examine the results of the three algorithms, we see that each one produces different outcomes due to its unique optimization strategy. GA provides a balanced solution by keeping humidity within an acceptable range and maintaining costs at a reasonable level. GA's ability to explore a wide solution space allowed it to balance both humidity and cost-effectively and also performed well in temperature regulation. GWO has shown effectiveness in maintaining a low humidity level; however, this has caused an increase in energy costs. GWO's focus on localized solutions may have contributed to the rise in energy costs while optimizing humidity. In contrast, PSO focuses on reducing overall energy costs and achieving the lowest cost. However, this focus led to a humidity level slightly above the target. PSO's rapid convergence has been effective in lowering costs, but it came with a compromise in humidity control. Based on our approach and experimental results, GWO is useful for humidity management, PSO for cost reduction, and GA for a balanced solution. Since the primary goal of our study is to optimize energy costs, PSO offers the best results for this purpose while also maintaining an acceptable temperature and humidity level.

Future Directions

There are some key research questions to enhance our methodology: How does the efficacy of PSO, GWO, and GA algorithms for energy optimization in intelligent residences fluctuate under varying climatic and environmental circumstances? In what manner can the generalizability of these algorithms be augmented through more extensive and diversified datasets? When contrasting the efficacy of PSO, GWO, and GA algorithms with alternative evolutionary algorithms such as Differential Evolution, Simplex Optimization, or Ant Colony Optimization Algorithms for energy optimization, what distinctions become apparent? What are the ramifications of alternative evolutionary methodologies on solution quality and optimization velocity? What influence do multi-objective optimization functions, which reconcile user comfort with energy efficiency, exert on energy optimization in intelligent residences? How can these multi-objective optimization functions be modified to accommodate diverse user behaviors and comfort thresholds? How can PSO, GWO, and GA algorithms be engineered to adapt in real-time to fluctuating environmental conditions in intelligent residences? How can adaptive optimization algorithms reconcile real-time energy cost mitigation and user comfort? In which situations can deep learning-based predictive models furnish more efficacious solutions for energy optimization in intelligent residences? How does the efficacy of AI-enhanced predictive models in forecasting energy demand affect the performance of optimization algorithms?

Conclusion

This chapter aims to increase energy efficiency in IoT-based smart home applications, fills an important gap in the literature, and demonstrates the applicability of energy optimization-based solutions. The use of algorithms such as Gray Wolf Optimization, Genetic Algorithm, and Particle Swarm Optimization on IoT-based data sets provides significant contributions in terms of minimizing energy consumption and preserving user comfort. The findings obtained reveal that energy-efficient management of IoT devices offers both environmental and economic benefits. In future studies, more in-depth analyses can be made in the field of energy consumption management in smart homes by using different data sets and optimization algorithms, and energy management solutions can be addressed from a broader perspective.

Acknowledgments

This study is supported by the TUBITAK 2209-A University Students Research Projects Support Program with project number 1919B012338526.

References

Al-Ali, A. R., Zualkernan, I. A., Rashid, M., Gupta, R., & AliKarar, M. (2017). A smart home energy management system using IoT and big data analytics approach. *IEEE Transactions on Consumer Electronics*, *63*(4), 426-434.

Appliances energy prediction. (2017, September 16). Kaggle. https://www.kaggle.com/datasets/loveall/appliances-energy-prediction/data

Bahmanyar, D., Razmjooy, N., & Mirjalili, S. (2022). Multi-objective scheduling of IoT-enabled smart homes for energy management based on Arithmetic Optimization Algorithm: A Node-RED and NodeMCU module-based technique. *Knowledge-Based Systems*, *247*, 108762.

Devi, M., Muralidharan, S., Elakiya, R., & Monica, M. (2023). Design and Implementation of a Smart Home Energy Management System Using IoT and Machine Learning. In *E3S Web of Conferences* (Vol. 387, p. 04005). EDP Sciences.

Erdoğan, F. (2023). *Application Of Binary Grey Wolf Optimization Algorithm To Binary Optimization Problems* (Master's thesis, Necmettin Erbakan University (Turkey)).

Eroglu, A. & Unlu Eroglu, H. (2023). Topological Data Analysis for Intelligent Systems and Applications. In S. Kocer. & O. Dundar (Eds.), Artificial Intelligence Applications in Intelligent Systems (pp. 27–60). ISRES Publishing.

Eroglu, A. (2022). 5G and Beyond Networks for Digital Transformation: Opportunities and Productivity, M.H. Calp. & R. Butuner (Eds .), Current Studies in Digital Transformation and Productivity (pp. 98–122). ISRES Publishing.

Francis, A., Madhusudhanan, S., Jose, A. C., & Malekian, R. (2023). An Intelligent IoT-based Home Automation for Optimization of Electricity Use. *Przeglad Elektrotechniczny*, *99*(9), 123-127.

Geraldo Filho, P. R., Villas, L. A., Gonçalves, V. P., Pessin, G., Loureiro, A. A., & Ueyama, J. (2019). Energy-efficient smart home systems: Infrastructure and decision-making process. *Internet of Things*, *5*, 153-167.

Goudarzi, S., Anisi, M. H., Soleymani, S. A., Ayob, M., & Zeadally, S. (2021). An IoT-based prediction technique for efficient energy consumption in buildings. *IEEE Transactions on Green Communications and Networking*, *5*(4), 2076-2088.

Karakoyun, M. (2021). *Developing a hybrid method based on Shuffled Frog Leaping and Gray Wolf Optimization algorithms to solve multi-objective optimization problems* (Doctoral dissertation, Dissertation Doctoral Thesis, Konya Technical University).

Karakoyun, M., & Özkış, A. (2021). Development of binary moth-flame optimization algorithms using transfer functions and their performance comparison. *Necmettin Erbakan University Journal of Science and Engineering*, *3*(2), 1-10.

Kaya, M. C., Eroglu, A., Karamanlioglu, A., Onur, E., Tekinerdogan, B., & Dogru, A. H. (2019). Runtime adaptability of ambient intelligence systems based on component-oriented approach. *Guide to Ambient Intelligence in the IoT Environment: Principles, Technologies and Applications*, 69-92.

Leitao, J., Gil, P., Ribeiro, B., & Cardoso, A. (2020). A survey on home energy management. *IEEE Access*, *8*, 5699-5722.

Machorro-Cano, I., Alor-Hernández, G., Paredes-Valverde, M. A., Rodríguez-Mazahua, L., Sánchez-Cervantes, J. L., & Olmedo-Aguirre, J. O. (2020). HEMS-IoT: A big data and machine learning-based smart home system for energy saving. *Energies*, *13*(5), 1097.

Mahmood, Z., Cheng, B., Butt, N. A., Rehman, G. U., Zubair, M., Badshah, A., & Aslam, M. (2023). Efficient scheduling of home energy management controller (hemc) using heuristic optimization techniques. *Sustainability*, *15*(2), 1378.

Makhadmeh, S. N., Khader, A. T., Al-Betar, M. A., Naim, S., Abasi, A. K., & Alyasseri, Z. A. A. (2021). A novel hybrid grey wolf optimizer with min-conflict algorithm for power scheduling problem in a smart home. *Swarm and Evolutionary Computation*, *60*, 100793.

Malik, S., Lee, K., & Kim, D. (2020). Optimal control based on scheduling for comfortable smart home environment. *IEEE Access*, *8*, 218245-218256.

Mirjalili, S., Mirjalili, S. M., & Lewis, A. (2014). Grey wolf optimizer. *Advances in engineering software*, *69*, 46-61.

Önder, E. (2011). Araç rotalama problemlerinin parçacık sürü ve genetik algoritma ile optimizasyonu. *PhD, İstanbul Üniversitesi, İstanbul, Türkiye*.

Priyadarshini, I., Sahu, S., Kumar, R., & Taniar, D. (2022). A machine-learning ensemble model for predicting energy consumption in smart homes. *Internet of Things*, *20*, 100636.

Shah, A. S., Nasir, H., Fayaz, M., Lajis, A., & Shah, A. (2019). A review on energy consumption optimization techniques in IoT-based smart building environments. *Information*, *10*(3), 108.

Wang, X., Mao, X., & Khodaei, H. (2021). A multi-objective home energy management system based on the Internet of Things and optimization algorithms. *Journal of Building Engineering*, *33*, 101603.

About the Authors

Gamze AKÇAY is an undergraduate student at the Department of Computer Engineering at Necmettin Erbakan University.
E-mail: 20010011039@ogr.erbakan.edu.tr, **ORCID:** 0009-0003-5314-1544

Behice Gülsüm ÇELİK, received her B.Sc. degree from the Department of Electrical and Electronics Engineering at KTO Karatay University, Konya, Turkey, in 2021 and completed a double major in Computer Engineering the same year. She is currently a research assistant and pursuing her M.Sc. degree in Computer Engineering at Necmettin Erbakan University, Konya, Turkey. Her research interests are in deep learning and optimization.
E-mail: behicegulsum.celik@erbakan.edu.tr, **ORCID:** 0000-0003-4008-1258

Alperen EROĞLU, received his B.Sc. degree from the Department of Electronics and Computer Education as the valedictorian of the department from Fırat University, Elazığ, Turkey in 2009, the M.Sc. and Ph.D. degrees in computer engineering from METU, Ankara, Turkey in 2015 and 2020, respectively. During his M.Sc. and Ph.D. studies, he researched and studied robotics, artificial intelligence, software engineering, system architecture and modeling, embedded systems, wireless networks, and computer networks. He accomplished his M.Sc. and Ph.D. studies in the field of wireless networks. He has a post-doctorate degree in Computer Engineering at METU. He is currently an assistant professor at the Department of Computer Engineering at Necmettin Erbakan University. His research interests are in 5G mobile networks, the Internet of Things, and wireless and computer networks. He is a member of the Wireless Systems, Networks, and Cybersecurity Laboratory (WINSLab). He is a member of IEEE.
E-mail: aeroglu@erbakan.edu.tr, **ORCID:** 0000-0002-1780-7025

Similarity Index
The similarity index obtained from the plagiarism software for this book chapter is 7%.

Controller Design Optimization of a Multi-DOF System Using Response Optimizer Toolbox

Saliha KÖPRÜCÜ
Necmettin Erbakan University

Engin Hasan ÇOPUR
Necmettin Erbakan University

Hasan Huseyin BILGIC
Necmettin Erbakan University

To Cite This Chapter

Köprücü, S., Çopur, E. H., & Bılgıc, H. H. (2024). Controller Design Optimization of a Multi-DOF System Using Response Optimizer Toolbox. In S. Koçer & Ö. Dündar (Eds.), *Intelligent Systems and Optimization in Engineering* (pp. 67-79). ISRES Publishing.

Introduction

Control applications for multi-degree-of-freedom (multi-DOF) systems play a crucial role today, especially in the stabilization and guidance of complex dynamic systems such as aircraft. In this context, the modeling, simulation, and control of multi-DOF systems are frequently addressed in both academic research and industry. Studies on these systems provide engineering students and researchers with in-depth knowledge of dynamic system control, while also enabling engineers to apply this knowledge in real-world applications. This section discusses the use of MATLAB/Simulink Response Optimizer, an effective tool for controlling multi-DOF systems, and explains the design of a controller for a multi-DOF system.

A review of the literature reveals that optimal controllers are often used in the control of multi-DOF systems when specific performance criteria, such as energy consumption, error rate, response time, or stability, need to be optimized (Hassaan 2021; Ekinci et al. 2021; İzci et al. 2021). However, constraining the control responses of these systems to a desired range often requires precise tuning. To achieve this, artificial intelligence algorithms are usually preferred for fine-tuning (Bilgic et al. 2016; Bilgic et al. 2021). At this stage, designing a multi-objective function, which requires expertise, becomes necessary (Rodríguez-Molina et al. 2020). For these reasons, MATLAB/Simulink's Response Optimizer stands out as an effective tool for adjusting the controller parameters of multi-DOF systems under specific constraints (Azeez et al. 2023).

Mahmoud and AlRamadhan (2021) examined two different SMC controller structures with the aim of achieving smooth transient and robust steady-state responses to track the reference rotor position when the hybrid stepper motor is subjected to load disturbances. They optimized the controller parameters using Simulink Response Optimizer application. Lachekhab et al. (2022), in their study on creating an accurate mathematical model of a quadcopter, controlled the position and attitude of the quadcopter in three-dimensional space. For this purpose, roll, pitch, and yaw angles, as well as altitude and position, were all controlled using SMC. They also used the Simulink Response Optimizer tool to optimize several control parameters in the SMC controller design. Wu et al. (2023) proposed a modeling and tracking control strategy based on fractional calculus to achieve high-accuracy control of dielectric elastomer actuators (DEAs). Since tracking control accuracy depends on the controller parameters, they applied a two-stage optimization process to optimize the FFCC parameters. In the first stage, the controller parameters were optimized based on a reduced-order model of the DEA using a gradient descent algorithm and Simulink's Response Optimizer application in the simulation environment. Following this, the control experiment was conducted, with root-mean-square errors in the experiment all recorded below 0.7%.

Gopi et al. (2023) introduced a new optimization technique, Simulink Design Optimization (SDO), for calculating optimal PID coefficients in automatic voltage regulators (AVRs). In their study, the performance of the SDO-PID controller is compared to PID controllers using water cycle algorithms, genetic algorithms, and local unimodal sampling algorithms. Results demonstrate that the SDO-PID controller, utilizing the response optimizer, not only outperforms traditional methods but also provides a broader stability range. This makes it a promising solution for enhancing the efficiency and reliability of automatic voltage regulation systems, contributing to more effective voltage management in various applications. In a study aimed at developing more efficient, robust, and user-friendly teleoperation systems, Nour et al. (2023) optimized the controller and parameter set in two bilateral teleoperation control schemes using the Simulink Response Optimizer tool. They performed a comparative analysis of each architecture's performance in terms of transparency and stability, considering both constant and variable time delays. The response optimizer tool was used to obtain the best set of tuning parameters that optimized the specified objective function and, in this case, minimized the tracking position error between master and slave devices.

Alhattab et al. (2023) used a controller called D-STATCOM to reduce voltage fluctuations in electrical power systems. In their study, they developed an AI-based control system with fuzzy logic and ANFIS to achieve faster, more stable control compared to the conventional PI controller. The gain parameters of the PI controller were optimized using Simulink Response Optimizer tool, enabling the system to adapt more quickly to varying load conditions. This approach effectively restored voltage to its nominal value during voltage sag and swell events, with simulation results demonstrating the success of the optimization-supported control system in enhancing power quality.

Example Study: As an example application, a Quanser 3DOF Hover system with three degrees of freedom was tested (Figure 1).

Figure 1
Quanser 3DOF Hover System (Quanser, 2024)

The Quanser 3DOF Hover system is a platform capable of movement in three axes (roll, pitch, and yaw) (Quanser, 2024). It is an ideal experimental system designed to test various control theories. The system's three-degree-of-freedom movement capability offers a significant advantage for simulating guidance and stabilization problems, especially for unmanned aerial vehicles (UAVs). The 3DOF Hover system consists of a motorized propeller mechanism and a platform that allows free rotation along three axes. The primary objective of the system is to develop a suitable control algorithm to maintain specific angular positions stably. Controlling this type of system is challenging from an engineering perspective due to its non-linear dynamics, making the 3DOF Hover system valuable for both educational and research purposes (Quanser, 2024).

Anna Prach et al. evaluated the control performance of nonlinear systems using the forward propagating Riccati equation (FPRE) and linear-quadratic regulator (LQR) methods in their paper. FPRE provides more flexible and effective control over a wider operating range by updating feedback gains in real time. In experiments conducted on the Quanser 3DOF Hover test system, FPRE demonstrated superior performance compared to LQR, particularly in handling high-amplitude commands. The study shows that FPRE offers a more robust control approach for nonlinear systems. Fethi Ouerdane et al. (2024) aimed to test the visual servoing systems of low-cost UAVs using Quanser's 3DOF hover quadcopter in an indoor environment without GPS. The system reads commands specified by QR codes and processes positional information using LabVIEW and hardware-in-the-loop (HIL) simulation. It employs PID controllers for horizontal and vertical position tracking and uses the LQR algorithm for roll, pitch, and yaw control. In the experiments, the Quanser quadcopter successfully performed positioning and tracking via QR codes, maintaining its tracking capability despite some detection limitations. The results indicate that this method can be adapted for advanced applications such as asset tracking and indoor inspections with 6-DOF UAVs. Normann (2023) investigated event-triggered (ETC) and self-triggered (STC) control strategies on the Quanser 3DOF Hover system to conserve energy and processing power in resource-constrained control systems. ETC continuously monitors the state of the Quanser hover system and updates only when necessary, while STC conserves energy by establishing a predictive update schedule. Simulations and experiments showed that the ETC strategy successfully maintained stability in the real system, while STC was more sensitive to modeling errors and external disturbances. It was found that ETC improved energy and bandwidth efficiency while keeping the Quanser hover system stable, whereas STC performed well in simulations but did not meet expectations in real-world applications.

Mathematical Modelling of 3Dof Hower System

Free-Body Diagram of the System

To perform a valid simulation, the created system must be accurately represented in the computer environment. Achieving this requires that the mathematical model of the system closely reflects reality. A mathematical model, in its simplest form, is a representation of a real system, whether planned or already existing, using only mathematical expressions. In the process of mathematical modeling, fundamental physical laws are employed to translate the real-world problem into a mathematical framework.

When dealing with a nonlinear system, as in this study, achieving a highly accurate mathematical model becomes challenging. Consequently, certain assumptions are made, and the system is linearized as much as possible to obtain an accurate model. Below are some of the assumptions made during the modelling process:

- The 3-DOF hover experiment set is rigid and symmetrical.
- The center of gravity of the 3-DOF hover experiment set is at the intersection of the

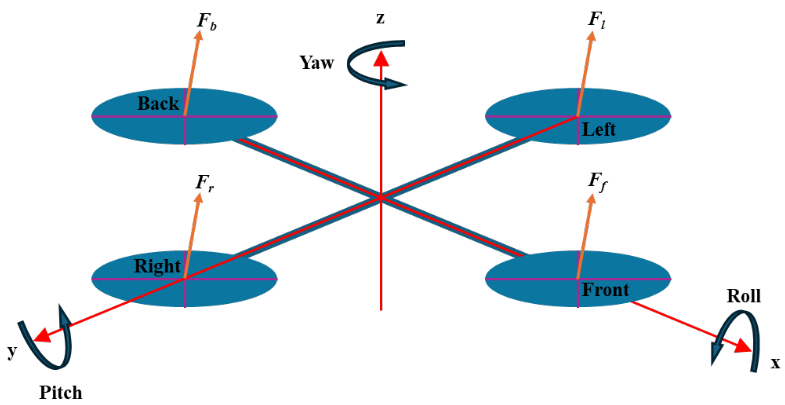
X, Y, and Z axes.

- The motors of the experiment set are rigidly fixed to the body.
- The experiment set is parallel to the ground when the pitch and roll angles are zero $(\theta_p = 0, \theta_r = 0)$.
- When the body of the experiment set rotates counterclockwise, the yaw angle increases.
- When the experiment set rotates counterclockwise around the Y-axis, the pitch angle increases.
- When the experiment set rotates clockwise around the X-axis, the roll angle increases.

The free-body diagram of the isometric view of the 3-DOF hover experiment set used in this study's simulation files is shown in Figure 2, along with the positive rotation directions of the X-Y-Z axes, defined respectively as the Roll-Pitch-Yaw axes.

Figure 2

Free-Body Diagram of the Isometric View of the 3DOF Hover Experimental Set



The 3-DOF hover experiment set has four DC motors positioned at the front, back, right, and left, as illustrated in Figure 2. When a positive voltage is applied to these motors, they provide the thrust required to activate the system. The thrust generated by these motors is denoted as F_f, F_b, F_r and F_l respectively.

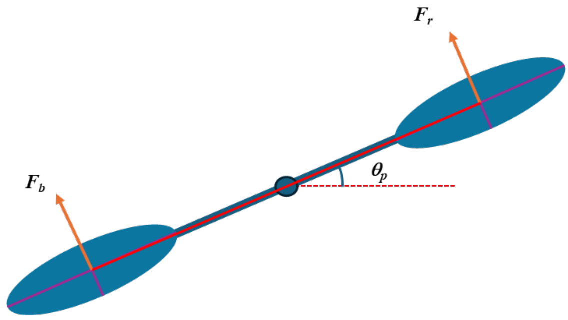
Pitch, Roll and Yaw-Axis Model of the System

The motors placed at the front and back are primarily responsible for controlling the pitch angle. When the thrust generated by the front motor exceeds that of the back motor, the pitch angle increases $(F_f > F_b)$.

Figure 3 provides a free-body diagram of this scenario:

Figure 3

Free-Body Diagram of the 3DOF Hover Experimental Set Viewed from the Right Side



The general equation for the dynamics of each axis is described as follows:

$$J\ddot{\theta} = \Delta FL \tag{1}$$

where J is the moment of inertia with respect to the axis, $\ddot{\theta}$ is the angular acceleration, ΔF is the differential thrust force, and L is the distance of the motor from the centre.

When Equation (1) is applied to the system's free-body diagram shown in Figure 3, the following equation is obtained:

$$J_p\ddot{\theta}_p = K_f(V_f - V_b) \tag{2}$$

where K_f is the thrust force constant, V_f and V_b are the voltages supplied to the front and back motors, respectively, θ_p is the pitch angle, and J_p is the moment of inertia about the Y-axis (pitch-axis).

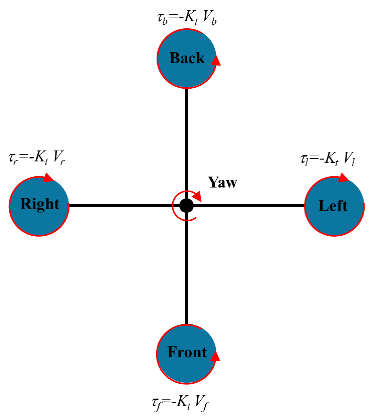
Similarly, for the X-axis (roll-axis), we have:

$$J_r\ddot{\theta}_r = K_f(V_r - V_l) \tag{3}$$

where V_r and V_l are the voltages supplied to the right and left motors, θ_r is the roll angle, and the J_r is the moment of inertia about the roll-axis. These equations describe the movements in the roll and pitch axes, while the yaw axis differs from these two.

Figure 4

Free-Body Diagram of the Top View of the 3DOF Hover Experimental Set



As shown in Figure 4, opposite motors rotate in the same direction, while consecutive motors rotate in opposite directions. Since the motors are equidistant from the centre, this design symmetry maintains force and moment balance. To initiate motion along the Z-axis (yaw axis), this balance must be disrupted. The equation for motion in the yaw axis is given as follows:

$$J_y\ddot{\theta}_y = \Delta\tau = \tau_l + \tau_r - \tau_f - \tau_b \tag{4}$$

where τ_l and τ_r are the torques generated by the clockwise-rotating propellers on the left and right, respectively, and τ_f and τ_b are the torques generated by the counterclockwise-rotating propellers at the front and back. Letting K_t be the thrust-torque constant and V_m the motor voltage, the total torque generated by all propellers can be represented as:

$$\tau = K_t V_m \tag{5}$$

Thus, the yaw-axis equation of motion can be written as:

$$J_y\ddot{\theta}_y = K_t(V_r + V_l) - K_t(V_f + V_b) \tag{6}$$

State-Space Model of the System

To apply the Linear Quadratic Regulator (LQR) method to the system, whose equations of motion are given in the previous section, the system's state-space model must be derived. As we know, the general representation of a state-space model can be written as:

$$\begin{aligned} \dot{x} &= Ax + Bu \\ y &= Cx + Du \end{aligned} \tag{7}$$

where x represents the state variable. The state vector of the 3DOF hover experiment set is as follows:

$$x^T = \begin{bmatrix} \theta_y & \theta_p & \theta_r & \dot{\theta}_y & \dot{\theta}_p & \dot{\theta}_r \end{bmatrix} \tag{8}$$

The output vector of the 3DOF hover experiment set is as follows:

$$y^T = \begin{bmatrix} \theta_y & \theta_p & \theta_r \end{bmatrix} \tag{9}$$

The control vector of the 3DOF hover experiment set is as follows:

$$u^T = \begin{bmatrix} V_f & V_b & V_r & V_l \end{bmatrix} \tag{10}$$

Using the equations of motion given above, the state-space matrices are defined as follows:

$$A=\begin{bmatrix}0&0&0&1&0&0\\0&0&0&0&1&0\\0&0&0&0&0&1\\0&0&0&0&0&0\\0&0&0&0&0&0\\0&0&0&0&0&0\end{bmatrix},$$

$$B=\begin{bmatrix}0&0&0&0\\0&0&0&0\\0&0&0&0\\-K_t/J_y&-K_t/J_y&K_t/J_y&K_t/J_y\\LK_f/J_p&-LK_f/J_p&0&0\\0&0&LK_f/J_r&-LK_f/J_r\end{bmatrix},$$

$$C=\begin{bmatrix}1&0&0&0&0&0\\0&1&0&0&0&0\\0&0&1&0&0&0\end{bmatrix},$$

$$D=\begin{bmatrix}0&0&0&0\\0&0&0&0\\0&0&0&0\end{bmatrix}.$$

$$\begin{bmatrix}\dot{\theta}_y\\\dot{\theta}_p\\\dot{\theta}_r\\\ddot{\theta}_y\\\ddot{\theta}_p\\\ddot{\theta}_r\end{bmatrix}=\begin{bmatrix}0&0&0&1&0&0\\0&0&0&0&1&0\\0&0&0&0&0&1\\0&0&0&0&0&0\\0&0&0&0&0&0\\0&0&0&0&0&0\end{bmatrix}\begin{bmatrix}\theta_y\\\theta_p\\\theta_r\\\dot{\theta}_y\\\dot{\theta}_p\\\dot{\theta}_r\end{bmatrix}+\begin{bmatrix}0&0&0&0\\0&0&0&0\\0&0&0&0\\-K_t/J_y&-K_t/J_y&K_t/J_y&K_t/J_y\\LK_f/J_p&-LK_f/J_p&0&0\\0&0&LK_f/J_r&-LK_f/J_r\end{bmatrix}\begin{bmatrix}V_f\\V_b\\V_r\\V_l\end{bmatrix}\tag{11}$$

$$\begin{bmatrix}\theta\\\theta\\\theta\end{bmatrix}=\begin{bmatrix}1&0&0&0&0&0\\0&1&0&0&0&0\\0&0&1&0&0&0\end{bmatrix}\begin{bmatrix}\theta\\\theta\\\theta\\\dot{\theta}\\\dot{\theta}\\\dot{\theta}\end{bmatrix}+\begin{bmatrix}0&0&0&0\\0&0&0&0\\0&0&0&0\end{bmatrix}\begin{bmatrix}\ \\\ \\\ \end{bmatrix}\tag{12}$$

Optimisation Process by Using Response Optimizer Toolbox

MATLAB/Simulink Response Optimizer Toolbox

The MATLAB System Optimizer Toolbox allows engineers and researchers to improve performance by optimizing system parameters. This toolbox aims to automatically adjust the control parameters of complex systems to achieve the desired performance. In control systems, manual tuning of dynamic parameters is often time-consuming and prone to error. By providing precise parameter optimization, this toolbox accelerates the tuning process.

The toolbox allows users to define cost functions and constraints to set optimization objectives. For example, cost functions can be created based on criteria such as minimizing the system's settling time or ensuring that an output signal converges to a specific value. MATLAB uses different optimization algorithms to determine the optimal parameters based on these criteria.

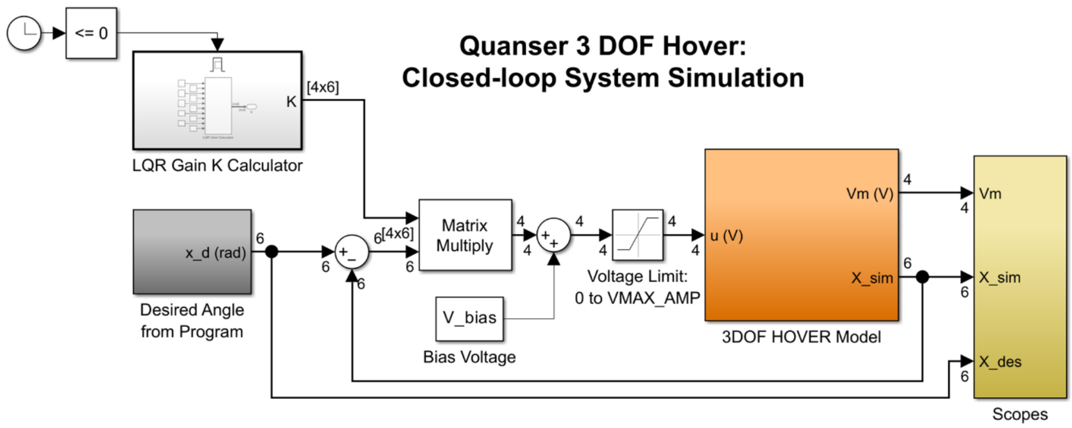
The compatibility of the toolbox with Simulink enables users to directly perform optimization on Simulink models. This functionality makes it possible to automatically optimize control system parameters and view the results graphically. For instance, in designing a PID controller, the PID gains can be optimized using the System Optimizer Toolbox to enhance system stability. With comprehensive analysis and simulation capabilities, the MATLAB Response Optimizer Toolbox is a powerful tool that increases efficiency in fields such as control engineering and system analysis.

Case Study: Tuning of LQR State Weighting Matrix Gains

As an example application, the control block diagram of the Quanser 3DOF Hover system is presented in Figure 5.

Figure 5

Control Block Diagram of Quanser 3DOF Hover System (Quanser, 2024)



The controller's gains based on the state-space model are determined by adjusting the weighting matrices using the Response Optimizer Toolbox. The ten parameters of the Q and R matrices, shown in Equation 13 as $q1$, $q2$, $q3$, $q4$, $q5$, $q6$, $r1$, $r2$, $r3$, and $r4$, are adjusted to meet the response requirements.

$$Q = \begin{bmatrix} q_1 & 0 & 0 & 0 & 0 & 0 \\ 0 & q_2 & 0 & 0 & 0 & 0 \\ 0 & 0 & q_3 & 0 & 0 & 0 \\ 0 & 0 & 0 & q_4 & 0 & 0 \\ 0 & 0 & 0 & 0 & q_5 & 0 \\ 0 & 0 & 0 & 0 & 0 & q_6 \end{bmatrix},$$

$$R = \begin{bmatrix} r_1 & 0 & 0 & 0 \\ 0 & r_2 & 0 & 0 \\ 0 & 0 & r_3 & 0 \\ 0 & 0 & 0 & r_4 \end{bmatrix} \tag{13}$$

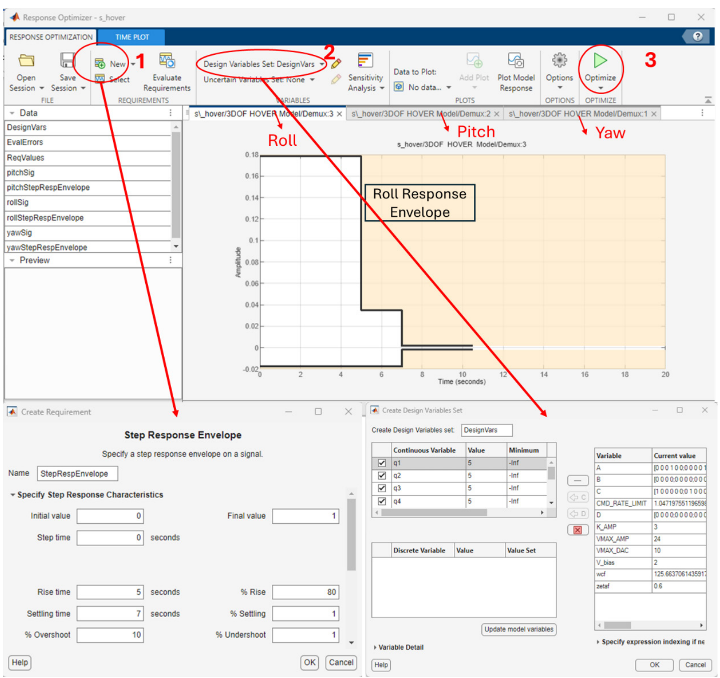
When tuning the controller parameters with the Response Optimizer Toolbox, the following steps are applied in sequence:

- The variables to be adjusted are defined symbolically in Simulink.
- A response envelope is created for the variables to be controlled in the output.
- A set of design variables is specified, with lower and upper limits set if applicable.
- The optimization process is initiated.

The optimization then proceeds using the gradient descent approach until the conditions fall within the response envelope. If conditions are not met, the optimization process automatically stops upon reaching the predetermined number of iterations.

Figure 6

MATLAB/Simulink Response Optimizer Toolbox



Results

In this study, fine-tuning of the weighting matrices for the LQR controller was conducted for a regulator problem. The objective of the regulator problem is to control the transition of a four-rotor system from specified pitch, roll, and yaw angles to a hover position. The system's initial conditions are set as , $\theta_p(0) = 10°$ and $\theta_y(0) = 20°$. To achieve precise tuning of the LQR controller's weighting matrices, constraints on the time response envelope for each rotational axis were specified as follows: a rise time of 5 seconds, a settling time of 7 seconds, a maximum overshoot of 10%, and a steady-state error within ±1%. Based on these constraints, the optimized Q and R matrices were determined as follows:

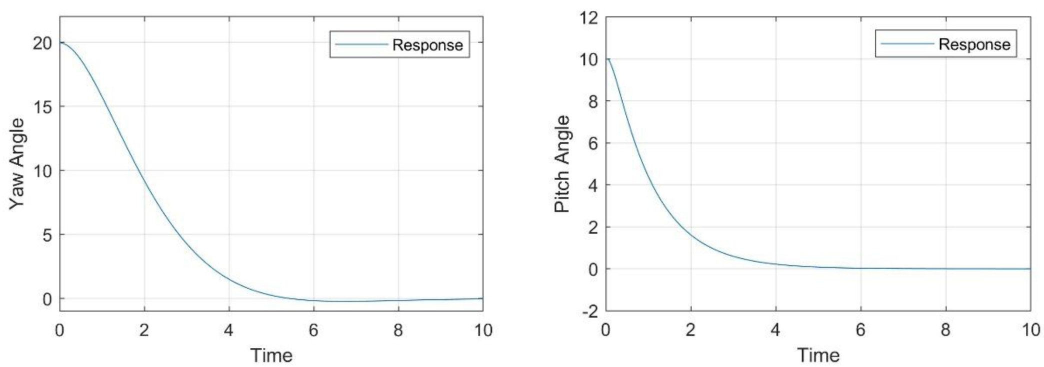
$$Q = \begin{bmatrix} 23.433 & 0 & 0 & 0 & 0 & 0 \\ 0 & 10.191 & 0 & 0 & 0 & 0 \\ 0 & 0 & 36.484 & 0 & 0 & 0 \\ 0 & 0 & 0 & 21.825 & 0 & 0 \\ 0 & 0 & 0 & 0 & 4.7375 & 0 \\ 0 & 0 & 0 & 0 & 0 & 3.246 \end{bmatrix} \tag{14}$$

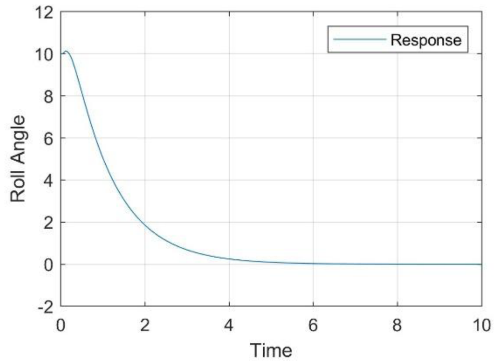
$$R = \begin{bmatrix} 3.225 & 0 & 0 & 0 \\ 0 & 3.811 & 0 & 0 \\ 0 & 0 & 4.3816 & 0 \\ 0 & 0 & 0 & 3.488 \end{bmatrix} \tag{15}$$

The simulation results for the system's pitch, roll, and yaw angle responses over time, obtained using these optimized Q and R matrices, are shown in Figure 7. The results indicate that the system successfully transitioned to the hover position while meeting the defined time response constraints.

Figure 7

Controller Responses Under Initial Angular Positions

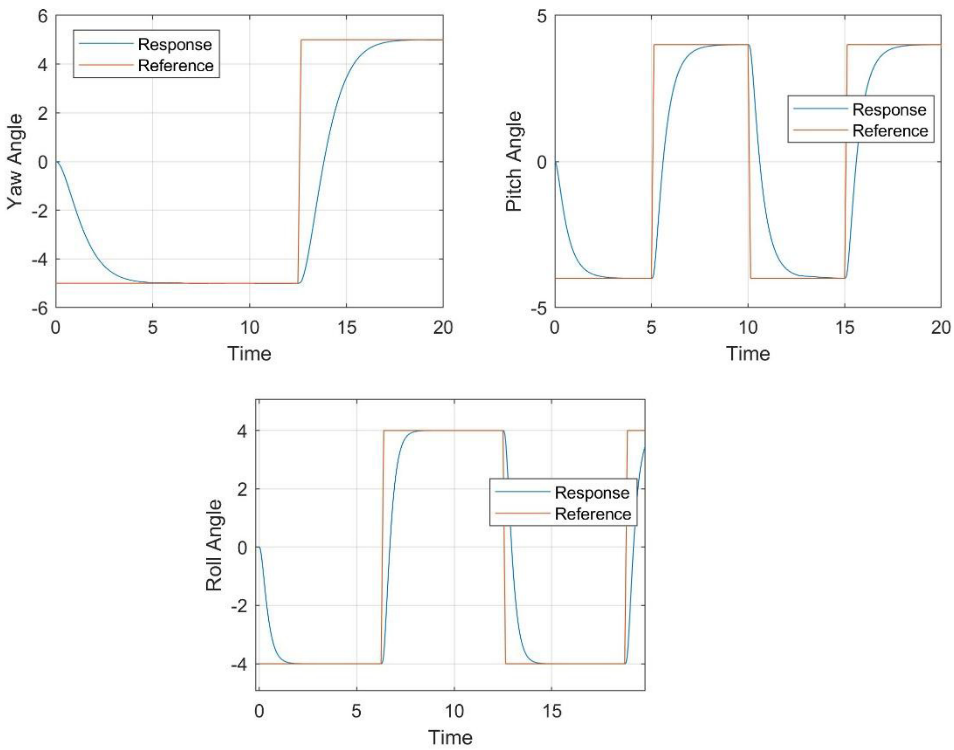




Additionally, another simulation was conducted to evaluate the performance of the optimized control parameters in reference tracking. In this simulation, the objective was for the system to follow predetermined square trajectories for each rotational angle. The square wave inputs were set as follows: an amplitude of 4 degrees and a frequency of 0.08 Hz for the roll angle, an amplitude of 4 degrees and a frequency of 0.1 Hz for the pitch angle, and an amplitude of 5 degrees and a frequency of 0.04 Hz for the yaw angle. The results are presented in Figure 8. The simulation results indicate that the optimized controller successfully tracked each reference input in all three axes.

Figure 8

Controller Responses for Square Trajectories



Conclusion

In this study, the effectiveness of the MATLAB/Response Optimizer Toolbox in solving control problems for multi-degree-of-freedom systems was examined. Since dynamic systems are often structured with multiple degrees of freedom, designing controllers for these systems poses various challenges. Precise tuning of control parameters is especially crucial for successfully meeting control criteria. The MATLAB/ Response Optimizer Toolbox provides an effective method for addressing these issues in multi-degree-of-freedom systems.

This study offers foundational information on using the MATLAB/Response Optimizer Toolbox and demonstrates its application to a control problem, providing users with practical insights. Furthermore, based on the results obtained from the control application in this study, it has been shown that control parameters can be successfully optimized according to control design criteria.

References

Alhattab, A. S., Alsammak, A. N. B., & Mohammed, H. A. (2023). An intelligent mitigation of disturbances in electrical power system using distribution static synchronous compensator. *Indonesian Journal of Electrical Engineering and Computer Science*, *30*(2), 633-642.

Azeez, M. I., Abdelhaleem, A. M. M., Elnaggar, S., Moustafa, K. A., & Atia, K. R. (2023). Optimized sliding mode controller for trajectory tracking of flexible joints three-link manipulator with noise in input and output. *Scientific reports*, *13*(1), 12518.

Bilgic, H. H., Sen, M. A., & Kalyoncu, M. (2016). Tuning of LQR controller for an experimental inverted pendulum system based on The Bees Algorithm. *Journal of Vibroengineering*, *18*(6), 3684-3694.

Bilgic, H. H., Sen, M. A., Yapici, A., Yavuz, H., & Kalyoncu, M. (2021). Meta-heuristic tuning of the LQR weighting matrices using various objective functions on an experimental flexible arm under the effects of disturbance. *Arabian Journal for Science and Engineering*, *46*(8), 7323-7336.

Ekinci, S., Izci, D., & Hekimoğlu, B. (2021). Optimal FOPID speed control of DC motor via opposition-based hybrid manta ray foraging optimization and simulated annealing algorithm. *Arabian Journal for Science and Engineering*, *46*(2), 1395-1409.

Gopi, P., Srinivasa Varma, P., Sai Kalyan, C. N., Ravikumar, C. V., Srinivasulu, A., Bohara, B., ... & Sathish, K. (2023). Dynamic behavior and stability analysis of automatic voltage regulator with parameter uncertainty. *International Transactions on Electrical Energy Systems*, *2023*(1), 6662355.

Hassaan, G. A. (2021). Tuning of a PD-PI Controller to Control Overdamped Second-Order-Like Processes. *Journal homepage: www. ijrpr. com ISSN*, *2582*, 7421.

Izci, D., Ekinci, S., Kayri, M., & Eker, E. (2022). A novel improved arithmetic optimization algorithm for optimal design of PID controlled and Bode's ideal transfer function based automobile cruise control system. *Evolving Systems*, *13*(3), 453-468.

Lachekhab, F., Zamoum, R. B., Belatreche, R., Kouzou, A., & Acheli, D. (2022, May). Modified Sliding Mode Control of Autonomous Quadrotor. In *2022 19th International Multi-Conference on Systems, Signals & Devices (SSD)* (pp. 1248-1254). IEEE.

Mahmoud, M. S., & AlRamadhan, A. H. (2021). Optimizing the parameters of sliding mode controllers for stepper motor through Simulink response optimizer application. *International Journal of Robotics and Control Systems*, *1*(2), 209-225.

Normann, V. (2023). *Event-triggered and self-triggered control of a 3 DOF hover system* (Master's thesis, uis).

Nour, M., Farid, F., & Adel, O. (2023, October). Optimizing Bilateral Teleoperation Control Architectures: A Comparative Study Under Constant and Variable Time Delays. In *2023 International Conference on Networking and Advanced Systems*

(ICNAS) (pp. 1-6). IEEE.

Ouerdane, F., & Mysorewala, M. F. (2024, June). Visual Servoing of a 3 DOF Hover Quadcopter using 2D Markers. In *2024 IEEE 33rd International Symposium on Industrial Electronics (ISIE)* (pp. 1-6). IEEE.

Quanser 3 DoF Hover - Flight Dynamics and Control of Vertical Lift-off Vehicles. Quanser, 2024. URL https://www.quanser.com/products/3-dof-hover/. Accessed 04 October 2024.

Prach, A., Kayacan, E., & Bernstein, D. S. (2016, July). An experimental evaluation of the forward propagating Riccati equation to nonlinear control of the Quanser 3 DOF Hover testbed. In *2016 American Control Conference (ACC)* (pp. 3710-3715). IEEE.

Rodríguez-Molina, A., Mezura-Montes, E., Villarreal-Cervantes, M. G., & Aldape-Pérez, M. (2020). Multi-objective meta-heuristic optimization in intelligent control: A survey on the controller tuning problem. *Applied Soft Computing*, *93*, 106342.

Wu, J., Xu, Z., Zhang, Y., Su, C. Y., & Wang, Y. (2023). Modeling and tracking control of dielectric elastomer actuators based on fractional calculus. *ISA transactions*, *138*, 687-695.

About the Authors

Saliha KÖPRÜCÜ graduated from the Department of Aeronautical Engineering at Necmettin Erbakan University, located in Konya, in 2023, and subsequently began her master's studies in the same department. She is currently continuing her graduate studies and has been working as a research assistant at Necmettin Erbakan University since December 2023. Her research areas include Aerospace Vehicle Dynamics, Control, and Autonomous Systems, in which she aims to gain expertise.
E-mail: salihakoprucu@gmail.com, **ORCID:** 0009-0007-9685-0008

Engin Hasan ÇOPUR gradauted as a mechanical engineer in 2007 from Gazi University which is located in Ankara, the capital city of Türkiye. Then started my MSc study in the same university and completed it in 2011. During this period, the author was also working as a research assistant there. After completing MSc study, author was able to obtain a state funding for my PhD study in UK and gained a Phd position in University of Southampton. Then author returned Turkey and started working as a lecturer in 2017 in Necmettin Erbakan University which is located in Konya Türkiye. Since 2017, author held the same position.
E-mail: engincopur@gmail.com, **ORCID:** 0000-0003-0837-1255

Hasan Huseyin BILGIC is received MSc. from Department of Mechanical Engineering, Mustafa Kemal University, Hatay, Turkey, in 2014 and then received PhD from Department of Mechanical Engineering, İskenderun Technical University, Hatay, Turkey, in 2018. Now he is Assistant Professor at Department of Aeronautical Engineering, Necmettin Erbakan University, Konya, Turkey. His current research interests include command/input shaping, sliding mode control, artificial intelligence, linear and nonlinear control systems, UAV system design and control.
E-mail: bilgichh@gmail.com, **ORCID:** 0000-0001-6006-8056

Similarity Index
The similarity index obtained from the plagiarism software for this book chapter is 13%.

On the Advanced Optimization Techniques for the Aerodynamic Design of Turbomachinery

Mehmet Numan KAYA

Necmettin Erbakan University

Bilal ERVURAL

Necmettin Erbakan University

To Cite This Chapter

Kaya, M. N., & Ervural, B. (2024). On the Advanced Optimization Techniques for the Aerodynamic Design of Turbomachinery. In S. Koçer & Ö. Dündar (Eds.), *Intelligent Systems and Optimization in Engineering* (pp. 80-97). ISRES Publishing.

Introduction

Optimization in engineering has become an essential tool for improving design efficiency and performance across various fields, leveraging advancements in computational power and algorithmic strategies (Bian & Priyadarshi, 2024; Kose & Kaya, 2018; Velasco et al., 2024; Ayaz & Kamisli Ozturk, 2021). Turbomachinery design optimization has evolved significantly in recent decades and is driven by these developments. The complex nature of turbomachinery flows, involving three-dimensional viscous effects, secondary flows, and multiple performance objectives, necessitates sophisticated optimization approaches. Traditional design methods, mainly relying on empirical correlations and engineering experience, are increasingly being supplemented or replaced by automated optimization techniques.

The evolution of optimization methods in turbomachinery design has progressed from simple gradient-based approaches to advanced machine-learning algorithms (Xu et al., 2024). This progression has been enabled by increased computational power and an improved understanding of complex flow physics. Modern optimization techniques can simultaneously handle multiple design variables and constraints while considering various performance metrics such as efficiency, pressure ratio, and structural integrity (Li & Zheng, 2017; Xu, 2024). Recent developments in artificial intelligence and machine learning have introduced new possibilities in turbomachinery optimization (Zou et al., 2024). These methods offer potential advantages in handling high-dimensional design spaces and reducing computational costs through surrogate modeling. However, selecting and implementing appropriate optimization strategies remains challenging and requires careful consideration of problem-specific requirements.

This study presents a comprehensive analysis of advanced optimization techniques applied to turbomachinery design. The strengths and limitations of different optimization strategies, their practical implementation challenges, and emerging trends in the field are examined. The following sections explore various advanced optimization methods used in turbomachinery design, with a discussion of their principles, advantages, and applications. The third section provides an in-depth analysis of the strengths and weaknesses of these optimization techniques, followed by a detailed look at the practical challenges encountered in their implementation. The study concludes with insights into future directions in optimization research, highlighting promising trends and areas for further exploration.

Advanced Optimization Methods in Turbomachinery Design

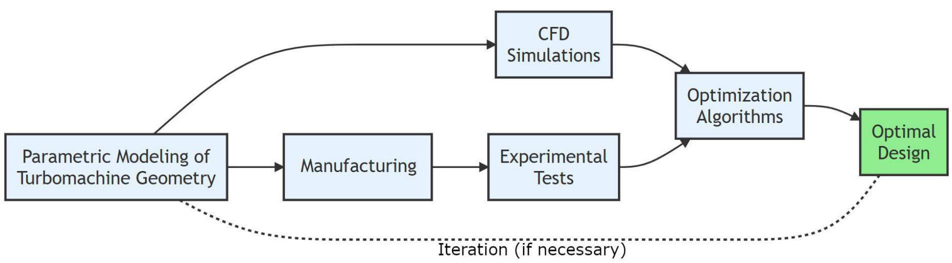
The aerodynamic optimization of turbomachinery has evolved significantly, moving from traditional trial-and-error methods to advanced computational approaches that allow for the systematic and efficient exploration of complex design spaces (Lavimi et al., 2024). As illustrated in Figure 1, the modern turbomachinery optimization workflow begins with the parametric modeling of component geometry, where flexible design parameters are established. This is followed by performance evaluation through computational fluid dynamics (CFD) simulations or experimental testing, providing critical insights into how the design performs under various conditions. Insights from these steps feed into optimization algorithms, which generate the optimal design. If necessary, iterations are performed to refine the design further, ensuring convergence to the best possible solution.

Recent advancements in optimization methods have significantly expanded the capabilities of turbomachinery design (Sagebaum et al., 2023). Techniques such as surrogate modeling, adjoint-based methods, and machine learning-driven approaches have transformed how engineers approach high-dimensional and nonlinear design challenges (LI et al., 2023; J. Luo et al., 2022). Surrogate models enable rapid design space exploration by approximating the results of computationally expensive high-fidelity simulations, drastically reducing the time required for optimization. Adjoint methods, on the other hand, leverage gradient information with exceptional precision, allowing for the optimization of intricate geometries and flow characteristics with minimal computational cost. Meanwhile, machine learning and neural network-based approaches have introduced data-driven solutions that are particularly effective in handling large datasets and predicting performance metrics under complex and dynamic conditions (Zou et al., 2024).

The following sub-sections provide an in-depth examination of these advanced methods, highlighting their strengths and applications in turbomachinery design. Adjoint and gradient-based techniques are emphasized for their precision in high-dimensional optimization problems. Metaheuristics, such as genetic algorithms and particle swarm optimization, are noted for their flexibility in addressing nonlinear and multimodal objectives. Surrogate-based approaches offer computational efficiency, while neural networks and deep learning (introduce the scalability needed for modern aerodynamic challenges. Hybrid methods combine the benefits of these techniques, enabling robust and versatile solutions. Together, these methodologies represent a paradigm shift in turbomachinery optimization, allowing engineers to push the boundaries of performance and innovation.

Figure 1

Schematic representation of optimization methodology in turbomachinery design process



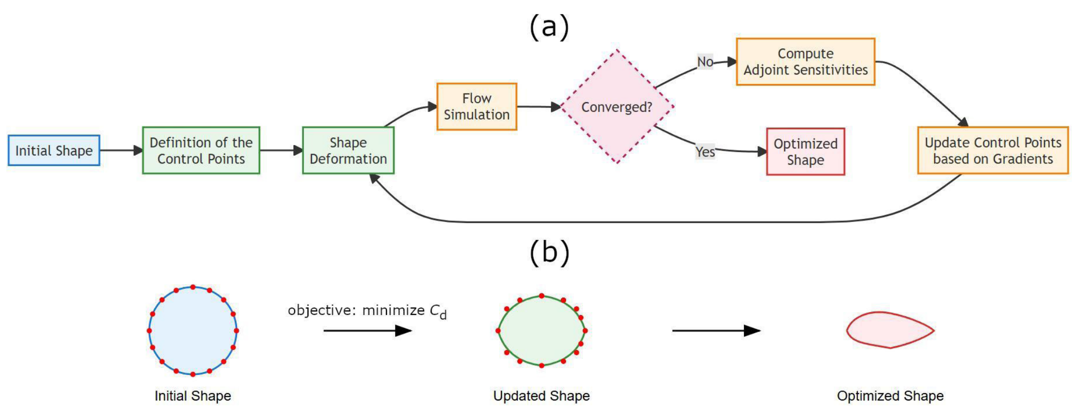
Adjoint and Gradient-Based Optimization

In turbomachinery design, classical optimization methods have played a significant role; however, their limitations have prompted the exploration of more advanced techniques, as they often struggle with the complexity and high dimensionality of turbomachinery design problems. These methods can be computationally expensive and may become trapped in local optimal, failing to identify the global optimum (Kim et al., 2019; T. Liu et al., 2019). As a result, researchers have increasingly turned to gradient-based methods, particularly the adjoint method, which offers a more efficient means of calculating sensitivity functions and derivatives of objective functions independent of the number of design variables (Lavimi, 2023; Rubino et al., 2021; Walther & Nadarajah, 2015; L. Wu et al., 2021). One prominent development area involves using adjoint-based techniques for calculating gradients. These techniques enable designers to optimize blade shapes for multiple objectives, such as minimizing pressure losses, maximizing efficiency, and ensuring robust performance under varying operating conditions.

In adjoint optimization, control points are defined around the shape, and their positions are modified within specified limits during optimization. As the shape changes, the flow around it must be recalculated. This requires completely re-meshing the domain or moving the existing mesh points from their initial positions. The first approach, re-meshing, is more suitable for significant geometric changes but is computationally expensive. In contrast, the second approach, mesh deformation, is better suited for minor shape modifications and is less time-intensive. Regardless of the method used, it is crucial that the generated mesh ensures smooth variations in the objective function to minimize noise in the computed gradients (Schramm et al., 2018). Figure 2 shows the systematic process of adjoint-based shape optimization for turbomachinery applications. The flowchart in Figure 2(a) illustrates the iterative optimization procedure, beginning with an initial shape definition and control point placement, followed by shape deformation, flow simulation, and adjoint sensitivity computation until convergence is achieved. As a representative example, Figure 2(b) demonstrates how the geometry might evolve from a simple circular initial shape to the final optimized airfoil profile through strategic manipulation of control points. This example transformation illustrates the concept of shape evolution guided by the objective of minimizing the drag coefficient (C_d), with the intermediate updated shape representing a transitional stage in the optimization process.

Figure 2

(a) Flowchart of the adjoint-based shape optimization process (b) Representative example showing the progression of shape optimization from initial circular shape to optimized shape with the objective of minimizing drag coefficient (C_d)



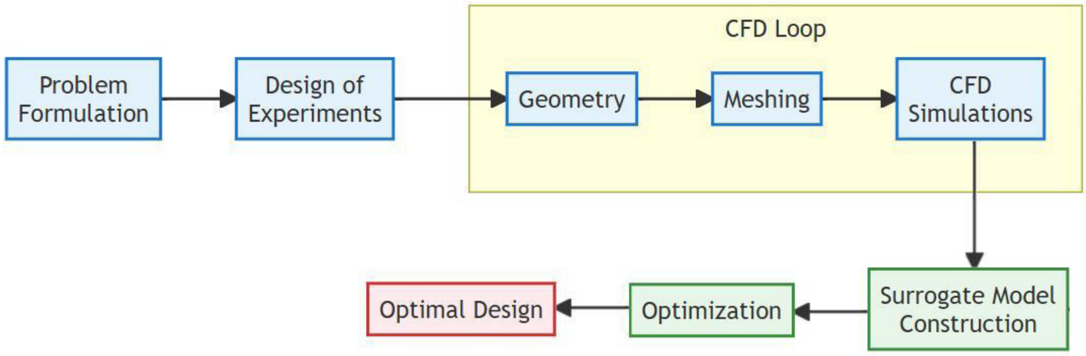
A discrete adjoint framework has been shown to facilitate efficient aerodynamic optimization by leveraging adaptive polynomial chaos expansion to mitigate uncertainties in flow conditions, improving design robustness significantly (Zhang et al., 2023). Recent studies have emphasized the importance of incorporating unsteady flow dynamics into optimization frameworks, recognizing the inherent transient nature of turbomachinery operations. Adjoint methods combined with harmonic balance solvers have effectively addressed unsteady aerodynamic damping and vibration stability in compressor cascades, improving operational reliability (H. Huang & Ekici, 2014; Rubino et al., 2020). Similarly, fully turbulent adjoint approaches leveraging time-domain methods have demonstrated accurate gradients for multirow configurations, enabling enhanced efficiency gains compared to traditional steady-state assumptions (Ntanakas et al., 2018). Unsteady optimization frameworks have also extended to multistage environments, showing remarkable capabilities in improving compressor performance by optimizing transient flow behavior (C. Ma et al., 2017). Another noteworthy development is incorporating robust optimization strategies to address flow conditions and design parameter uncertainties. Gradient-based methods assisted by surrogate models or adaptive response surfaces have proven effective in quantifying and mitigating the impact of such uncertainties. For example, surrogate-assisted gradient-based optimization methods have been validated for improving aerodynamic robustness in transonic turbine blades, outperforming traditional deterministic approaches (J. Luo et al., 2022). Furthermore, advanced frameworks utilizing polynomial chaos expansions have enhanced the efficiency of robust aerodynamic design processes (Zhang et al., 2023).

Advancements in geometry parameterization techniques have also complemented gradient-based methods by providing flexible and accurate representations of blade shapes. Methods based on Non-Uniform Rational B-Splines (NURBS) and shape derivatives have facilitated the seamless integration of parameterization into optimization workflows, ensuring smooth transitions between baseline and optimized geometries (Agromayor et al., 2021). Multi-objective optimization, aided by gradient-based Pareto front approximation, has enabled designers to balance conflicting objectives, such as efficiency and pressure loss, with reduced computational costs (Vasilopoulos et al., 2021). Recent studies highlight a clear trajectory toward integrating gradient-based optimization with high-fidelity modeling, robust uncertainty quantification, and advanced parameterization techniques. These developments underscore the transformative potential of gradient-based approaches in achieving high-performance and resilient turbomachinery systems.

Surrogate-Based Optimization

Surrogate-based optimization (SBO) is designed to tackle computationally expensive optimization problems by approximating the original high-fidelity model with a computationally efficient surrogate model (Koziel et al., 2011). The surrogate model acts as a proxy, reducing the computational burden while retaining reasonable accuracy. The working principle of SBO involves constructing surrogate models—such as polynomial regression, kriging, radial basis functions, or neural networks—based on a limited number of high-fidelity simulations. These models are then iteratively refined by adding new samples in areas of interest, balancing design space exploration with the exploitation of known high-performance regions (Queipo et al., 2005). Figure 3 illustrates the critical steps in a surrogate-based optimization process that utilizes CFD simulations. Kaya et al. (2021) employed a similar approach for the aerodynamic optimization of a wind turbine blade using CFD.

Figure 3
Flowchart of Surrogate-based Optimization based on CFD Simulations



SBO has found widespread applications in engineering that require complex simulations and high-dimensional optimization, especially in aerospace and turbomachinery design. Its effectiveness has been demonstrated in diverse disciplines, including aerodynamics, structures, and propulsion.

The effectiveness of surrogate models in turbomachinery optimization is underscored by their ability to facilitate rapid evaluations of design alternatives while maintaining accuracy in performance predictions. Zhao et al. (2024) developed a prescreening surrogate-model-assisted multi-objective differential evolution optimizer for highly loaded axial compressors, demonstrating notable improvements with efficiency increases and surge margin improvements. In addressing manufacturing uncertainties, Cheng et al. (2023) introduced a novel surrogate model combining self-organizing mapping and neural networks, improving efficiency and reducing performance variability.

The application of surrogate models spans various turbomachinery types and optimization challenges. Kim et al. (2010) demonstrated their effectiveness in centrifugal compressor impeller optimization using three-dimensional Reynolds-averaged Navier-Stokes equations while Heo et al. (2016) applied these techniques to mixed-flow pump optimization. In the context of low-pressure turbines, Baert et al. (2020) tackled high-dimensional design spaces with 350 parameters, achieving efficiency gains of 0.5 points through online surrogate-based optimization. Kong et al. (2021) further demonstrated the method's versatility in low-pressure axial fan design, achieving efficiency improvements through kriging-based surrogate models.

Advanced applications have shown promise in complex design scenarios. Persico et al. (2019) developed a sophisticated approach for non-conventional turbomachinery, achieving a 50% reduction in cascade loss coefficient for supersonic turbine nozzles. Mondal et al. (2019) introduced a multi-fidelity global-local approach for transonic compressor optimization, effectively combining rapid low-fidelity evaluations with targeted high-fidelity simulations. Q. Wang et al. (2022) successfully applied surrogate-based optimization to counter-rotating open rotors while Cao et al. (2022) implemented non-parametric surrogate models for low-pressure steam turbine exhaust systems.

The integration of surrogate models with advanced optimization algorithms has further enhanced their effectiveness. Song et al. (2014) demonstrated this by combining genetic algorithms with artificial neural networks for radial compressor optimization. Kozaket al. (2020) coupled high-fidelity flow modeling with a surrogate management framework for gas turbine optimization. These hybrid approaches have proven particularly effective in managing the trade-off between computational cost and design accuracy, enabling more efficient exploration of complex design spaces while maintaining solution quality.

Metaheuristic-Based Optimization

Metaheuristic algorithms are widely employed in turbomachinery design to address the complexity and nonlinearity of design optimization problems. These algorithms handle multidimensional, multimodal, and highly constrained design spaces shared in turbomachinery applications. Unlike gradient-based methods, metaheuristics do not rely on derivatives, making them highly versatile for problems with discontinuities or non-smooth objective functions. This section reviews the applications, strengths, and advancements of metaheuristic techniques in turbomachinery design, along with insights from the literature.

Metaheuristic algorithms have been found to be extensively valuable for optimizing various components of turbomachinery, including blades, compressors, turbines, and casings. For example, optimizing radial flow turbines has demonstrated the efficacy of coupling metaheuristic algorithms with CFD simulations. Studies have shown that Grey Wolf Optimizer (GWO) outperforms other algorithms in achieving higher temperature drops by optimizing blade inlet angles and improving casing design for better pressure recovery (Mehrnia et al., 2020). Similarly, the aerodynamic performance of single-stage transonic axial compressors has been enhanced by hybrid algorithms like the combination of genetic algorithms (GA) and particle swarm optimization (PSO), which optimize parameters such as stall margin and peak efficiency (Dinh et al., 2024; Vuong & Kim, 2021).

Swarm-based algorithms such as artificial bee colony and PSO have proven effective in multi-disciplinary optimization frameworks. These methods, combined with high-order CFD solvers, have been employed for addressing aero-mechanical challenges, providing reliable and efficient designs under complex constraints (Ampellio et al., 2016). Furthermore, bio-inspired algorithms like Genetic Algorithms, Flower Pollination Algorithm, and Cuckoo Search have been utilized to optimize geometries for axial turbomachinery, highlighting their adaptability to diverse design conditions and constraints (Ait Chikh et al., 2018).

Metaheuristic algorithms offer several advantages in turbomachinery design. Their flexibility allows them to tackle multi-objective optimization problems, such as maximizing efficiency while minimizing pressure losses. Advanced variants, such as hierarchical dynamic switching PSO, introduce adaptive mechanisms that enhance convergence rates and global search capabilities, effectively preventing premature convergence (Yan et al., 2024). These improvements are particularly beneficial for complex optimization tasks like turbine blade profile design, where global optima are challenging to identify.

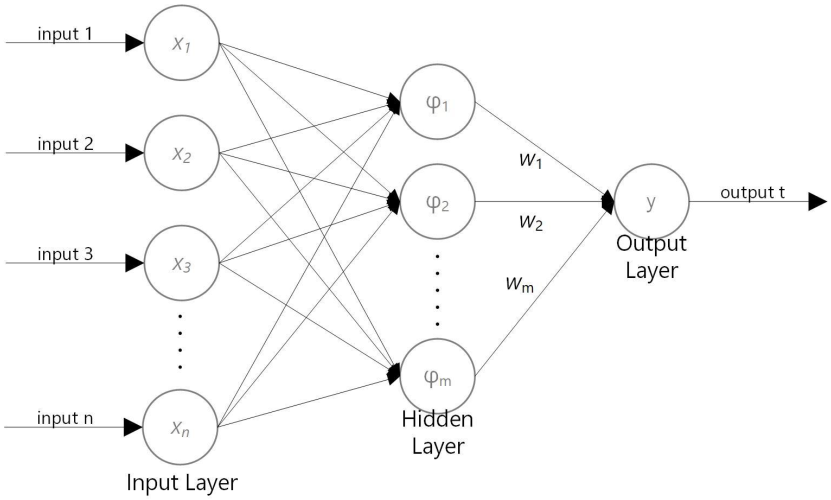
Another strength lies in their ability to incorporate surrogate models, which reduce computational costs without compromising accuracy. For instance, surrogate-based optimization methods, combined with PSO, have been used to design dual-bleeding recirculation channels in compressors, resulting in significant improvements in stall margin and operational stability (Vuong & Kim, 2021). This integration of metaheuristics with machine learning techniques, such as neural networks, has also enabled advancements in reliability prediction and dynamic modeling for turbomachinery systems (Bai et al., 2021).

Neural Networks

Neural networks (NNs) have become an indispensable tool in engineering design and optimization, offering exceptional capabilities for performance prediction, design exploration, and uncertainty quantification (Ünler & Seyfi, 2022). Their ability to model complex, nonlinear relationships between design parameters and performance metrics has made them a cornerstone for surrogate modeling, optimization, and robust analysis. Various NN architectures have been developed to suit applications, including multilayer perceptrons (MLPs), learning vector quantization, and radial basis function networks (RBFNNs). These networks are categorized based on data flow as feed-forward or recurrent and by learning approaches such as supervised training or self-organizing techniques. Figure 4 illustrates the general architecture of an RBFNN.

Figure 4

A General Architecture of a Radial Basis Function Neural Network (RBFNN)



One of the primary roles of NNs is as surrogate models for predicting performance metrics based on geometric or operational parameters. Ghorbanian and Gholamrezaei (2009) demonstrated this through an artificial neural network that effectively mapped the relationship between compressor design variables and aerodynamic performance. This work established the foundation for using NNs as predictive tools, reducing the dependency on full-scale simulations. Similarly, Barsi et al. (2021) employed NNs to optimize the design of a hydraulic propeller turbine, showing how these models can drive significant performance improvements through geometric modifications.

The integration of NNs with advanced optimization algorithms has further expanded their utility. For instance, Sakaguchi et al. (2016) combined NNs with genetic algorithms to enhance optimization efficiency by rapidly evaluating design alternatives. This hybrid approach leverages the speed of NNs for surrogate modeling and the global search capabilities of genetic algorithms, enabling faster convergence to optimal solutions. Another notable example is Du et al. (2022) applied series convolutional neural networks to optimize the end-wall profile of turbine stator blades. This method improved aerodynamic performance and required minimal training data, outperforming traditional surrogate models in accuracy and efficiency.

Neural networks are also pivotal in addressing uncertainties in turbomachinery operations, such as variations in operating conditions or material properties. Dual Graph Neural Networks (DGNNs) have been used for robust aerodynamic optimization, accurately predicting flow field behavior under multi-source uncertainties. Li et al. (2023) demonstrated that incorporating DGNNs into optimization frameworks led to designs that enhanced power output and efficiency while minimizing performance variability, ensuring robust performance across diverse operating scenarios.

Another significant NNs application is modeling fluid-structure interactions, such as blade flutter and aeroelastic stability. Graph Convolutional Neural Networks (GCNNs) have been employed to predict aerodynamic damping and stability margins with remarkable precision and speed. By replacing traditional high-cost simulation methods, GCNNs enable rapid analysis of aeroelastic phenomena, allowing for faster iteration during the design phase.

In scenarios with limited access to high-fidelity data, Multi-Fidelity Graph Neural Networks (MFGNNs) have demonstrated exceptional capability by integrating low- and high-fidelity datasets (Li et al. (2023) and Liu et al. (2024) showed how MFGNNs achieve accurate predictions for flow field characteristics while minimizing computational costs. This approach bridges the gap between computational efficiency and predictive accuracy, making it a powerful tool for turbomachinery optimization.

Deep Learning Approaches

Deep learning (DL) has emerged as a transformative tool in turbomachinery optimization, demonstrating unparalleled capabilities in managing the intricacies of large datasets and highly nonlinear relationships inherent in complex aerodynamic and thermodynamic systems. Its ability to learn directly from data without requiring explicit physics-based modeling makes it particularly well-suited for addressing challenges in modern turbomachinery design.

Recent studies underscore the diverse applications and advantages of deep learning across different aspects of turbomachinery. Shrivastava et al. (2022) employed deep learning models combined with nonlinear optimization techniques to dramatically reduce turbocharger rotor design cycle times—from days to hours—while preserving the accuracy of dynamic performance predictions. This represents a significant leap in accelerating the iterative design process, a key challenge in industrial applications.

For predictive modeling in aerodynamic systems, Fesquet et al. (2024) showcased the superior capabilities of U-net architectures over traditional surrogate models like POD-Kriging. By leveraging deep neural networks, their approach effectively predicted 2D wake-flow fields and critical performance metrics for fan rotor blades, delivering both precision and computational efficiency. This advancement highlights how deep learning can address challenges in high-fidelity flow field simulations while reducing reliance on costly numerical computations.

Geometric deep learning has also gained traction as a specialized branch within DL applications for turbomachinery. Gouttiere et al. (2023) applied geometric convolutional neural networks to optimize the Rotor 37 test case, achieving a notable improvement in isentropic efficiency, verified through CFD validation. This study illustrates the power of continuous learning and geometry-aware models in tackling three-dimensional optimization challenges. Building on this foundation, Du et al. (2022) introduced dual convolutional neural networks for turbine blade profile optimization. The method achieved exceptional accuracy, with prediction errors below 0.5% for 99% of validation samples, and reduced computation times to a staggering 3 milliseconds per evaluation.

Transfer learning has proven highly effective when faced with limited training data—often a constraint in engineering domains. Deng et al. (2024) demonstrated this by employing deep transfer learning to optimize transonic rotor performance. By fine-tuning pre-trained models, they improved tip-loading distribution, achieving significant aerodynamic gains without requiring extensive datasets. Such techniques hold promise for applications where data collection is constrained by cost or physical limitations.

In addition to design optimization, deep learning has shown its strength in operational efficiency enhancements. Huang et al. (2024) introduced a data-driven multi-agent deep reinforcement learning framework for optimizing air compressors in industrial aerodynamic systems. By integrating historical operational data with advanced reinforcement learning principles, their approach addressed the challenges posed by cyclic production schedules and dynamic load profiles. The model outperformed conventional energy efficiency and operational cost strategies while ensuring system stability and security.

Moreover, DL's potential extends to monitoring and diagnostics. Cao et al. (2021) applied deep neural networks to predict gas path degradation in gas turbines. Their work demonstrated that DL models can extract meaningful patterns from complex datasets, enabling early detection of performance anomalies and informing proactive maintenance strategies.

Hybrid Methods

Hybrid optimization methods have gained attention in turbomachinery design due to their ability to combine the strengths of multiple approaches, such as surrogate modeling, advanced simulation techniques, and optimization algorithms. These methods effectively balance computational efficiency and design accuracy, which is critical in optimizing complex and computationally expensive systems like turbomachinery. By integrating data-driven techniques with physics-based models, hybrid methods enable faster convergence, improved performance predictions, and enhanced exploration of design spaces.

For instance, artificial neural networks (ANNs) are frequently hybridized with evolutionary algorithms to exploit the predictive accuracy of ANNs and the global search capabilities of evolutionary methods. ANNs excel in approximating nonlinear relationships within the design space, while evolutionary algorithms are adept at exploring diverse regions of the space to find global optima. Villar et al. (2018) demonstrated this synergy by employing feedforward neural networks and evolutionary algorithms to optimize the aerodynamic performance of counter-rotating open rotors. Additionally, the work of Lavimi (2024) highlights the role of NNs alongside other surrogate models, such as polynomial response surface methods and Kriging, in aerodynamic optimization tasks. Moreover, integrating NNs with advanced optimization algorithms has led to significant advancements in turbomachinery design. For example, Song et al. employed a multidisciplinary design optimization approach that combined NNs with a self-adaptive multi-objective differential evolution algorithm to enhance the aerodynamic performance of a transonic turbine stage (Y. Wang et al., 2020). This synergy between NNs and optimization algorithms not only improves the accuracy of performance predictions but also accelerates the convergence of optimization processes.

Similarly, methods integrating surrogate models like Kriging with optimization techniques offer computationally efficient solutions for high-dimensional design problems. Bellary et al. (2016) compared various Kriging variants, including ordinary, universal, and blind Kriging, in optimizing a centrifugal impeller. The study highlighted how hybridizing Kriging models with hybrid genetic algorithms provided both accuracy and computational efficiency, enabling significant performance improvements in impeller design. Blind Kriging, in particular, achieved the best results by effectively modeling the complex flow characteristics of the system, reducing recirculation, and increasing efficiency. Luo et al. (2017) utilized proper orthogonal decomposition-based hybrid models for flow reconstruction and aerodynamic optimization in turbomachinery blades. Integrating POD modes, derived via singular value decomposition, with nonlinear regression techniques and adaptive Latin hypercube sampling ensured precision and computational efficiency.

Combining manual and automatic differentiation (AD), hybrid differentiation techniques also exhibit significant advantages in sensitivity analysis and gradient-based optimization. Wu et al. (2023) developed a hybrid adjoint solver by integrating AD with manually optimized code to reduce memory consumption and computational cost. Applied to NASA Stage 35 and Aachen turbines, the method efficiently optimized multi-row turbomachinery designs, highlighting its practical utility in large-scale, computationally intensive problems.

In addition to these examples, hybrid approaches that integrate radial basis function networks with dimensionality reduction techniques, such as principal component analysis, further exemplify the potential of hybrid methods. Ma et al. (2010) employed such a combination to optimize centrifugal compressor impellers, demonstrating the adaptability and effectiveness of hybrid approaches in turbomachinery design optimization.

Strengths and Weaknesses of Optimization Techniques in Turbomachinery Design

Optimization techniques play an essential role in turbomachinery design by enabling engineers to address complex, high-dimensional challenges and achieve optimal performance. These methods facilitate the navigation of multimodal and nonlinear design spaces, accommodating diverse constraints and performance metrics. However, each technique has specific strengths and limitations that influence its suitability for different design problems. Factors such as computational cost, sensitivity to noise, and the ability to handle uncertainty often determine their effectiveness in real-world scenarios.

Table 1 provides a detailed summary of the primary optimization methods employed in this domain, outlining their advantages, limitations, and notable applications in the literature. Genetic algorithms, for example, are renowned for their global search capabilities, excelling in finding solutions for highly complex and multimodal problems. Neural networks, in contrast, offer unmatched speed and precision in surrogate modeling, rapidly predicting performance metrics based on geometric or operational parameters. Deep learning techniques further expand these capabilities by efficiently processing large datasets and handling intricate geometries. Hybrid methods combine these strengths, leveraging the complementary advantages of different techniques to tackle multifaceted optimization challenges.

Despite their utility, these methods face challenges such as scalability in high-dimensional design spaces, overfitting in data-driven approaches, and computational intensity in gradient-free methods. Addressing these limitations often requires innovative integration of techniques, such as embedding physical constraints into machine learning models or employing multi-fidelity approaches to balance accuracy and computational cost. The analysis presented in this section provides a roadmap for selecting and tailoring optimization strategies to meet the demands of turbomachinery design.

Table 1
Strengths and Weaknesses of Optimization Techniques in Turbomachinery Design

Optimization Method		Key Strengths	Key Weaknesses	Example study in Turbomachinery Design
Meta-Heuristic Optimization	**Genetic Algorithms (GA)**	- Global search capability	- Computationally expensive	- Sakaguchi et al. (2016) have combined GA with NNs for turbine design optimization
		- Handles complex, multi-modal problems	- Convergence to local optima possible without proper tuning	
		- Flexible with non-linear objectives		
Neural Networks	**Neural Networks (NNs)**	- Fast surrogate modeling	- Requires large datasets for training	- Ghorbanian & Gholamrezaei (2009) have used NN for compressor performance prediction
		- Captures non-linear relationships	- Sensitive to overfitting and hyperparameter choices	
		- Reduces simulation costs		
	Physics-Informed Neural Networks (PINNs)	- Integrates physical laws	- High computational cost for training	- Salz et al. (2023) have used PINNs for airfoil optimization
		- Reduces reliance on data-driven methods	- Complex implementation	
		- Improves prediction reliability		
Deep Learning	**Deep Learning (e.g., CNNs, DCNNs)**	- Handles large datasets effectively	- Requires extensive computational resources	- Du et al. (2022) have applied DCNNs for turbine blade profile optimization
		- Adaptable to complex geometries	- Overfitting risks with limited data	
		- Predicts performance with high precision		
	Reinforcement Learning (e.g., DMA-DRL)	- Adapts to dynamic environments	- Long training times	- Huang et al. (2024) have used DMA-DRL for operational optimization of air compressors
		- Excels in operational efficiency optimization	- Requires detailed reward function design	
		- Handles multi-agent scenarios		
Hybrid Approaches	**Hybrid Approach**	- Combines strengths of different methods	- Implementation complexity	- Villar et al. (2018) have applied NN and evolutionary algorithms for counter-rotating rotor optimization
		- Balances exploration and exploitation	- Requires careful tuning of combined models	
		- Accelerates convergence		
Adjoint Optimization	**Adjoint Optimization**	- Provides high sensitivity accuracy	- Limited to differentiable models	- Wu et al. (2023) have used an adjoint solver for multi-row turbomachinery design
		- Efficient gradient-based optimization	- Can be computationally expensive for multi-row problems	
Surrogate-Based Methods	**Surrogate-Based Optimization (SBO)**	- Reduces computational costs	- Relies heavily on surrogate model accuracy	- Shrivastava et al. (2022) have used surrogate models for turbocharger rotor design optimization
		- Effective for expensive simulations	- Requires careful sampling design	
		- Provides interpretable models		
	Kriging (Ordinary, Blind, etc.)	- Accurate interpolation for small datasets	- Computationally expensive for high-dimensional problems	- Bellary et al. (2016) have compared Kriging variants for centrifugal impeller optimization
		- Provides uncertainty quantification	- Limited scalability	

Challenges and Future Directions

While significant advancements have been made in turbomachinery optimization, critical challenges persist that demand innovative solutions. Managing high-dimensional design spaces and capturing complex flow physics remain formidable tasks, often requiring a delicate balance between computational efficiency and solution accuracy. Additionally, robust design optimization under uncertainty—essential for ensuring reliable performance across varying operating conditions—is an area requiring further exploration. Validation of advanced optimization techniques through experimental studies is also critical for bridging the gap between theoretical advancements and practical implementation.

Recent developments in metaheuristic algorithms tailored for turbomachinery applications have begun addressing these challenges (Hakan Cetin & Zhu, 2023). For instance, neuroevolutionary strategies that integrate ant colony optimization with long short-term memory neural networks have shown promise in predictive maintenance, particularly for predicting turbine engine vibrations (ElSaid et al., 2018). Experimental validation has further bolstered the role of metaheuristics in design optimization. Studies on high-load axial flow compressors have demonstrated the effectiveness of multi-objective PSO algorithms, achieving notable gains in peak efficiency and stall margin (S. Huang et al., 2024). Similarly, integrating metaheuristics with dynamic weight strategies, as exemplified by the SDWPSO-BPNN models, has significantly improved reliability predictions for turbochargers, outperforming conventional methods (Bai et al., 2021).

The future of turbomachinery optimization lies in hybrid methodologies that combine the strengths of metaheuristics, surrogate models, and machine learning. Physics-informed neural networks (PINNs), which embed domain-specific knowledge such as the Navier-Stokes equations, hold the potential to improve the accuracy and reliability of optimization outcomes. Transfer learning, enabling the reuse of pre-trained models with limited data, offers a promising avenue for reducing computational demands in scenarios with sparse high-fidelity datasets. Focusing on sequential decision-making, reinforcement learning presents a compelling solution for optimizing operational strategies and diagnostics in real-time. Moreover, experimental validation will continue to play a pivotal role in ensuring that advancements translate effectively into real-world performance, fostering the development of more reliable and efficient turbomachinery systems.

Conclusion

Advanced optimization techniques have revolutionized turbomachinery design, enabling engineers to navigate complex design spaces and achieve unprecedented performance. Gradient-based methods, metaheuristics, surrogate models, neural networks, deep learning, and hybrid approaches have each contributed to this progress. The key strengths of specific optimization methods include genetic algorithms excelling at global optimization for complex problems, neural networks rapidly predicting performance, deep learning effectively handling large datasets and complex geometries, and hybrid methods synergistically combining multiple techniques. However, challenges remain in computational efficiency, uncertainty quantification, and experimental validation. Future research should explore hybrid methods, physics-informed models, transfer learning, and reinforcement learning to push the boundaries of turbomachinery optimization further. By addressing these challenges and leveraging emerging techniques, engineers can design the next generation of highly efficient, reliable, and sustainable turbomachinery systems.

References

Agromayor, R., Anand, N., Müller, J. D., Pini, M., & Nord, L. O. (2021). A Unified

Geometry Parametrization Method for Turbomachinery Blades. *Computer-Aided Design*, *133*, 102987. https://doi.org/10.1016/J.CAD.2020.102987

Ait Chikh, M. A., Belaidi, I., Khelladi, S., Paris, J., Deligant, M., & Bakir, F. (2018). Efficiency of bio- and socio-inspired optimization algorithms for axial turbomachinery design. *Applied Soft Computing*, *64*, 282–306. https://doi.org/10.1016/J.ASOC.2017.11.048

Ampellio, E., Bertini, F., Ferrero, A., Larocca, F., & Vassio, L. (2016). Turbomachinery design by a swarm-based optimization method coupled with a CFD solver. *Advances in Aircraft and Spacecraft Science*, *3*(2), 149–170. https://doi.org/10.12989/AAS.2016.3.2.149

Ayaz, H. I., & Kamisli Ozturk, Z. (2021). A mathematical model and a heuristic approach for train seat scheduling to minimize dwell time. *Computers & Industrial Engineering*, *160*, 107590. https://doi.org/10.1016/J.CIE.2021.107590

Baert, L., Chérière, E., Sainvitu, C., Lepot, I., Nouvellon, A., & Leonardon, V. (2020). Aerodynamic Optimization of the Low-Pressure Turbine Module: Exploiting Surrogate Models in a High-Dimensional Design Space. *Journal of Turbomachinery*, *142*(3). https://doi.org/10.1115/1.4046232

Bai, B., Zhang, J., Wu, X., wei Zhu, G., & Li, X. (2021). Reliability prediction-based improved dynamic weight particle swarm optimization and back propagation neural network in engineering systems. *Expert Systems with Applications*, *177*, 114952. https://doi.org/10.1016/J.ESWA.2021.114952

Barsi, D., Ubaldi, M., Zunino, P., & Fink, R. (2021). Optimized Design of a Novel Hydraulic Propeller Turbine for Low Heads. *Designs*, *5*(1), 20. https://doi.org/10.3390/designs5010020

Bellary, S. A. I., Samad, A., Couckuyt, I., & Dhaene, T. (2016). A comparative study of kriging variants for the optimization of a turbomachinery system. *Engineering with Computers*, *32*(1), 49–59. https://doi.org/10.1007/S00366-015-0398-X/TABLES/6

Bian, K., & Priyadarshi, R. (2024). Machine Learning Optimization Techniques: A Survey, Classification, Challenges, and Future Research Issues. *Archives of Computational Methods in Engineering 2024 31:7*, *31*(7), 4209–4233. https://doi.org/10.1007/S11831-024-10110-W

Cao, J., Li, Q., Xu, L., Yang, R., & Dai, Y. (2022). Non-Parametric Surrogate Model Method Based on Machine Learning With Application on Low-Pressure Steam Turbine Exhaust System. *Journal of the Global Power and Propulsion Society*, *6*, 165–180. https://doi.org/10.33737/jgpps/151661

Cao, Q., Chen, S., Zheng, Y., Ding, Y., Tang, Y., Huang, Q., Wang, K., & Xiang, W. (2021). Classification and Prediction of Gas Turbine Gas Path Degradation Based on Deep Neural Networks. *International Journal of Energy Research*, *45*(7), 10513–10526. https://doi.org/10.1002/er.6539

Cheng, H., Zhou, C., Lu, X., Zhao, S., Han, G., & Yang, C. (2023). Robust aerodynamic optimization and design exploration of a wide-chord transonic fan under geometric and operational uncertainties. *Energy*, *278*. https://doi.org/10.1016/j.energy.2023.128011

Deng, H., Zhang, S., Xia, K., Qiang, X., Zhu, M., & Teng, J. (2024). Performance prediction and design optimization of a transonic rotor based on deep transfer learning. *Physics of Fluids*, *36*(9). https://doi.org/10.1063/5.0221767

Dinh, C.-T., Vuong, T.-D., & Kim, K.-Y. (2024). Aerodynamic Optimization of a Single-Stage Transonic Axial Compressor with a Circumferential Feedback Channel.

Journal of Aerospace Engineering, *37*(3), 04024005. https://doi.org/10.1061/JAEEEZ.ASENG-4056

Du, Q., Li, Y., Yang, L., Liu, T., Zhang, D., & Xie, Y. (2022). Performance prediction and design optimization of turbine blade profile with deep learning method. *Energy*, *254*. https://doi.org/10.1016/j.energy.2022.124351

Du, Q., Yang, L., Li, L., Liu, T., Zhang, D., & Xie, Y. (2022). Aerodynamic design and optimization of blade end wall profile of turbomachinery based on series convolutional neural network. *Energy*, *244*, 122617. https://doi.org/10.1016/J.ENERGY.2021.122617

ElSaid, A. E. R., El Jamiy, F., Higgins, J., Wild, B., & Desell, T. (2018). Optimizing long short-term memory recurrent neural networks using ant colony optimization to predict turbine engine vibration. *Applied Soft Computing*, *73*, 969–991. https://doi.org/10.1016/J.ASOC.2018.09.013

Fesquet, J., Bauerheim, M., Rojda, L., Bousquet, Y., & Binder, N. (2024). APPLICATION OF DEEP LEARNING FOR FAN ROTOR BLADE PERFORMANCE PREDICTION IN TURBOMACHINERY. *Proceedings of the ASME Turbo Expo*, *12D*. https://doi.org/10.1115/GT2024-127348

Ghorbanian, K., & Gholamrezaei, M. (2009). An artificial neural network approach to compressor performance prediction. *Applied Energy*, *86*(7–8), 1210–1221. https://doi.org/10.1016/J.APENERGY.2008.06.006

Gouttiere, A., Alessi, G., Wunsch, D., Zampieri, L., & Hirsch, C. (2023). REALTIME CFD BASED SHAPE OPTIMIZATION USING GEOMETRIC DEEP LEARNING FOR FAMILIES OF TURBOMACHINERY APPLICATIONS. *Proceedings of the ASME Turbo Expo*, *13D*. https://doi.org/10.1115/GT2023-101938

Hakan Cetin, T., & Zhu, J. (2023). Aerodynamic design and optimization of sCO2 turbomachinery: Two-Stage optimization approach. *Energy Conversion and Management*, *297*, 117711. https://doi.org/10.1016/J.ENCONMAN.2023.117711

Heo, M.-W., Kim, K.-Y., Kim, J., & Choi, Y. S. (2016). High-Efficiency Design of a Mixed-Flow Pump Using a Surrogate Model. *Journal of Mechanical Science and Technology*, *30*(2), 541–547. https://doi.org/10.1007/s12206-016-0107-8

Huang, H., & Ekici, K. (2014). A discrete adjoint harmonic balance method for turbomachinery shape optimization. *Aerospace Science and Technology*, *39*, 481–490. https://doi.org/10.1016/J.AST.2014.05.015

Huang, S., Yang, C., & Wang, P. (2024). Aerodynamic optimization design and experimental verification of a high-load axial flow compressor. *Aerospace Science and Technology*, *155*, 109638. https://doi.org/10.1016/J.AST.2024.109638

Huang, Y., Zhong, L., Liu, Y., Zhao, J., & Wang, W. (2024). Data Driven Multi-Agent Deep Reinforcement Learning Based Operational Optimization of Industrial Aerodynamic System. *Proceedings of 2024 IEEE 13th Data Driven Control and Learning Systems Conference, DDCLS 2024*, 464–471. https://doi.org/10.1109/DDCLS61622.2024.10606700

Kaya, M. N., Köse, F., Uzol, O., Ingham, D., Ma, L., & Pourkashanian, M. (2021). Aerodynamic Optimization of a Swept Horizontal Axis Wind Turbine Blade. *Journal of Energy Resources Technology, Transactions of the ASME*, *143*(9). https://doi.org/10.1115/1.4051469/1111083

Kim, J., Choi, J.-H., & Kim, K. (2010). Surrogate Modeling for Optimization of a Centrifugal Compressor Impeller. *International Journal of Fluid Machinery and Systems*, *3*(1), 29–38. https://doi.org/10.5293/ijfms.2010.3.1.029

Kim, J., Ma, S.-B., Kim, S. H., Choi, Y.-S., & Kim, K. (2019). Design and Verification of a Single-Channel Pump Model Based on a Hybrid Optimization Technique. *Processes*, *7*(10), 747. https://doi.org/10.3390/pr7100747

Kong, C., Wang, M., Jin, T., & Liu, S. (2021). An optimization on the stacking line of low-pressure axial-flow fan using the surrogate-assistant optimization method. *Journal of Mechanical Science and Technology*, *35*(11), 4997–5005. https://doi.org/10.1007/s12206-021-1018-x

Kose, F., & Kaya, M. N. (2018). Wind-Hydro Pumped Storage Power Stations to Meet the Energy Demands of Irrigation: Feasibility, Optimal Design and Simulation of a System Airiti Library Huayi Online Library. *CHINESE SOCIETY MECHANICAL ENGINEERS*, *39/2*, 223–232. https://www.airitilibrary.com/Article/Detail/02579731-201804-201806140004-201806140004-223-232

Kozak, N., Rajanna, M. R., Wu, M. C. H., Murugan, M., Bravo, L., Ghoshal, A., Hsu, M.-C., & Bazilevs, Y. (2020). Optimizing gas turbine performance using the surrogate management framework and high-fidelity flow modeling. *Energies*, *13*(17). https://doi.org/10.3390/en13174283

Koziel, S., Ciaurri, D. E., & Leifsson, L. (2011). Surrogate-Based Methods. *Studies in Computational Intelligence*, *356*, 33–59. https://doi.org/10.1007/978-3-642-20859-1_3

Lavimi, R. (2023). *Derivation of the Full Continuous Adjoint Method Based on the K–? SST Turbulence Model for Turbomachinery Shape Optimization*. https://doi.org/10.17118/11143/20987

Lavimi, R. (2024). A Review on Aerodynamic Optimization of Turbomachinery Using Adjoint Method. *Proceedings of the Institution of Mechanical Engineers Part C Journal of Mechanical Engineering Science*, *238*(13), 6405–6441. https://doi.org/10.1177/09544062231221625

Lavimi, R., Benchikh Le Hocine, A. E., Poncet, S., Marcos, B., & Panneton, R. (2024). A review on aerodynamic optimization of turbomachinery using adjoint method. *Https://Doi.Org/10.1177/09544062231221625*, *238*(13), 6405–6441. https://doi.org/10.1177/09544062231221625

Li, J., Li, Y., Liu, T., Zhang, D., & Xie, Y. (2023). Multi-fidelity graph neural network for flow field data fusion of turbomachinery. *Energy*, *285*, 129405. https://doi.org/10.1016/J.ENERGY.2023.129405

Li, J., Liu, T., Zhu, G., Li, Y., & Xie, Y. (2023). Uncertainty quantification and aerodynamic robust optimization of turbomachinery based on graph learning methods. *Energy*, *273*, 127289. https://doi.org/10.1016/J.ENERGY.2023.127289

LI, X., MENG, T., LI, W., ZHOU, L., & JI, L. (2023). Aerodynamic adjoint optimization of turbomachinery with direct control on blade design parameters. *Chinese Journal of Aeronautics*, *36*(11), 119–134. https://doi.org/10.1016/J.CJA.2023.09.022

Li, Z., & Zheng, X. (2017). Review of design optimization methods for turbomachinery aerodynamics. *Progress in Aerospace Sciences*, *93*, 1–23. https://doi.org/10.1016/J.PAEROSCI.2017.05.003

Liu, T., Zhang, Y., Li, K., Wang, Y., & Li, J. (2019). Aerodynamic Optimization of a Centrifugal Fan Using Response Surface Methodology. *Open Journal of Fluid Dynamics*, *09*(01), 92–105. https://doi.org/10.4236/ojfd.2019.91007

Liu, Y., Li, Y., Li, L., Xie, Y., & Zhang, D. (2024). A fast prediction model of blade flutter in turbomachinery based on graph convolutional neural network. *Aerospace Science and Technology*, *148*, 109119. https://doi.org/10.1016/J.AST.2024.109119

Luo, J., Chen, Z., & Zheng, Y. (2022). A gradient-based method assisted by surrogate model for robust optimization of turbomachinery blades. *Chinese Journal of*

Aeronautics, *35*(10), 1–7. https://doi.org/10.1016/j.cja.2021.07.019

Luo, J. Q., Zhu, Y. L., Tang, X., & Liu, F. (2017). Flow reconstructions and aerodynamic shape optimization of turbomachinery blades by POD-based hybrid models. *Science China Technological Sciences*, *60*(11), 1658–1673. https://doi.org/10.1007/S11431-016-9093-Y/METRICS

Ma, C., Su, X., & Yuan, X. (2017). An efficient unsteady adjoint optimization system for multistage turbomachinery. *Journal of Turbomachinery*, *139*(1). https://doi.org/10.1115/1.4034185/378754

Ma, Y., Engeda, A., Cave, M., & Di Liberti, J. L. (2010). Improved centrifugal compressor impeller optimization with a radial basis function network and principle component analysis. *Http://Dx.Doi.Org/10.1243/09544062JMES1635*, *224*(4), 935–945. https://doi.org/10.1243/09544062JMES1635

Mehrnia, S., Miyagawa, K., Kusaka, J., & Nakamura, Y. (2020). Radial turbine optimization under unsteady flow using nature-inspired algorithms. *Aerospace Science and Technology*, *103*, 105903. https://doi.org/10.1016/J.AST.2020.105903

Mondal, S., Joly, M. M., & Sarkar, S. (2019). Multi-fidelity global-local optimization of a transonic compressor rotor. *Proceedings of the ASME Turbo Expo*, *2D-2019*. https://doi.org/10.1115/GT2019-91778

Ntanakas, G., Meyer, M., & Giannakoglou, K. C. (2018). Employing the time-domain unsteady discrete adjoint method for shape optimization of three-dimensional multirow turbomachinery configurations. *Journal of Turbomachinery*, *140*(8). https://doi.org/10.1115/1.4040564/366301

Persico, G., Rodriguez-Fernandez, P., & Romei, A. (2019). High-Fidelity Shape Optimization of Non-Conventional Turbomachinery by Surrogate Evolutionary Strategies. *Journal of Turbomachinery*, *141*(8). https://doi.org/10.1115/1.4043252

Queipo, N. V., Haftka, R. T., Shyy, W., Goel, T., Vaidyanathan, R., & Kevin Tucker, P. (2005). Surrogate-based analysis and optimization. *Progress in Aerospace Sciences*, *41*(1), 1–28. https://doi.org/10.1016/J.PAEROSCI.2005.02.001

Rubino, A., Colonna, P., & Pini, M. (2021). Adjoint-Based Unsteady Optimization of Turbomachinery Operating With Nonideal Compressible Flows. *Journal of Propulsion and Power*, *37*(6), 910–918. https://doi.org/10.2514/1.b37920

Rubino, A., Vitale, S., Colonna, P., & Pini, M. (2020). Fully-turbulent adjoint method for the unsteady shape optimization of multi-row turbomachinery. *Aerospace Science and Technology*, *106*, 106132. https://doi.org/10.1016/J.AST.2020.106132

Sagebaum, M., Blühdorn, J., Gauger, N. R., Backhaus, J., Frey, C., Geiser, G., Bayer, F., Kügeler, E., Battistoni, C., & Nagel, M. (2023). On Recent Developments for Efficient Turbo-Machinery Design using Algorithmic Differentiation. *Www.Euroturbo.Eu//*. https://doi.org/10.29008/ETC2023-346

Sakaguchi, D., Tun, M. T., Numakura, R., & Wang, B. T. (2016). Global Optimization of Recirculation Flow Type Casing Treatment in Centrifugal Compressors of Turbochargers. *Proceedings of the Institution of Mechanical Engineers Part C Journal of Mechanical Engineering Science*, *232*(24), 4461–4471. https://doi.org/10.1177/0954406216679435

Schramm, M., Stoevesandt, B., & Peinke, J. (2018). Optimization of Airfoils Using the Adjoint Approach and the Influence of Adjoint Turbulent Viscosity. *Computation*, *6*(1). https://doi.org/10.3390/computation6010005

Shrivastava, S., Sinha, A., Ray, S., Du, I., Begin, L., Shrivastava, S., Sinha, A., Ray, S., Du, I., & Begin, L. (2022). Automotive Turbocharger Rotor Optimization Using Machine Learning Technique. *SAE Technical Papers*, *2022*. https://doi.

org/10.4271/2022-01-0216

Song, P., Sun, J., & Wang, K. (2014). Axial Flow Compressor Blade Optimization Through Flexible Shape Tuning by Means of Cooperative Co-Evolution Algorithm and Adaptive Surrogate Model. *Proceedings of the Institution of Mechanical Engineers Part a Journal of Power and Energy*, *228*(7), 782–798. https://doi.org/10.1177/0957650914541647

Ünler, T., & Seyfi, L. (2022). Experimental Investigation of Range Measurement Error for Radar under Different Weather Conditions. *Proceedings of the Bulgarian Academy of Sciences*, *75*(9), 1325–1333. https://doi.org/10.7546/CRABS.2022.09.10

Vasilopoulos, I., Asouti, V. G., Giannakoglou, K. C., & Meyer, M. (2021). Gradient-based Pareto front approximation applied to turbomachinery shape optimization. *Engineering with Computers*, *37*(1), 449–459. https://doi.org/10.1007/S00366-019-00832-Y/FIGURES/10

Velasco, L., Guerrero, H., & Hospitaler, A. (2024). A Literature Review and Critical Analysis of Metaheuristics Recently Developed. *Archives of Computational Methods in Engineering*, *31*(1), 125–146. https://doi.org/10.1007/S11831-023-09975-0/TABLES/9

Villar, G. M., Lindblad, D., & Andersson, N. (2018). *Multi-Objective Optimization of an Counter Rotating Open Rotor Using Evolutionary Algorithms*. https://doi.org/10.2514/6.2018-2929

Vuong, T. D., & Kim, K. Y. (2021). Design Optimization of a Dual-Bleeding Recirculation Channel to Enhance Operating Stability of a Transonic Axial Compressor. *Energies 2022, Vol. 15, Page 159*, *15*(1), 159. https://doi.org/10.3390/EN15010159

Walther, B., & Nadarajah, S. (2015). Optimum Shape Design for Multirow Turbomachinery Configurations Using a Discrete Adjoint Approach and an Efficient Radial Basis Function Deformation Scheme for Complex Multiblock Grids. *Journal of Turbomachinery*, *137*(8). https://doi.org/10.1115/1.4029550

Wang, Q., Li, Z., & Wang, Z. (2022). *Surrogate-Based Aerodynamic Shape Optimization of Contra Rotating Open Rotor*. https://doi.org/10.33737/gpps21-tc-76

Wang, Y., Li, L., Zhang, D., & Xie, Y. (2020). Aerodynamic Optimization of a SCO_2 Radial-Inflow Turbine Based on an Improved Simulated Annealing Algorithm. *Proceedings of the Institution of Mechanical Engineers Part a Journal of Power and Energy*, *235*(5), 1039–1052. https://doi.org/10.1177/0957650920976666

Wu, H., Da, X., Wang, D., & Huang, X. (2023). Multi-Row Turbomachinery Aerodynamic Design Optimization by an Efficient and Accurate Discrete Adjoint Solver. *Aerospace*, *10*(2). https://doi.org/10.3390/AEROSPACE10020106

Wu, L., Wilson, A. G., Kim, J. W., Radford, D., & Shahpar, S. (2021). *Low-Noise Blade Design Optimization for a Transonic Fan Using Adjoint-Based MDO Approach*. https://doi.org/10.2514/6.2021-3055

Xu, J. (2024). A Rapid Method for Turbomachinery Aerodynamic Design and Optimization Using Active Subspace. *Journal of the Global Power and Propulsion Society*, *8*, 24–38. https://doi.org/10.33737/jgpps/184214

Xu, L., Jin, S., Ye, W., Li, Y., & Gao, J. (2024). A Review of Machine Learning Methods in Turbine Cooling Optimization. *Energies 2024, Vol. 17, Page 3177*, *17*(13), 3177. https://doi.org/10.3390/EN17133177

Yan, C., Kang, E., Liu, H., Li, H., Zeng, N., & You, Y. (2024). Efficient aerodynamic optimization of turbine blade profiles: an integrated approach with novel HDSPSO algorithm. *Multidiscipline Modeling in Materials and Structures*, *20*(4), 725–745. https://doi.org/10.1108/MMMS-02-2024-0051/FULL/PDF

Zhang, J., Li, L., Dong, X., Zhang, Z., Zhang, Y., & Lu, X. (2023). A discrete adjoint framework coupled with adaptive PCE for robust aerodynamic optimization of turbomachinery under flow uncertainty. *Aerospace Science and Technology*, *142*, 108592. https://doi.org/10.1016/J.AST.2023.108592

Zhao, W., Chen, J., Liu, Y., Xiang, H., & Li, B. (2024). Prescreening surrogate-model-assisted multi-objective aerodynamic optimization design of highly loaded axial compressor in heavy-duty gas turbine. *Applied Thermal Engineering*, *254*. https://doi.org/10.1016/j.applthermaleng.2024.123813

Zou, Z., Xu, P., Chen, Y., Yao, L., & Fu, C. (2024). Application of artificial intelligence in turbomachinery aerodynamics: progresses and challenges. *Artificial Intelligence Review 2024 57:8*, *57*(8), 1–46. https://doi.org/10.1007/S10462-024-10867-3

About the Authors

Dr. Mehmet Numan KAYA is an Assistant Professor in the Department of Mechanical Engineering at Necmettin Erbakan University, Turkey. He earned both his bachelor's and master's degrees in mechanical engineering from Selçuk University. He completed his doctoral studies in Mechanical Engineering at Konya Technical University and served as a visiting researcher at the University of Sheffield during his PhD. Afterward, he pursued postdoctoral research at the RWTH Aachen University. His research interests include Aerodynamics, Aeroacoustics, Computational Fluid Dynamics, Fluid Flow and Turbomachinery.
E-mail: mnkaya@erbakan.edu.tr , **ORCID:** 0000-0003-3116-9442

Dr. Bilal ERVURAL is an Assistant Professor in the Department of Industrial Engineering at Necmettin Erbakan University, Turkey. His research interests include combinatorial optimization, manufacturing systems, group decision-making, and simulation-based analysis. Dr. Ervural has contributed to various academic and industrial projects, combining theoretical advancements with practical applications to address complex problems in engineering and decision sciences. His work focuses on developing innovative methodologies to enhance efficiency and decision-making in dynamic and uncertain environments.
E-mail: bervural@erbakan.edu.tr , **ORCID:** 0000-0002-5206-7632

Similarity Index
The similarity index obtained from the plagiarism software for this book chapter is 6%.

Operating Principles and Future Potential of Next-Generation Gunpowder-Free Launchers: Coil Guns

Tarık ÜNLER

Necmettin Erbakan University

Mehmet DURSUN

Necmettin Erbakan University

To Cite This Chapter

Ünler, T., & Dursun, M. (2024). Operating Principles and Future Potential of Next-Generation Gunpowder-Free Launchers: Coil Guns. In S. Koçer & Ö. Dündar (Eds.), *Intelligent Systems and Optimization in Engineering* (pp. 98-105). ISRES Publishing.

Introduction

Coil guns, also known as electromagnetic launchers, represent a significant advancement in projectile propulsion technology, utilizing electromagnetic fields to accelerate projectiles without the need for traditional chemical propellants. The historical development of coil guns can be traced back to the early explorations of electromagnetic propulsion, with initial designs focusing on single-stage systems. However, as research progressed, the introduction of multi-stage coil guns allowed for increased projectile velocities and improved efficiency, making them attractive for various applications, including military, aerospace, and even recreational uses (Dayi & Vijayan, 2017; Abdo, 2023; Çoramık et al., 2020). The operational principle of coil guns involves the sequential activation of electromagnetic coils that generate magnetic fields, propelling a ferromagnetic projectile along a barrel. This method offers several advantages over conventional firearms, such as reduced wear on components due to the absence of physical contact between the projectile and the coils, as well as the potential for higher muzzle velocities (Coşkun et al., 2010; Giès et al., 2020). Recent studies have demonstrated that optimizing coil configurations and utilizing advanced materials can significantly enhance the performance of coil guns. For instance, splitting a single coil into multiple smaller coils activated in sequence has been shown to improve energy transfer and efficiency (Giès & Soriano, 2019; Giès et al., 2020). Looking towards the future, the potential applications of coil guns are expanding rapidly. Innovations in design and materials are paving the way for their use in launching microsatellites from aircraft, which could revolutionize the aerospace industry by providing a cost-effective and efficient means of deploying payloads into orbit (Abdo, 2023). Additionally, hybrid systems that combine features of both coil guns and railguns are being explored, potentially leading to even greater versatility and performance (Domin, 2021). The integration of advanced computational modeling techniques, such as finite element analysis, is also expected to play a crucial role in the design and optimization of coil guns, allowing for more precise control over performance parameters (Liu et al., 2009; Abdo et al., 2016). Furthermore, the military applications of coil guns are being actively researched, with the potential for hypervelocity projectiles that could exceed conventional weapon capabilities

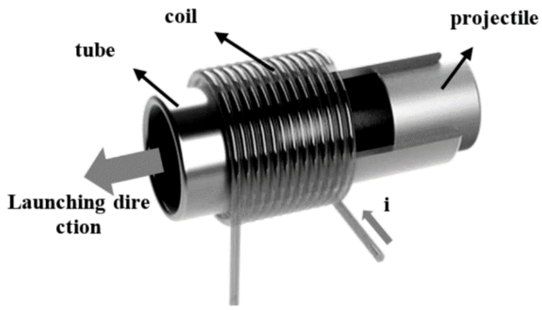
(Darnse & Singh, 2003; Dayi & Vijayan, 2017). The development of systems capable of achieving muzzle velocities exceeding 2000 m/s has been reported, although widespread deployment in military contexts may still be some time away (Darnse & Singh, 2003). The ongoing advancements in power electronics and energy storage technologies are likely to enhance the feasibility of these systems, making them more practical for real-world applications (Coşkun et al., 2010; Lee et al., 2013). In conclusion, the evolution of coil gun technology reflects a dynamic interplay between historical advancements and future possibilities. As research continues to refine designs, optimize performance, and explore new applications, coil guns are poised to become a significant component of modern propulsion systems across various fields.

The Working Principle of a Coil Gun

A coil gun, also referred to as a Gauss gun, is an electromagnetic launcher that utilizes a series of coils to accelerate a ferromagnetic or conductive projectile to high velocities. The device operates on the principles of electromagnetic induction, where the magnetic fields generated by electric currents flowing through the coils exert a force on the projectile. The primary mechanism involves the sequential activation of these coils, with each coil energized in a precisely timed manner to ensure continuous acceleration along the length of the launcher. Coil guns are predominantly designed as multistage systems, in which multiple coils are arranged in series along the barrel to maximize performance. At each stage, the coil closest to the projectile is activated momentarily, creating a strong magnetic field that pulls the projectile forward. Once the projectile reaches the center of the active coil, the current is switched off to prevent magnetic drag, and the next coil in sequence is energized. This process is repeated at carefully calibrated intervals, with each subsequent stage further accelerating the projectile. The multistage design offers several advantages, particularly by maintaining high acceleration over a longer distance, thereby increasing the projectile's terminal velocity. Furthermore, the efficient use of multiple stages reduces energy losses that typically occur in single-stage systems, leading to improved overall energy efficiency.

Figure 1 illustrates the structure of the coil used in each stage of the coil gun.

Figure 1
General Structure of Coil Gun Stages (Abdo et al. ,2023)



In Figure 1, the tube represents the barrel through which the projectile moves, while the coil refers to the wire, typically made of copper, that generates the electromagnetic force responsible for propelling the projectile forward. The coil creates a magnetic field that accelerates the projectile, enabling its movement through the barrel.

Components and Structure

A typical coil gun consists of several interdependent components that work together to achieve electromagnetic acceleration. At the core of the system are the coils, also known as solenoids, which are composed of tightly wound conductive wires. When an electric current flows through these coils, they generate magnetic fields capable of exerting force on a nearby ferromagnetic object. The projectile is usually made of materials such as iron or steel, which are strongly attracted to magnetic fields, allowing the coils to pull the projectile toward their center. To supply the necessary energy for rapid acceleration, the system relies on a power source, typically a set of capacitors or high-capacity batteries, which can discharge high-current pulses in short bursts.

The precise control of these pulses is critical for efficient operation, and this is managed by a switching system. The system typically includes sensors, such as photoelectric or Hall effect sensors, and control circuits that detect the position of the projectile and trigger the sequential activation and deactivation of the coils. This ensures that each coil is turned off before the projectile reaches its center to prevent deceleration, and that the next coil in sequence is energized at the right moment to continue accelerating the projectile. All of these components are housed within a barrel, which serves as a guiding structure, keeping the projectile aligned with the intended trajectory as it moves through the gun. The coordinated operation of these elements allows the coil gun to convert electrical energy into kinetic energy, propelling the projectile at high speeds.

Figure 2

Multi-Layer Coil Gun Base Components and Working Diagram (Lee et al. ,2013)

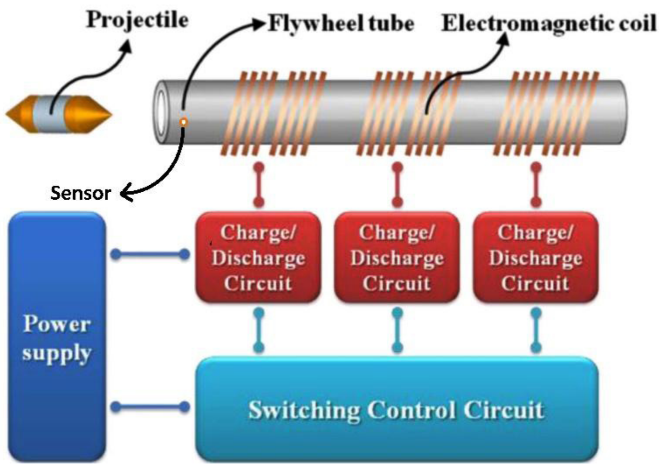
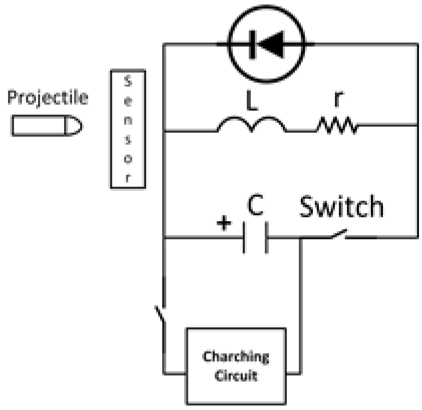


Figure 2 illustrates the structure of a multi-layered coil gun. In this figure, a single projectile is shown along with several key components: coils to generate a magnetic field, capacitor charging circuits, switching control circuits to produce an instantaneous magnetic field across the coils, and a power supply. Additionally, to detect the projectile's approach and activate the triggering mechanism, a sensor is positioned before each layer of the coil gun. This sensor setup enables precise timing and control, enhancing the efficiency of the magnetic propulsion as the projectile progresses through each stage.

Working principle of the coil gun

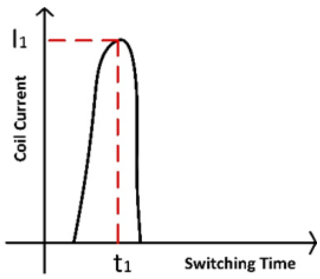
A classic coil gun comprises several key electrical components, including a coil, diode, charging circuit, switching element, and power supply. These elements work together to generate the magnetic force needed to propel a projectile through the coil gun's barrel. When the capacitor in the charging circuit discharges, the switching element rapidly directs current through the coil, creating a magnetic field that accelerates the projectile forward. The basic electrical circuit diagram of a coil gun is shown in Figure 3, illustrating how these components are integrated to achieve projectile motion.

Figure 3
The Electrical Circuit Used for Each Stage in a Coil Gun System



When examining the working principle of a coil gun, it becomes clear that as soon as the sensor detects the projectile, it generates a triggering signal for the switching circuit. This signal activates the solid-state switch within the switching circuit, transferring the energy stored in the capacitor to the coil. The instantaneous current generated through the coil creates a strong magnetic field, drawing the ferromagnetic projectile into the coil. Once the projectile moves past the sensor, the switching circuit cuts off the current passing through the coil, ending the magnetization process. This switching action results in an instantaneous current through the coil, as illustrated in Figure 4. This cycle is then repeated for each stage of the coil gun, with the next sensor triggering the corresponding switching circuit as the projectile progresses. By precisely timing each stage's activation, the coil gun enhances both acceleration and energy efficiency, achieving a powerful and controlled launch.

Figure 4
Coil Gun Coil Switching Timing



At time t1, the maximum current will flow through the coil, causing the highest level of magnetization to occur at this moment. If the projectile has passed through the center of the coil and the switch has not yet opened, the magnetic field within the coil will reverse, leading to a reversal of the magnetization. This causes the coil to exert a force on the projectile in the opposite direction, effectively attempting to decelerate or stop the projectile. The operation of the coil gun can be explained step by step, and this process can be broken down into four main stages for clarity. Each stage plays a crucial role in efficiently accelerating and directing the projectile, ensuring optimal performance of the system.

Magnetic Attraction and Acceleration

The operation of a coil gun relies heavily on the magnetic attraction generated by the coils. When an electric current passes through a coil, a magnetic field is created around it. If the coil is energized while the projectile is at a specific distance, the magnetic field attracts the projectile toward the coil. As the projectile approaches the center of the coil, the magnetic attraction increases and reaches its maximum at the coil's midpoint. This phase is critical because the magnetic force provides the primary acceleration that propels the projectile forward.However, if the projectile remains in the coil's magnetic field for too long, it may begin to slow down due to the symmetrical nature of the magnetic force. Therefore, to maintain efficient acceleration, careful control over the duration and timing of the coil's energization is required.

Deactivation of Coils to Prevent Deceleration

To prevent deceleration, the coil must be turned off just before the projectile reaches the coil's center. If the coil continues to be energized after the projectile crosses the midpoint, the magnetic force will act in the opposite direction, slowing the projectile instead of accelerating it. The precise deactivation of the coils at the right moment is essential to avoid this opposing force. This is achieved through sensors and control circuits that monitor the projectile's position in real-time. Sensors, such as Hall effect sensors, optical sensors, or infrared detectors, are positioned along the barrel to detect the projectile's approach to the coil. Once the sensors identify the optimal point, they trigger the control system to deactivate the coil. This step ensures that the projectile exits the coil with maximum speed, preparing it for the next stage of acceleration.

Sequential Activation for Continuous Acceleration

In multi-stage coil guns, several coils are arranged along the barrel in sequence to provide continuous acceleration throughout the launcher. As the projectile moves through the first coil and reaches maximum speed, the subsequent coil is energized to further accelerate the projectile. This process repeats across multiple coils, with each stage adding to the kinetic energy of the projectile. The synchronized activation of coils is vital to maintain a smooth transfer of energy between stages. If the coils are not activated in the correct sequence, the projectile may lose momentum or experience irregular motion. The control system ensures that each coil is activated precisely when the projectile reaches the ideal point, providing uninterrupted acceleration throughout the length of the barrel.

Energy Storage and Discharge

Coil guns typically rely on capacitors to store the required electrical energy, as they are capable of releasing high currents in short bursts. These capacitors accumulate energy over time and discharge it rapidly when needed, producing intense magnetic fields. This quick release of energy is essential for generating the powerful magnetic forces required to accelerate the projectile efficiently. In some advanced systems, multiple capacitors are used in combination to form capacitor banks, ensuring that sufficient energy is available for each stage of the coil gun. Additionally, the discharge process must be precisely timed to coincide with the projectile's position within the coil. Any delay or misfire could result in a loss of energy, reducing the overall efficiency of the system. Some coil guns also incorporate energy recovery mechanisms to improve efficiency. For instance, the back electromotive force (back EMF) generated when coils are turned off can be redirected back into the capacitors, allowing for partial energy reuse. Proper management of this back EMF is crucial to prevent it from interfering with the operation of the control circuits. The efficient operation of a coil gun relies on the careful coordination of magnetic attraction, precise deactivation of coils, and synchronized multi-stage activation, allowing for the effective conversion of electrical energy into kinetic energy. Each stage of acceleration requires meticulous timing to ensure that the magnetic field reaches its peak when the projectile is optimally positioned within the coil. To avoid deceleration, the coil must be deactivated just before the projectile reaches the midpoint, as any delay would result in opposing forces. In multi-stage systems, multiple coils are arranged sequentially along the barrel, with each coil activating at the precise moment the projectile reaches it, ensuring smooth energy transfer and continuous acceleration. Capacitors play a crucial role by storing and rapidly releasing electrical energy in high-current bursts, aligning with the projectile's movement to generate intense magnetic fields. Advanced systems may use capacitor banks to ensure sufficient energy availability, while energy recovery mechanisms, such as the reuse of back electromotive force (back EMF), further improve efficiency. Performance optimization also requires addressing practical challenges related to heat dissipation, power stability, and material selection. The rapid discharge of high currents generates significant heat, which, if unmanaged, can impair performance or damage components. To prevent this, advanced designs incorporate cooling systems or heat sinks. Additionally, the continuous charge-discharge cycles of the capacitors demand a stable and reliable power supply to ensure consistent operation. The material properties of the projectile are equally important, as high magnetic permeability is needed for maximum attraction, while a lightweight structure helps achieve higher velocities. In some designs, non-magnetic or conductive projectiles are used with customized coil configurations to meet specific performance requirements. By effectively managing magnetic forces, optimizing energy storage, and incorporating advanced control mechanisms, modern coil guns push the boundaries of electromagnetic acceleration, offering a robust platform for both research and practical applications.

Coil Guns Current and Future Perspectives

Coil guns, classified as gunpowder-free weaponry, continue to be developed and refined in various research and experimental contexts. They offer distinct advantages, including low projectile costs, extended barrel life, and the ability to increase muzzle velocity by adding multiple stages along the barrel. Currently, coil guns are predominantly used in experimental and demonstration settings, such as scientific research, non-lethal defense applications, and educational laboratories, where they serve as practical examples of electromagnetic propulsion principles. For instance, their deployment in scientific environments showcases their capacity for precision and controlled acceleration, while educational demonstrations highlight their potential as a non-traditional propulsion system. Given these attributes, coil guns are seen as promising candidates for future applications in defense, industrial material processing, and even space exploration. Their potential to revolutionize various fields continues to grow as new research in materials, energy efficiency, and switching technologies advances.

Current Status of Coil Guns

Coil guns, also known as electromagnetic guns or Gauss rifles, have shown significant potential in various fields, including military applications and scientific research. Currently, their use is primarily limited to experimental setups and educational environments, where they effectively demonstrate the core principles of electromagnetism. These devices function by using electromagnetic coils to accelerate ferromagnetic projectiles, eliminating the need for traditional propellants like gunpowder. Recent progress in coil gun technology has focused on improving the design of electromagnetic coils, capacitor banks, and switching circuits. Innovations in high-energy capacitors, more efficient switching mechanisms, and better cooling techniques have led to improved system performance. Despite these advances, coil guns are still largely restricted to controlled settings for non-lethal defense, scientific demonstrations, and hobbyist applications, with limited adoption in commercial or military settings. There are still a few challenges that need to be addressed to unlock the full potential of coil guns, particularly in terms of energy efficiency, scalability, and practical implementation. A key issue is the high energy consumption required for larger projectiles and higher velocities. Furthermore, achieving precise control over projectile acceleration and targeting remains a significant engineering challenge.

Future Prospects of Coil Guns

The future of coil guns offers intriguing possibilities, with the potential to revolutionize numerous sectors, particularly in defense, space exploration, and medical technologies. As advancements in coil gun design continue to address current limitations—such as energy efficiency, precision control, and scalability their application scope is expected to broaden significantly. These future applications of coil guns can be categorized into four main domains, each representing a promising area of impact where electromagnetic acceleration technology could provide substantial advantages.

Military and Defense

The potential for coil guns to impact military and defense applications is significant, particularly in naval and space defense systems. The advantage of launching projectiles without the use of explosive propellants offers safety benefits, such as reduced risks related to weapon storage and transportation. Additionally, coil guns may be adapted for kinetic energy weapons, potentially replacing conventional firearms for non-lethal crowd control or enabling more precise targeting in combat situations.

Space Exploration

One of the most promising applications of coil gun technology lies in space exploration. Unlike traditional rocket propulsion systems, which rely heavily on fuel, coil guns could provide a more energy-efficient means of launching payloads into orbit. This innovation could drastically reduce the cost of space missions and facilitate the transport of materials for space stations, satellites, or even lunar or Martian colonies.

Energy and Efficiency Enhancements

Future advancements in energy storage technologies will likely lead to significant improvements in the efficiency of coil guns. Innovations in high-density energy storage systems, such as superconducting magnetic energy storage (SMES) and advanced capacitors, could enable the development of coil guns capable of launching heavier payloads at greater speeds and with enhanced accuracy. Furthermore, advancements in power electronics may minimize energy losses during operation, further improving system efficiency.

Medical Applications

The principles underlying coil gun technology are also being explored for medical purposes, particularly in drug delivery and tissue repair. Electromagnetic propulsion could enable precise, non-invasive delivery of medical capsules or micro-robots to specific locations within the human body. This application holds potential for minimally invasive procedures, targeted therapies, and improved patient outcomes, while also reducing recovery times.

Conclusion

In summary, although coil gun technology remains in the early stages of development, its potential applications are extensive and span multiple fields. With continued advancements in materials science, energy storage, and electronic control systems, coil guns are expected to play an increasingly significant role in various industries. As innovations in high-capacity capacitors, efficient switching circuits, and durable magnetic materials progress, the practical capabilities of coil guns will likely expand, making them viable for an even broader array of uses. Ultimately, coil guns stand on the cusp of pushing the limits of electromagnetic propulsion, offering transformative and sustainable possibilities across defense, space exploration, and scientific research, among other applications.

Reference

Abdo, M. (2023). Design of a new electromagnetic launcher based on the magnetic reluctance control for the propulsion of aircraft-mounted microsatellites. Applied System Innovation, 6(5), 81. https://doi.org/10.3390/asi6050081

Abdo, M. M. M., El-Hussieny, H., Miyashita, T., & Ahmed, S. M. (2023). Design of A New Electromagnetic Launcher Based on the Magnetic Reluctance Control for the Propulsion of Aircraft-Mounted Microsatellites. Applied System

Innovation, 6(5), 81.

Abdo, T., Elrefai, A., Adly, A., & Mahgoub, O. (2016). Performance analysis of coil-gun electromagnetic launcher using a finite element coupled model. https://doi.org/10.1109/mepcon.2016.7836938

Coşkun, İ., Kalender, O., Ege, Y., & Nazlibilek, S. (2010). Optimization of parameters acting on a projectile velocity within a four stage induction coil-gun. Measurement, 43(10), 1656-1660. https://doi.org/10.1016/j.measurement.2010.09.020

Darnse, R. and Singh, A. (2003). Advanced concepts of the propulsion system for the futuristic gun ammunition. Defence Science Journal, 53(4), 341-350. https://doi.org/10.14429/dsj.53.2279

Dayi, T. and Vijayan, A. (2017). Design and simulation of an electromagnetic launcher for ordnance applications. International Journal of Engineering and Technology, 9(4), 3183-3190. https://doi.org/10.21817/ijet/2017/v9i4/170904127

Domin, J. (2021). Laser measurement system dedicated to the determination of rails deformation of a hybrid electromagnetic launcher. International Journal of Applied Electromagnetics and Mechanics, 64, S191-S202. https://doi.org/10.3233/jae-209515

Giès, V. and Soriano, T. (2019). Modeling and optimization of an indirect coil gun for launching non-magnetic projectiles. Actuators, 8(2), 39. https://doi.org/10.3390/act8020039

Giès, V., Soriano, T., Marzetti, S., Barchasz, V., Barthélémy, H., Glotin, H., & Hugel, V. (2020). Optimisation of energy transfer in reluctance coil guns: application to soccer ball launchers. Applied Sciences, 10(9), 3137. https://doi.org/10.3390/app10093137

Lee, S., Kim, J., Song, B., & Kim, J. (2013). Coil gun electromagnetic launcher (eml) system with multi-stage electromagnetic coils. Journal of Magnetics, 18(4), 481-486. https://doi.org/10.4283/jmag.2013.18.4.481

Liu, S., Ruan, J., Huang, D., & Wan, Z. (2009). Analysis of inductive coil gun performance based on field coupling circuit method. https://doi.org/10.1109/ipemc.2009.5157502

About the Authors

Tarik ÜNLER, PhD, is an Assistant Professor of Aviation Electrical and Electronic at Necmettin Erbakan University in Konya, Türkiye. He holds a PhD in Electric-Electronic Engineering from Konya Technical University. His main areas of interest are embedded systems, autonomous systems, electronics, image processing RADAR and aviation applications.
E-mail: tunler@erbakan.edu.tr, **ORCID:** 0000-0002-2658-1902

Mehmet DURSUN, PhD, is an Assistant Professor of Aviation Electrical and Electronic at Necmettin Erbakan University in Konya, Türkiye. He holds a PhD in Electric-Electronic Engineering from Selçuk University. His main areas of interest are signal processing, image processing, electronics and fiber optics communication systems.
E-mail: mehmet.dursun@erbakan.edu.tr, **ORCID:** 0000-0002-0558-6309

Similarity Index
The similarity index obtained from the plagiarism software for this book chapter is 15%.

Chapter 9

Use of UAV in Agriculture

Mehmet DURSUN
Necmettin Erbakan University

Tarık ÜNLER
Necmettin Erbakan University

To Cite This Chapter

Dursun, M., & Ünler, T. (2024). Use of UAV in Agriculture. In S. Koçer & Ö. Dündar (Eds.), *Intelligent Systems and Optimization in Engineering* (pp. 106-114). ISRES Publishing.

Introduction

Unmanned aerial vehicles (UAVs) are autonomous aircraft capable of executing missions planned before takeoff or adapting to various scenarios based on evolving conditions during flight, all controlled and commanded from a remote location, even though they do not physically carry a pilot. These vehicles are generally referred to as "drones."

In contemporary contexts, UAVs are widely used in both civilian and military sectors. Although initially developed for military purposes, UAVs have found extensive application in civilian life, especially in recent years. UAVs are considered platforms onto which integrated systems, equipped with the latest technologies, have begun to be mounted, allowing them to discover new areas of application day by day.

This study examines the widespread use of UAVs in civilian areas, focusing particularly on the agricultural sector.

History of Unmanned Aerial Vehicles

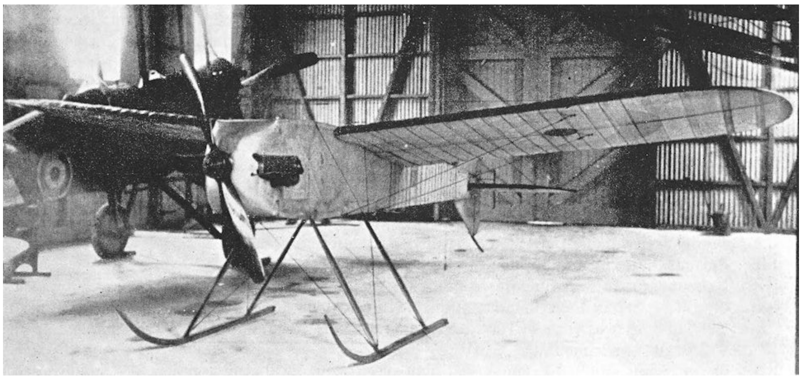
Unmanned aerial vehicles (UAVs) first emerged in the late 19th and early 20th centuries. The earliest examples of UAVs were, in fact, flying balloons. Although these balloons, armed with bombs, were directed toward specific targets with the aim of destroying them, they often failed to reach their intended targets due to uncontrollable weather conditions. Therefore, these early aerial vehicles differ significantly from today's UAVs (Stamp & Jeffrey, 2015).

The first examples of structures similar to today's UAVs began to appear during World War I. At that time, UAVs were used to gain superiority over enemy forces on the battlefield. These initial models lacked any control mechanisms and were launched from a ramp toward a designated target with the intent of destroying it. Although more effective than balloons, they often failed to achieve the desired effect due to difficulties in controlling them (Zaloga, 2011).

Figure 1
V-1 Flying Bomb

In subsequent years, advancements in electronics enabled the remote control of UAVs for the first time. This made it possible to control UAVs from a distance within certain limits (Flying Machines, 2024).

Figure 2
The First Radio-Controlled Aircraft: Aerial Target.



In the following years, UAVs equipped with mounted camera systems were used for reconnaissance and surveillance, transforming them into aerial vehicles that provide critical intelligence information.

Figure 3
Tadiran Mastiff UAV, which made its first flight in 1970

Since the early 21st century, UAVs have evolved into systems capable of remaining airborne for extended periods, enabling communication over long distances, minimizing radar visibility, and executing successful operations with advanced camera systems and munitions mounted on them (Baykar Tech, 2024).

Figure 4
Bayraktar Akıncı UCAV (Baykar Tech, 2024).

The Use of Unmanned Aerial Vehicles in Agriculture

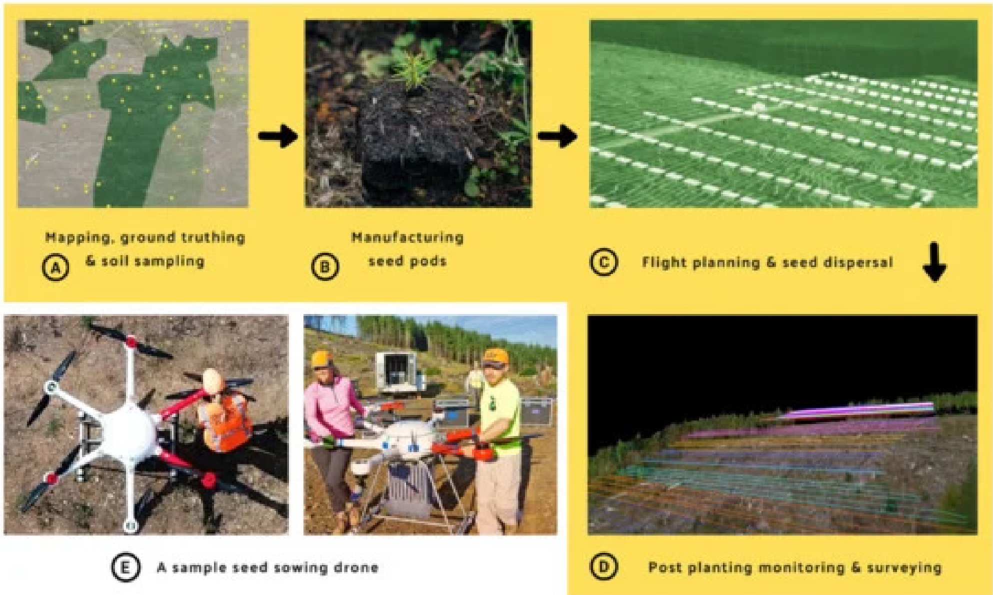
UAV Use in Seed and Seedling Planting

Today, the increase in human population, industrialization, rising consumption, and the lack of necessary measures in waste management have caused significant damage to ecosystems. Urbanization and the establishment of new residential areas, particularly in cultivated lands, along with intentional or accidental forest fires, have considerably reduced vegetation cover. Preserving forested areas and expanding their surface area are therefore essential for mitigating these negative impacts. Although these planting activities are typically conducted by the state or through public efforts organized by non-governmental organizations, planting in unsafe or hard-to-reach areas can sometimes be unfeasible. At this stage, the use of UAVs has recently made it possible to survey vegetation diversity, identify areas suitable for planting, and conduct aerial seeding to develop and expand forested areas (Mohan et al., 2021).

Figure 5
UAV-Assisted Seed Planting Process

Additionally, with the help of UAVs, low-altitude imaging and measurements over cultivated areas allow for the differentiation of planted seed and seedling types, as well as monitoring their growth. If any issues are detected at this stage, necessary measures can be taken within response timeframes to address potential problems (Buters, Belton & Cross, 2019).

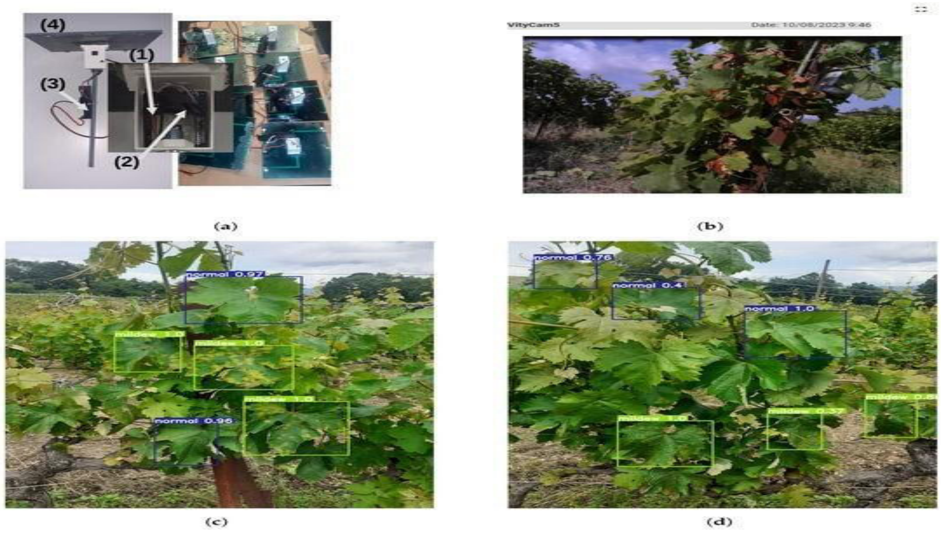
Figure 6
UAV Seeding and Seed Planting Processes Assisted by UAVs (Courtesy of DroneSeed



UAV Use in Detecting Crop Pests and Plant Diseases

In agricultural practices, crop yield can decrease, and plants may even be lost during the flowering and fruiting stages due to pests and diseases. Monitoring and identifying these issues early for timely intervention can increase yield and reduce production costs (Kılıç & Karakoyun, 2023). For this purpose, UAVs equipped with thermal, hyperspectral, and high-resolution cameras and sensors make it possible to sustainably monitor plant growth and health (Kontogiannis et al., 2024).

Figure 7
Plant Image Captured with UAV Assistance; System for Detecting Healthy and Diseased Plant Leaves (Kontogiannis et al., 2024).



UAV Use in Mapping

In sustainable agriculture, controlling weeds around plants is as important as managing plant diseases. The success of this control depends on the accurate and timely mapping of weeds, allowing weed control efforts to be conducted with minimal time, chemical usage, and cost. For this mapping process, UAVs are equipped with high-resolution cameras as well as multispectral cameras. These systems enable accurate classification and detection of weed species, facilitating effective mapping. The image below illustrates weed detection in a cornfield based on camera footage from a UAV (Gao et al., 2024).

Figure 8
Plant Image Captured with UAV Assistance; Detection of Weeds (Gao et al., 2024).

UAV Use in Pesticide and Fertilizer Spraying

In crop cultivation, combating harmful factors that damage plants is essential for increasing yield and reducing economic losses. Chemical control is the most widely used method in plant protection (Demir, 2015).

While mechanical sprayers have made it easier to control and eliminate these pests, unplanned and indiscriminate use of chemicals can harm plants, beneficial organisms around them, and the environment, leading to ecosystem disruption. Additionally, unregulated chemical application increases production costs and, most critically, endangers human health (Demir, 2015).

At this stage, UAV-integrated spraying systems enable targeted application of chemicals from low altitudes, minimizing environmental dispersal and chemical usage. UAVs can identify specific areas that require treatment, ensuring that chemicals are applied only where necessary. This method reduces the amount of chemicals used, thereby lowering spraying and production costs. Minimizing chemical release into nature also helps preserve the environment (Wei et al., 2024).

Moreover, UAVs can distribute fertilizer—crucial for plant growth—uniformly across cultivated areas. Similar to spraying, UAVs apply fertilizer within predefined boundaries directly onto plants, allowing for maximum yield with minimal fertilizer use. This targeted fertilization approach significantly reduces production costs. In Figure 9, trees were detected on the image taken from the UAV in a garden with apple trees.

Figure 9
Identification of Spraying Areas in an Orchard of Apple Trees Using UAV Assistance (Wei et al., 2024).

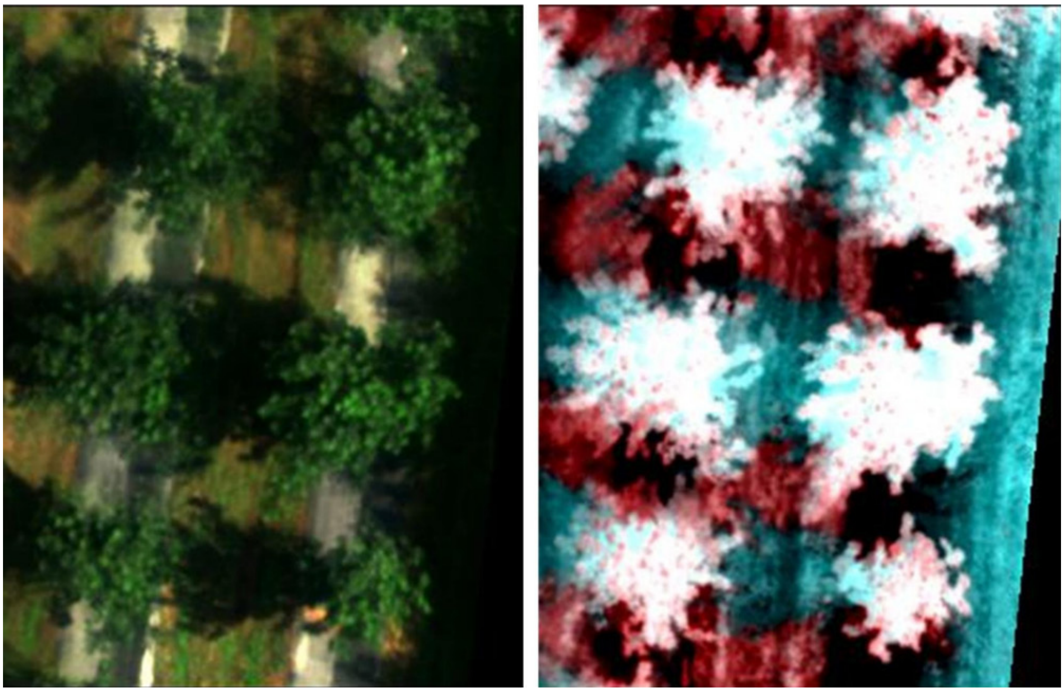
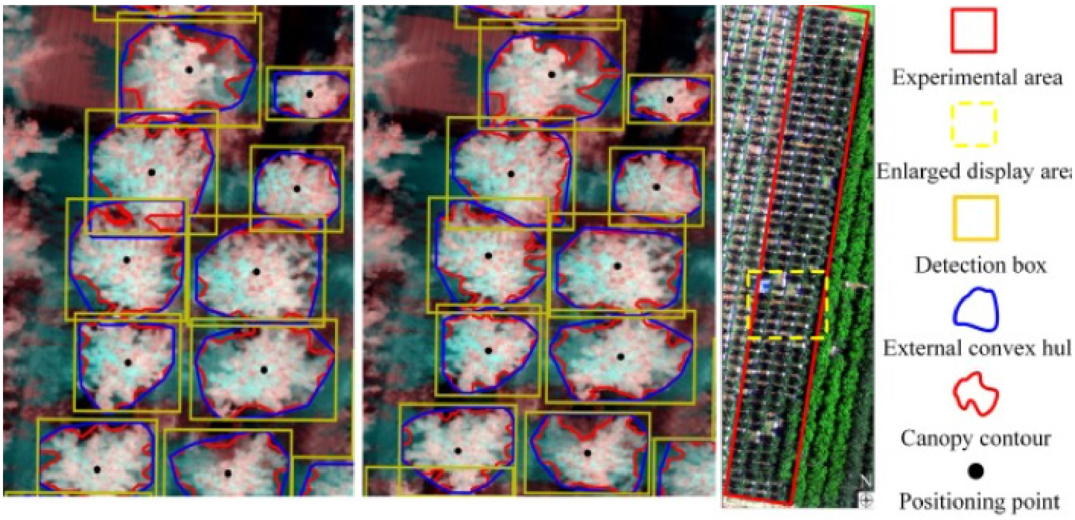
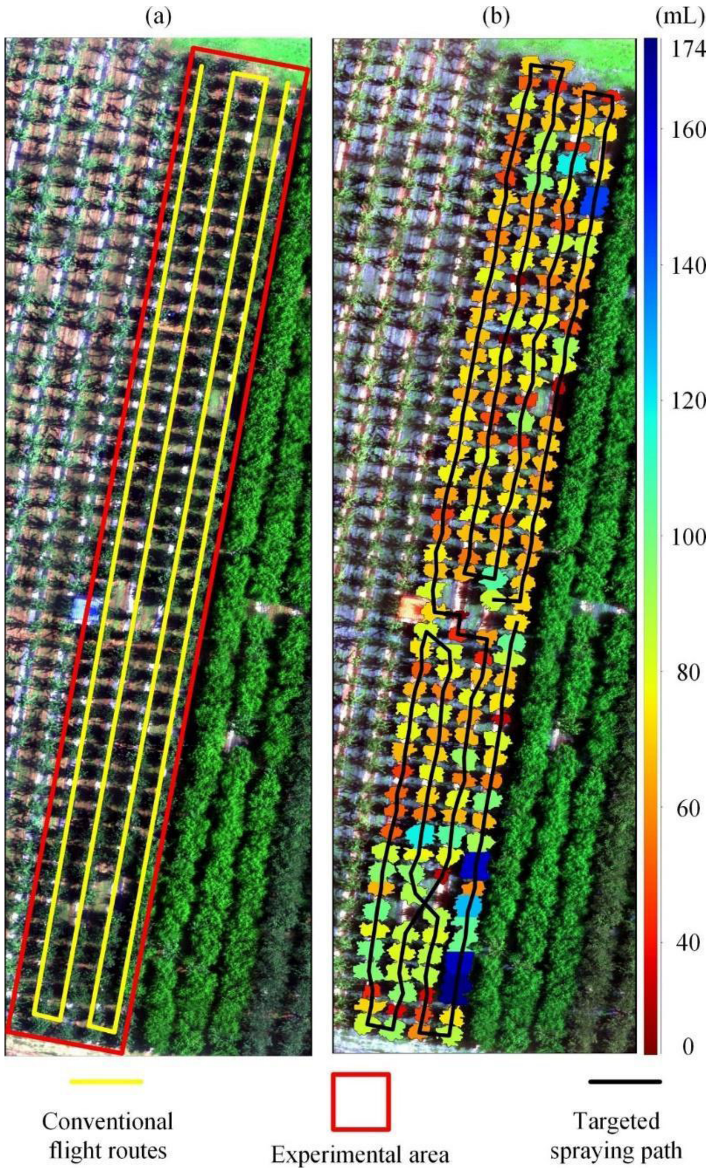


Figure 10
Identification of UAV Positions for Spraying Areas in an Orchard of Apple Trees Using UAV Assistance (Wei et al., 2024).



In Figure 10, the areas where the UAV will be positioned for spraying are identified. Finally, in Figure 11, the shortest flight path that the UAV will follow for spraying the entire area is planned. (a) shows the traditional flight path, while (b) illustrates the shortest path determined for spraying based on the study results.

Figure 11
Planning the Shortest Flight Path for UAV Spraying in an Orchard of Apple Trees (Wei et al., 2024).



UAV Use in Crop Yield and Harvest Prediction

Regular monitoring of plant growth and the ability to accurately predict crop yield and harvest in advance play a crucial role in management and the control and regulation of production stages. Predictions made using traditional methods may differ from actual values. Recently, UAV-based remote sensing systems have been used to predict yield. During key growth stages, such as flowering and fruiting, images captured by UAV cameras are processed using various image processing techniques and classifiers to estimate yield. Studies have shown that yield predictions based on UAV-derived images can accurately reflect actual production values (Feng et al., 2020; Shvorov et al., 2020).

Conclusion

Although UAVs were initially developed for military purposes and their advancements have largely been in this field, they are now extensively used in civilian sectors. One of the most common applications in these civilian areas is agriculture. UAVs are used across a wide range of tasks, from seed planting and plant protection to species identification and yield prediction. The benefits gained from these applications make a significant contribution to sustainable, efficient, and low-cost production.

References

Buters, T., Belton, D., & Cross, A. (2019). Seed and seedling detection using unmanned aerial vehicles and automated image classification in the monitoring of ecological recovery. *Drones*, *3*(3), 53.

Demir, B. (2015). İç Anadolu bölgesinin bitki koruma makineleri projeksiyonu. *Alinteri Journal of Agriculture Science*, *28*(1), 27-32.

Feng, A., Zhou, J., Vories, E. D., Sudduth, K. A., & Zhang, M. (2020). Yield estimation in cotton using UAV-based multi-sensor imagery. *Biosystems Engineering*, *193*, 101-114.

Gao, J., Liao, W., Nuyttens, D., Lootens, P., Xue, W., Alexandersson, E., & Pieters, J. (2024). Cross-domain transfer learning for weed segmentation and mapping in precision farming using ground and UAV images. *Expert Systems with applications*, *246*, 122980.

https://baykartech.com/tr/insansiz-hava-araci-sistemleri/

https://flyingmachines.ru/Site2/Arts/Art62254.htm

https://www.xagaustralia.com.au/aerial-seeding

Kilic A.E., Karakoyun M., (2023). Plant Diseases Detection Using Explainable Deep LearningMethods and Its Applicability in Real Life. In S. Kocer. & O. Dundar (Eds.), *Artificial Intelligence Applications in Intelligent Systems* (pp. 1–15). ISRES Publishing.

Kontogiannis, S., Konstantinidou, M., Tsioukas, V., & Pikridas, C. (2024). A Cloud-Based Deep Learning Framework for Downy Mildew Detection in Viticulture Using Real-Time Image Acquisition from Embedded Devices and Drones. *Information*, *15*(4), 178.

Mohan, M., Richardson, G., Gopan, G., Aghai, M. M., Bajaj, S., Galgamuwa, G. P., ... & Cardil, A. (2021). UAV-supported forest regeneration: Current trends, challenges and implications. *Remote Sensing*, *13*(13), 2596.

Shvorov, S., Lysenko, V., Pasichnyk, N., Opryshko, O., Komarchuk, D., Rosamakha, Y., ... & Martsyfei, A. (2020, February). The method of determining the amount of yield based on the results of remote sensing obtained using UAV on the example of wheat. In *2020 IEEE 15th International Conference on Advanced Trends in Radioelectronics, Telecommunications and Computer Engineering (TCSET)* (pp. 245-248). IEEE.

Stamp, Jeffrey (2015). “Aero-Statik Savaş: Ondokuzuncu Yüzyılın Ortalarındaki Kuşatma Savaşlarında Balonculuğun Kısa Bir İncelemesi”

Wei, P., Yan, X., Yan, W., Sun, L., Xu, J., & Yuan, H. (2024). Precise extraction of targeted apple tree canopy with YOLO-Fi model for advanced UAV spraying plans. *Computers and Electronics in Agriculture*, *226*, 109425.

Zaloga, S. J. (2011). *V-1 Flying Bomb 1942–52: Hitler's infamous "doodlebug"* (Vol. 106). Bloomsbury Publishing.

About the Authors

Mehmet DURSUN, PhD, is an Assistant Professor of Aviation Electrical and Electronic at Necmettin Erbakan University in Konya, Türkiye. He holds a PhD in Electric-Electronic Engineering from Selçuk University. His main areas of interest are signal processing, image processing, electronics and fiber optics communication systems.
E-**mail:** mehmet.dursun@erbakan.edu.tr, **ORCID:** 0000-0002-0558-6309

Tarik ÜNLER, PhD, is an Assistant Professor of Aviation Electrical and Electronic at Necmettin Erbakan University in Konya, Türkiye. He holds a PhD in Electric-Electronic Engineering from Konya Technical University. His main areas of interest are embedded systems, autonomous systems, RADAR and aviation applications.
E-mail: tunler@erbakan.edu.tr , **ORCID:** 0000-0002-2658-1902

Similarity Index
The similarity index obtained from the plagiarism software for this book chapter is 19 %.

Texture analysis for Mobiles Phones in Data Compression

Sabri KOÇER

Necmettin Erbakan University

Özgür DÜNDAR

Necmettin Erbakan University

To Cite This Chapter

Koçer, S., & Dündar, Ö. (2024). Texture analysis for Mobiles Phones in Data Compression. In S. Koçer & Ö. Dündar (Eds.), *Intelligent Systems and Optimization in Engineering* (pp. 115-126). ISRES Publishing.

Introduction

Storing and transmitting information are frequently important issues for businesses, states and different organizations. These organizations can handle additional information by packaging it, which likewise reduces the expected space and cost for storage. In the event that you record or send information as part of your responsibilities, it can be valuable to be aware of compression capabilities and the advantages it can offer you and your association (Kocer vd., 2022).

Compression for mobile phones is the most important area because we take care of almost all our needs over the phone. 2D, 3D images are used and in graphics 2D, 3D images can be called textures with different compression methods than normal images. Mostly all techniques are enhanced for human eyes brightness, not only the color in texture compression for mobile phones.

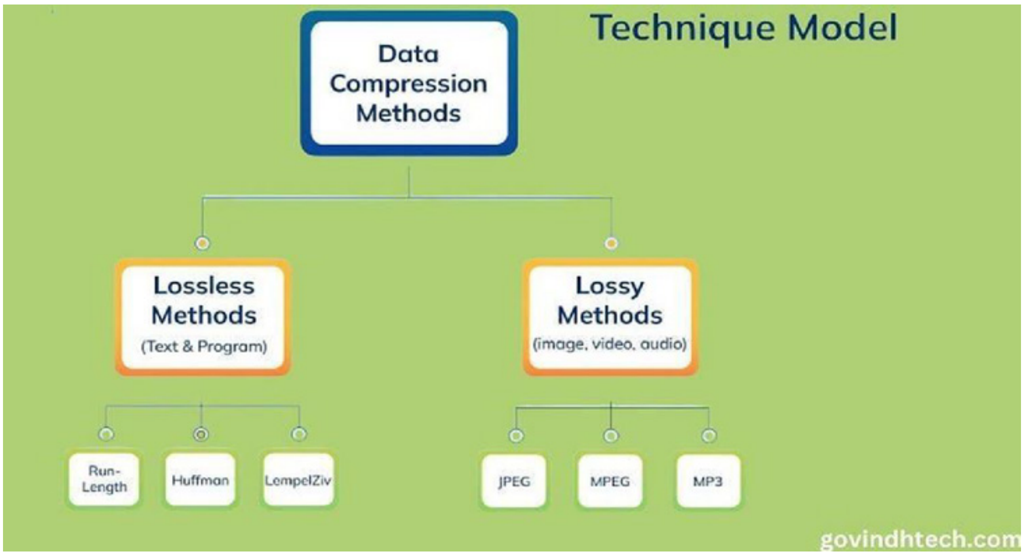
Nowadays when everything becomes computerized, the need for storage of huge amount of data becomes more crucial. The problem is that even if the capacity of storage increases the demand for extra storage will keep growing even twice (Jayasankar, 2021). Therefore, in order to deal with possible storage problems some data compression methods were introduced. The main thing to know before using some compression method is the level of compression the one wants to apply which is divided into categories such as Tuple level, Page level, Column Level, Block Level and Table Level compression. The data compression started its beginning from the compression of messages in telegraphs by Morse Code which was taken as origin for Huffman coding (Anuradha, D.,2016, Said, A.2003, Moffat, A., 2019)

Data Compression Techniques

Usually, compression methods based on data quality are classified as lossless and lossy. However, a hybrid type of compression which is called near-lossless compression exists in practice. Usually, Lossy compression methods are not used in traditional database systems but only at the application level, since in database systems both data integrity and accuracy are principal because when the data loss happens the database can't deduce what has to be thrown away and what does not (Pavlo, 2024). Nevertheless, lossy compression still can be applied in particular database-related contexts. In NearLossless technique the difference between the original and decompressed data differ by no more than a precise amount known as maximum absolute distortion (MAD) in Fig 1

Figure.1

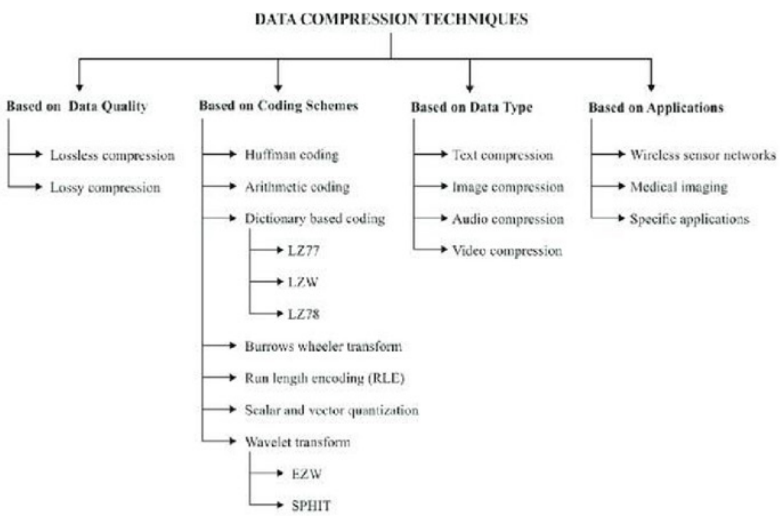
Data Compression Techniques' main Classification (Khalid A, 2006)



Depending on what is needed to be compressed in database system, distinct compression schemes can be applied (Sayood, K., 2017). As was stated in abstract part of this review paper the main rule is to have some goals such as producing fixed-length values which means to have the predetermined length of compressed output regardless of the original size or complexity of the input data which is crucial for Storage Efficiency, Simplified Data Access, Improved Query Performance in Fig 2. Another goal is to have a lossless scheme which means that input data and output data have to correspond to each other. In addition, another solution for improving query performance is decompressing data until necessary during query execution to reduce I/O operations, optimize memory and CPU usage (Salomon, D. 2007).

Figure 2

Data Compression Techniques (Jayasankar, 2021).



Levels of compression

As discussed before there are several compression levels in database systems. Compression granularity refers to the size of data to be compressed in the database and can vary depending on specific requirements (Jayasankar, 2021).

Tuple or row level compression is compression which covers individual rows in database. It is usually selected when there are rows that have repetitive terms. Besides inconsiderable performance impact, another positive thing about it is the fact that when the data is updated in row level it is much simpler than decompression and recompression of higher-level data compression.

Page level compression is held on the database pages the size of each is $2n$KB usually 8KB. Pages are categorized into 3 types Data Pages, Index pages, Overflow pages. When a table is created each row is stored in tables. Here the balance between performance and effective storage usage is achieved. Since This level of compression covers more data than tuple level it is easier to reduce redundancy (Severance, D. G. 1983).

Block level compression is the technique when database management system compresses the data before transferring to the disc and decompress when there is need to read it. It stands for reducing I/O operations and saving space for more data to be stored in physical space. Depending on different factors and DBMS capabilities, algorithms such as LZ77, Huffman, RLE can be applied to blocks.

Columnar level compression is another level of compression when the data is kept in the form of columns. It is effective when columns have similar items and algorithms such as RLE, dictionary Encoding are good for achieving great ratio of compression.

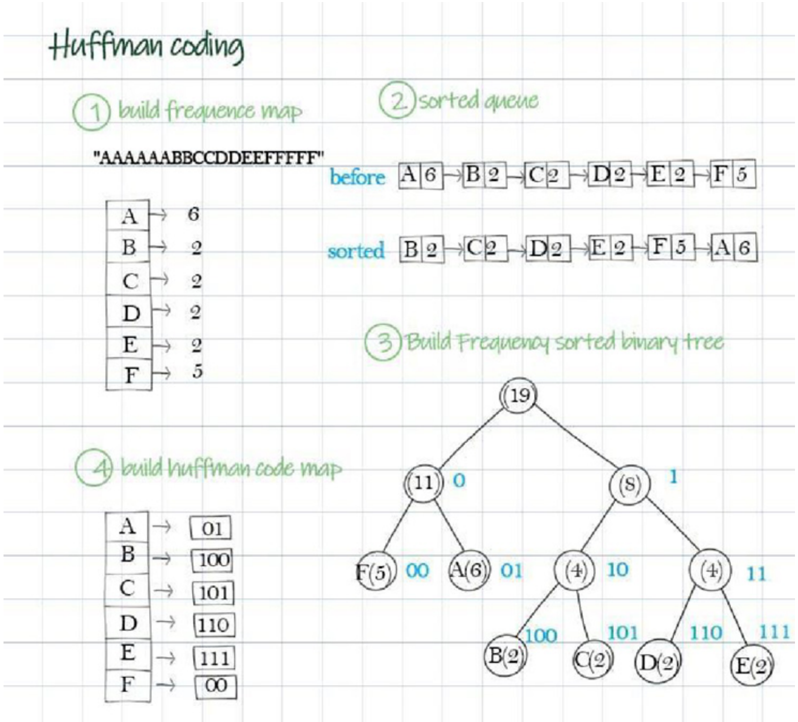
Tablespace level compression is the most significant storage reduction method which ameliorates performance, since tablespace contains all the tables, indexes and other objects.

Huffman Coding

Huffman Coding is one of the essential techniques used in lossless compression methods first introduced in 1952 (Moffat, A. 2019). Even so, its simplicity continues to make it popular. primary concept involves allocating tables with codes according to each character's frequency of appearance, which is determined by $log\ p$, where p is the probability of occurrence, that’s why shorter codes represent most frequently appearing character and are on the upper branch of the binary tree while longer codes identify less frequently seen characters in Fig 3.

In the creation of tree, the thing that should be considered is the fact that elements with high frequency have to stand on the upper branches. And starting from the root node which is the overall number of all occurrences of all elements in the original data, left and right branches represent 0s and 1s and this process keep going until leaf node. Eventually Huffman code map is built.

Figure 3
Example Of Huffman Coding (Lavivienpost,2024)

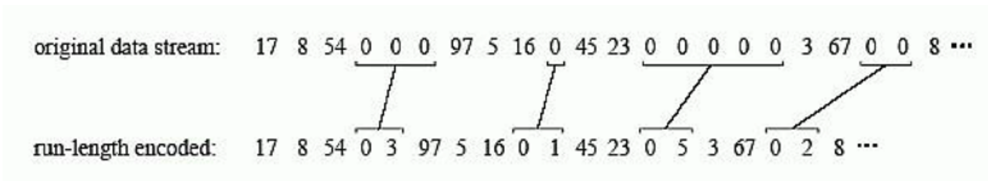


There are multiple variants of Huffman coding, including Minimum Variance Huffman Code, Length-Limited Huffman Code, Non-Binary Huffman Code, Canonical Huffman Code Adaptive Huffman Code and others. Many compression algorithms, such as Deflate, JPEG, and MP3, utilize Huffman coding as their underlying technique (Pigeon, S., 2003).

Run Length Encoding

One of the simplest forms of compression is Run Length Encoding (Fig 4) which was found by Capon in 1959. It is used particularly in cases when the data brings significant redundancy and repetitive, otherwise it can even increase the size of the compressed file. In database systems it is applicable in columnar database where for instance column identifying the gender of people and terms female and male repeat (Severance, D. G. 1983).

Figure 4
Run Length Encoding

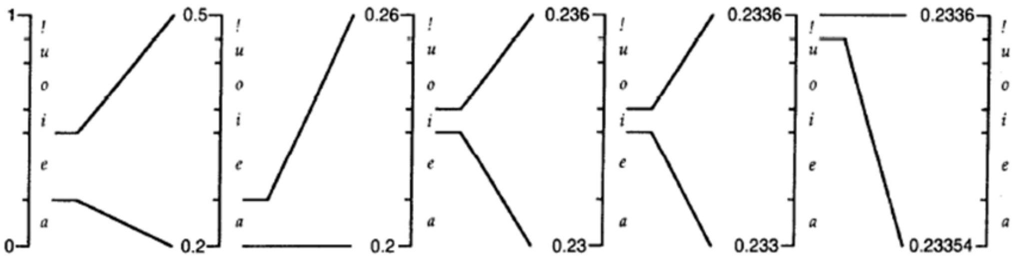


Arithmetic Coding

Arithmetic coding, introduced by Langdon in 1984, is a notable technique and differs from traditional coding methods since it represents an entire message as a single number that is a fraction between 0 and 1 which allows compression rates to approach the theoretical limits defined by the entropy of the input data in Fig 5. Despite its high effectiveness, it is more challenging to implement than some other methods. Arithmetic coding is especially advantageous over Huffman coding when handling sources with small alphabets and imbalanced probability distributions. A key advantage of arithmetic coding is its ability to separate the modeling and coding phases of the compression process, providing greater flexibility and efficiency. Here with increasing the frequency of symbol the bits encoded are decreasing. Arithmetic Coding is categorized in two types – adaptive and binary (Said, A. 2023).

Figure 5

Arithmetic coding (Witten,1986)



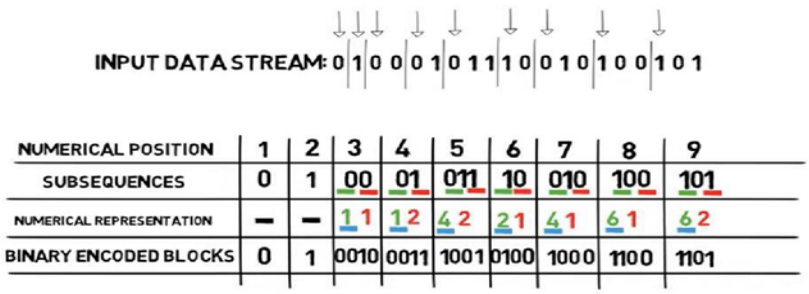
Dictionary based coding

Dictionary based coding is used when we deal with long text or any other file format with repetitive components inside it. This method uses a dictionary where most frequently sequences of characters are kept. The stored sequences now have indexes and each time when new character or sequence of characters which has to be encoded is the same as the one in the dictionary encoder indexes them from dictionary in Fig 6. Otherwise, when the character is not repetitive the character or sequence of characters is sent in uncompressed form to dictionary. The dictionary exists in two forms-static which is built before process of encoding and is left unchanged, meanwhile the dynamic dictionary is adjustable can change in the process, leading to enhanced compression ratios. Also, dictionaries's scope can be chosen I block level, table level and multi table level or so-called database level. One of the most renowned and effective methods for larger files is Lempel–Ziv algorithm (LZ) is lossless compression method which replaces frequent characters by indexes in the dictionary (Das, D, 2005).

It has two versions developed in 1977 and 1978, first employs a sliding window technique to look for matches within a specified range of past data from the current position. It encodes sequences by referencing earlier occurrences within the window, using (distance, length) pairs, occasionally including a literal character. For instance, Open-source MySQL employs the LZ77 algorithm for compressing its InnoDB tables.

Figure 6

Example of Lempel Ziv Compression Method



LZ78 on the other hand, uses a dictionary-based approach. Here characters or ssequence of characters are appended to the dictionary dynamically and as an output references to the built dictionary created. Lempel–Ziv-Welch (LZW) is the most recent version which is used very often and developed by Terry Welch inspired by LZ77 and LZ78 (Subathra, S.,2005).

Brief History of Texture Compression for Mobile Phones

In image pixilation, the bandwidth of the system is what limits performance. (Aila vd., 2003). One of the techniques that reduce bandwidth is texture compression. (Knittel vd., 1996). One of the must-have features in the system for texture compression is that textures can be stored compressed in the cache in order to make better use of the cache. Second, a constant compression ratio is needed for simple addressing and random access. Some losses occur due to fixed rate compression. Thirdly, texture-dependent lookup tables (LUT, Look Up Table) are avoided, eliminating the update that the LUT needs. (Hakura & Gupta, 1997).

While initially texture compression was targeted at mobile phones, it was later used in computer graphics cards and game consoles, using PACKMAN texture compression, which offers low complexity and reasonable image quality. (Ström & Akenine-Möller, 2004) .

With Block Truncation Coding (BTC), gray images were compressed into blocks of 4 × 4 pixels at a time. Two 8-bit grayscale values were stored for each block, and each pixel in the block was indexed to one of these grayscales, resulting in 2 bits per pixel (2 bpp – bit per bit).(Delp & Mitchell, 1979). Although BTC is not a tissue compression technique, the basis of other techniques is based on it.

The method called Color Cell Compression (CCC) uses an 8-bit color palette as an index instead of the grayscale value of BTC. The color palette is limited and requires memory searching (Campbell vd., 1986). Later CCC was implemented in hardware and was also used as texture cache (Knittel vd., 1996).

An evolution of CCC and one of the most popular is the S3TC texture compression method, which has been used in DirectX and has extensions for OpenGL. The block size is 4×4 and compressed to 64 bits. The two primary colors are stored in 16 bits. Additionally, it stores a two-bit index between two local base colors. All colors lie on a single line in the RGB space. The compression rate is 4 bpp. One disadvantage of S3TC is that only four colors can be used per block (Mccabe & Brothers, 1998) (Iourcha vd., 2003).

S3TC's problem was solved by using colors in neighboring blocks, but this increased the memory bandwidth (Ivanov & Kuzmin, 2000). The scheme, called POOMA, uses a variation of S3TC but compresses the 3×2 block size to 32 bits. Another difference from S3TC is that it uses fewer bits and only one intermediate color (Akenine-Moller vd., 2003). With the method called vector quantization, textures can be compressed up to 1 bpp and 2 bpp, and additional memory is required for access to vector quantization. This is not a very suitable solution for the high performance expected from computer graphics (Beers vd., 1996).

Later, a very different approach was taken. Two low-resolution images derived from the original texture are stored. During the process of decompressing textures, the textures are enlarged bidirectionally and a blend between the two is made to create the final color of the textures. It has two modes: 4 bpp and 2 bpp. In the 4 bpp version, two base colors are stored per 4x4 block. For bidirectional linear growth, 2x2 adjacent blocks are needed. Once textures are caught, decompression should be fast (Fenney, 2003).

In another method, mipmapping (the process of reducing large textures to increase the visibility of distant objects) and texture compression are combined. Each 4×4 block was compressed in YUV space and box filtering technique was used. Bitrate is approximately 4.6 bpp (Pereberin, 1999).

How Data Compression Works

3D computer graphics is whole with textures. This is a feature that increases the image quality and detail of 3D objects. Textures contain not only color but also information such as height and direction.

Textures are two-dimensional images, and they are applied to 3D surfaces. Each pixel in the images is called a texture element. Standard compression algorithms (RLE, LZW, Deflate) and popular image formats (JPEG, PNG, TIFF) are not suitable for textures. Access to tissue is done randomly, and this process occurs on the tissues needed, not sequentially. Textures corresponding to pixels that are next to each other do not mean textures that should be next to each other. Therefore, the performance of graphics systems is highly dependent on texture access. Texture compression formats are characterized by random access (Paltashev1 & Perminov2,2014).

Four compression techniques are applied for real-time applications.

Decoding Speed: Since textures are presented in a compressed format, decoding speed is very important. The decoding algorithm should be simple so that the processor finds each texture quickly and easily when it searches for it.

Random Access: Quick access to any texture is important because it is unknown how to access the texture.

Compression Ratio and Visual Quality: Compression ratio and visual quality are interrelated. Because as the texture is compressed, visual artificiality will increase.

Encoding Speed: Encoding speed is not very important as it uses textures before they are stored (Beers vd., 1996).

Textures are compressed in two ways: lossy and lossless. Since lossy texture compression techniques will cause errors in the texture, the result obtained will be an approximation of the original. In this technique, processing is done according to the compression ratio, and as the amount of compression increases, artifacts increase and quality decreases. Lossless compression technique always returns a replica of the original (Strom & Wennersten, 2011) & (Eskicioglu & Fisher, 1993).

When textures are compressed at a fixed rate, they always give the same value bit per pixel (bpp). This means that a texture with a size of N*N will always be compressed to the same file size. Non-constant compression techniques have also been shown to be slower in random access since there is no structure defined to access the texture data, regardless of the content.

Texture Compression Techniques for Mobile Phones

ETC (Ericsson Texture Compression) Family

ETC format, a standard compression scheme for Android-based devices, was first developed for use on mobile devices. PACKMAN is the first version (Ström & Akenine-Möller, 2004) then ETC1(Ström & Akenine-Möller, 2005)/ETC2(Ström & Martin, 2007) versions are enhanced. Based on the fact that the human eye is more sensitive to brightness than color, ETC has made texture compression based on this idea. Therefore, only a basic color is stored in each subblock (ETC1/ETC2 consists of two subblocks) with only the brightness information tied to a single integer value. In a sub-block, there are only four different brightness offsets and, accordingly, only four different colors are available.

PACKMAN

If the data path occupies the same number of bits as its width, interruptions in the data path are avoided, which simplifies hardware implementation. Memory sizes and bus widths of mobile devices are limited, so 2x4 sized blocks were designed and each block was compressed to 32 bits and the compression rate was 4 bpp.

Only one RGB color value is stored in a block. Four bits per R, G, B represent the basic color for a total of twelve bits (also referred to as RGB444 since each has 4 bits). The remaining 20 bits represent brightness. For each pixel, the base color is additionally modified by a constant taken from the four-input index table.

The same constant is added to all color components. This requires a pixel index of two bits per pixel to indicate which of the four values should be used. Coding can be done by exhaustively searching all parameters for least mean square error. It takes about a minute for a 64×64 texture on a 1.2GHz PC. A comprehensive search of other parameters based on average color is much faster and only takes about 30 milliseconds (Ström & Akenine-Möller, 2004).

ETC1 (iPACKMAN)

An image compressed with PACKMAN has significantly fewer luminance bands than an image where all pixels are 12 bit. To improve this, instead of a single block as in RGB444, two 2x4 sized blocks were used side by side, and 4x4 sized blocks in total were used.

The 4x4 ETC1 block is divided into two smaller blocks. Blocks can be stacked vertically or horizontally. As a result, it can choose the basic color with greater freedom. The RGB (R1 − R2, G1 − G2, B1 − B2) difference was made, and the color component that displayed the greatest deviation was noted. RGB555, or a 5-bit field from each color, was then made. This difference is represented by the black areas in the diagram below. It goes without saying that some blocks cannot be encoded in this manner by difference. In order to decide whether or not to use differential coding, a bit was added to the 4x4 block.

- Therefore, a difference bit coding is also used to indicate whether it is differential or normal,
- Flip bit indicating whether portrait (flip bit=0) or landscape (flip bit=1) orientation is used,
- 16 pieces 2-bit pixel indexes (one for each texture),
- two 3-bit table code words (one for each sub block) specifying which table to use
- two color code words used (independently or together) to encode the base color of the first sub block and the base color of the second sub block (Ström & Akenine-Möller, 2005).

ETC2

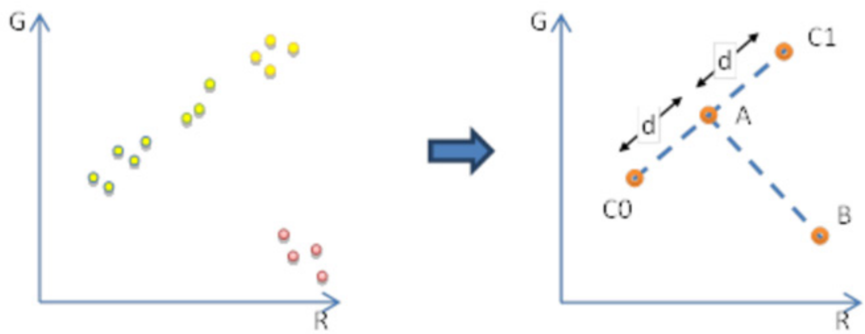
Although the diff blocks with RGB color differences increased the quality, since there was a single base color in each block, there were many errors due to compression in textures that were much more colorful. Therefore, these problems were tried to be corrected with extra blocks in ETC2.

In ETC1, when the sum of the base color and the offset exceeds the valid 5-bit range [0, 31], the blocks are decoded using one of the new modes, as invalid combinations occur in the differential block. Of the 64 bits, 1 bit is spent on the diff-bit and 8 bits are spent on the R0 and dR1 values, giving 55 bits. However, it is possible to use the lowest two bits of R0 and dR1 to encode additional information. There are 3 modes for this; T, H, Planar mode. These modes are caused by the overflow of each color (R, G, B) (Ström & Martin, 2007).

In T mode, two colors A and B are compressed in RGB444 format. The remaining 3 bits encode the d value. Then the other colors are calculated as C0 = (A - (d, d, d)) and C1 = (A + (d, d, d)). This mode is applied when the colors are on a single line in Fig 7.

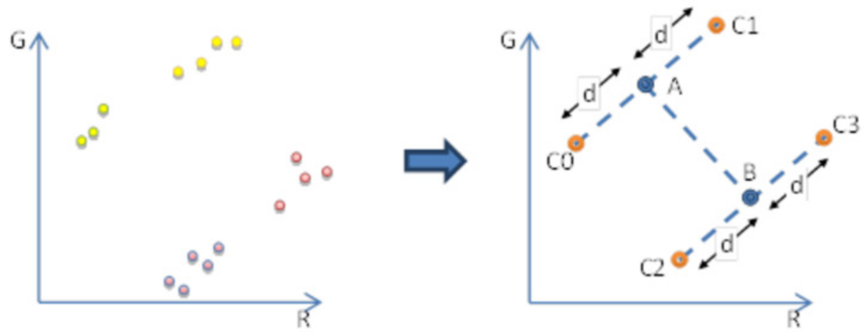
Figure 7

T Blocks in ETC2 (Zeng, 2016)



In H mode, unlike T3, C0, C1, C2, C3 colors are used. This mode is applied when the colors are on two lines. But the capacity of the H block is one bit less than that of the T block. Since A and B colors are symmetrical, they can be changed, and the problem is solved in Fig 8.

Figure 8
H Blocks in ETC2 (Paltashev1 & Perminov2,2014)



In planar mode, C0, CH, CV colors are available and are compressed as RGB676. And it is calculated according to the formula below (Ström & Martin, 2007);

$$C(x,y) = \frac{x(C_H - C_0)}{4} + \frac{y(C_V - C_0)}{4} + C_0$$

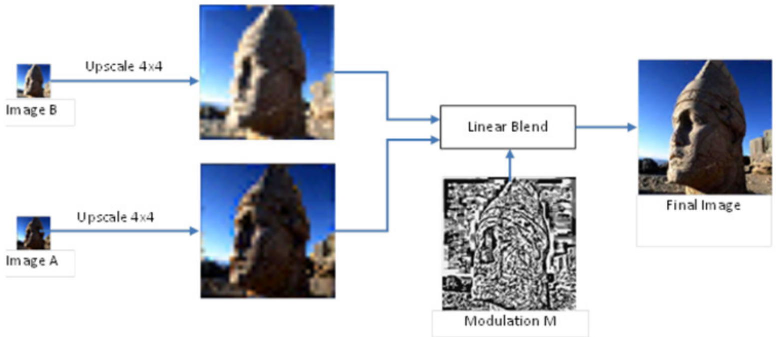
PVRTC Family

The PVRTC (PowerVR Texture Compression) format is patented by Imagination Technologies and is designed for the PowerVR family of graphics cores (Simon, 2003). It is used on Apple mobile devices such as iPhones and iPads. There is almost no publicly available information about the PVRTC technique.

Based on PVRTC, a high-frequency and low-frequency signal are represented by two low-resolution images, A and B, scaled down by a factor of 4 in both dimensions in Fig 9. To decompress an image, images A and B are first amplified and then processed with the M signal, which indicates the processing weight for each tissue (Simon, 2003).

Figure 9

PVRTC technique (Paltashev1 & Perminov2,2014)



The main idea has something in common with wavelet compression, where the entire image is split into high-frequency and low-frequency signals. A low-frequency signal is represented by two low-resolution images A and B, scaled down by a factor of 4 in both dimensions. A high frequency signal is a modulation signal M with full resolution but low precision. To decompress an entire image, images A and B must first be amplified and then blended using the modulation signal M, which specifies the blending weights per texture (Simon, 2003).

ASTC Family

ASTC (Adaptive Scalable Texture Compression) was jointly developed by ARM and AMD and introduced in 2012 and is an open royalty-free format. ASTC format has a fixed size 128-bit block. However, it has a 4x4 to 12x12 size structure for 2D textures and a 3x3x3 to 6x6x6 size structure for 3D textures. ASTC is one of the most flexible formats because it also supports LDR, HDR, 2D and 3D textures. ASTC is one of the most flexible formats because it also supports LDR, HDR, 2D and 3D textures (Jorn, 2012).

Conclusion

Texture Compression has an important place both in computer graphics and in the use of mobile devices. The mobile platform still has much weaker hardware than desktop computers. Therefore, existing resources must be used in the most efficient way. Computational power, memory size and memory bandwidth are some of the limited resources. Texture compression is an optimization technique used to reduce the demand on memory size.

Most textures on the GPU use “block compression” formats to save sampling bandwidth, memory usage, and for faster texture loads. The common theme between them is that they are all lossy and have fixed compression ratio and memory bandwidth.

References

Aila, T., Miettinen, V., Nordlund, P., & Oy, B. (2003). Delay Streams for Graphics Hardware.

Akenine-Moller, T., Strom, J., & Research, E. (2003). Graphics for the Masses: A Hardware Rasterization Architecture for Mobile Phones.

Anuradha, D., & Bhuvaneswari, S. (2016). A detailed review on the prominent compression methods used for reducing the data volume of big data. Annals of Data Science, 3, 47-62.

Beers, A. C., Agrawala, M., & Chaddha, N. (1996). Rendering from compressed textures. Proceedings of the 23rd Annual Conference on Computer Graphics and Interactive Techniques, 373-378.

Campbell, G., DeFanti, T. A., Frederiksen, J., Joyce, S. A., Leske, A., Lindberg, J. A., & Sandin, J. (1986). Two Bit/Pixel Full Color Encoding.

Das, D., Kumar, R., & Chakrabarti, P. P. (2005, January). Dictionary based code compression for variable length instruction encodings. In 18th International Conference on VLSI Design held jointly with 4th International Conference on Embedded Systems Design (pp. 545-550).

Delp, E., & Mitchell, O. (1979). Image Compression Using Block Truncation Coding. IEEE Transactions on Communications, 27(9), 1335-1342.

Eskicioglu, A., & Fisher, P. (1993, Ekim 19). A survey of quality measures for gray scale image compression. 9th Computing in Aerospace Conference. 9th Computing in Aerospace Conference, San Diego, CA, U.S.A. https://doi.org/10.2514/6.1993-4514

Fenney, S. (2003). Texture Compression using Low-Frequency Signal Modulation.

Hakura, Z. S., & Gupta, A. (1997). The Design and Analysis of a Cache Architecture for Texture Mapping.

Huffman Coding. (2024). Lavivienpost.com.

lavivienpost https://www.lavivienpost.com/wp-content/uploads/2021/12/huffman-steps1.jpg.webp

Iourcha, K., Nayak, K., & Hong, Z. (2003). System and Method for Fixed-Rate Block-based Image Compression with Inferred Pixels Value.

Ivanov, D., & Kuzmin, Ye. (2000). Color Distribution - a New Approach to Texture Compression.pdf. Computer Graphics Forum.

Jayasankar, U., Thirumal, V., & Ponnurangam, D. (2021). A survey on data compression techniques: From the perspective of data quality, coding schemes, data type and applications. Journal of King Saud University-Computer and Information Sciences, 33(2), 119-140.

Jorn, N. (2012). Adaptive Scalable Texture Compression. Eurographics Association. High Performance Graphics, pp. 105-114.

Kocer, S., Jama, B.S.A ., Er Y. , Dundar, B. (2022). Analysing Trends and Applications in Data Compression Techniques, In S. Kocer, O. Dundar (Eds.), Current Studies in Basic Sciences Engineering and Technology (pp. 114–128). ISRES Publishing.

Khalid A "Introduction to data compression", Third edition-2006

Knittel, G., Schilling, A., Kugler, A., & Straßer, W. (1996). Hardware for superior texture performance. Computers & Graphics, 20(4), 475-481.

Lavivienpost, https://www.lavivienpost.com/huffman-coding-and-decoding/,2024

Mccabe, D., & Brothers, J. (1998). DirectX 6 Texture Map Compression. Game Developer Magazine 5, 42-46.

Moffat, A. (2019). Huffman coding. ACM Computing Surveys (CSUR), 52(4), 1-35.

Paltashev1, T., & Perminov2, I. (2014). Texture Compression Techniques. sv-journal.org/2014-1/06/en/index.php?lang=en

Pavlo, A., (2024). Lecture# 03: Data Formats & Encoding II. https://15721.courses.cs.cmu.edu/spring2023/slides/05-compression.pdf

Pereberin, A. V. (1999). Hierarchical Approach for Texture Compression.

Pigeon, S. (2003). Huffman coding. Lossless Compression Handbook, 79-100.

Said, A., & Sayood, K. (2003). Arithmetic coding (pp. 101-152). Academic Press, San

Diego, CA.

Salomon, D. (2007). A concise introduction to data compression. Springer Science & Business Media.

Sayood, K. (2017). Introduction to data compression. Morgan Kaufmann.

Severance, D. G. (1983). A practitioner's guide to database compression tutorial. Information Systems, 8(1), 51-62.

Simon, F. (2003). Texture compression using low-frequency signal modulation. Eurographics Association. Graphics Hardware. pp. 84-91.

Strom, J., & Wennersten, P. (2011). Lossless compression of already compressed textures. Proceedings of the ACM SIGGRAPH Symposium on High Performance Graphics, 177-182. https://doi.org/10.1145/2018323.2018351

Ström, J., & Akenine-Möller, T. (2004). PACKMAN: Texture compression for mobile phones. ACM SIGGRAPH 2004 Sketches on - SIGGRAPH '04, 66.

Ström, J., & Akenine-Möller, T.(2005). iPACKMAN: High-quality, Low-complexity Texture Compression for Mobile Phones. New York, NY, USA: ACM. Proceedings of the ACM SIGGRAPH/EUROGRAPHICS Conference on Graphics Hardware. pp. 63-70.

Ström, J. & Martin, P. (2007). ETC2: Texture Compression Using Invalid Combinations. Aire-la-Ville, Switzerland: Eurographics Association. Proceedings of the 22Nd ACM SIGGRAPH/EUROGRAPHICS Symposium on Graphics Hardware. pp. 49-54.

Subathra, S., Sethuraman, M., & Babu, J. V. (2005, December). Performance analysis of dictionary-based data compression algorithms for high speed networks. In 2005 Annual IEEE India Conference-Indicon (pp. 361-365). IEEE

Witten, I. H., Neal, R. M., & Cleary, J. G. (1987). Arithmetic coding for data compression. *Communications of the ACM*, *30*(6), 520-540.

Zeng, J., Fang, L., Pang, J., Li, H., & Wu, F. (2016). Subpixel image quality assessment syncretizing local subpixel and global pixel features. *IEEE Transactions on Image Processing*, *25*(12), 5841-5856.

About the Authors

Sabri KOÇER, PhD, He graduated from the Electrical Engineering Department of Selcuk University. He completed his graduate and his doctorate in Gazi University. Currently, Necmettin Erbakan University, Faculty of Engineering, Computer Engineering is working. Electronics, Computer, Telecommunication, Signal Processing and Biomedical studies in the area.
E-mail: skocer@erbakan.edu.tr , **ORCID:** 0000-0002-4849-747X

Özgür DÜNDAR, PhD, works at Necmettin Erbakan University, Department of Astronautical Engineering. He graduated from the Electrical and Electronics Engineering Department of Selcuk University. He worked as an Automation Engineer for a while. His master's and doctorate degrees are from Selçuk University, Institute of Science and Technology, Department of Electrical and Electronics Engineering. Special fields of study are Automation, Robotic, Communication, Electromagnetic and Micro Strip Patch Antenna designs.
E-mail: ozdundar@erbakan.edu.tr , **ORCID:** 0000-0002-4142-4446

Similarity Index
The similarity index obtained from the plagiarism software for this book chapter is 17%..

The Use of Artificial Intelligence Techniques to Improve Production Quality in Machining Processes

Mehmet Ali GÜVENÇ

Iskenderun Technical University

Selçuk MISTIKOĞLU

Iskenderun Technical University

To Cite This Chapter

Güvenç, M. A., & Mıstıkoğlu, S. (2024). The Use of Artificial Intelligence Techniques to Improve Production Quality in Machining Processes. In S. Koçer & Ö. Dündar (Eds.), *Intelligent Systems and Optimization in Engineering* (pp. 127-140). ISRES Publishing.

Introduction

Overview of Industrial Revolutions and Their Impact on Manufacturing

Industrial revolutions have been among the most significant transformation processes in human history. Beginning in the 18th century with the first industrial revolution driven by steam power and mechanization, it marked the initial transition from manual labor to machine use in production processes. Subsequently, the second industrial revolution, characterized by the advent of electric power and mass production techniques, dramatically increased production speed and scale. The third industrial revolution introduced digitalization and automation, bringing greater flexibility to manufacturing (Vogel and Hess, 2016).

The Fourth Industrial Revolution (Industry 4.0) is currently revolutionizing manufacturing processes through the incorporation of cyber-physical systems, the Internet of Things (IoT), artificial intelligence (AI), and big data analytics. These advancements contribute to improved efficiency, flexibility, and product quality, while simultaneously lowering costs and fostering sustainability in production systems (Thoben et al., 2017).

All stages of the industrial revolutions have profoundly impacted precision and quality-dependent manufacturing processes, such as machining. The transition from traditional methods to intelligent manufacturing systems has become critical for gaining a competitive edge and meeting customer expectations (Guo et al., 2018).

The Role of Advanced Technologies in Enhancing Production Quality

Advanced technologies play a crucial role in enhancing quality in production processes. Precision manufacturing techniques and advanced control systems optimize the dimensional accuracy and surface quality of products while minimizing errors. In particular, big data analytics enables proactive management of processes by identifying potential issues on production lines in advance (Guvenc, 2022).

Artificial intelligence (AI) is one of the most significant technologies developed to improve production quality. For example, machine learning algorithms analyze sensor data to enhance product quality and minimize machine downtime (Cakir, 2022). Robotic automation systems, on the other hand, accelerate production processes, helping to achieve more consistent and higher-quality products (Gilchrist, 2016).

Modern production processes face numerous challenges, including high competition, diversity in customer demands, and sustainability requirements (Li et al., 2017). Key challenges include:

- **Production Efficiency**: Competing with traditional methods has become increasingly difficult. Shorter production cycles and high-quality demands necessitate continuous improvement of production processes.
- **Resource Management**: Rising raw material and energy costs drive businesses toward more sustainable production methods.
- **Technology Adaptation**: Transitioning to Industry 4.0 technologies poses financial and operational challenges, particularly for small and medium-sized enterprises (SMEs).
- **Workforce Shortages**: High-tech requirements demand a more skilled and equipped workforce, complicating the recruitment and training of qualified personnel.
- **Environmental Impact**: Sustainability goals encourage the adoption of innovative production methods to reduce the carbon footprint.

These challenges necessitate the integration of innovative technologies and AI-powered solutions. Manufacturers must overcome these barriers to establish more sustainable, flexible, and efficient processes (Li et al., 2015).

Additionally, technologies such as virtual reality (VR) and augmented reality (AR) are being employed in production design and process planning, offering new methods to reduce errors and enhance quality. The combined use of these technologies significantly raises quality standards, particularly in complex and detail-oriented processes like machining.

Importance of Machining Processes in Modern Manufacturing

Machining is an indispensable part of modern manufacturing technologies. In industries where factors such as precision, surface quality, and material properties are critical, machining methods offer high accuracy. Sectors requiring advanced technologies, such as automotive, aerospace, defense, and medical devices, heavily rely on machining processes. Machining methods provide flexibility to work on materials like metal, plastic, and composites. These methods include milling, turning, grinding, and drilling. Each process must be optimized to meet the dimensions, surface characteristics, and functional requirements of the product (Guvenc, 2022; Bilgic, 2019).

In addition to traditional approaches, the integration of AI-supported systems into machining processes in modern production offers significant potential for preventing manufacturing defects, increasing production speed, and improving final product quality. In this context, the role of technology in machining processes is becoming increasingly critical for the continuous development and competitiveness of the industry (Hoe et al., 2018; Guvenc, 2023).

Industrial Developments and Their Influence on Manufacturing Efficiency

The transition from the Third Industrial Revolution to the Fourth Industrial Revolution has created a radical transformation in manufacturing processes. While Industry 3.0 marked the beginning of digitalization, it also witnessed the widespread adoption of automation, programmable logic controllers (PLCs) and computer-operated machines. During this era, manufacturing processes became faster, more reliable, and more repeatable (Gilchrist, 2016).

Industry 4.0, however, goes beyond this digital foundation by leveraging cyber-physical systems and the Internet of Things (IoT) to make manufacturing processes smarter and more interconnected. Machines can now communicate with one another and make decisions based on real-time data. This transition has enabled manufacturing processes to become more efficient and flexible (Kim et al., 2018).

For instance, digital twin technology creates a simulation of the production line, allowing for the identification of potential errors and the optimization of processes. Furthermore, big data analytics and AI-based systems enhance manufacturing efficiency while continuously learning and adapting to improve quality (Khuntia, 2019).

This transition has also impacted the workforce. Manufacturing processes now require less human intervention but demand higher levels of technical expertise and knowledge. Consequently, the effects of Industry 4.0 extend beyond manufacturing processes to reshape workforce training and management strategies (Thoben, 2017).

Smart manufacturing encompasses the most significant trends shaping the future of production processes. By transcending traditional automation, it integrates artificial intelligence, robotics, IoT, and big data analytics, leading to a complete transformation of manufacturing systems (Thoben, 2017; Guvenc, 2022). The key trends include:

- **Robotic Automation**: The shift from traditional production lines to robot-assisted flexible systems ensures speed and precision in manufacturing. For instance, collaborative robots (cobots) operate in conjunction with human workers, augmenting overall productivity and improving operational efficiency.
- **Cyber-Physical Systems**: Full integration of manufacturing machines with the digital world enables real-time monitoring and optimization of processes.
- **Big Data and Analytics**: The acquisition and examination of manufacturing data facilitate predictive and proactive management of production processes. For example, predictive maintenance algorithms can identify potential machine malfunctions prior to their occurrence, enabling timely interventions.
- **Sustainability-Oriented Technologies**: Practices such as using recyclable materials, reducing energy consumption, and minimizing waste are integral components of smart manufacturing.
- **Artificial Intelligence and Machine Learning**: Applications of AI in quality control, process optimization, and decision-support systems are becoming increasingly prevalent.

These trends make manufacturing processes more flexible, efficient, and sustainable. Additionally, they provide businesses with a competitive advantage by enabling the delivery of high-quality products that meet customer expectations (Teti et al., 2010).

Factors Influencing Product Quality in Machining Processes

Material Properties and Their Effects on Machining

Material properties are among the most critical factors determining product quality in machining processes. Characteristics such as hardness, strength, ductility, and thermal conductivity of the workpiece material directly influence both machining parameters and the resulting surface quality (Teti et al., 2010).

- **Hardness**: Harder materials can accelerate tool wear and increase surface roughness. In such cases, selecting the appropriate cutting tool and fine-tuning machining parameters are crucial.

- **Strength**: High-strength materials require more power and energy during machining, which can impose additional loads on machine performance and potentially affect machining accuracy.
- **Ductility**: Ductile materials may hinder chip breaking during machining, leading to reduced surface quality. Chip breaker designs or specialized cooling fluids can be employed to address this issue.
- **Thermal Properties**: The thermal conductivity of a material impacts its ability to dissipate heat during machining. Materials with low thermal conductivity may cause overheating in the cutting zone, which can compromise dimensional accuracy.

Understanding material properties and making informed material selections are essential for producing high-quality products. Employing machining methods tailored to the material's characteristics is necessary to optimize surface roughness and dimensional accuracy (Zhang et al., 2015).

Machine Tool Parameters: Spindle Speed, Feed Rate, and Depth of Cut

The parameters of machine tools are crucial in influencing both the performance and the quality of products in machining processes. It is vital to optimize factors such as spindle speed, feed rate, and depth of cut to enhance the efficiency and effectiveness of cutting operations:

- **Spindle Speed**: Influences surface roughness and machining time. Higher speeds can improve surface smoothness but may accelerate tool wear.
- **Feed Rate**: Determines the material's movement speed across the machine. Higher feed rates can reduce production time but may increase surface roughness and cutting forces.
- **Depth of Cut**: Defines the amount of material removed in a single pass. Greater depths of cut can shorten machining time but may increase tool load, reducing tool life.

An optimal combination of these parameters enhances product quality while minimizing production costs. Experimental studies or AI-assisted modeling methods are often used to determine the best parameter settings (Guvenc, 2022; Abouelattta, 2001).

Environmental and Operational Factors: Temperature, Lubrication, and Tool Wear

Environmental and operational factors significantly influence the efficiency of machining processes and product quality:

- **Temperature**: Heat generated during cutting has a profound effect on machining quality. High temperatures can cause thermal expansion of the material, leading to dimensional inaccuracies. Additionally, thermal damage may occur to the cutting tool.
- **Cooling and Lubrication**: Coolants control temperature in the cutting zone, reduce friction, and facilitate chip removal. However, improperly selected or insufficiently applied coolants can lead to undesirable surface roughness.
- **Tool Wear**: Tool wear is inevitable over the tool's lifespan. A worn tool degrades surface quality and requires higher cutting forces. Monitoring and timely replacement of worn tools are vital for ensuring process continuity.

Effectively managing these factors is a critical step toward achieving consistent production of high-quality products (Upase and Ambhore, 2020).

Integration of Artificial Intelligence in Machining Processes

Applications of Artificial Intelligence in Machining

AI has a variety of applications in machining processes, offering significant potential to enhance production quality, optimize process efficiency, and reduce costs:

- **Predictive Maintenance**: AI-based predictive models analyze data from machine sensors to detect equipment failures in advance. This reduces unplanned downtimes and extends machine lifespan.

- **Process Optimization**: AI algorithms are employed to optimize key machining parameters, including spindle speed, feed rate, and depth of cut, thereby facilitating the efficient production of high-quality products within reduced time intervals.

- **Quality Control**: Image processing and deep learning algorithms assess surface quality and dimensional accuracy of products. These systems can detect defects that are too small for the human eye to perceive.

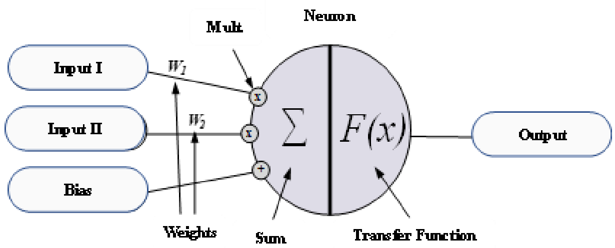
- **Chip Formation and Management**: In machining processes, controlling the shape and size of chips influences machining stability. AI analyzes chip formation mechanisms and develops optimal chip-breaking strategies (Kim, 2018).

Types of AI Used in Machining: ANN, ANFIS, and Metaheuristic Algorithms (A Practical Approach)

Various artificial intelligence methods are applied in machining processes to perform prediction and optimization tasks across different stages of production:

- **Artificial Neural Networks (ANNs)**: ANNs modeled after the human nervous system, are composed of interconnected units referred to as artificial neurons. Similar to biological neurons, artificial neurons evaluate incoming information and transmit it to other neurons or output units. As illustrated in Figure 2, ANNs are widely used to model complex process relationships and perform data analytics. A schematic representation of the artificial neuron model is shown in Figure 3.15. If formalized, Equation 1 can be derived to describe this system (Jain and Mao, 1996).

Figure 1
The Structure of Artificial Neuron (Güvenç, 2022)



$$y(k) = F(\sum_{i=0}^{m} \quad x_i . w_i + b) y(k) = F(\sum_{i=0}^{m} \quad x_i . w_i + b) \tag{1}$$

In the equation:

- ***xi*** represents the input value at time *k*, ranging from 0 to m.

- ***wi*** denotes the weight value at time *k*, ranging from 0 to m.
- ***b*** is the bias term.
- ***F*** is the transfer function.
- ***y(k)*** is the output value at time *k*.

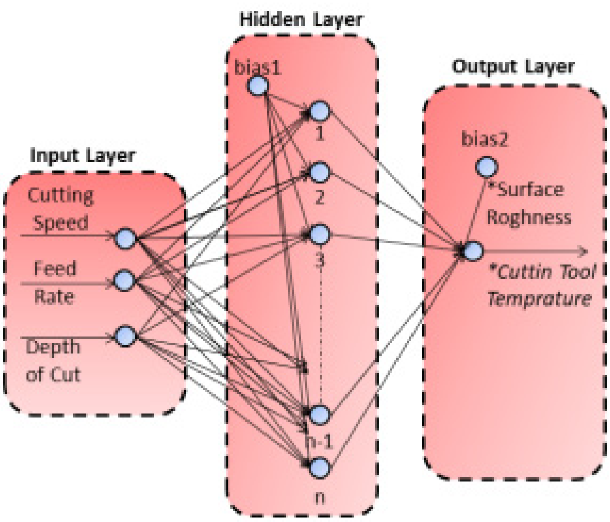
ANNs have extensive applications in machining processes. ANNs are highly effective for modeling datasets with complex and nonlinear relationships. Key applications of ANN in machining processes include:

- **Prediction of Surface Roughness**: Surface roughness during machining is influenced by factors such as spindle speed, feed rate, and depth of cut. ANN models learn the complex relationships between these parameters to predict surface quality (Bilgic, 2019).
- **Process Optimization**: ANN is employed to determine optimal machining parameters in production processes. For instance, it can predict parameters that maximize quality while minimizing production costs.
- **Monitoring Tool Wear**: Tool wear affects product quality and machining time. ANN-based systems analyze sensor data to predict tool wear in advance and determine optimal replacement times (Upase and Ambhore, 2020).

In a study conducted by Güvenç (2022), ANN was used to predict surface roughness and tool-tip temperature during the turning of S235JR alloy material. An image captured during the study is provided in Figure 2.

Figure 2
Image taken during turning (Güvenç, 2022)

Figure 3
Structure of ANN Model (Bilgic, 2019)



The results obtained from the Artificial Neural Network (ANN) model were compared with those derived from the Multiple Linear Regression Model (MLRM). The schematic representation of the ANN model is illustrated in Figure 3. In this model, cutting speed, feed rate, and depth of cut were provided as inputs to predict surface roughness and tool-tip temperature (Güvenç, 2022).

Table 1 presents the performance criteria for training, testing, and overall data for both the MLRM and ANN models. As shown in the table, the coefficient of determination (*R*) for tool-tip temperature evaluations was approximately 92% for the ANN, while it was only 76% for the MLRM. Similarly, for surface roughness predictions, the *R* value of the ANN model was 93%, whereas it remained at 43% for the MLRM (Güvenç, 2022).

Table 1

The performance results of ANN (Güvenç 2022, Bilgic, 2019)

Tool Temperature						
	Train		**Test**		**All Data**	
	ANN	MLRM	ANN	MLRM	ANN	MLRM
MSE	1,0611	1,1994	0,0186	4,2618	0,8006	2,3400
MAE	0,5219	1,0008	0,0847	1,8641	0,4126	1,2166
RMSE	1,0301	1,3036	0,1374	2,0664	0,8947	1,5297
R	0,8889	0,8100	0,9986	0,8228	0,9189	0,7555
Surface Roughness						
	Train		**Test**		**All Data**	
	ANN	ÇLRM	ANN	ÇLRM	ANN	ÇLRM
MSE	0,1484	0,7693	0,0529	1,8960	0,1245	1,0510
MAE	0,2566	0,7059	0,1863	1,2388	0,2390	0,8391
RMSE	0,3852	0,8771	0,2300	1,3770	0,3528	1,0252
R	0,9228	0,5941	0,9825	0,0001	0,9320	0,4276

To better visualize the model performances, the surface roughness and tool-tip temperature predictions are depicted as graphs in Figures 4 and 5, respectively. Based on these figures, as well as evaluations of metrics such as Mean Squared Error (MSE), Mean Absolute Error (MAE), Root Mean Squared Error (RMSE), and *R*, ANN—a widely used artificial intelligence technique demonstrates significantly higher accuracy than the traditional MLRM (Guvenc, 2022).

Figure 4
Model Achievement Results for Surface Roughness (Güvenç, 2022)

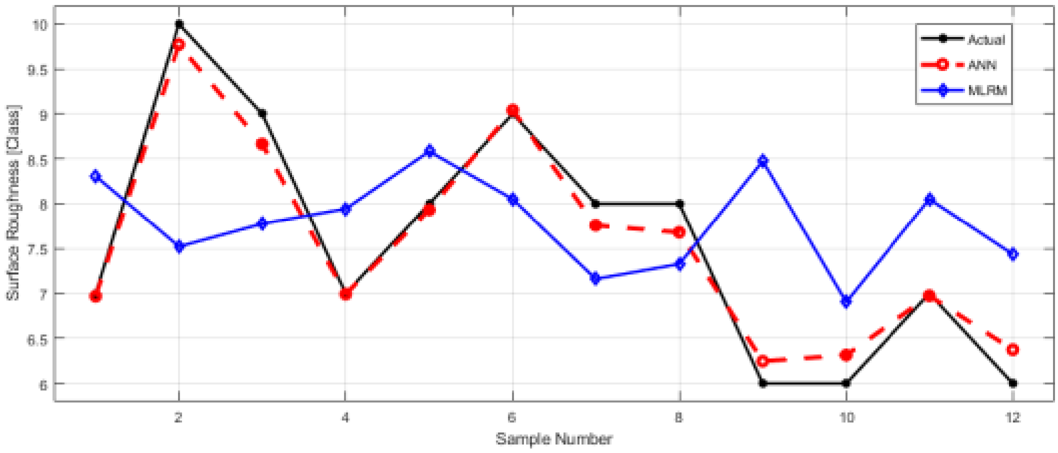
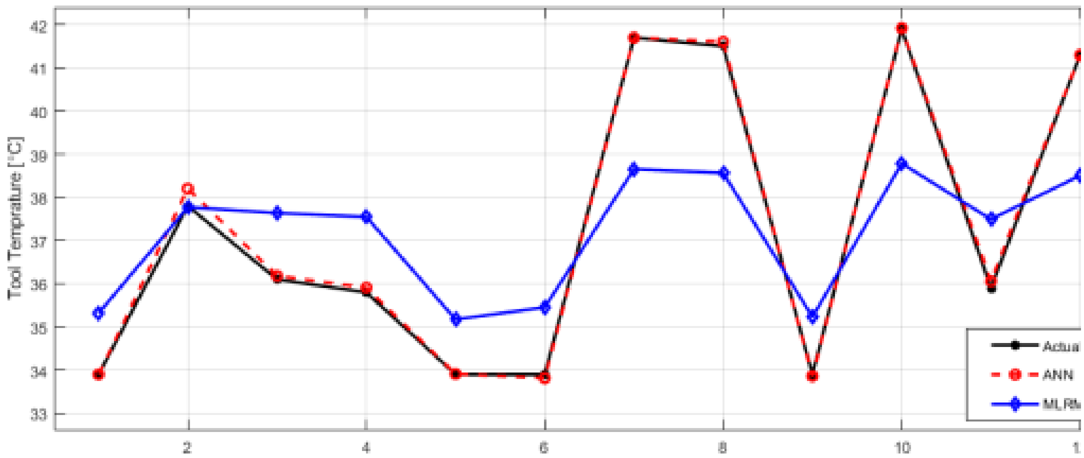


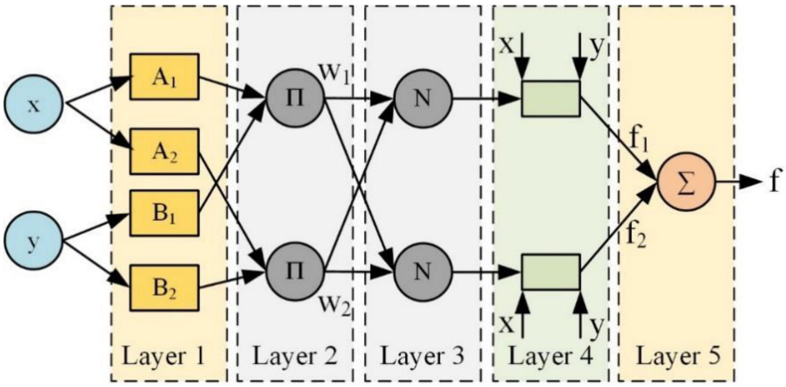
Figure 5.
Model Achievement Results for Tool Temperature (Güvenç, 2022)



As evident from the aforementioned study, ANN serves as a powerful tool for making production processes more efficient and consistent.

- **Adaptive Neuro-Fuzzy Inference System (ANFIS)**: ANFIS integrates the capabilities of fuzzy logic and artificial neural networks, leveraging the advantages of both methodologies in a synergistic framework (Siddhpura and Paurobally, 2012). It is highly effective for analyzing and optimizing complex relationships between parameters in machining processes (Guvenc, 2022). ANFIS operates through five key stages. Bunlar sırasıyla **Input Fuzzification** (Layer 1), **Fuzzy Operators** (Layer 2), **Application Method** (Layer 3), **Aggregation of All Outputs** (Layer 4) ve **Defuzzification** (Layer 5) (Jang, 1991). The basic structure of ANFIS is illustrated in Figure 6.

Figure 6
Basic ANFIS structure with two inputs and one output (Güvenç, 2022)

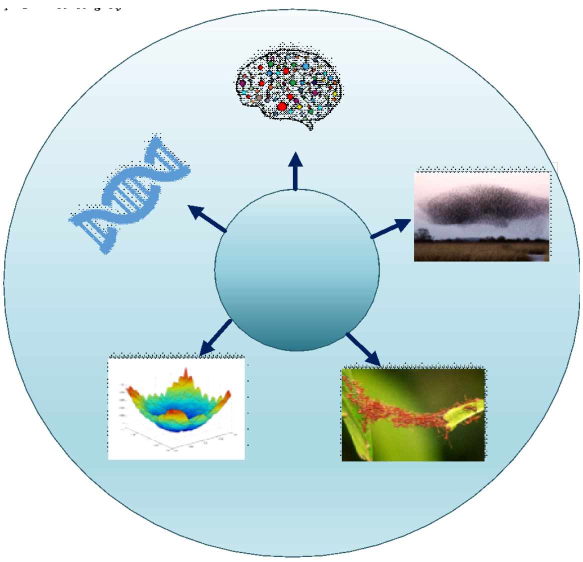


- **Metaheuristic Algorithms**:

Optimization involves finding the best solution among various alternatives to solve a problem. Metaheuristic algorithms have been developed through the effective application of different heuristic methods. Numerous well-established and frequently used metaheuristic algorithms exist in the literature, including Genetic Algorithms (GA), Ant Colony Optimization (ACO), Differential Evolution (DE), Artificial Bee Colony Algorithm (ABC), and Teaching-Learning-Based Optimization (TLBO). Many applications utilize metaheuristic algorithms in combination with methods like ANFIS (Güvenç, 2022).

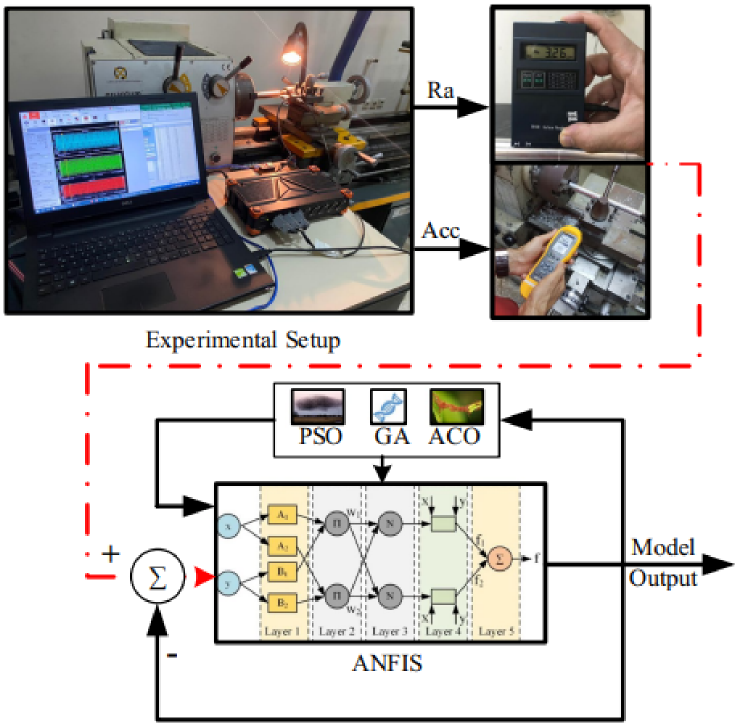
In a study conducted by Güvenç (2022), tool vibrations and surface roughness during the turning of AA6013 alloy were predicted using ANFIS optimized with metaheuristic algorithms. The optimization process incorporated three different metaheuristic algorithms. One of these is an Ant Colony Optimization (ACO), modeled on how ants find the shortest path to a food source (Paul et al., 2015). The other one is these a Genetic Algorithm (GA), Based on Darwin's theory of evolution (Akdagli et al., 2006). The last one is a Particle Swarm Optimization (PSO), Inspired by the social behavior of bird flocks (Wahid et al., 2019). The schematic representation of this study is presented in Figure 7.

Figure 7
Schematic view of study



The experimental setup and block diagram used in the study are illustrated in Figure 8. For the turning process, different values of cutting depth, feed rate, and rotational speed were selected from the system. Using these varied parameters, data for surface roughness (Ra) and tool vibration (Acc) were collected (Güvenç, 2022).

Figure 8
Experimental setup and block scheme of study (Guvenc, 2022)



The results obtained from the study are summarized in Table 2. According to the findings:

- For **Acc prediction**, the Multiple Linear Regression Model (MLRM), with an RRR value of 0.726, was identified as the least effective model among the proposed approaches. On the other hand, the **GA-ANFIS model**, achieving an RRR value of 0.946, was found to be the most effective model for Acc prediction.

- In the context of Ra prediction, the **ACO-ANFIS** model demonstrated the lowest efficacy among the proposed methodologies, achieving an RRR value of 0.810. Conversely, the PSO-ANFIS model emerged as the most proficient approach, attaining an RRR value of 0.916.

The GA-ANFIS and PSO-ANFIS models exhibited superior performance relative to conventional approaches, highlighting their potential as robust and reliable predictive tools for analogous applications (Güvenç, 2022).

Table 2
Comparison of Forecast Models

		MSE		MAE		R	
		Train	Test	Train	Test	Train	Test
Acc	ÇLRM	0.068847	0.083466	0.217571	0.241197	0.725249	0.726901
	ACO	0.066702	0.078225	0.217374	0.23551	0.720807	0.761294
	GA	0.030968	0.025861	0.15536	0.146241	0.892455	0.946903
	PSO	0.031444	0.049664	0.139707	0.162712	0.883234	0.88679

Ra	**ÇLRM**	0.044783	0.049864	0.174726	0.193745	0.755374	0.8244
	ACO	0.042983	0.053632	0.167134	0.196744	0.758745	0.810425
	GA	0.025824	0.034109	0.123148	0.144264	0.876478	0.911402
	PSO	0.020494	0.031704	0.109358	0.13957	0.894505	0.916647

Challenges and Opportunities in Implementing Artificial Intelligence Techniques

While predictive and optimization analytics using artificial intelligence methods offer great opportunities, they can also bring some challenges (Sivarajah et. al., 2017). These include:

Challenges:

- **Data Quality:** The integrity and reliability of the data employed in the analysis are of paramount importance, as inaccuracies or omissions in the dataset may lead to erroneous predictive outcomes.
- **Infrastructure Requirements:** Predictive analytics requires high-performance computers and sensor infrastructure.
- **Operator Training:** Operators must have adequate training to understand and utilize these technologies effectively.

Opportunities:

- **Faster Decision Making:** Predictive analytics accelerates decision-making during processes, thus increasing production speed.
- **Cost Reduction:** Preventing process errors and enabling scheduled maintenance helps reduce operational costs.
- **Competitive Advantage:** Predictive analytics allows companies to respond more quickly to market demands.

Future Directions and Research Opportunities

The continuous evolution of machining technologies presents new opportunities for research and the application of cutting-edge techniques to improve production quality and efficiency. Emerging technologies, particularly those related to Artificial Intelligence (AI) and Digital Twins, are transforming the optimization of machining processes. AI is rapidly advancing, offering innovative solutions for enhancing machining processes. New AI techniques, especially deep learning (DL) and advanced machine learning (ML) algorithms, are expected to play a key role in improving process efficiency and product quality in machining. These techniques enable more precise optimization of machining parameters, predictive maintenance, and real-time process adjustments.

The integration of autonomous process control and dynamic parameter adjustment through AI algorithms will enhance the adaptability and efficiency of machining systems. Furthermore, computer vision and natural language processing (NLP) technologies are increasingly being incorporated to streamline the analysis of machining data, enabling faster anomaly detection and decision-making. Research into these emerging AI techniques holds the potential to enhance production flexibility, reduce energy consumption, and improve surface quality, positioning AI as a central technology for the future of machining (Khuntia, 2019; Kim, 2018).

Conclusion

The quality and efficiency improvements in machining processes are undergoing a significant transformation with the integration of Artificial Intelligence (AI) and other advanced technologies. This section summarizes the key findings of the study, explores their implications for both industry and academia, and outlines the future path for AI-driven machining.

This study has examined the application of AI techniques for improving quality and efficiency in machining processes. The key findings are as belows:

- **AI** offer substantial potential in optimizing machining parameters and improving production processes. Specifically, deep learning and advanced machine learning algorithms are shown to yield effective results in process optimization, predictive maintenance, and real-time process adjustments.
- **Digital Twins** and **Simulation technologies** create virtual models of physical machining processes, enabling real-time monitoring and optimization. These technologies provide valuable insights into potential inefficiencies, tool wear, and deviations before they impact production, enhancing process reliability.
- **Metaheuristic Algorithms** and **Multi-Objective Optimization** methods allow for the simultaneous optimization of multiple conflicting objectives. These methods provide solutions in more complex and broader parameter spaces compared to traditional optimization techniques, offering better adaptability and efficiency in machining operations.

Acknowledgements

This study is supported by the Iskenderun Technical University scope of PhD Thesis of Mehmet Ali GUVENC. We would like to thank Iskenderun Technical University, Faculty of Engineering and Natural Sciences, for their support. Finally, we would like to thank Konya Technical University and Prof. Dr. Mete Kalyoncu for allowing us to use the laboratory facilities.

References

Abouelatta, O. B., ve Madl, J. (2001). Surface roughness prediction based on cutting parameters and tool vibrations in turning operations. Journal of materials processing technology, 118(1-3), 269-277.

Akdagli AA, Guney K, Karaboga D. 2006. Touring ant colony optimization algorithm for shaped-beam pattern synthesis of linear antenna. Electromagnetics 26:615–628.

Bilgic, H. H., Guvenc, M. A., Cakır, M., & Mıstıkoglu, S. (2019). A study on prediction of surface roughness and cutting tool temperature after turning for s235jr steel. Konya Journal of Engineering Sciences, 7, 966-974.

Cakir, M., Guvenc, M. A., & Mistikoglu, S. (2021). The experimental application of popular machine learning algorithms on predictive maintenance and the design of IIoT based condition monitoring system. Computers & Industrial Engineering, 151, 106948.

Gilchrist, A. (2016). Industry 4.0: the industrial internet of things. Apress

Güvenç, M. A. (2022). Tornalama süreçlerinde aktif titreşim kontrolü, yapay zeka uygulamaları ve nesnelerin interneti uygulamaları (PhD thesis, İskenderun Teknik Üniversitesi/Lisansüstü Eğitim Enstitüsü/Makine Mühendisliği Ana Bilim Dalı).

Guo, K., Lu, Y., Gao, H., ve Cao, R. (2018). Artificial intelligence-based semantic internet of things in a user-centric smart city. Sensors, 18(5), 1341.

Guvenc, M. A., Bilgic, H. H., Cakir, M., & Mistikoglu, S. (2022). The prediction of surface roughness and tool vibration by using metaheuristic-based ANFIS during dry turning of Al alloy (AA6013). Journal of the Brazilian Society of Mechanical Sciences and Engineering, 44(10), 474.

Hoe, C. H., Reddy, M. M., Lee, V. C. C., ve Debnath, S. (2018). Chatter Behavior in the Milling Process of Inconel 718: Effects of Tool Edge Radius. In MATEC Web of Conferences (Vol. 202, p. 02006). EDP Sciences

Jain, Anil K. and Jianchang Mao. 1996. "Artificial Neural Networks: A Tutorial." IEEE Xplore.

Jang J (1991) Fuzzy Modeling Using Generalized Neural Networks and Kalman Filter Algorithm. Proceedings of the 9th National Conference on Artificial Intelligence 91:762–767.

Khuntia, S. R., Rueda, J. L., ve van der Meijden, M. A. (2019). Smart asset management for electric utilities: Big data and future. In Asset intelligence through integration and interoperability and contemporary vibration engineering technologies (pp. 311-322). Springer, Cham.

Kim, D. H., Kim, T. J., Wang, X., Kim, M., Quan, Y. J., Oh, J. W., ... ve Ahn, S. H. (2018). Smart machining process using machine learning: A review and perspective on machining industry. International Journal of Precision Engineering and Manufacturing-Green Technology, 5(4), 555-568.

Li, B. H., Hou, B. C., Yu, W. T., Lu, X. B., ve Yang, C. W. (2017). Applications of artificial intelligence in intelligent manufacturing: a review. Frontiers of Information Technology ve Electronic Engineering, 18(1), 86-96.

Li, Y., Liu, Q., Tong, R., ve Cui, X. (2015). Shared and service-oriented CNC machining system for intelligent manufacturing process. Chinese Journal of Mechanical Engineering, 28(6), 1100-1108.

Paul SK, Azeem A, Ghosh AK. 2015. Application of adaptive neuro-fuzzy inference system and artificial neural network in inventory level forecasting. International Journal of Business Information Systems 18:268–284.

Siddhpura, M. and R. Paurobally. 2012. "A Review of Chatter Vibration Research in Turning." International Journal of Machine Tools and Manufacture 61:27–47.

Sivarajah, U., Kamal, M. M., Irani, Z., ve Weerakkody, V. (2017). Critical analysis of Big Data challenges and analytical methods. Journal of Business Research, 70, 263-286.

Teti, R., Jemielniak, K., O'Donnell, G., ve Dornfeld, D. (2010). Advanced monitoring of machining operations. CIRP annals, 59(2), 717-739.

Thoben, K. D., Wiesner, S., ve Wuest, T. (2017). "Industrie 4.0" and smart manufacturing-a review of research issues and application examples. International journal of automation technology, 11(1), 4-16.

Upase, R., ve Ambhore, N. (2020). Experimental investigation of tool wear using vibration signals: An ANN approach. Materials Today: Proceedings, 24, 1365-1375.

Vogel-Heuser, B., ve Hess, D. (2016). Guest editorial Industry 4.0–prerequisites and visions. IEEE Transactions on Automation Science and Engineering, 13(2), 411-413.

Wahid MA, Masood S, Khan ZA, et al. 2019. A simulation-based study on the effect of underwater friction stir welding process parameters using different evolutionary optimization algorithms. Proceedings of the Institution of Mechanical Engineers, Part C: Journal of Mechanical Engineering Science 234:643–657.

Zhang, S. J., To, S., Zhang, G. Q., ve Zhu, Z. W. (2015). A review of machine-tool vibration and its influence upon surface generation in ultra-precision machining. International journal of machine tools and manufacture, 91, 34-42.

About the Authors

Mehmet Ali GÜVENC completed his BSc degree as a mechanical engineer in 2012 at Selcuk University which is located in Konya, Türkiye. Then started MSc study in the same university and completed it in 2015. In the same year, He began his doctoral studies at the same university; however, in 2017, he transferred his doctoral studies to Iskenderun Technical University. Subsequently, he earned his PhD from Iskenderun Technical University in 2021. Now he is Asst. Prof. at Department of Aeronautics and Astronautics Engineering, Iskenderun Technical University, Hatay, Türkiye. His research areas include artificial intelligence, mechanical system design and control, mechanical vibrations, smart manufacturing applications and internet of things.
E-mail: gvnc.mehmetali@gmail.com, **ORCID:** 0000-0002-4652-3048

Selçuk MISTIKOĞLU completed his doctoral studies at Cukurova University, earning the title of Doctor of Engineering in the field of Mechanical Engineering. He served as the Director of İskenderun Vocational School and as the founding Dean of the Faculty of Aeronautics and Astronautics at Iskenderun Technical University. With numerous published scientific works, he is Prof. Dr. in the Department of Mechanical Engineering at the Faculty of Engineering and Natural Sciences at Iskenderun Technical University.
E-mail: selcuk.mistikoglu@iste.edu.tr, **ORCID:** 0000-0003-2985-8310

Similarity Index

The similarity index obtained from the plagiarism software for this book chapter is 19%..

Optimization of Microstrip Patch Antenna Parameters with Artificial Neural Networks

Özgür DÜNDAR
Necmettin Erbakan University

Sabri KOÇER
Necmettin Erbakan University

To Cite This Chapter

Dündar, Ö., & Koçer, S. (2024). Optimization of Microstrip Patch Antenna Parameters with Artificial Neural Networks. In S. Koçer & Ö. Dündar (Eds.), *Intelligent Systems and Optimization in Engineering* (pp. 141-162). ISRES Publishing.

Introduction

With the development of technology, it has become very important to transmit information from one point to another in an error-free and accurate manner, at low cost and by selecting the appropriate transmission medium. Antenna, which is very important in the transmission of information, is a circuit element that converts electrical signals into electromagnetic waves and transfers them to the transmission medium or converts electromagnetic waves from the transmission medium into field and electrical signals. It can be said that the microstrip antenna (MBA) is an element that provides these transformations at microwave frequencies.

Microstrip patch antennas are a geometrical structure consisting of a conductive metal patch that radiates on a metal-coated dielectric substrate called a ground plane (Garg at al., 1995). Microstrip patch antennas are one of the most popular types of antennas today due to their simple geometry, easy fabrication, low cost, light weight and ease of installation. Microstrip patch antennas have the potential for a wide range of applications over a wide frequency range. The widespread use of microstrip patch antennas in the industrial field means that they can be used in almost every field of communication and has brought significant contributions to the sector in terms of cost.

Today, the need for broadband and high data rate communication is increasing, especially in mobile communication systems. UHF band covers a part of satellite communication technology. It is evaluated that X/Ku/Ka band technologies can be used in the next generation over-the-horizon communication systems due to the low number of satellites with a transponder, also called UHF trasponder, and the high cost of circuit rental (Uğurlu, 2016).

In particular, developments in microwave technology have been driven by the advantages of antennas to be used in related devices such as large bandwidth, being as small as possible and easy to manufacture. Today, antenna designers direct their work by considering these factors.

Factors affecting the bandwidth of a microstrip patch antenna are generally the shape of the patch, the feeding method, the placement of the base and radiating patches and the parasitic elements. Various bandwidth enhancement methods have been used in the literature over the years to address these factors. The bandwidth of a microstrip patch antenna can be increased due to a low Q quality factor and well-excited multiple resonant modes. Another bandwidth enhancement method is the choice of a thick base with a low dielectric constant. The use of a parasitic patch is another bandwidth enhancement method (Panayi at al., 1999). The use of stack elements is another bandwidth enhancement method (Tong at al., 2000). Finally, changing the dimensions of the ground plane is another bandwidth enhancement method (Huynh & Stutzman 2003, Dilek Uzer at al., 2016).

As the studies on microstrip antennas have increased, the number of antennas with various geometric structures has also increased. Today, microstrip antennas are named according to the patch geometry. The study on rectangular microstrip patch antennas, which is considered as an important source of information in the literature, was done by Howell (Howell, 1972). Rectangular patches are the first and most widely used microstrip patch antenna geometry. It is larger in size than other patch geometries and therefore has a larger bandwidth.

The radiation pattern of a single element antenna is relatively broad, but the single element provides low gain. In order to meet the needs of long distance communication, it is necessary to design high gain antennas in most applications (Roy & Chakraborty, 2011). The gain of the antenna can be increased by increasing the electrical size of the antenna. However, the antenna gain can also be increased by placing the radiating element in a specific electrical and geometrical arrangement. This structure consisting of more than one radiating element is called an antenna array. Array antennas were first designed by Sanford as surface-matched array antennas for L Band applications (Sanford, 1974). The use of microstrip antennas as array antennas is quite common (Balanis, 1982). Antenna arrays are used to synthesize the required pattern that cannot be achieved with a single element, to improve directivity, beam scanning and other functions that are difficult to achieve with a single element.

Haeng Sook Noh, Jae Seung Yun, Jong Myen Kim, Soon-Ik Jeon (2004); In this study, a bidirectional high gain and broadband 1x8 rectangular cross-section microstrip patch antenna design for Ku band was realized and Rx/Tx feeds were formed for bidirectional communication. As a result, it was observed that the bandwidth increased by 10% in Rx feed and 11% in Tx feed.

Dowon Kim, Moonil Kim, Tanaka M., Matsugatani K. (2006); In this study, an array antenna with 12 patches in the shape of a hexagonal with smoothed corners was designed to increase the bandwidth. At 6.8 GHz, a material with a thickness of 1.27 mm and a dielectric constant of 9.8 was used, and the results showed that the bandwidth increased by 1.7% for each patch according to the measured values.

Shah, Suaidi, Aziz, Rose, Kadir, Ja'afar, Sidek, Rahim M.K.A., (2008); In this study, three (1x2, 1x4, 2x2) array antennas were designed. First, -45 and +45 degree inclined antennas were designed, then the same antenna was combined 2 times side by side and in the third design, 1x2 antennas were combined one below the other. FR4 material with dielectric constant (ε) 4 and h height 1.6 was used in the design. As a result of the measurements, it was observed that the gain of the 1x2 array antenna was 9.5 dB, while the gain of the 1x4 and 2x2 array antennas was approximately 20 dB. As a result, it is seen that antenna combinations will increase the gain.

Ang Yu, Xuexia Zhang (2002); In this study, a 2x1 rectangular patch antenna was first designed. Then 4x(2x1) patch antennas are combined to form a 1x8 combined patch antenna. Then, 8 interference patches with the same patch size were formed on the front side of the 1x8 array antenna. The parasitic patch antenna is compared with the formed patch antenna. Operating at 2.4 GHz resonant frequency, the antenna uses 2 mm thick dielectric material with a dielectric constant of 2.58. As a result, the gain of the 1x8 antenna with parasitic patch increases by 2.13 dB compared to the 1x8 patch antenna.

Such an antenna is recommended for wireless LAN networks.

Gültekin S.S. (2002); This study was conducted as a doctoral thesis and the parameters of rectangular, circular and triangular microstrip antennas were calculated by ANN and compared with the literature results. In this study, rectangular, circular and triangular geometric shapes are evaluated in a single geometric form and the results obtained are interpreted. Eleven different training algorithms were used in the ANN.

Türkmen M., Yıldız C., Sağıroğlu Ş. (2003); In this study, the characteristic impedance and effective dielectric constant of coplanar waveguides with an infinitely long dielectric base are calculated with a single Artificial Neural Network (ANN) model. The ANN structure was trained using five different learning algorithms and the algorithm performances were evaluated among themselves. It is observed that the obtained results are in good agreement with the existing results in the literature. Comparison of the results obtained from the presented ANN model with the results obtained with the Conformal Transformation Technique used in the literature for the analysis of such structures has led to the conclusion that ANNs can be used as a new alternative for solving such problems.

Ataş İ., Kurt MB., Ataş M. (2013); In this study, Open Coupled Microstrip Patch Antenna design based on ANN model for frequency values between 1 GHz and 3.5 GHz was carried out. A total of 500 patch antennas with different geometric structures were designed with the Electromagnetic Field Simulator software using the Finite Element Method and the resonant frequency value of each antenna was designed. The ANN model developed on the basis of the Levenberg Marquard learning algorithm was trained with the examples realized with the Electromagnetic Field Simulator and its accuracy was measured using the test data set that it did not see during the training period. The 5-fold crossover accuracy method was used to measure the success of the developed ANN model. As a result, in terms of time efficiency, the proposed method was found to work at least 100 times faster than the Electromagnetic Field Simulator software.

Microstrip Patch Antennas

Microstrip patch antennas that fulfill many wireless system requirements are widely used in handheld mobile devices and wireless communication systems. According to their advantages, these antennas find wider applications in mobile and satellite communications.

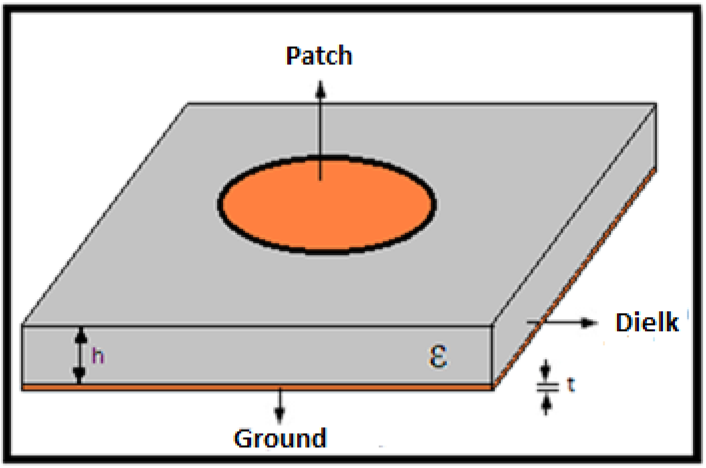
Advantages of microstrip patch antennas; light profile, low scattering interference, linear and circularly polarized radiation with small changes in feed position, easy construction of two or more frequency antennas, no need for cavity support, suitable mounting on planar and non-planar surfaces, simple and inexpensive to produce using modern printed circuit technology, mechanically strong when mounted on solid surfaces, compatible with MMIC (Monolithic Microwave Integrated Circuit Design) designs and regular conductor structures, solid state devices such as oscillators, amplifiers, variable attenuators, switches, modulators, mixers, phase shifters, etc. can be added to the substrate of microstrip antennas and composite systems can be developed, feeder lines and matching circuits can be manufactured at the same time with the antenna, and when modes are selected, compound systems can be developed. etc., solid state devices such as oscillators, amplifiers, variable attenuators, switches, modulators, mixers, phase shifters, etc. can be added to the substrate of microstrip antennas, feeder lines and matching circuits can be produced at the same time with the antenna, and when modes are selected; resonance frequency, polarization, pattern and impedance are versatile (Dundar, 2012).

In addition, microstrip patch antennas have significant disadvantages. These include; very narrow bandwidth, low efficiency and low power, high Q (sometimes more than a hundred), insufficient polarization purity, insufficient scattering performance, spurious feed radiation, large physical dimensions of designs at low frequencies (Dundar, 2012).

The applications of microstrip patch antennas can be listed as wireless systems, satellite communications, defense industry systems, biomedical systems, environmental instrumentation and remote sensing systems, mobile communications, doppler and radars, guided missiles.

Physically, a microstrip patch antenna in its simplest form, as shown in Figure 1, consists of a metal patch radiating on a dielectric base covered with a metal called a ground plane. The metal parts are usually selected from a good conductive metal such as copper or gold. The thickness (t) of the ground and the patch varies between 50 µm and 200 µm. The thickness of the dielectric structure h varies between 0.25 mm and 25 mm.

Figure 1
Microstrip Patch Antenna Structure



The first step in microstrip antenna design is the selection of a suitable dielectric base. The dimensions and electrical properties of the dielectric base, which physically occupies the largest volume in the microstrip antenna, are very important for antenna performance. The dielectric base acts as a part of the transmission line that mechanically supports the circuit elements on the microstrip antenna by providing ease of assembly. Its dielectric permittivity and thickness determine the resonant frequency, resonant resistance and other electrical properties of the antenna (Sainati, 1996). Researchers have found that the most important parameters affecting the performance of microstrip antennas are the dielectric constant of the dielectric material and the tolerance values specified by the manufacturers for this dielectric. Therefore, materials with known dielectric constants are preferred in designs.

There are a large number of base materials used on the market. Their dielectric constant ranges from 1.7 to 25 (Traut, 1980; Olyphant & Nowicki, 1980). However, for high antenna performance, the dielectric constant εr should be less than 3 (Balanis 1982).

Microstrip Antenna Feed Type

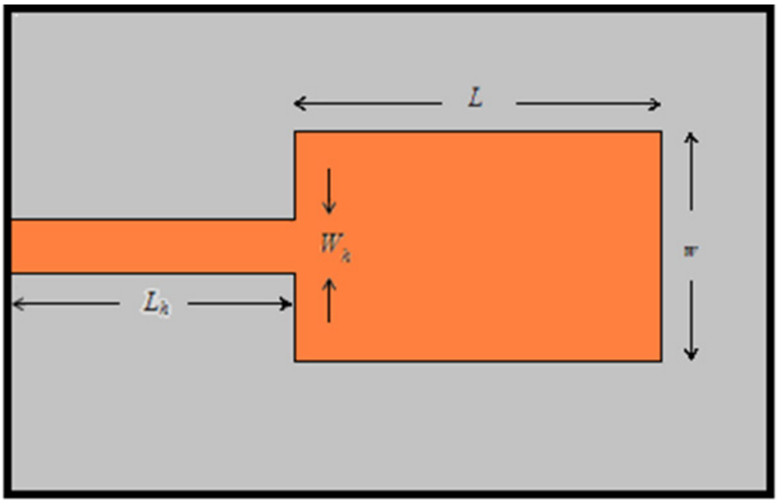
The most common techniques used for feeding microstrip patch antennas are microstrip line feed and coaxial line feed. Impedance matching is the most important factor to be considered for maximum power transfer in microstrip antennas. The feed line impedance is fixed and is 50 Ω. If the antenna impedance is different, a matcher must be installed between the feed line and the antenna to ensure impedance matching.

Microstrip Line Feed

One of the original excitation techniques for microstrip patch antennas is the edge feed or microstrip line feed technique (Munson 1974). As shown in Figure 2, a microstrip line of length Lh and width Wh is directly connected to a rectangular patch conductor of dimensions L and W using a printed circuit.

Figure 2

Edge-Fed Microstrip Patch Antenna Demonstration

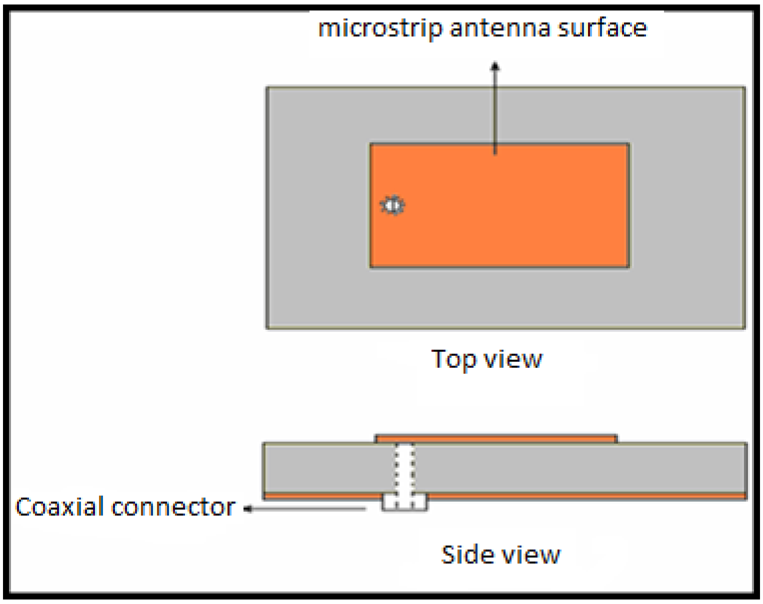


Coaxial Line Feed

Feeding a microstrip patch antenna with a coaxial coupler is another original form of excitation technique proposed in the 1970s. Coupling power through a coaxial coupler is one of the most fundamental mechanisms for microwave power transfer. A probe is usually used as a coaxial coupler. Figure 3 shows a coaxial line feed diagram fed by a coaxial coupler.

Figure 3

Microstrip Antenna With Coaxial Line Feed

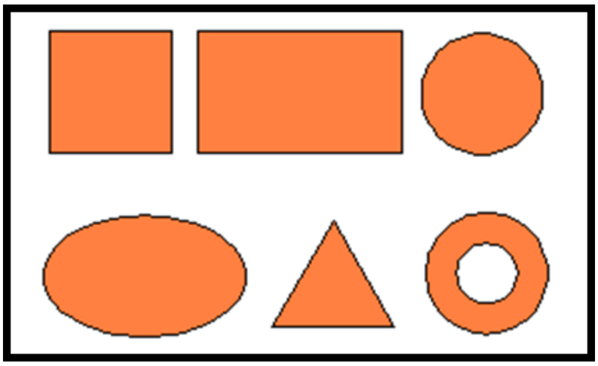


The coaxial feed pin is usually the inner conductor of a coaxial line. This is why probe feed is also known as coaxial feed. The coaxial conductor is connected from the ground side of the patch antenna, and the center conductor of the coax is soldered to the metal of the patch after passing the base.

Microstrip Antenna Patch Types

In microstrip antennas, the radiating part of the patch can be in different geometric shapes and sizes. As shown in Figure 4, there are patch antennas designed in square, rectangular, circle, triangular, etc. shapes.

Figure 4
Common Patch Shapes Used İn Microstrip Antennas

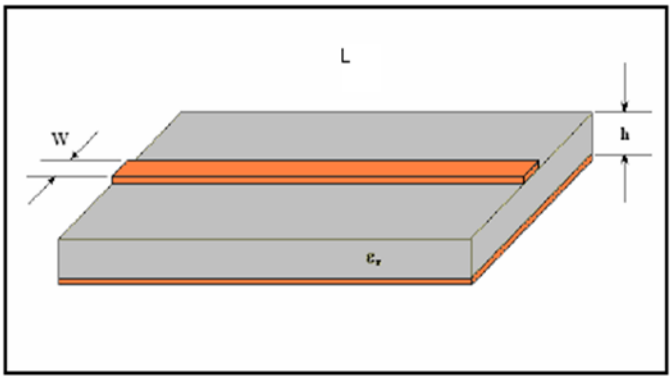


In microstrip patch antennas, studies have shown that the bandwidth is directly proportional to the patch volume. In other words, bandwidth can be increased by increasing the patch size. However, this can also lead to some disadvantages. For example, since the patch length is related to the wavelength of the designed frequency, the size can be increased by reducing the dielectric coefficient of the base. Considering that the feed line will emit more energy in this case, attention should be paid to the cross-polarization and side beam levels in the radiation. It should also be taken into account that increasing the patch width may lead to the excitation of higher order modes in the cavity. Increasing the base thickness increases the surface waves and losses at the surface. For these reasons, the addition of parasitic elements on the base is preferred rather than increasing the patch width (Yazgan 2006).

Rectangular Microstrip Patch Antenna Design

The rectangular microstrip patch antenna is the simplest microstrip patch structure. As shown in Figure 5, the basic antenna element is a conducting strip of dimensions (LxW) on a base of thickness h and dielectric constant εr, with the back side covered by the ground plane and the extension of the transmission line.

Figure 5
Basic Microstrip Patch Structure Used İn Microstrip Antennas



W Patch width can be found by the following formula (James 1989).

$$W = \frac{c}{2f_r}\left(\frac{\varepsilon_r + 1}{2}\right)^{-\frac{1}{2}} \quad (1)$$

In the formula c is the propagation speed of light in a vacuum, εr is the dielectric constant of the material and fr is the operating frequency.

L The patch length is obtained by subtracting the fringe field length (Δ) from the half-wavelength length.

$$L = \frac{c}{2 f_r \sqrt{\varepsilon_e}} - 2\Delta l \qquad (2)$$

In the formula εe is the effective dielectric constant and for (w/h) > 1;

$$\varepsilon_e = \frac{\varepsilon_r + 1}{2} + \frac{\varepsilon_r - 1}{2}\left[1 + \frac{12t}{W}\right] \qquad (3)$$

(Schneider, 1969). The value t in the equation is the microstrip line thickness.

The line expansion Δ is;

$$\Delta l = 0.412h \frac{(\varepsilon_e + 0.3)\left(\frac{W}{h} + 0.264\right)}{(\varepsilon_e - 0.258)\left(\frac{W}{h} + 0.8\right)} \qquad (4)$$

is expressed as (Hammerstad & Bekkadal 1975).

In microstrip lines, the characteristic impedance depends on the width of the line and the thickness of the dielectric base. The formulas for the calculation of the effective dielectric coefficient and characteristic impedance are given in the following equations

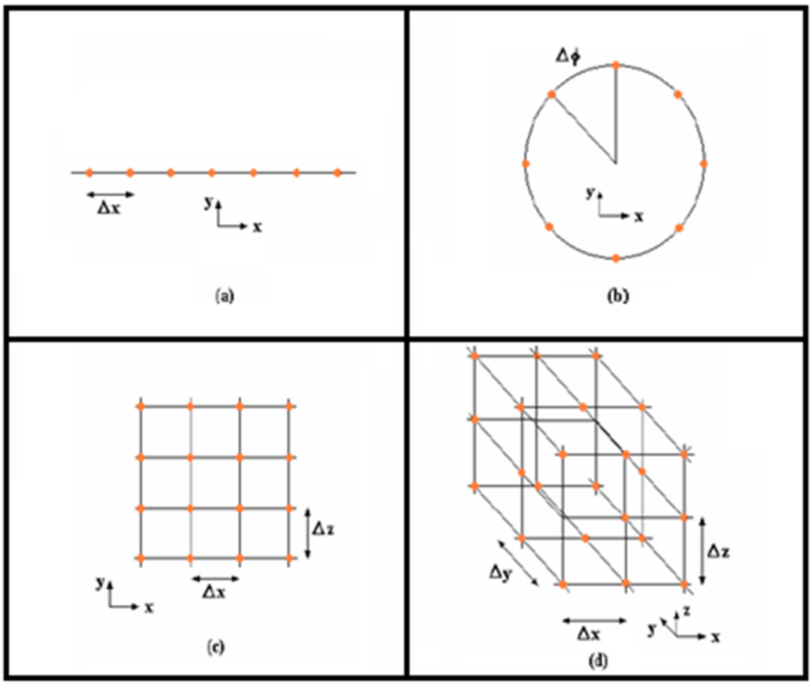
$$.\,\varepsilon_{eff} = \begin{cases} \dfrac{\varepsilon_r + 1}{2\left(1 - \dfrac{B}{C}\right)^2} & \dfrac{W}{h} < 1.3 \\[2ex] \dfrac{\varepsilon_r + 1}{2} + \left(\dfrac{\varepsilon_r - 1}{2}\right)\left(1 + \dfrac{10h}{W}\right)^{-\frac{1}{2}} & \dfrac{W}{h} \geq 1.3 \end{cases} \qquad (5)$$

Array Antennas

The radiation pattern of a single element antenna is relatively broad, but the single element provides low gain. To meet the needs of long-distance communication, it is necessary to design high gain antennas for most applications. The gain of the antenna can be increased by increasing the electrical size of the antenna. However, the antenna gain can also be increased by placing the radiating element in a specific electrical and geometrical arrangement. This structure consisting of more than one radiating element is called an antenna array. The radiation pattern of an array of identical antenna elements depends on the geometrical shape of the array (linear, circular, spherical, etc.), the distance between the array elements, the amplitude and phase of the feed of the array elements, and the radiation pattern of the array elements alone (Balanis C. A., 1997).

Microstrip antenna arrays can be designed as one-dimensional, two-dimensional or three-dimensional, depending on the space dimensions in which radiation is desired. Figure 6 shows the array antenna geometries.

Figure 6
Microstrip Array Antenna Geometries, a) One Dimensional Uniform Linear Array, b) Circular Dimensional Array, c) Two Dimensional Array, d) Three Dimensional Array



The structure shown in Figure 6 a is a one-dimensional uniform linear array geometry for azimuthal beamforming in the horizontal plane only. These arrays are the most basic structures used for beamforming in azimuth. The structure shown in Figure 6 b is a circular array geometry representation, which is also used only for azimuthal beamforming in the horizontal plane. Figure 6 c shows a two-dimensional array antenna geometry that can be shaped in both azimuth and elevation angles. Figure 6 d shows a three-dimensional array antenna geometry that can be shaped in both azimuth and elevation angles. Two and three dimensional array antenna geometries are generally preferred in densely populated and indoor environments (Çakır G., 2004).

Artificial Neural Networks (ANNs)

Artificial neural networks have a theoretical structure inspired by the biological structure of the human nervous system. That is, artificial neural networks are composed of basic elements that function as neurons, which are nerve cells in the human nervous system. These elements are organized similar to the anatomy of the human brain. In addition to this great similarity, neural networks have many surprising features of the human brain.

Learning

Artificial neural networks can change their output behavior according to the inputs they receive from the environment. In the learning process, when a certain input is given to the network, the network has to modify itself in order to produce consistent responses. Various learning algorithms have been developed to train artificial neural networks. Each of these algorithms has its own strengths and weaknesses.

Generalization

A network that has been taught can be, to some extent, insensitive to small changes in the inputs given to it. That is, it always reacts in the same way. This ability is important in the real world to be able to recognize inputs that are slightly distorted by factors from the environment. This is a system that goes beyond the logic used in computers; it is a system developed to understand the imperfect world we live in. ANNs do generalization automatically because of their structure.

Summarization

Some artificial neural networks are capable of extracting the essence of a given set of input information. For example, a network may be given various distorted forms of the letter ‘A’ as input. After sufficient teaching, the network will be able to give a proper letter ‘A’ in response to a corrupted letter ‘A’. In a sense, the network will have learned to produce something it has never seen or learned before. Here, the ability to create ideal prototypes is an important and useful human trait. Today, the use of this feature in ANNs is on the agenda. ANNs, of course, are not suitable for tasks such as calculating payrolls, which is what computers do. However, in the field of pattern-recognition, which is difficult or very limited for traditional computers, artificial neural networks have found a wide range of applications (Burr D.J., 1987).

Biological Structure

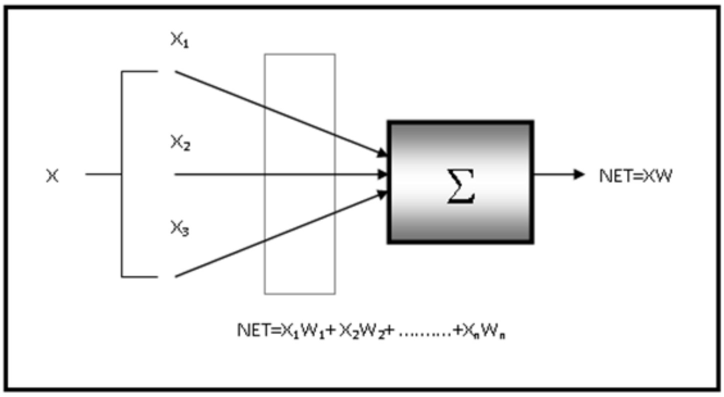
Artificial neural networks are inspired by biological neurons and researchers have studied the organization of the brain when thinking about the network shape and algorithm. However, knowledge about the brain’s working system is very limited. The opportunity to guide those working on this subject is also very limited. In this respect, network design scientists have been forced to go beyond the existing biological knowledge and look for concepts that will help them find useful functions. In many cases this important change ignores biological facts, the brain becomes virtual, and networks cannot be organically applied to brain anatomy and function.

Despite this subtlety, comparisons between ANNs and the brain reveal similarities, even though they are not biologically very similar. ANN functions are reminiscent of human perception. For this reason, it is very difficult to avoid making analogies. The estimated 100 billion neurons in the brain have about 1 quadrillion connectors. Each neuron shares many common characteristics with other cells in the body. But the role of the neurons that make up the brain’s communication system is to receive, process and transmit electrochemical signals in neural networks.

Artificial Neuron

The artificial neuron is designed to mimic the input, processing and output characteristics of a biological neuron. Here each input is multiplied by its own weight and all these multiplications are summed up. This sum can be likened to synaptic strength. This sum is used to determine the activation level of the neuron. Figure 7 shows this model.

Figure 7
Artificial Neuron

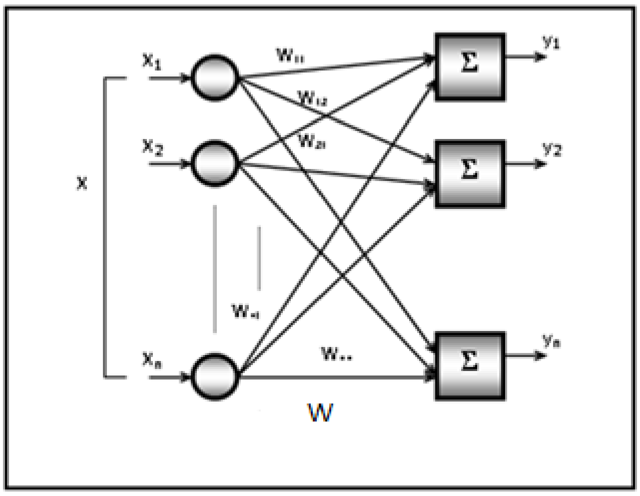


Although network sequences vary widely, most are based on the representation in Figure 7. Here the input set is represented as x1, x2, ..., xn and applied to the artificial nerve. These inputs are summed to form the vector X and sent as signals to the biological synapses. Each signal is multiplied by its associated weight value w1,w2,...,wn and transferred to the block denoted by Σ. Each weight represents the strength of a single biological synaptic connection. Here the sum of the set of weights forms a vector called W.

Single Layer ANN

The power and success of an ANN consisting of an artificial neuron and the interconnection of neurons will not be as perfect as that of a real nervous system. The simplest network structure is shown in Figure 8. Round representations only distribute the inputs to the neuron and do not perform any analysis. Therefore, in order to express the inputs differently from the neurons, it is necessary to represent them as round. Here, each element of X, the set of inputs, is connected to the ANN with a weight value. The first ANNs created were not so complex, with each neuron providing a weighted sum of the inputs to the network through a simple calculation. Artificial and biological neural networks may have some connections deleted, but complete connections are preferred to generalize the structure of the ANN. Where W is the weight matrix, m is the number of inputs and n is the number of neurons.

Figure 8
Single Layer Neural Networks

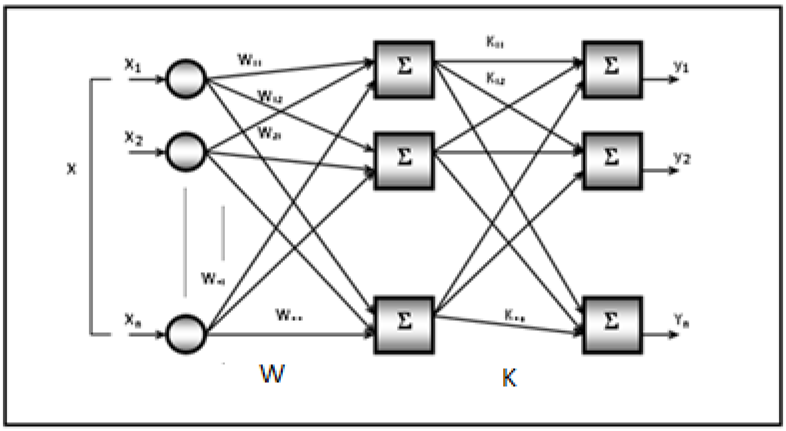


Multilayer ANN

Larger and more complex networks also need more computational capacity. ANNs are built to reflect a part of the brain. Multilayer networks have been proven to be better than single-layer networks and all studies have focused on them. Multilayer networks are a combination of single layer networks. Figure 9 shows the multilayer network structure.

Figure 9

Double Layer Neural Networks



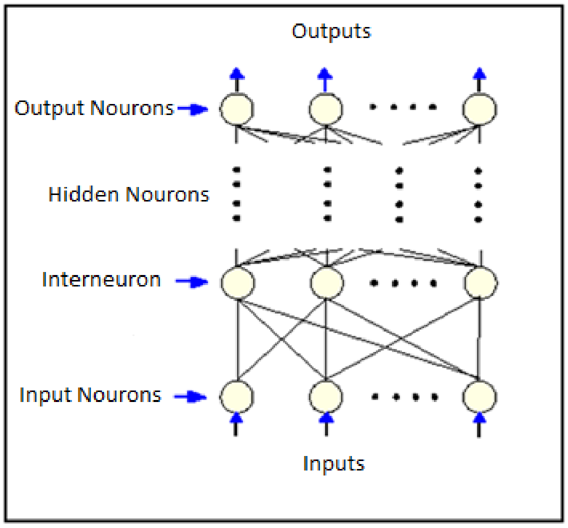
In multilayer networks, if a non-linear activation function is used, the computational power of these networks will not be perfect. If the activation function is linear, the output can be calculated as $(XW_1)W_2$ where X is the input vector, W_1 is the first weight matrix and W_2 is the second weight matrix. The terms are regrouped until the matrix product converges and written as $X(W_1W_2)$.

Multilayer Perseptron (MLP) ANN Structure

The MLP is a type of feed-forward ANN consisting of an input layer, one or more intermediate layers and an output layer, as shown in Figure 10. The processing elements in the input layer distribute the input signals to the processing elements in the intermediate layer. The processing elements in the intermediate layer sum the inputs from the input layer after multiplying them by the connection weights and pass them through a transfer function to the output layer. The processing elements in the output layer act like the intermediate layer elements and calculate the output value of the network.

Figure 10

A Common MLP Structure



In the so-called feed-forward neural network model, the information flow is forward and there is no feedback. The number of processing elements in the input layer depends on the number of inputs of the applied problem. The number of intermediate layers and the number of processing elements in the intermediate layers are found by trial and error.

ANN Learning Algorithms

There are many learning algorithms in the literature. These algorithms vary according to ANN structures. Some of them are Delta-Bar-Delta (DBD), Extended Delta-Bar-Delta (EDBD) and Levenberg-Marquardt (LM) learning algorithms.

The Delta-Bar-Delta training algorithm is a heuristic approach to increase the convergence rate of the weights in the MLP learning algorithm developed by Jacobs (1988). Experimental studies have shown that each dimension of the weight space can be very different in terms of error surfaces. In order to identify these differences in error surfaces, each weight of the network must have its own learning coefficient (Jacobs 1988). However, in this approach, the learning coefficient generated may be appropriate for a single weight but not for all weights. However, assigning a learning coefficient to each weight and allowing this learning coefficient to change over time will also provide more degrees of freedom to reduce the convergence time of the network.

The Extended Delta-Bar-Delta (EDBD) algorithm (Minai and Williams 1990) is an extension of the DBD algorithm (Jacobs 1988) and is based on the principle of reducing the training time of MLPs. The EDBD algorithm differs from the DBD algorithm in that it uses heuristic momentum, eliminates large jumps in the weight space, and is fast in avoiding large jumps that exceed the goal of geometric reduction. Levenberg-Marquardt (LM) learning algorithm, a highly successful optimization method, is one of the different learning techniques of the backpropagation algorithm used in learning. Based on the idea of a large number of neighbors, the LM algorithm is a least square estimation method (Levenberg 1944; Marquardt 1963).

Array Patch Antenna Design and Improvement of Antenna Parameters with Artificial Neural Networks

Antenna arrays can be used to increase antenna gain, directivity and bandwidth. Since it is simpler and more practical, identical array elements are chosen in many applications. However, array elements can also be selected in different structures (Özen B., Afacan E., 2014).

Design of 1x4 Rectangular Section Microstrip Patch Antenna at 2500 MHz Frequency

Today, the internet has entered every aspect of our lives and a life without the internet is unthinkable. WiMax (Wordwide Interoperability for Microwave Access), which is a worldwide interoperable microwave access, is an amplified wireless internet access (Wi-Fi) technology in the simplest terms. In this study, a rectangular 1x4 microstrip patch antenna design has been realized at 2500 MHz frequency for WiMax wireless high speed internet access. As in all microstrip antenna designs, material selection is the primary parameter in this design. The FR4 material properties selected for the design are given in Table 1.

Table 1

FR4 Base Material Properties

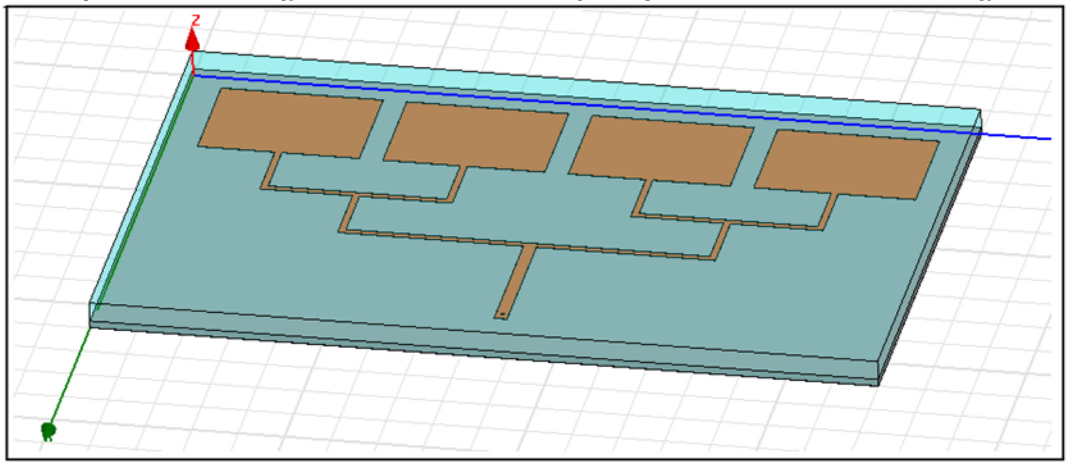
Material Properties	Value
Base	FR4
Dielectric Constant (εr)	4.9
Dielectric Thickness (h)	1.6 mm
Copper Thickness	0.035 mm

The first step in microstrip array antenna design is to create a basic array element that can radiate at 2500 MHz. In this design, the physical dimensions of the microstrip patch antenna are calculated using the design formulas given above.

This design forms the basis of the array antenna. In order to form an array, a half wavelength distance is left between two patches with the same physical properties. Thus, a 2x1 array antenna is formed. Two 2x1 array antennas are combined to form a 1x4 microstrip array antenna. Again, the distance between the patches is half a wavelength. The simulated 1x4 array antenna is given in Figure 11 (Dundar O, 2018).

Figure 11

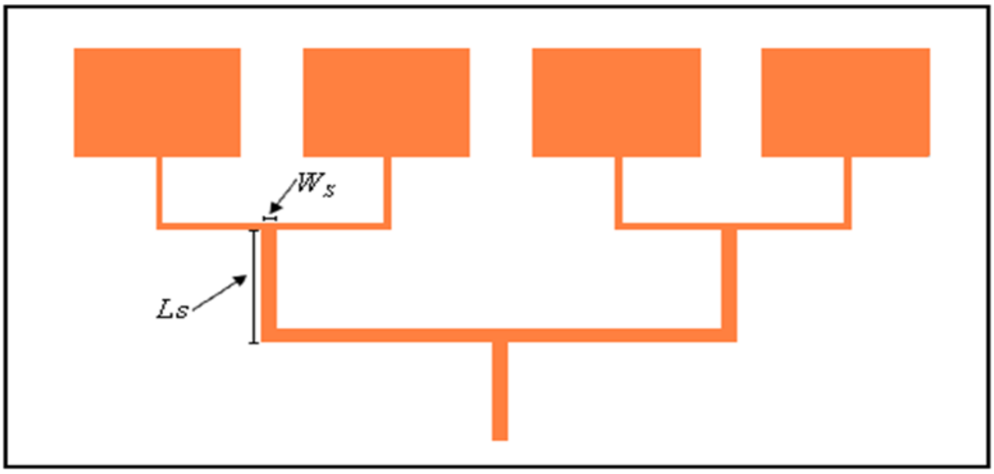
1x4 Array Antenna Designed at 2500 MHz Frequency in HFSS Simulation Program



After this stage, new array antennas were fabricated for refinement and optimization. The best optimization is achieved by ensuring impedance matching in the feed line between the patch and the feed point. For this purpose, the array antenna simulations were started by changing the dimensions of the feed line length Ls and width Ws. The 1x4 array antenna with Ls and Ws dimensions is shown in Figure 4.7. The 1x4 array antenna was designed in the HFSS simulation program by changing the Ls value between 13 mm and 16 mm at 0.5 mm intervals and the Ws value between 1 mm and 1.5 mm at 0.05 mm intervals. The feed path length Ls and feed path thickness Ws values were changed to form the basis of the antenna designed in Figure 12 (Dundar O, 2018).

Figure 12

Electrical Parameter Improvement of 1x4 Array Antenna with 2500 Mhz Resonant Frequency



In this way, 77 array antennas were designed in the HFSS simulation program and the results are given in Table 2.

Table 2

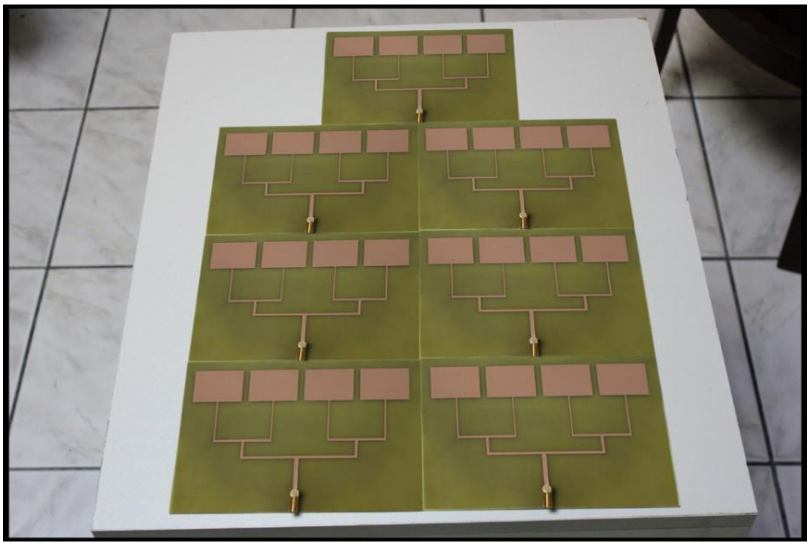
HFSS Result Outputs of 77 Array Antennas Designed (Dundar O, 2018).

Order	L_S (mm)	W_S (mm)	f_{rs} (MHz)	S_{11} (dB)	BW (MHz)	Directivite	Gain	Efficiency
1	16.0	1.50	2650	-31.74	68	7.5153	3.2344	0,7271
2	16.0	1.45	2640	-31.87	74	7.3728	3.1841	0,7341
3	16.0	1.40	2640	-32.88	72	7.4663	3.2627	0,7476
4	16.0	1.35	2640	-35.99	72	7.4630	3.2367	0,7497
5	16.0	1.30	2640	-37.40	73	7.3092	3.1873	0,7480
6	16.0	1.25	2640	-38.00	75	7.4033	3.2067	0,7524
7	16.0	1.20	2640	-30.73	73	7.2396	3.1315	0,7385
8	16.0	1.15	2630	-37.53	77	7.4609	3.2235	0,7698
9	16.0	1.10	2640	-28.15	75	7.4385	3.2654	0,7945
10	16.0	1.05	2630	-26.64	79	7.4316	3.2525	0,7951
11	16.0	1.00	2630	-25.00	75	7.4697	3.2119	0,7866
12	15.5	1.50	2650	-36.27	67	7.6181	3.2937	0,7637
13	15.5	1.45	2650	-29.09	73	7.5078	3.2443	0,7589
14	15.5	1.40	2650	-25.39	71	7.4988	3.2437	0,7492
15	15.5	1.35	2650	-32.63	73	7.2856	3.1436	0,7390
16	15.5	1.30	2650	-25.79	75	7.4862	3.2703	0,7758
17	15.5	1.25	2620	-22.91	72	6.6886	2.7548	0,7375
18	15.5	1.20	2640	-26.83	75	7.4380	3.2086	0,7582
19	15.5	1.15	2640	-26.83	78	7.4514	3.2540	0,7922
20	15.5	1.10	2520	-33.00	62	7.5152	3.3923	0,7966
21	15.5	1.05	2540	-38.92	63	7.6296	3.4057	0,8045
22	15.5	1.00	2540	-49.28	62	7.7745	3.4221	0,8129
23	15.0	1.50	2660	-32.37	72	7.6498	3.3426	0,7864
24	15.0	1.45	2660	-29.00	71	7.6539	3.3376	0,7852
25	15.0	1.40	2650	-29.00	73	7.5870	3.3082	0,7967
26	15.0	1.35	2650	-29.29	73	7.5870	3.3082	0,7967
27	15.0	1.30	2660	-7.80	0	3.6200	0.9268	0,4414

28	15.0	1.25	2650	-25.00	75	7.4741	3.2930	0,8032
29	15.0	1.20	2650	-23.22	78	7.6520	3.3704	0,8117
30	15.0	1.15	2640	-21.41	75	7.5954	3.3265	0,8095
31	15.0	1.10	2640	-23.14	78	7.5468	3.3189	0,8350
32	15.0	1.05	2630	-19.33	76	7.2822	3.1730	0,7964
33	15.0	1.00	2640	-20.00	84	7.6627	3.4056	0,8988
34	14.5	1.50	2665	-21.13	80	7.4787	3.2557	0,7874
35	14.5	1.45	2660	-22.20	73	7.5482	3.2723	0,7909
36	14.5	1.40	2660	-23.38	74	7.7271	3.4170	0,8506
37	14.5	1.35	2660	-23.62	72	7.7271	3.4170	0,8506
38	14.5	1.30	2640	-18.70	75	7.4014	3.2330	0,7972
39	14.5	1.25	2640	-18.89	73	7.4014	3.2330	0,7972
40	14.5	1.20	2650	-20.47	75	7.5349	3.3116	0,8362
41	14.5	1.15	2640	-20.00	77	7.4690	3.2604	0,8199
42	14.5	1.10	2640	-18.39	77	7.4248	3.3014	0,8699
43	14.5	1.05	2650	-18.64	78	7.6282	3.3860	0,8802
44	14.5	1.00	2650	-18.76	79	7.6282	3.3860	0,8802
45	14.0	1.50	2660	-21.09	71	7.2584	3.1177	0,8526
46	14.0	1.45	2610	-32.24	67	5.9829	2.3855	0,8863
47	14.0	1.40	2660	-25.94	71	7.5764	3.3668	0,9050
48	14.0	1.35	2660	-20.70	75	7.5813	3.3094	0,8309
49	14.0	1.30	2660	-18.39	76	7.5015	3.2870	0,8499
50	14.0	1.25	2660	-18.33	76	7.5015	3.2870	0,8499
51	14.0	1.20	2660	-15.12	78	7.6573	3.3986	0,9224
52	14.0	1.15	2650	-17.25	76	7.5107	3.3138	0,8665
53	14.0	1.10	2650	-17.82	80	7.5764	3.3668	0,9050
54	14.0	1.05	2650	-16.56	79	7.6219	3.3970	0,9061
55	14.0	1.00	2650	-16.81	79	7.6219	3.3970	0,9061
56	13.5	1.50	2580	-21.00	52	6.0839	2.3205	0,7784
57	13.5	1.45	2670	-18.45	74	7.4940	3.2724	0,8196
58	13.5	1.40	2670	-18.33	73	7.5766	3.3097	0,8479
59	13.5	1.35	2670	-18.26	74	7.5766	3.3097	0,8479
60	13.5	1.30	2660	-18.20	75	7.7134	3.4283	0,8947
61	13.5	1.25	2660	-18.26	75	7.7134	3.4283	0,8947
62	13.5	1.20	2660	-14.74	73	7.4830	3.3280	0,8945
63	13.5	1.15	2650	-16.31	78	7.5178	3.3165	0,8878
64	13.5	1.10	2660	-15.93	78	7.6350	3.4326	0,9441
65	13.5	1.05	2650	-15.93	79	7.6763	3.4368	0,9557
66	13.5	1.00	2650	-15.87	81	7.6763	3.4368	0,9557
67	13.0	1.50	2660	-17.56	73	7.6337	3.3814	0,8635
68	13.0	1.45	2670	-17.44	73	7.6671	3.3817	0,8859
69	13.0	1.40	2670	-17.38	74	7.6123	3.3755	0,9091
70	13.0	1.35	2670	-16.69	74	7.6123	3.3755	0,9091
71	13.0	1.30	2670	-15.26	74	7.6566	3.4170	0,9284
72	13.0	1.25	2670	-15.32	73	7.6566	3.4170	0,9284
73	13.0	1.20	2660	-12.57	65	7.3898	3.2466	0,9085
74	13.0	1.15	2660	-14.95	73	7.5886	3.3882	0,9228
75	13.0	1.10	2530	-16.76	54	7.9852	3.4213	0,9622
76	13.0	1.05	2660	-14.38	75	7.6643	3.4162	0,9576
77	13.0	1.00	2660	-14.45	76	7.6643	3.4162	0,9576

When these designs are examined, it is seen that each data is different. Within these data, the array designs with the best data results for each Ls were taken and the manufacturing process was started. The array antennas with the lowest return response numbers 6, 22, 23, 37, 46, 56 and 67 in the Table were selected for manufacturing.
For 7 different Ls values, the array antennas with the best result data were fabricated. The fabricated array antenna with row number 6, which performed well, is shown in Figure 4.8. The other array antennas are collectively given in Figure 13 (Dundar O, 2018).

Figure 13
Manufacturing of 7 1x4 Microstrip Array Antennas



Measurements of the array antennas were performed with VNA. The values obtained as a result of the measurements are given in Table 3.

Table 3
Simulation and Measurement Results of Fabricated Array Antennas

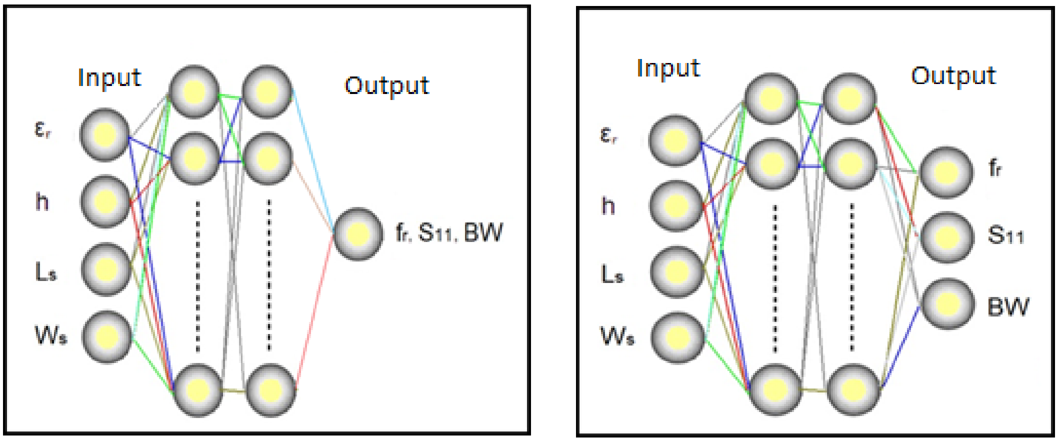
Order	Sequence Number in Table 2	L_S (mm)	W_S (mm)	Simulation Results			Measurement Results		
				S_{11} (dB)	f_r (MHz)	BW (MHz)	S_{11} (dB)	f_r (MHz)	BW (MHz)
1	6	16,0	1,25	-38.00	2640	75	-15,2	2580	85
2	22	15,5	1.00	-49,28	2540	62	-14.0	2612	77
3	23	15.0	1,50	-32,37	2660	72	-15.0	2609	85
4	37	14,5	1,35	-23,62	2660	72	-25.0	2594	86
5	46	14,0	1,45	-32,24	2610	67	-22.0	2610	92
6	56	13,5	1,50	-21.00	2580	52	-26.0	2599	82
7	67	13,0	1,50	-17,56	2660	73	-22,5	2600	178

Calculation of Return Response, Frequency and Bandwidth of 1x4 Rectangular Section Microstrip Patch Antenna at 2500 MHz Frequency Using ANN

In this section, the return response, frequency and bandwidth of a 1x4 rectangular microstrip patch antenna at 2500 MHz are calculated by ANN. The simulation results of 77 array antennas given in Table 4.3 are used to train the ANN structure and 7 measurement results given in Table 4.4 are used to test the ANN structure. The calculations were performed using the "Multilayer Perseptron" (MLP)-ANN network structure. In addition, two network structures were used separately. The first is a single output network structure (Figure 14a) and the second is a two-layer three output network structure (Figure 14b). Four input parameters were used in both structures. The input parameters are dielectric constant (εr), dielectric base thickness (h), feed line length (LS) and feed width (WS). The output parameters are resonance frequency (frs), return response (S11) and bandwidth (BW). Levenberg-Marquardt was used as the training algorithm in both ANN network structures (Dundar O, 2018).

Figure 14

2500 Mhz MLP-ANN Network Structures a) Single Output Network Structure, b) Three Output Network Structure



a) b)

Table 4 shows the input output test set and ANN test results for the single output network structure for the calculation of the resonant frequency (fr). The differences obtained from the results are indicated. Table 5 shows the input output test set and ANN test results of the single output network structure for the calculation of the return response (S_{11}). The differences obtained from the results are indicated. Table 6 shows the input output test set and ANN test results for the single output network structure for bandwidth (BW) calculation. The differences obtained from the results are indicated. Table 7 shows the input output test set and ANN test results of the three output network structure for the calculation of resonant frequency (fr), return response (S_{11}) and bandwidth (BW). The differences obtained from the results are indicated. Table 8 shows the number of inputs, outputs, iterations and intermediate layers for the single output and three output network MLP-ANN structure (Dundar O, 2018).

Table 4

Input Output Test Set and ANN Test Results of Single Output Network Structure for Calculation of Resonant Frequency (fr)

Inputs				Out f_r (GHz)	ANN f_r (GHz)	Difference Absolute Error (GHz)
ε_r	h (mm)	Ls (mm)	Ws (mm)			
4.900	1.600	16.000	1.250	2.6400	2.6399	0.0001
4.900	1.600	15.500	1.000	2.5400	2.5400	0.0000
4.900	1.600	15.000	1.500	2.6600	2.6611	0.0011
4.900	1.600	14.500	1.350	2.6600	2.6588	0.0012
4.900	1.600	14.000	1.450	2.6100	2.6100	0.0000
4.900	1.600	13.500	1.500	2.5800	2.5800	0.0000
4.900	1.600	13.000	1.500	2.6600	2.6600	0.0000
Total Test Absolute Error Average						**0.000343**

Table 5
Input Output Test Set and ANN Test Results of Single Output Network Structure for Calculation of Return Response (S_{11})

Inputs				Out S_{11} (dB)	ANN S_{11} (dB)	Difference Absolute Error (dB)
ε_r	h (mm)	Ls (mm)	Ws (mm)			
4.900	1.600	16.000	1.250	-38.0000	-38.0000	0.0000
4.900	1.600	15.500	1.000	-49.2800	-49.2800	0.0000
4.900	1.600	15.000	1.500	-32.3700	-32.3716	0.0016
4.900	1.600	14.500	1.350	-23.6200	-23.6187	0.0013
4.900	1.600	14.000	1.450	-32.2400	-32.2396	0.0004
4.900	1.600	13.500	1.500	-21.0000	-21.0002	0.0002
4.900	1.600	13.000	1.500	-17.5600	-17.5663	0.0063
Total Test Absolute Error Average						**0.00140**

Table 6
Input Output Test Set and ANN Test Results for Single Output Network Structure for Bandwidth (BW) Calculation

Inputs				Out BW (MHz)	ANN BW (MHz)	Difference Absolute Error (MHz)
ε_r	h (mm)	Ls (mm)	Ws (mm)			
4.900	1.600	16.000	1.250	75.00	75.00	0.00
4.900	1.600	15.500	1.000	62.00	62.00	0.00
4.900	1.600	15.000	1.500	72.00	72.00	0.00
4.900	1.600	14.500	1.350	72.00	72.10	0.10
4.900	1.600	14.000	1.450	67.00	67.00	0.00
4.900	1.600	13.500	1.500	52.00	52.00	0.00
4.900	1.600	13.000	1.500	73.00	73.00	0.00
Total Test Absolute Error Average						**0.01428**

Table 7
Input Output Test Set and ANN Test Results of Three Output Network Structure for Calculation of Resonance Frequency (fr), Return Response (S_{11}) and Bandwidth (BW)

Inputs				Outputs			ANN Outputs			Difference Absolute Error		
ε_r	h (mm)	Ls (mm)	Ws (mm)	f_r (GHz)	S_{11} (dB)	BW (MHz)	f_r (GHz)	S_{11} (dB)	BW (MHz)	f_r (GHz)	S_{11} (dB)	BW (MHz)
4.900	1.600	16.000	1.250	2.6400	-38.0000	75.0000	2.6350	-38.0000	71.6000	0.0050	0.0000	3.4000
4.900	1.600	15.500	1.000	2.5400	-49.2800	62.0000	2.5390	-49.2800	61.5000	0.0010	0.0000	0.5000
4.900	1.600	15.000	1.500	2.6600	-32.3700	72.0000	2.6614	-32.3700	74.0000	0.0014	0.0000	2.0000
4.900	1.600	14.500	1.350	2.6600	-23.6200	72.0000	2.6535	-23.6200	71.7000	0.0065	0.0000	0.3000
4.900	1.600	14.000	1.450	2.6100	-32.2400	67.0000	2.6382	-32.2401	69.8000	0.0282	0.0001	2.8000
4.900	1.600	13.500	1.500	2.5800	-21.0000	52.0000	2.5787	-21.0000	59.0000	0.0013	0.0000	7.0000

4.900	1.600	13.000	1.500	2.6600	-17.5600	73.0000	2.6601	-17.5600	73.0000	0.0001	0.0000	0.0000
Total Test Absolute Error Average										**0.00621**	**0.000014**	**2.2857**

Table 8

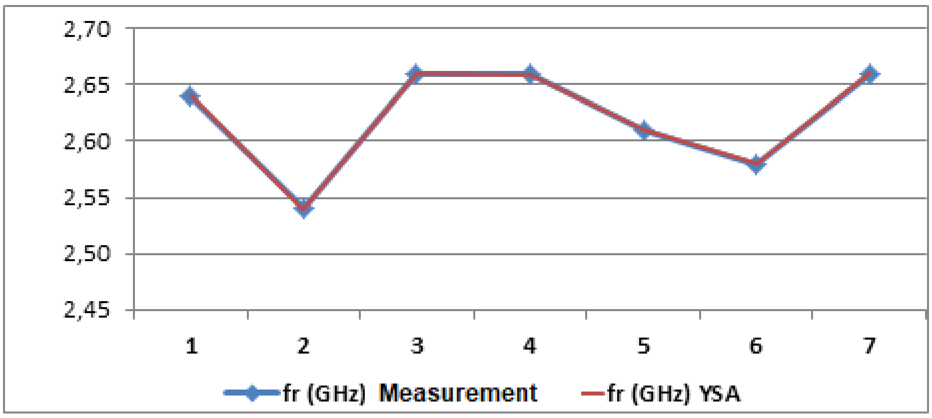
Three Output ANN Network Structure of frs, S11 and BW Output Parameters for 2500 Mhz Operating Frequency

ANN Network Structure	Inputs	Outputs	Interlayer Neuron Numbers		Number of Training Iterations
Single output structure	ε_r, h, L_S, W_S, $f_{rç}$	f_{rs}	8	6	3000
	ε_r, h, L_S, W_S, $f_{rç}$	S_{11}	8	7	2000
	ε_r, h, L_S, W_S, $f_{rç}$	BW	8	8	400
Three output structure	ε_r, h, L_S, W_S, $f_{rç}$	f_{rs}, S_{11}, BW	10	9	2000

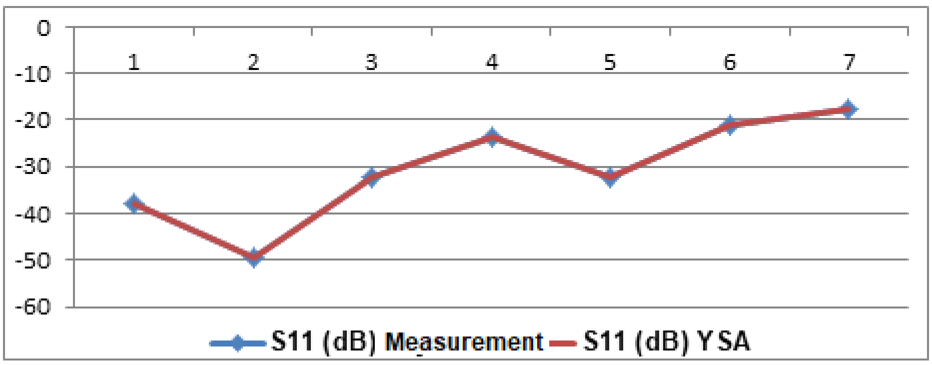
Figure 15 (a, b, c) shows 7 measurements and ANN test curves obtained from a single output network structure and Figure 16 (a, b, c) shows a three output network structure (Dundar O, 2018).

Figure 15

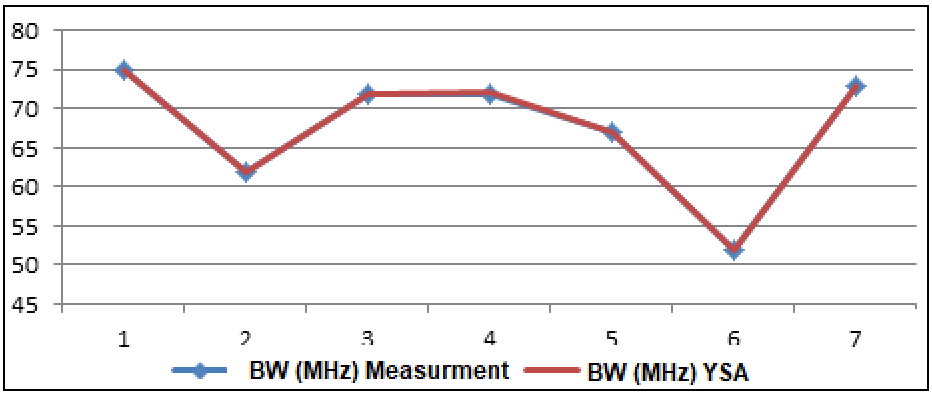
Measurement and ANN Test Curves Obtained with Single Output CCP-ANN Network Structure, a) fr, b) S_{11}, c) BW



a)



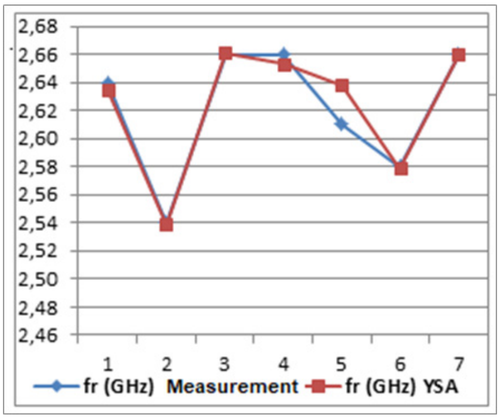
b)

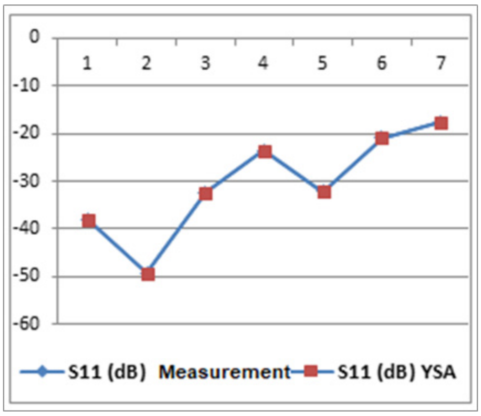


c)

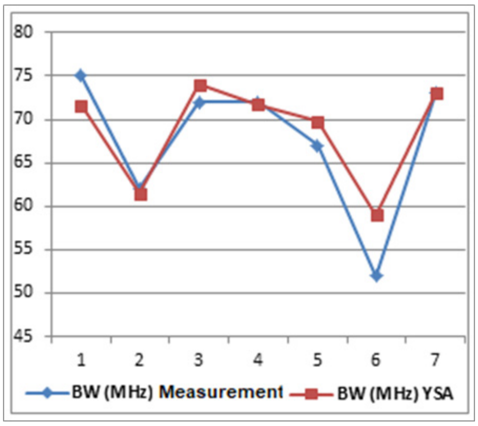
Figure 16

Measurement and ANN Test Curves Obtained with Three Output MLP-ANN Network Structure, a) fr, b) S_{11}, c) BW





a) **b)**



c)

Conclusion

As can be seen from the tables and graphs, the return response, frequency and bandwidth of the 1x4 rectangular section microstrip patch antenna at 2500 MHz frequency trained with the Levenberg-Marquardt training algorithm in single output and three output MLP-ANN network structure were calculated and very good results were obtained. The results show that ANN can calculate these parameters.

Referances

Ataş İ., Kurt MB., Ataş M., 2013, Dicle Üniversitesi Mühendislik Dergisi, Cilt 4, Sayı 2, Sayfa 69-75, Kasım 2013, Diyarbakır

Bahl I.J., Bhartia P., 1980, MigrostripAntennas,DedhamMA,Artegh House, London.

Balanis C. A., 1997, *Antenna theory Analysis and Desing*, 2nd ed. John Wiley and Sons, New York 1997.

Balanis CA, 1982. *Antenna theory: analysis and design*

Burr, D.J., 1987, Experiments With A Connectionist Text Reader, In Proceedings of *The First International Conference on Neural Networks*, pp. 717-724, San Diego, 1987

Çakır G., 2004, *Gezgin haberleşme sistemleri için hüzme yönlendirmeli mikroşerit dizi anten tasarımı: analitik hesaplamalar,bilgisayar benzetimleri ve ölçmeler*, Doctoral thesis, Kocaeli Üniversitesi.

Dundar O, Uzer D, Gultekin SS, Bayrak M, (2012). Effects of microstrip feed line width on 1× 4 rectangular microstrip antenna array electrical parameters and estimation with artificial neural networks. *Progress in Electromagnetic Research (PIERS)*. Kuala Lumpur, Malaysia: 656.

Dundar O, 2018. *Determınatıon Of Desıng Parameters Of Patch Antenna Array Usıng Neurel Networks.* Doctoral thesis, Secuk University.

Garg R., Bhartia P., Bahl I. and Ittipiboon A.,1995, *Microstrip Antenna Design*

Gültekin S.S., 2002, *Çeşitli Mikroşerit Antenlerin Karakteristik Parametrelerinin Farklı Tipteki Algoritmalarla Eğitilen Yapay Sinir Ağları İle Hesaplanması(2002)*, Doktora Tezi, Selçuk Universty

Howell, J. Q. (1972). *Microstrip antennas*. In Antennas and Propagation Society

Huang J, 1983. *The Finite Ground Plane Effect on the Microstrip Antenna Radiation-Patterns*. IEEE T Antenn Propag, 31, 4, 649-53.

Huynh MC, Stutzman W. Ground plane effects on planar inverted-F antenna (PIFA) performance. Microwaves, *Antennas and Propagation*, IEE Proceedings, 209-13. International Symposium, 1972. AP-G. Digest, s. 177–180.

Jacobs R.A., 1988, Increased Rate of Convergence Through Learning Rate Adaptation, *Neural Networks*, 1, s. 295-307, 1988.

Kim D., Kim M., Tanaka M., Matsugatani K., 2006, A broadband planar patch array resonator antenna, *Microwave Conference*, 2006. APMC 2006. Asia-Pacific, IEEE

Levenberg K., 1944, A Method For the Solution of Certain Nonlinear Problems in Least Squares, Quart.Appl. Math., 2, s. 164-168, 1944.

Marquardt D.W., 1963, An Algorithm For Least-Squares Estimation Of Nonlinear Parameters, J. Soc. Ind. Appl. Math., 11, s. 431-441, 1963.

Minai A.A., Williams, R.D., “Accelaration of Backpropagation Through Learning Rate and Momentum Adaptation”, *International Joint Conference on Neural Networks*, 1, s. 676-679, 1990.

Munson R., “Conformal microstrip antennas and microstrip phased arrays”, IEEE *Transactions on Antennas and Propagation*, 22:74–78, 1974.

Noh H.S., Yun J.S., Kim J.M., Jeon S., “Microstrip patch array antenna with high gain and wideband for Tx/Rx dual operation at Ku-band”, *Antennas and Propagation Society International Symposium*, IEEE, 2004.

Özen B., Afacan E., 2014 Elektrik Elektronik ve Biyomedikal Mühendisliği Sempozyumu, 27-29 Kasım ELECO, 2014 Bursa

Panayi P., Al-Nuaimi M., Ivrissimtzis L., "Tuning techniques for the planar inverted-F antenna", *Antennas and Propagation*, IEE National Conference, 259-62, 1999.

Roy G.G., Chakraborty P.N. 2011., Desing of Non-Uniform Circular Antenna Arrays Using a Modified Invasive Weed Optimization Algorithm. IEEE, *Antennas and Propagation Magazine*, 59, 110-118

Sanford, G. G. (1974), Conformal microstrip phased array for aircraft test with ats. *6. In National Electronics Conference, 30 th, Chicago*, Ill, Proceedings, volume 29, pp. 252–257.

Shah M., Suaidi M.K., Aziz M., Rose M.R.C., Kadir M., Ja'afar A.S., Sidek, M., Rahim M.K.A.,"Dual Polarization Microstrip Patch Array Antenna Telecommunication", IST, IEEE, *International Symposium*, 2008.

Tong KF, Luk KM, Lee KF, Lee RQ., A broad-band U-slot rectangular patch antenna on a microwave substrate. IEEE, *Antenn Propagation*, 48, 6, 954-60, 2000.

Traut, G., Clad laminates of ptfe composites for microwave antennas, *Microwave Journal*, 23:47–51, 1980.

Türkmen M., Yıldız C., Sağıroğlu Ş., "Sonsuz Uzunluktaki Dielektrik Tabana Sahip Geleneksel Eş Düzlemli Dalga Kılavuzlarının Quasi-Statik Analizlerinin Yapay Sinir Ağlari İle Gerçekleştirilmesi", *Elektrik Elektronik Bilgisayar Mühendisliği 10. Ulusal kongresi,* s389-392, 2003.

Uğurlu E., *Çok Amaçlı Açıklığı Daralan Yarık Anten Tasarımı ve Performans Artırımına Yönelik Anten Parametrelerinin Optimizasyonu* (2016), Doktora Tezi. Selçuk Üniversitesi.

Uzer D., 2016, *Geniş Band Mikro Şerit Yama Anten Tasarımları İçin Uygun Yöntemlerin Araştırılması*, Doktora Tezi. Selçuk Üniversitesi.

Yazgan E, 2006, Mikroşerit Antenler. Elektrik Mühendisliği, 2006, 262-8.

Yu A., Zhang X., 2002. A broadband patch antenna array for wireless LANs, *Antennas and Propagation Society International Symposium*. IEEE

About the Authors

Özgür DÜNDAR, PhD, works at Necmettin Erbakan University, Department of Astronautical Engineering. He graduated from the Electrical and Electronics Engineering Department of Selcuk University. He worked as an Automation Engineer for a while. His master's and doctorate degrees are from Selçuk University, Institute of Science and Technology, Department of Electrical and Electronics Engineering. Special fields of study are Automation, Robotic, Communication, Electromagnetic and Micro Strip Patch Antenna designs.

Email: ozdundar@erbakan.edu.tr , **ORCID:** 0000-0002-4142-4446

Sabri KOÇER, PhD, He graduated from the Electrical Engineering Department of Selcuk University. He completed his graduate and his doctorate in Gazi University. Currently, Necmettin Erbakan University, Faculty of Engineering, Computer Engineering is working. Electronics, Computer, Telecommunication, Signal Processing and Biomedical studies in the area.

Email: skocer@erbakan.edu.tr , **ORCID:** 0000-0002-4849-747X

Similarity Index

The similarity index obtained from the plagiarism software for this book chapter is 19**%..**

Computer Vision: Object Recognition and Classification

Yunus Emre GÖKTEPE
Necmettin Erbakan University

Yusuf UZUN
Necmettin Erbakan University

To Cite This Chapter

Dündar, Ö., & Koçer, S. (2024). Optimization of Microstrip Patch Antenna Parameters with Artificial Neural Networks. In S. Koçer & Ö. Dündar (Eds.), *Intelligent Systems and Optimization in Engineering* (pp. 163-171). ISRES Publishing.

Introduction

The advent of computer vision has ushered in a transformative era in technology, with its potentiality to augment the accuracy and efficiency of operations across distinct domains (Szeliski, 2022). This field, rooted in the interdisciplinary convergence of computer science, artificial intelligence, and signal processing, has evolved rapidly, offering innovative solutions to complex visual perception challenges. One of the most promising aspects of computer vision is its capacity for real-time applications, which are poised to become more prevalent with the ongoing advancements in technology (Huang et al., 2017).

A remarkable tendency in the realm of computer vision is the escalating use of autonomous systems. Autonomous vehicles and drones, for instance, are increasingly leveraging sophisticated computer vision techniques for tasks such as environmental sensing, object recognition, and classification. These technologies enable machines to interpret and interact with their surroundings, facilitating safer and more efficient operations. The proliferation of these systems is indicative of a future where autonomous systems are an integral part of our daily lives, transforming industries such as transportation, agriculture, and logistics.

The healthcare industry is also experiencing a revolution driven by computer vision. Medical imaging, powered by advanced computer vision algorithms, is enhancing diagnostic accuracy and treatment planning. For example, automated image analysis can detect diseases at early stages, support radiologists in interpreting medical images, and provide personalized treatment recommendations. This integration of computer vision in healthcare not only improves patient outcomes but also optimizes workflow efficiency in medical facilities (Litjens et al., 2017).

However, the rise of these technologies necessitates a greater emphasis on personal data privacy and security. As computer vision systems become more ubiquitous, safeguarding of the data they collect and analyze will be paramount. This necessitates the formulation of robust security measures and policies to protect individual privacy (Zarsky, 2016). This necessitates the creation of robust security measures and guidelines to protect individual privacy. Ethical considerations, such as transparency in data usage and algorithmic fairness, must also be addressed to ensure that these technologies are implemented responsibly.

In addition to privacy concerns, significant technical hurdles must be addressed to fully harness the capabilities of computer vision. These include improving the robustness of algorithms to varying environmental conditions, enhancing the interpretability of models, and reducing the computational resources required for real-time processing. Addressing these challenges will pave the broader and more effective implementation of computer vision technologies.

In conclusion, computer vision systems hold immense potential across various industries and are set to become more widespread with the progression of technology. Advances in object recognition and classification, in particular, are expected to find increased application in everyday life, contributing significantly to various application areas (Krizhevsky, Alex, et al., 2017). As we continue to innovate and address the associated challenges, the future of computer vision promises a more intelligent and interconnected world, where machines not only see but also understand and interact with their environment in ways that enhance human capabilities.

Basic Principles of Object Recognition and Classification

Object recognition involves detecting and assigning labels to particular objects within an image or video sequence. This process enables computer vision systems to understand and interpret visual data by identifying distinct elements and categorizing them accordingly. Classification is the assignment of these defined objects to certain categories. These processes typically include feature extraction, feature selection, model training and prediction, and classification steps. Each step is critical to the success of the object recognition and classification process.

Image Acquisition and Filtering

Image acquisition is the initial step in computer vision, capturing visual data through cameras or sensors. The quality of captured images significantly impacts subsequent analysis. Factors like resolution, lighting, and noise levels influence the effectiveness of image processing algorithms.

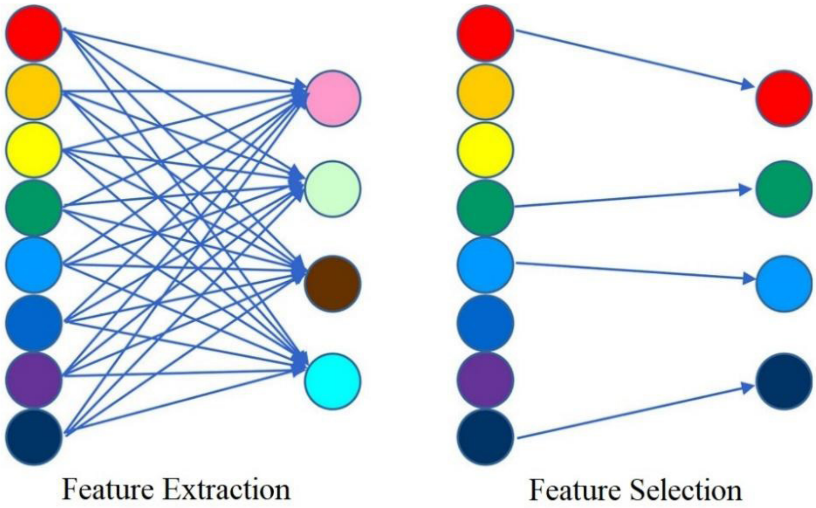
Image preprocessing enhances image quality and prepares it for analysis. Techniques such as noise reduction, edge detection, and contrast enhancement are commonly applied. Noise reduction filters minimize image degradation, while edge detection highlights object boundaries. Contrast enhancement improves image clarity, making features more discernible (Tabik et al., 2017).

Filtering is crucial for preparing images for further stages such as feature extraction and classification. By enhancing image quality, filtering can significantly improve the performance and accuracy of computer vision models.

Feature Extraction and Feature Selection

Feature extraction and selection are complementary techniques for dimensionality reduction. Feature extraction converts the initial data into a different feature space, often through linear or nonlinear combinations of the initial features. In contrast, feature selection chooses a specific portion of the initial features (Ding et al., 2020). While feature extraction creates entirely new features, feature selection retains a subset of the original ones as represented in Figure 1.

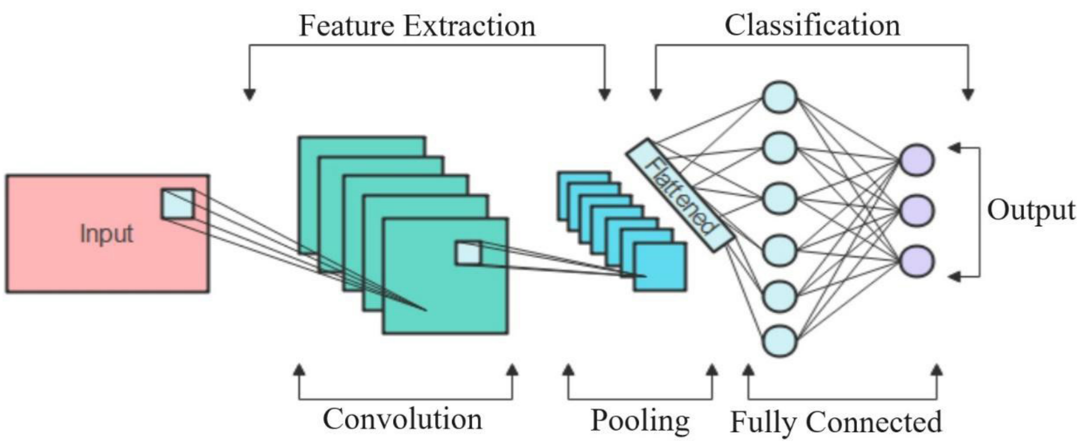
Figure 1
Difference of feature extraction and feature selection algorithms (Ding et al., 2020)



Feature extraction refers to the process of extracting meaningful features from visual data. Traditional methods employ techniques such as edge detection, corner detection, and shape analysis. For instance, edge detection algorithms like the Canny edge detector are utilized to identify the edges of objects in images, while the Harris corner detector is used to find corners. Shape analysis methods, such as the Hough transform, can detect specific shapes like lines and circles (Szeliski, 2022).

Deep learning approaches, approaches convolutional neural networks (CNNs), have transformed the process of feature extraction through automation. Figure 2 depicts the framework of a CNN, which typically includes a sequence of convolutional and pooling stages, culminating in fully connected layers. Convolutional layers identify hierarchical features in the input image using various filters, while pooling layers decrease the spatial dimensions of the resulting feature maps. The fully connected layer at the end integrates these features to perform classification or regression tasks (Aggarwal, 2023; Hor Yan et al., 2024). CNNs can extract both low-level (such as edge and texture) and high-level features (such as object components and full objects) from images, providing a hierarchical feature representation. LeCun et al. emphasize that deep learning methods can achieve high accuracy rates, especially on large data sets (LeCun et al., 2015).

Figure 2
Architecture of a Convolutional Neural Network (Hor Yan et al., 2024)



Feature selection involves identifying the most relevant features for classification among the extracted features. This step is crucial for enhancing model performance and reducing computational costs. The selection process can be carried out using statistical methods or deep learning algorithms. Effective feature extraction and selection significantly influence the accuracy of the final model.

Model Training and Deep Learning

Model training is a pivotal phase in the cultivation of computer vision systems, involving the application of extracted and selected features to a machine learning model. Object recognition and classification tasks have extensively utilized traditional machine learning techniques, including support vector machines (SVM), k-nearest neighbors (k-NN), and decision trees. These methods, while effective, often require manual feature engineering and are limited in their ability to handle the high-dimensional and complex data typical of computer vision applications.

Over the last few years, deep learning approaches, and in particular Deep Convolutional Neural Networks (DCNNs), have made great progress in computer vision applications. DCNNs can automatically extract complex feature hierarchies using raw image data, and this feature allows them to achieve superior results to traditional approaches in many cases.

DCNNs consist of a series of layers: convolutional layers, pooling layers, and fully connected layers. Together, these layers allow the network to produce complex features from images. Convolutional layers apply various filters to the input images, detecting features such as edges, textures, and patterns. Pooling layers increase the efficiency of the model by reducing the spatial dimension of the data and decreasing the sensitivity of the model to small changes in the input. Fully connected layers allow the integration of the extracted features for the final classification process.

A major benefit of deep learning models is their strong capacity to generalize well from large datasets. As the volume of training data increases, DCNNs can learn more robust and discriminative feature representations, leading to improved performance. For example, deep learning models trained on the ImageNet dataset, comprising over one million labeled images across 1,000 categories, can classify a wide range of objects with high accuracy (Russakovsky et al., 2015).

Another powerful method in deep learning is transfer learning, which comprises fine-tuning pre-trained techniques on extensive datasets for specialized tasks using smaller datasets. This technique shortens the training time. It also boosts the performance of the models by leveraging the knowledge acquired from the pre-trained models. For instance, pre-trained models such as VGGNet, ResNet, and Inception have been successfully adapted to various applications, demonstrating the versatility of deep learning.

Beyond DCNNs, other deep learning architectures like RNNs and generative adversarial networks (GANs) are also gaining traction in computer vision. RNNs, notably Long Short-Term Memory (LSTM) networks, excel in handling sequential data, making them particularly effective for applications like video analysis and action recognition. GANs, on the other hand, have shown remarkable capabilities in generating realistic images, data augmentation, and unsupervised learning.

The training of deep learning models involves several critical steps:

- Data Preprocessing: Preparing the dataset by normalizing pixel values, resizing images, and augmenting the data to improve generalization.
- Model Initialization: Setting up the initial weights and biases, often using techniques like Xavier or He initialization to ensure proper convergence.
- Forward Propagation: Passing the input data through the network to compute the output predictions.
- Loss Computation: Calculating the loss using functions like cross-entropy or mean squared error, which measure the disparity between the predicted and actual labels.
- Backpropagation: This procedure entails determining how much each parameter

in the model contributed to the overall loss. The gradients of the loss are computed relating to each model parameter, and these gradients are then applied to renew the weights of the network. Optimization algorithms such as stochastic gradient descent (SGD) or Adam are typically employed to adjust the weights, aiming to minimize the loss and improve the model's accuracy.

- Model Evaluation: Assessing the efficiency of the trained model on a validation set using metrics like accuracy, precision, recall, and F1 score.

Deep learning models, particularly DCNNs, have exhibited superior performance in object recognition and classification, achieving high accuracy rates when trained on large datasets. LeCun et al. emphasize that deep learning methods can achieve remarkable results, especially on extensive datasets, because of their capacity to understand intricate and abstract feature representations (LeCun et al., 2015).

Gauging a model's efficacy requires a rigorous evaluation process using diverse performance metrics. Accuracy, precision, recall, specificity, and the F1 score offer quantitative insights into a model's capabilities. Accuracy measures overall correctness, while precision quantifies the proportion of correct positive predictions among all positive predictions. Recall assesses the model's ability to identify true positives, and specificity measures its capacity to correctly identify true negatives. The F1 score is the harmonic mean of precision and sensitivity.

In summary, the training and evaluation of deep learning models are fundamental to the success of computer vision systems. By leveraging large datasets, advanced architectures, and sophisticated optimization techniques, these models continue to push the horizons of object recognition and classification, which paves the way for innovative applications across various domains.

Feature Extraction Techniques

Feature extraction is one of the fundamental building blocks of computer vision systems. They can be examined in two primary approaches: conventional methods and those based on deep learning.

Conventional methods usually extract low-level features of images. These methods include techniques such as edge detection, corner detection, and shape analysis. For example, the Canny algorithm which detects edges is a popular method used to define edges in images. Harris corner detection algorithm is used to identify corners in images. The Hough transform is a method used to detect certain shapes (for example, straight lines or circles) in images.

Deep learning-based methods typically extract high-level and abstract features from images. These methods can achieve high accuracy rates, especially when trained on large data sets. One of the frequently preferred methods in this field is convolutional neural networks (CNNs). CNNs contain multilayer structures that automatically extract and classify features in images. These networks perform the final classification by extracting certain features at each layer and combining these features. Researchers demonstrated that deep convolutional neural networks (DCNNs) can achieve higher accuracy rates by extracting more complex and abstract features and combining these features (Kaiming et al., 2016).

Pre-trained models can be used by retraining deep learning models on large datasets such as ImageNet on smaller and specific datasets using the transfer learning method. This method reduces training time and increases accuracy.

Feature Selection and Dimension Reduction

Feature selection is the procedure of determining the most useful features for classification among the extracted features. This process is done to increase the performance of the model and reduce the computational costs of the selection process. It can be achieved using statistical methods or deep learning algorithms. Statistical methods analyze the statistical properties of extracted features to determine the most useful ones. For example, correlation analysis can be used to analyze the impact of certain features on classification accuracy. Dimensionality reduction techniques mitigate the challenges posed by high-dimensional datasets by identifying and preserving the most critical features, such as principal component analysis (PCA) and t-distributed stochastic neighbor embedding (t-SNE) (Jolliffe, 2002; Van Der Maaten and Hinton, 2008).

Model Training and Evaluation

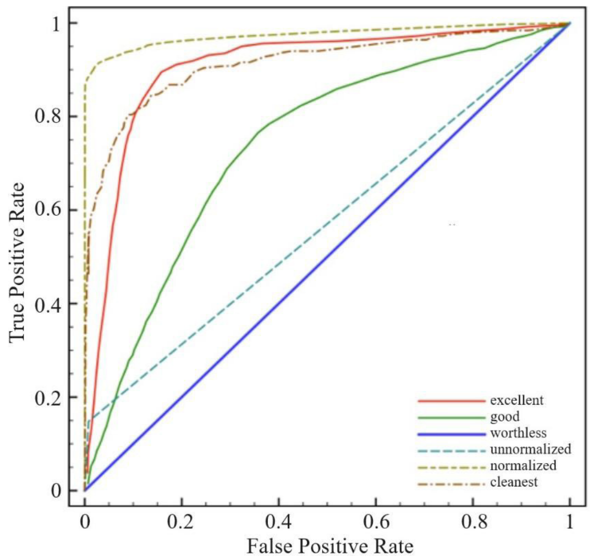
Model training applies the extracted and selected features to a machine learning model. Traditional methods include SVM, decision trees, and k-NN algorithms. SVM classifies data by separating it into certain boundaries, decision trees classify data by rules, and k-NN classifies data by comparing it to known data points.

Deep learning methods extract high-level and abstract features from images, achieving high accuracy rates when trained on large datasets. CNNs, with their multilayer structures, automatically extract and classify features in images. Researchers emphasize that deep learning methods can achieve high accuracy rates, especially on large datasets (LeCun et al., 2015).

Model evaluation assesses the effectiveness of the proposed model on new data, using performance metrics to measure accuracy, precision, sensitivity, and specificity. Accuracy is the proportion of truly classified examples to total examples, precision is the rate of truly classified positive examples to total positive examples, sensitivity is the ratio of true positive classifications to total true positives, and specificity is the proportion of true negative classifications to total true negatives. The F1 score is a balanced measure of a model's accuracy, considering both precision and recall.

In addition to these metrics, the Receiver Operating Characteristic (ROC) curve is a valuable tool to assess the effectiveness of classification approaches. This curve visually depicts a classifier's performance across different classification thresholds as seen in figure 3. The Area Under the ROC Curve (AUC) yields a single criterion of overall model efficiency. Scores closer to 1 in this curve signify superior performance. AUC is particularly useful for comparing various techniques and selecting the best one for a particular problem (Fawcett, 2006). For instance, an AUC of 0.9 suggests that the model has a good capability to correctly classify positive and negative instances.

Figure 3
Receiver Operating Characteristic (ROC) curve



In summary, the training and evaluation of deep learning models are fundamental to the success of computer vision systems. By leveraging large datasets, advanced architectures, and sophisticated optimization techniques, these models continue to push the boundaries of what is possible in object recognition and classification, allowing for innovative applications across various domains. The integration of ROC curves and AUC metrics further enhances the ability to evaluate and compare model performance, ensuring the selection of the most effective models for specific tasks.

Scope of Application

The success of computer vision techniques in object recognition and classification has found a diverse array of applications across various industries. Medical imaging stands as a milestone application within the realm of computer vision. Computer vision systems are used to detect abnormalities and classify diseases in medical images such as X-rays, MRIs, and CT scans. These systems contribute to reducing medical errors by assisting doctors in diagnosis processes. For example, a deep learning-based model can be used to identify pneumonia in chest X-rays and achieve high accuracy rates (Lakhani and Sundaram, 2017).

The automotive industry is heavily dependent on computer vision systems for autonomous vehicles and driver assistance systems. Autonomous vehicles rely on computer vision systems to recognize and classify objects in their environment. These systems ensure safe driving by detecting pedestrians, vehicles, traffic signs and road signs. For example, Tesla's autonomous driving systems use deep learning-based computer vision techniques for environmental sensing and object recognition (Bojarski et al., 2017).

In the field of security, facial recognition techniques are widely used. It is used in security applications such as facial recognition systems, security cameras, and access control systems. These technologies increase security by identifying specific individuals. For example, facial recognition systems are used to verify the identities of passengers at security checkpoints at airports (Parkhi et al., 2015).

In the entertainment industry, computer vision techniques are also used in video games and augmented reality (AR) applications. These systems provide more interactive and realistic experiences by recognizing and classifying in-game objects. For example, AR games such as Pokémon GO analyze real-world images and place virtual objects on these images (Xian et al., 2017).

Future Directions and Conclusion

Object recognition and classification technologies are rapidly developing in the realm of computer vision. In the future, these technologies are expected to develop further thanks to the use of larger and more diverse data sets, more powerful computing resources, and advanced algorithms. In particular, advances in artificial intelligence and deep learning are It will increase the accuracy and efficiency of operations. Future trends include more real-time applications, greater use of autonomous systems, and greater emphasis on personal data privacy and security. In particular, the use of computer vision techniques in autonomous systems such as autonomous vehicles and drones will increase. These systems will use further advanced computer vision techniques for tasks such as environmental sensing, object recognition, and classification.

In conclusion, computer vision systems have great potential in various industries and will become more widespread with the development of technology. Advances in object recognition and classification will find greater use in daily life and make significant contributions to various application areas.

References

Aggarwal, C.C., 2023. Neural Networks and Deep Learning: A Textbook. Springer International Publishing, Cham. https://doi.org/10.1007/978-3-031-29642-0

Bojarski, M., Yeres, P., Choromanska, A., Choromanski, K., Firner, B., Jackel, L., Muller, U., 2017. Explaining How a Deep Neural Network Trained with End-to-End Learning Steers a Car. https://doi.org/10.48550/ARXIV.1704.07911

Ding, Y., Zhou, K., Bi, W., 2020. Feature selection based on hybridization of genetic algorithm and competitive swarm optimizer. Soft Comput. 24, 11663–11672. https://doi.org/10.1007/s00500-019-04628-6

Fawcett, T., 2006. An introduction to ROC analysis. Pattern Recognit. Lett., ROC Analysis in Pattern Recognition 27, 861–874. https://doi.org/10.1016/j.patrec.2005.10.010

Hor Yan, T., Mohd Azam, S.N., Md. Sani, Z., Azizan, A., 2024. Accuracy study of image classification for reverse vending machine waste segregation using convolutional neural network. Int. J. Electr. Comput. Eng. IJECE 14, 366. https://doi.org/10.11591/ijece.v14i1.pp366-374

Huang, G., Liu, Z., Van Der Maaten, L., Weinberger, K.Q., 2017. Densely Connected Convolutional Networks, in: 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). Presented at the 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), IEEE, Honolulu, HI, pp. 2261–2269. https://doi.org/10.1109/CVPR.2017.243

Jolliffe, I.T., 2002. Principal component analysis, 2nd ed. ed, Springer series in statistics. Springer, New York.

Kaiming, H., Zhang, X., Ren, S., Sun, J., 2016. Deep Residual Learning for Image Recognition. Presented at the Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 770–778.

Krizhevsky, Alex, Sutskever, Ilya, Hinton, G.E., 2017. ImageNet classification with deep convolutional neural networks. Commun. ACM 60, 84–90. https://doi.org/10.1145/3065386

Lakhani, P., Sundaram, B., 2017. Deep Learning at Chest Radiography: Automated Classification of Pulmonary Tuberculosis by Using Convolutional Neural Networks. Radiology 284, 574–582. https://doi.org/10.1148/radiol.2017162326

LeCun, Y., Bengio, Y., Hinton, G., 2015. Deep learning. Nature 521, 436–444. https://doi.org/10.1038/nature14539

Litjens, G., Kooi, T., Bejnordi, B.E., Setio, A.A.A., Ciompi, F., Ghafoorian, M., van der Laak, J.A.W.M., van Ginneken, B., Sánchez, C.I., 2017. A survey on deep learning in medical image analysis. Med. Image Anal. 42, 60–88. https://doi.org/10.1016/j.media.2017.07.005

Parkhi, O., Vedaldi, A., Zisserman, A., 2015. Deep face recognition. BMVC 2015 - Proc. Br. Mach. Vis. Conf. 2015.

Russakovsky, O., Deng, J., Su, H., Krause, J., Satheesh, S., Ma, S., Huang, Z., Karpathy, A., Khosla, A., Bernstein, M., Berg, A.C., Fei-Fei, L., 2015. ImageNet Large Scale Visual Recognition Challenge. Int. J. Comput. Vis. 115, 211–252. https://doi.org/10.1007/s11263-015-0816-y

Szeliski, R., 2022. Computer Vision: Algorithms and Applications. Springer Nature.

Tabik, S., Peralta, D., Herrera-Poyatos, A., Herrera, F., 2017. A snapshot of image pre-processing for convolutional neural networks: case study of MNIST: Int. J. Comput. Intell. Syst. 10, 555. https://doi.org/10.2991/ijcis.2017.10.1.38

Van Der Maaten, L., Hinton, G., 2008. Visualizing Data using t-SNE. J. Mach. Learn. Res. 9, 2579–2605.

Xian, Y., Xu, Hanzhang, Xu, Haolin, Liang, L., Hernandez, A.F., Wang, T.Y., Peterson, E.D., 2017. An Initial Evaluation of the Impact of Pokémon GO on Physical Activity. J. Am. Heart Assoc. 6, e005341. https://doi.org/10.1161/JAHA.116.005341

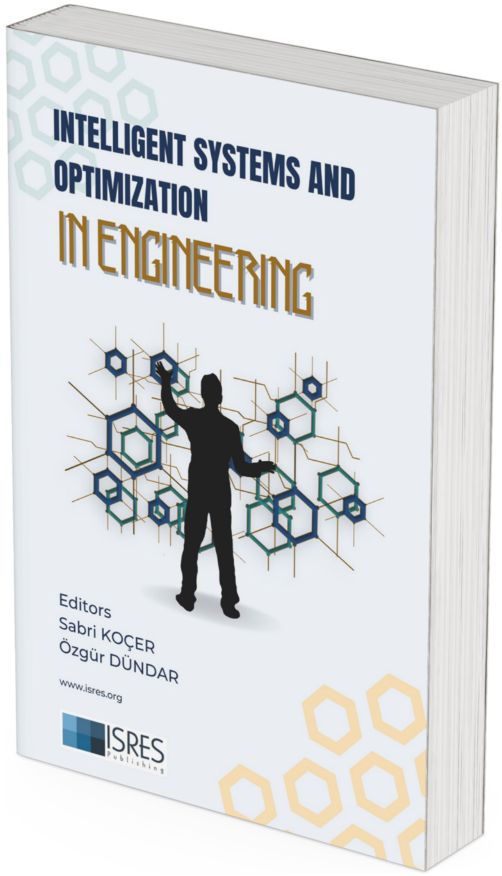
Zarsky, T., 2016. The Trouble with Algorithmic Decisions: An Analytic Road Map to Examine Efficiency and Fairness in Automated and Opaque Decision Making. Sci. Technol. Hum. Values 41, 118–132. https://doi.org/10.1177/0162243915605575

About the Authors

Yunus Emre GÖKTEPE, PhD, is an Assistant Professor at Necmettin Erbakan University. The author's areas of expertise are bioinformatics, artificial intelligence, and machine learning. He is still serving as a faculty member of the Department of Computer Engineering, Seydisehir Ahmet Cengiz Engineering Faculty at Necmettin Erbakan University in Konya, Turkey.
E-mail: ygoktepe@erbakan.edu.tr, **ORCID:** 0000-0002-8252-2616

Yusuf UZUN, PhD, is an Assistant Professor of Computer Engineering at Necmettin Erbakan University in Konya, Turkey. He holds a PhD in Mechanical Engineering from Necmettin Erbakan University. His main areas of interest are artificial intelligence, autonomous systems, and augmented reality applications. He also works as the Rector's Advisor at Selcuk University.
E-mail: mailto:yuzun@erbakan.edu.tr, **ORCID:** 0000-0002-7061-8784

Similarity Index
The similarity index obtained from the plagiarism software for this book chapter is 11%.



Intelligent Systems and Optimization in Engineering 2024 was developed from selected articles invited by the editors.

This edition comprises 13 sections covering various topics, including machine learning, artificial intelligence, optimization, image processing, data compression, health, and industrial applications widely utilized in modern technology. Each submission underwent a rigorous review process by at least two international reviewers. The book aims to offer readers a comprehensive scholarly publication in the fields of technology, artificial intelligence, and optimization.

Published by ISRES Publications, this volume provides valuable insights into the advancements and applications of intelligent systems in engineering.

ISRES
Publishing