AI-Empowered Interference Mitigation Technique for M2M Networks

Submitted By

Dhruvi Pancholi

22MCES08

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

SCHOOL OF TECHNOLOGY, INSTITUTE OF TECHNOLOGY

NIRMA UNIVERSITY

AHMEDABAD-382481

May 2024

AI-Empowered Interference Mitigation Technique for M2M Networks

Major Project - I

Submitted in partial fulfillment of the requirements

for the degree of

Master of Technology in Computer Science and Engineering (CYBER SECURITY)

Submitted By

Dhruvi Pancholi

(22MCES08)

Guided By

Dr. Sudeep Tanwar

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

SCHOOL OF TECHNOLOGY, INSTITUTE OF TECHNOLOGY

NIRMA UNIVERSITY

AHMEDABAD-382481

May 2024

Certificate

This is to certify that the major project entitled **"AI-Empowered Interference Mitigation Technique for M2M Networks"** submitted by **Dhruvi Pancholi (Roll No: 22MCES08)**, towards the partial fulfillment of the requirements for the award of degree of Master of Technology in Computer Science and Engineering (CYBER SECURITY) of Nirma University, Ahmedabad, is the record of work carried out by him under my supervision and guidance. In my opinion, the submitted work has reached a level required for being accepted for examination. The results embodied in this major project part-I, to the best of my knowledge, haven't been submitted to any other university or institution for award of any degree or diploma.

21/05/24

Dr. Sudeep Tanwar
Guide & Professor,
CSE Department,
Institute of Technology,
Nirma University, Ahmedabad.

Dr. Vijay Ukani
Associate Professor,
Coordinator M.Tech - CSE (Cyber Security)
Institute of Technology,
Nirma University, Ahmedabad

Dr. Madhuri Bhavsar
Professor and Head,
CSE Department,
Institute of Technology,
Nirma University, Ahmedabad.

Dr Himanshu Soni
Director,
School of Technology,
Nirma University, Ahmedabad

Statement of Originality

I, **Dhruvi Pancholi**, Roll. No. **22MCES08**, give undertaking that the Major Project entitled "**AI-Empowered Interference Mitigation Technique for M2M Networks**" submitted by me, towards the partial fulfillment of the requirements for the degree of Master of Technology in **Computer Science & Engineering (CYBER SECURITY)** of Institute of Technology, Nirma University, Ahmedabad, contains no material that has been awarded for any degree or diploma in any university or school in any territory to the best of my knowledge. It is the original work carried out by me and I give assurance that no attempt of plagiarism has been made.It contains no material that is previously published or written, except where reference has been made. I understand that in the event of any similarity found subsequently with any published work or any dissertation work elsewhere; it will result in severe disciplinary action.

Pancholi

Signature of Student

Date: 27-05-2024

Place: Ahmedabad.

21/5/24

Endorsed by

Dr. Sudeep Tanwar

(Signature of Guide)

Acknowledgements

The project's successful completion was made possible by the encouragement and support of many kind hearts.

I would like to extend my sincere gratitude to **Dr. Madhuri Bhavsar**, Head of the Department of Computer Science and Engineering at Nirma University, for providing resources throughout the course of the project.

I would especially want to thank **Dr. Himanshu Soni**, Honourable Director of the Institute of Technology at Nirma University in Ahmedabad, for his unwavering motivation during the course of the project.

I would like to take this opportunity to thank **Dr. Vijay Ukani**, MTech Cyber Security PG Coordinator at Nirma University's Department of Computer Science and Engineering, for granting permission and making the necessary facilities available so that the research could be carried out in a systematic manner.

I am extremely grateful to **Dr. Sudeep Tanwar**, professor in the department of Computer Science and Engineering at Nirma University, for his insightful suggestions. I deeply appreciate **Nilesh Jadav** and **Dr. Rajesh Gupta** for their invaluable guidance, passionate interest, and support during the project. I would also like to express our gratitude to all of the teaching and non-teaching staff at Nirma University's Department of Computer Science and Engineering for their earnest guidance and collaboration on the project.

I also express gratitude to the authors of the literatures cited in this project. Last but not least, I relish the opportunity to offer my heartfelt gratitude to my family and friends.

Dhruvi Pancholi

22MCES08

Abstract

Machine-to-machine, or M2M, communication has reshaped how industries function as a whole. Its advancement has fueled the growth of the Industrial Internet of Things (IIoT), transforming factories into smart, networked environments. M2M enables devices and machines to connect with one another in real-time, resulting in swifter data exchange, real-time surveillance, and optimized operations. M2M communication, on the other hand, relies primarily on wireless networks, which are prone to randomness and uncertainty. These networks are prone to noise and interference, which reduces the efficiency of communication and degrades performance. This may have a major effect on the dependability and resilience of smart industrial systems. Interference mitigation techniques are employed to improve the performance of M2M communication. In this paper, we propose a hybrid Artificial Intelligence (AI)-empowered interference technique. A virtual setup of the cellular network is simulated in MATLAB. The channel gain data generated from the simulation is fed as a dataset to the clustering algorithm DBSCAN (Density-Based Spatial Clustering of Applications with Noise). Furthermore, the best data cluster is fed as the initial population to the genetic algorithm (GA) to enhance the performance. Through each iteration of the GA process, the solution with the best fitness is chosen as the initial population for the next iteration. Ultimately, we have the population with the best fitness function as the solution. The GA employed in the proposed framework performs well, gradually decreasing execution time over the generations and converging at a throughput of 21.2 bps.

Abbreviations

M2M	Machine-to-Machine.
IIoT	Industrial Internet-of-Things.
AI	Artificial Intelligence.
DBSCAN	Density-based spatial clustering of applications with noise.
GA	Genetic Algoritm.
D2D	Device-to-Device.
NOMA	Non-Orthogonal Multiple Access.
SIC	Successive Interference Cancellation.
TPC	Transmit Power Control.
IWLAN	Industrial Wireless Networks.
RL	Reinforcement Learning.
QoS	Quality of Service.
COGA	Co-evolutionary Genetic Algorithm.
MIMO	Multiple-Input Multiple-Output.
D-mMIMO	Distributed massive MIMO.
CCI	Co-channel interference.
RS	Reference Signal.
OFDM	Orthogonal frequency-division multiplexing.
MSE	Mean Square Error.
BER	Bit Error Rate.
EBBDSA	Evolutionary Biogeography-based Dynamic Subcarrier Allocation.
CSI	Cross-tier Subcarrier Interference.
SINR	Signal-to-interference-plus-noise ratio.
DL	Deep Learning.
DRL	Deep Reinforcement Learning.
DNN	Deep Neural Network.
GAN	Generative Adversarial Network.
BIRCH	Balanced Iterative Reducing and Clustering using Hierarchies.
DBI	Davies Bouldin Index.
CHS	Calinski and Harabasz Score.

Contents

List of Figures

Chapter 1

Introduction

The Industrial Internet of Things (IIoT) has brought in a new era of efficiency and connection for smart industries, and it has significantly impacted machine-to-machine (M2M) communication. This groundbreaking journey began with the integration of IIoT technologies, specifically smart sensors, into industrial processes. These clever sensors are necessary to provide machine-to-machine communication, which speeds up and improves the efficiency of processes. The capacity of sensors to communicate data efficiently over communication channels is the core component of machine-to-machine (M2M) communication. This has revolutionised several sectors by enabling real-time data communication between machines, enhancing automation, and ultimately raising overall performance. In smart industries, the development of IIoT has proven particularly effective in creating intelligently integrated systems. Nevertheless, there are challenges with M2M communication, especially with regard to wireless communication methods. Interference and noise are two examples of the unpredictability and randomness that wireless networks bring. This volatility could have a significant impact on M2M communication systems' efficiency and performance. Noise and interference can impair the dependability of machine-to-machine communication by causing packet loss, transmission delays, or data corruption. Despite the fact that IIoT has undoubtedly transformed industrial communication, wireless media's intrinsic randomness adds a degree of unpredictability. This volatility may be harmful since it can produce disparities in M2M communication effectiveness and, on rare occasions, a reduction in overall performance.

Numerous interference reduction strategies have been presented by researchers to enhance M2M communication performance. These mostly consist of game theory and graph theory-based techniques. The purpose of a non-cooperative game is to simulate the communication between primary and secondary users. To satisfy the global fairness goals, heuristic power allocation algorithms (i.e., proportional and max-min) are applied. This method is scalable and effectively decreases interference, but it may also add to the total complexity. Problems with implementation can arise, particularly in scenarios involving real-time, dynamic communication.[1] In cellular networks, an inventive pricing-based game theory strategy aims to minimise interference and increase D2D communication efficiency. By using this tactic, (Device-to-Device) D2D users are encouraged to lower their transmit power in order to minimise interference with cellular users. Yet using this strategy in a real-world cellular network could pose challenges. It could be challenging to implement and ensure user compliance in real-world scenarios.[2]

Researchers created AI-based methods to address the drawbacks of game- and graph-based interference mitigation strategies as technology progressed. These methods require creating an objective function that accounts for factors like network coverage, interference, and power efficiency in order to arrive at an optimal or nearly optimal solution. These AI-based techniques perform better in terms of scalability and robustness to unfavourable conditions than game- and graph-based techniques. They do, however, have certain disadvantages. Notwithstanding their advantages, these techniques can be computationally taxing, especially when addressing large-scale optimisation problems.[3] The length of time required to identify a solution could be one potential limitation in real-time or resource-constrained applications.[4][5] This complexity may complicate implementation and require careful consideration of practical constraints in practical scenarios. Moreover, these approaches fail to take into account the uncertainty and diversity of real-world circumstances.

Motivated by the previously mentioned problems, we provide a hybrid AI method to reduce interference in M2M communications. MATLAB[6] simulation is used to get the channel gain matrix dataset. The dataset is subjected to an additional clustering algorithm, DBSCAN. Through clustering, devices with similar channel gain patterns or that are close to one another are clustered together. This makes network administration and organisation more effective. To maximise the mitigation of interference, a genetic algorithm is used. The genetic algorithm starts with the best cluster as its initial population. The goal of every iteration is to create a more superior set of solutions that, ideally, balance performance criteria. By applying genetic processes, including crossover and mutation, to the solutions, each cycle produces a fresh set of potential configurations. Crossover is the combining of elements from two parent solutions and the introduction of tiny, random modifications through mutation. The fitness of each solution is evaluated using the objective function. Solutions that meet the optimisation requirements successfully are given higher fitness scores. By combining genetic algorithms and clustering, the interference mitigation technique dynamically optimises M2M network organisation. It allows for flexibility in response to changing network conditions, ensuring efficient use of resources and improved overall network performance.

1.1 Research Contributions

The research contributions of the proposed work are as follows.

- We propose an AI-powered approach for mitigating interference in M2M networks.
- To do this, eight communicating entities are taken into consideration and a NOMA (Non-Orthogonal Multiple Access) with SIC(Successive Interference Cancellation)-based M2M network is simulated inside the MATLAB environment using a variety of toolboxes. The simulated scenario's channel gain matrix is extracted out, and a dataset is produced.
- The clustering algorithm consumes the generated data, and clusters are generated.
- These clusters then serve as the initial population for GA. The solution is assessed using a fitness function that is based on signal-to-interference-plus-noise ratio (SINR).
- The optimal cluster is determined iteratively by GA, and it eventually outputs it.
- In contrast to numerical optimization methods, GA is robust, unbiased, and random. It offers good scalability as well.

Chapter 2

Literature Survey

Das *et al.*[7] introduced a novel location-aware power regulation technique that uses reinforcement learning (RL) to autonomously choose the optimal frequency and power allocation for each M2M device in order to reduce co-tier interference in M2M communications over cellular networks. The scientists evaluated the suggested methodology using a simulation of a cellular network with M2M devices. The proposed location-aware power regulation system outperforms the traditional methods with enhanced scalability, spectrum efficiency, improved QoS parameters for all M2M devices, and greater network capacity. The suggested approach accounts for both co-tier and cross-tier interference. One drawback of the suggested method is that it necessitates accurate position information from every M2M device.

Yan *et al.*[8] introduced a novel successive interference cancellation (SIC) technique based on a co-evolutionary genetic algorithm (COGA) to optimise the power distribution of wireless nodes and eliminate interference. The SIC-COGA algorithm was tested by the authors using a simulation of a wireless network with several users. Throughput and energy efficiency are two areas where the recommended algorithm outperforms the alternatives. The suggested SIC-COGA algorithm effectively reduces interference and is resilient to changes in the wireless environment. One possible drawback of the suggested method is the usage of co-evolutionary genetic algorithms, which are more complex and maybe harder to adjust than standard genetic algorithms.

In order to mitigate interference in non-orthogonal multiple access (NOMA) distributed massive MIMO (Multiple-Input Multiple-Output) (D-mMIMO) wireless networks, Seimeni *et al.*[9] proposed a novel resource allocation and mitigation mechanism.

The proposed framework consists of two main components: an interference mitigation technique that uses a combination of successive interference cancellation (SIC) and power control to further reduce interference between users, and a resource scheduling algorithm that jointly optimises the subcarrier allocation and power allocation for each user in the network while taking into account their interference limitations. The proposed architecture was evaluated with a simulation of a D-mMIMO wireless network with NOMA transmission. It is power-efficient and resistant to CCI (co-channel interference) effects. It increases the throughput and spectrum efficacy of the network. Nevertheless, heterogeneous networks cannot be used with the suggested framework.

Dey *et al.*[10] introduced a novel reference signal (RS) design method that uses phase rotation and time-frequency spreading to provide unique RS sequences for different numerologies, therefore reducing co-channel interference (CCI) in 5G OFDM (Orthogonal frequency-division multiplexing) systems. The suggested method is contrasted by the authors with many state-of-the-art RS design techniques. Flexible and adaptable, the proposed RS design technique greatly enhances MSE and BER performance. The suggested method does, however, come at the expense of complexity and synchronisation.

Hasan *et al.*[11] introduced the Evolutionary Biogeography-based Dynamic Subcarrier Allocation (EBBDSA) algorithm for resource allocation in 5G heterogeneous networks with the goal of reducing cross-tier interference. Reducing interference between tiers is achieved by allocating resources to users according to the proposed EBBDSA algorithm. The method takes into account each user's SINR, the cross-tier interference they produce, and their QoS requirements when allocating resources. In addition to improving SINR performance, the proposed EBBDSA method also reduced outage probability to 88.1%, raised spectral efficiency to 67.5%, and improved total spectral efficiency to 83.6%. The EBBDSA algorithm is not scalable to other instances and can be computationally expensive, particularly in large-scale networks.

A variety of Deep Learning (DL)-based interference management techniques are examined by the authors[12], including deep reinforcement learning (DRL) for interference alignment, deep neural networks (DNNs) for interference suppression, and generative adversarial networks (GANs) for interference cancellation. The research focuses on recent advancements and the effectiveness of these techniques in mitigating interference in wireless networks. It comes to the conclusion that dynamic, flexible, and DL-based interference mitigation techniques improve signal detection. They do, however, have several disadvantages, including data dependency, high computational complexity, and constrained interpretability.

Cen *et al.*[13] presents a self-supervised DL framework for multi-interference reduction in Synthetic Aperture Radar (SAR) images. Apart from improved generalizability and scalability, the framework has minimal requirements for data annotation. In addition to its benefits, the framework's drawbacks include its high computational complexity, the potential for biases, and the challenge of establishing direct control over the suppression process.

Table 2.1: Literature Review

Year	Author	Objective	Methodology	Pros	Cons
2020	Das et al.[7]	reduce interference in cellular networks by optimizing power levels based on device locations to improve energy efficiency and network performance	location-aware power control mechanism to mitigate interference in M2M communications over cellular networks	successfully mitigated cross-tier and co-tier interference; higher throughput; improved the system and network performance; scalable	high energy consumption; requires location information for all M2M devices
2020	Yan et al.[8]	enhance the management of interference in wireless communication systems, potentially leading to improved network performance	SIC algorithm based on Co-Evolutionary Power Control for improving interference management in wireless communication systems	better performance in terms of optimality of solution; robust to uncertainties	increased communication overhead and delay

Table 2.1 continued from previous page

Year	Author	Objective	Methodology	Pros	Cons
2022	Seimeni et al.[9]	optimize resource allocation and reduce interference in distributed massive-MIMO wireless systems with NOMA transmission	resource scheduling and interference mitigation in distributed massive MIMO wireless systems using NOMA transmission for 5G networks	power efficiency is enhanced; resistance against intense CCI effects; power savings even in intense CCI scenarios; improved spectral efficiency; low complexity	not applicable to heterogeneous networks
2022	Dey et al.[10]	address the interference challenges in 5G OFDM systems to reduce co-channel interference in flexible and full-duplex communication scenarios	reference signal design techniques to mitigate co-channel interference in 5G OFDM systems	significant improvement in MSE and BER performance; flexible and adaptable	high complexity; compatibility issues; requires syncronization
2023	Hasan et al.[11]	develop a resource allocation technique to address interference issues in 5G heterogeneous networks, focusing on cross-tier interference	novel resource allocation technique based on EBBDSA to reduce CSI between macrocells and femtocells in 5G HetNets	higher SINR performance; lowered the outage probability to 88.1%; increased spectral efficiency to 67.5%; total spectral efficiency improved to 83.6%	the EBBDSA algorithm is not scalable to different scenarios

Table 2.1 continued from previous page

Year	Author	Objective	Methodology	Pros	Cons
2023	Arunprasath et al.[12]	examine DL-based methods for managing interference in wireless networks	examines a variety of DL methods for interference suppression, alignment, and cancellation, including MIMO and graph neural networks	employs cutting-edge DL techniques	lacks comprehensive experimental results
2023	Cen et al. [13]	develop a technique for self-supervised learning that suppresses various interferences in SAR data	employs DL and time-frequency analysis to suppress interference	novel technique; less reliance on labeled data; effective in complex situations	limited cross-domain applicability; time-frequency analysis demands substantial computer resources
2024	Proposed Work	address interference in cellular networks using AI techniques	a hybrid AI technique integrating clustering and genetic algorithm to mitigate interference in cellular networks	better performance due to amalgamation of multiple AI techniques; robust; scalable; unbiased	-

Chapter 3

System Model

The system model, as shown in Figure 3.1, consists of a base station (b) and various IIoT-enabled machines $\{m_1, m_2, \ldots, m_i\} \in M$. The exchange of information between the machines is enabled through M2M network.

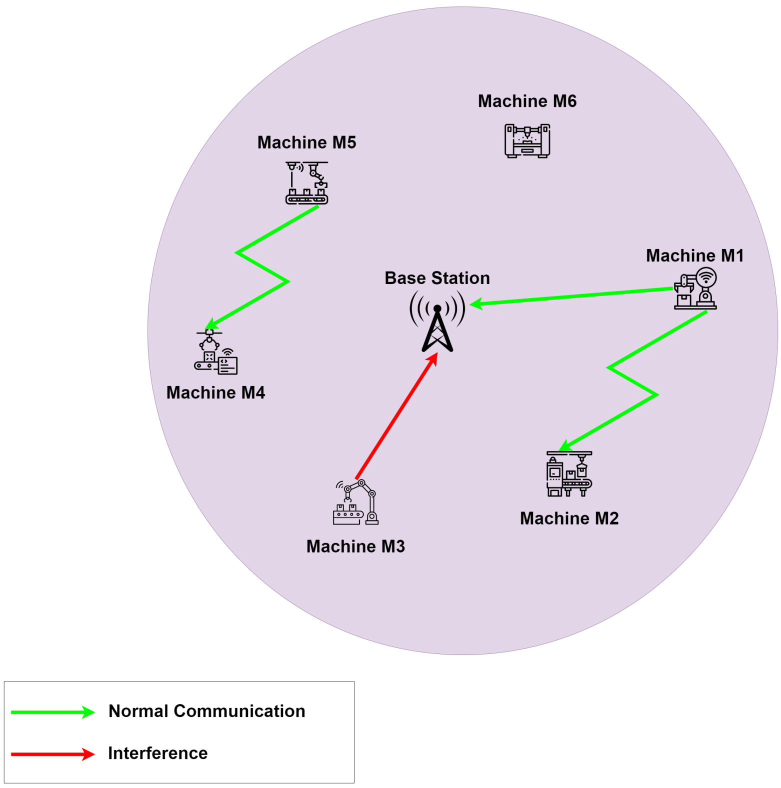

Figure 3.1: System Model

Data communication between machines is based on channel gain. Channel gain between machine m_i and base station b is given by Eq. 3.1. Similarly, channel gain between machine m_i and machine m_j is given by Eq. 3.2.

$$\theta_{m_i,b} = \sqrt{1-\varrho}\omega_{\bar{m_i,b}} + \sqrt{\varrho}\omega_{\hat{m_i,b}} \tag{3.1}$$

$$\theta_{m_i,m_j} = \sqrt{1-\varrho}\omega_{\bar{m_i,m_j}} + \sqrt{\varrho}\omega_{\hat{m_i,m_j}} \tag{3.2}$$

where θ represents the channel fading between transmitter and receiver. $\bar{\omega}$ represents the estimated value of the channel and $\hat{\omega}$ represents the estimation error. ϱ is the estimated channel variance of θ, where $\varrho \in [0,1]$.

Eq. 3.3 gives the transmitted signal χ_{m_i,m_j} from machine m_i to machine m_j.

$$\text{Machine } m_i \xrightarrow[\text{using channel gain } \theta_{m_i,m_j}]{\text{transmit signal } \chi_{m_i,m_j}} \text{Machine } m_j \tag{3.3}$$

where θ_{m_i,m_j} is the channel gain between machine m_i and machine m_j. Increased channel gain will result in improved transmission.

Eq. 3.4 represents the received signal ψ_{m_j,m_i} at machine m_j from machine m_i.

$$\psi_{m_j,m_i} = \theta_{m_i,m_j}\chi_{m_i,m_j} + \upsilon_{m_i,m_j} + \lambda_{m_i,m_j} \tag{3.4}$$

where θ_{m_i,m_j} is the channel gain between machine m_i and machine m_j, χ_{m_i,m_j} is the signal transmitted from machine m_i to machine m_j, υ_{m_i,m_j} is the interference and λ_m_i, b is the noise in the channel between machine m_i and machine m_j.

SINR received at machine m_j ($SINR_{m_j}$) is calculated as given in the Eq. 3.5.

$$SINR_{m_j} = \frac{\psi_{m_j,m_i}}{\upsilon_{m_i,m_j} + \lambda_{m_i,m_j}} \tag{3.5}$$

where ψ_{m_j,m_i} is the signal received at machine m_j from machine m_i, υ_{m_i,m_j} is the interference and $\lambda_{m_i,b}$ is the noise in the channel between machine m_i and machine m_j.

Chapter 4

Proposed Framework

The proposed framework, as shown in Figure 4.1, consists of three layers, namely the data collection layer, clustering layer, and GA layer. A detailed explanation of each layer is as follows.

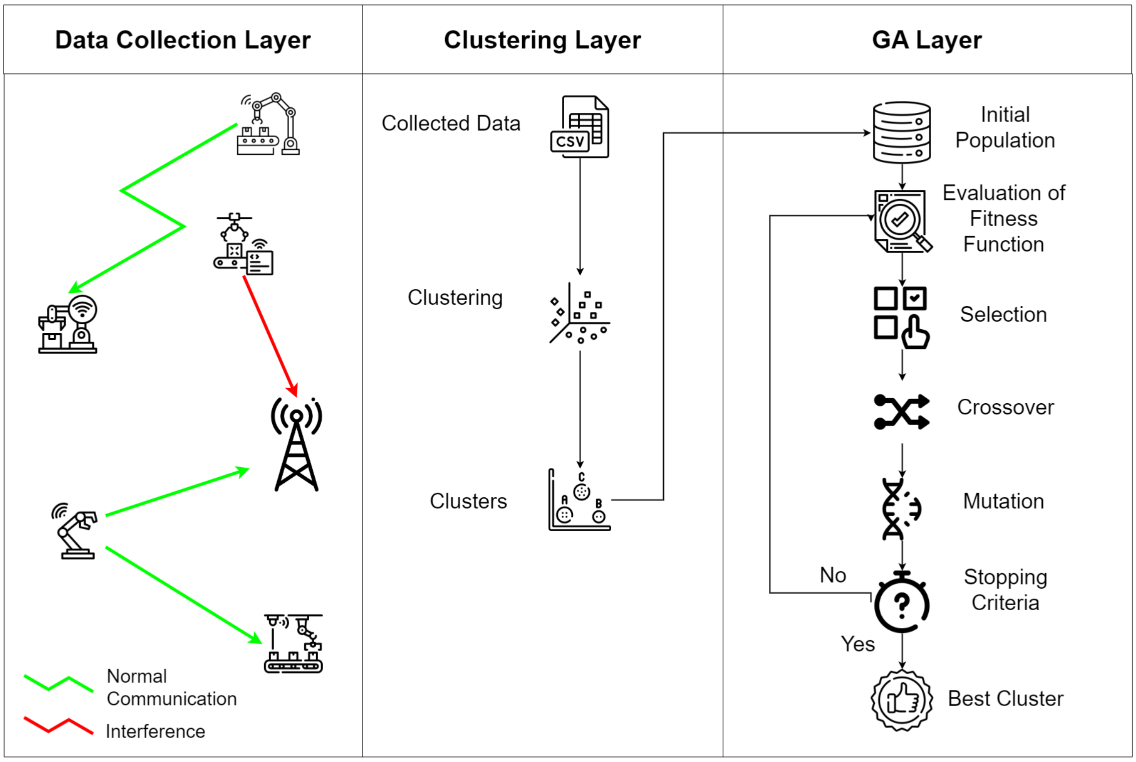

Figure 4.1: Proposed Framework

4.1 Data Collection Layer

The data collection layer consists of a base station (bs) and various machines $\{m_1, m_2, \ldots, m_i\} \in M$ in an IIoT environment. The channel gain matrix of this network is denoted by a matrix Δ, where each element $\delta_{i,j} \in \Delta$ represents the channel gain between machine m_i and machine m_j as denoted by Eq. 4.1.

$$\delta_{m_i,m_j} = \sqrt{1-\varphi}\psi_{m_i,m_j}^{-} + \sqrt{\varphi}\psi_{m_i,m_j}^{\hat{}} \tag{4.1}$$

where $\bar{\psi}$ is the channel estimated value, $\hat{\psi}$ is the estimated error and φ is the estimated channel variance of δ.

The transmitted signal β_{m_i,m_j} from machine m_i to machine m_j is denoted by Eq. 4.2.

$$\text{Machine } m_i \xrightarrow[\text{using channel gain } \delta_{m_i,m_j}]{\text{transmit signal } \beta_{m_i,m_j}} \text{Machine } m_j \tag{4.2}$$

where δ_{m_i,m_j} is the channel gain between machine m_i and machine m_j.

The signal received at machine m_j from machine m_i is denoted by Eq. 4.3.

$$\gamma_{m_j,m_i} = \delta_{m_i,m_j}\beta_{m_i,m_j} + \upsilon_{m_i,m_j} + \lambda_{m_i,m_j} \tag{4.3}$$

where δ_{m_i,m_j} is the channel gain between machine m_i and machine m_j, β_{m_i,m_j} is the signal transmitted from machine m_i to machine m_j, υ_{m_i,m_j} is the interference and $\lambda_{m_i,b}$ is the noise in the channel between machine m_i and machine m_j.

From Eq. 4.3, it is evident that the strength of the received signal is directly proportional to channel gain $\gamma_{m_j,m_i} \propto \delta_{m_i,m_j}$. Also, strength of $\gamma_{m_j,m_i} \Uparrow$, if $\upsilon_{m_i,m_j}(inference) \Downarrow$ and $\lambda_{m_i,m_j}(noise) \Downarrow$. Noise is an inherent element of wireless channels. So, mitigation of interference is the solution to improving communication.

4.2 Clustering Layer

The input to this layer is the channel gain data produced in the data collection layer. Clustering algorithm used in the proposed framework is DBSCAN. The channel gain data is fed into DBSCAN as input after being preprocessed.

DBSCAN accurately identifies clusters of any shape in spatial data. DBSCAN's functions based on density connectivity. Dense data point regions divided by less densely packed regions are referred to as clusters. In DBSCAN, every data point is classified into one of three categories.

- Core Point: If a minimal number of points (min_samples) are found within a specified radius (ε), then a data point is considered to be a core point. Stated otherwise, a core point has an adequate number of neighboring points in its immediate vicinity.
- Border Point: A data point is classified as a border point if it is situated close to a core point but is not a core point itself.
- Noise Point (or Outlier): Data points that don't fall into the core or border categories are known as outliers or noise points.

DBSCAN starts by choosing an unexplored data point at random. DBSCAN counts the number of neighboring points within a ε radius to calculate the ε-neighborhood of this picked point. If a point has more points in its neighborhood than the predefined cutoff (min_samples), it is categorized as core. It is believed that core points have sufficient density surrounding them to form a cluster. DBSCAN expands the cluster by iteratively visiting every point in the ε-neighborhood after identifying a core point. Every point in this neighborhood is examined to determine whether it is a core point. Should this be the case, a recursive investigation of its ε-neighborhood is conducted.

DBSCAN gradually expands the cluster by adding neighboring core points and their corresponding neighbors until no more core points are reachable within the ε-neighborhood. Points that are in the ε-neighborhood of a core point but are not core points are called border points. These points are considered to be part of the cluster even though they do not contribute to its expansion. Points that are neither core nor border points are known as outliers or noise points. These spots typically do not belong to any cluster and are surrounded by data sparsely or in isolation. Eventually, the algorithm creates clusters composed of both core points and boundary points that are accessible from core points. Each cluster is represented by a set of related boundary points and interconnected core points. The process stops once each point in the dataset has been categorized as a noise point or assigned to a cluster.

The processes performed in this layer are explained by Algorithm 1.

Algorithm 1 Working of the proposed clustering layer

Input: channel gain matrix Δ

Output: set of clusters C

procedure FETCH BEST CLUSTER(Δ, DBSCAN)
  $\{\delta_{m_1,m_2}, \delta_{m_1,m_3}, \ldots, \delta_{m_i,m_j}\} \in \Delta$
  Preprocess dataset Δ using standard preprocessing steps
  if data is not normalized **then**
    $\Delta \leftarrow \Gamma$, where $\Gamma = Z_{score}(\Delta)$
  end if
  $\Delta \xrightarrow{\text{input to}}$ DBSCAN
  DBSCAN $\xrightarrow{\text{gives}} C$, C is the set of clusters
end procedure

4.3 GA Layer

Natural selection and genetic principles are the foundations of the GA, an optimization method. The first step of the procedure is to create a population of individuals, or a set of clusters that the clustering layer has found. Every member of the population has the potential to solve the optimization problem. Every member of the population is evaluated to determine their fitness, which is a measure of their capacity to resolve issues. Every problem has a unique fitness function that measures the quality of the solution each individual represents. SINR (signal-to-interference-plus-noise ratio) serves as the fitness function for our problem. Individuals are selected from the existing population to be the parents of the next generation. Fitness value is usually the determining factor in the selection process, with fitter candidates having a higher probability of getting picked. This process is similar to the "survival of the fittest" theory of natural selection.

Crossover is the process by which some individuals (parents) combine to generate new individuals (offspring). The process of producing one or more children by transferring genetic material between the two parents is known as crossover. This method increases population diversity while combining promising solutions. Mutations (random alterations) are occasionally added to the offspring's gene pool in order to maintain genetic diversity and prevent an early convergence to less-than-ideal solutions. Mutation can be used to explore new regions of the solution space. The new children frequently replace some members of the current population, depending on their fitness. This ensures that the population size will remain constant and that only the fittest individuals will live to bear children.

The selection, crossover, mutation, and replacement processes are iterated through by the algorithm for a predefined number of generations or until a termination condition is met (like arriving at a satisfactory solution). The population tends to gravitate toward better solutions when the number of people who are more fit increases over successive generations. Eventually, the method converges to a best cluster solution that is either almost optimal or falls under the limitations of the problem.

The operations carried out in this layer are defined in Algorithm 2.

Algorithm 2 Working of the proposed GA layer

Input: set of clusters C
number of generation $num - gen$
mutation rate α
fitness threshold τ
Output: cluster ϕ having the best fitness

procedure CALC-SINR(individual)
    signal-power = $\sum(\sum(individual/2)^2)$
    sinr = $\frac{signal-power}{interference+noise}$
    datarate = $log_2(1 + sinr)$
    return datarate
end procedure
procedure FITNESS-VALUE(population)
    for each individual in population **do**
        calc-SINR(individual)
    end for
end procedure
procedure SELECT-INDIVIDUALS(population, fitness value)
    if fitness-value(individual) > mean-fitness-value **then**
        parents ← individual
    end if
    return parents
end procedure
procedure CROSSOVER-AND-MUTATION(mating-pool, α)
    Select 2 random individuals from mating-pool as parent1 and parent2
    offspring ← crossover(parent1, parent2)
    offspring ← mutate(offspring, α)
    return offspring
end procedure
procedure GA(C, num-gen, α, τ)
    initial-population ← C
    for each generation in num-gen **do**
        fitness-values ← fitness-value(initial-population)
        mating-pool ← select-individuals(initial-population, fitness-values)
        offspring ← crossover-and-mutation(mating-pool, α)
        initial-population ← offspring
        throughput ← max(fitness-values)
        if fitness-value(initial-population) > τ **then**
            solution ϕ ← initial-population
            break
        end if
    end for
end procedure

Chapter 5

Result Analysis

The result analysis of the proposed framework is addressed in this section. Numerous tools and technologies have been used to implement the suggested structure. Moreover, a range of metrics for performance are utilised to evaluate how effective the proposed approach is.

5.1 Experimental Setup and Tools

MATLAB[6] is used to simulate a M2M network of 8 machines, with a focus on NOMA with SIC. The dataset for the proposed framework is the channel gain matrix extracted from the simulation.

The proposed framework is put into implementation on Google Colaboratory, which provides a development environment for using various AI models. The development of the framework made use of a number of Python libraries, including sklearn ($v1.2.2$) for machine learning algorithms, Numpy ($v1.23.5$) for numerical computations, Pandas ($v1.5.3$) for data manipulation, and Matplotlib ($v3.7.1$) for data visualisation. The suggested framework runs on a PC with the following specifications: 500 GB solid-state drive, Intel iRISXe integrated graphics card, 8GB RAM, and Intel core i5.

5.2 Performance Analysis

Three performance criteria are used to compare the clustering algorithm used in the proposed framework, DBSCAN, against other clustering techniques like k-means, BIRCH (Balanced Iterative Reducing and Clustering using Hierarchies), agglomerative clustering and affinity propagation to evaluate its performance.

Table 5.1: DBSCAN-param

Parameter	**Value**	**Description**
eps (ε)	0.8	The maximum distance that separates two samples such that one is considered to be the neighbour of other.
min_samples	5	The minimum number of samples required in a neighbourhood for a point to be regarded as a core point. The point itself is included in it. DBSCAN will identify denser clusters if min_samples is set to a larger number; on the other hand, it will identify more sparse clusters if it is set to a lower value.

The parameters used in the proposed framework clustering algorithm DBSCAN are as listed in the Table 5.1.

The silhouette score is used to verify data cluster consistency. It calculates a data point's similarity to its own cluster relative to other clusters. A higher silhouette score indicates that the data point matches better within its own cluster and less well with other clusters. For incorrect grouping, the score is -1; for exceptionally dense clustering, it is +1. Overlapping clusters are suggested by scores close to zero. The score is high when clusters are densely packed and well separated, which is consistent with the traditional definition of a cluster. Figure 5.1 contrasts several clustering methods with the DBSCAN-based model on silhouette score that is employed in the suggested framework. With the highest silhouette score, DBSCAN outperforms the other clustering techniques in terms of performance.

The average similarity across clusters is represented by the Davies Bouldin Index (DBI), which is a metric that contrasts the size of the clusters with their distance from one another. Using the Davies Bouldin index, Figure 5.2 contrasts several clustering models with DBSCAN. A model with a lower Davies-Bouldin index performs better because it has a higher cluster separation. DBSCAN outperforms other clustering methods because it has the lowest DBI.

The ratio of intra-cluster to inter-cluster dispersion is measured by the Variance Ratio

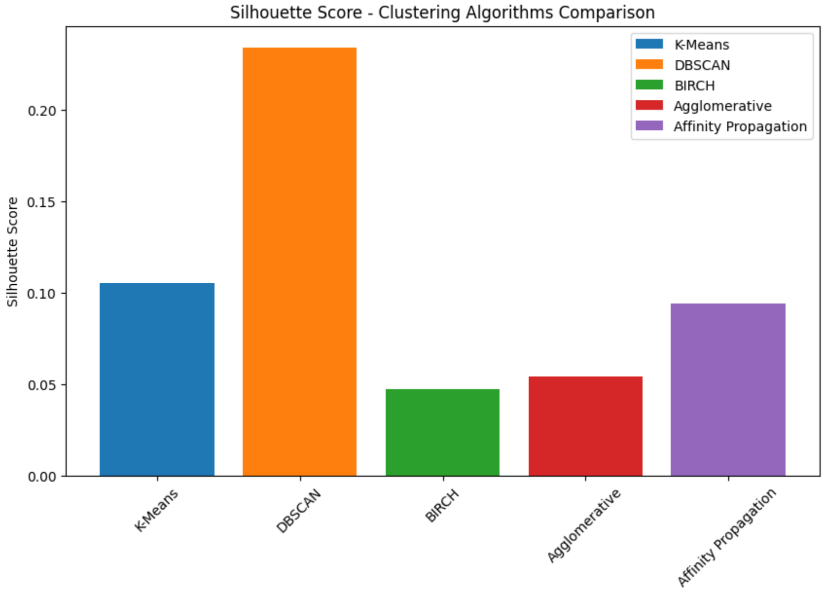

Figure 5.1: Silhouette Score

Table 5.2: GA-param

Parameter	**Value**	**Description**
num_generations	200	Each generation consist of a population of a number of individuals.
mutation rate (α)	0.05	Determines the likelihood that an individual will undergo the mutation process.

Criterion, often known as the Calinski and Harabasz Score (CHS). A higher index indicates better performance. Using CHS as a basis, Figure 5.3 contrasts several clustering models with DBSCAN. Once more, DBSCAN has the highest CHS, suggesting that it performs better than other clustering techniques.

Based on the assessment of three performance metrics—the silhouette score, DBI, and CHS we may thus conclude that DBSCAN, the clustering algorithm included in the suggested framework, performs better than any other clustering algorithm.

The parameters used in the proposed framework GA layer are as listed in the Table 5.2.

Figure 5.4 shows the convergence graph for GA. The best fitness vs. generation graph of a genetic algorithm shows how the fitness of the population's best individual varies as it becomes optimized over several generations. It can reveal details about the genetic algorithm's workings, including -

- Convergence: It can tell us whether the algorithm is moving in the direction of

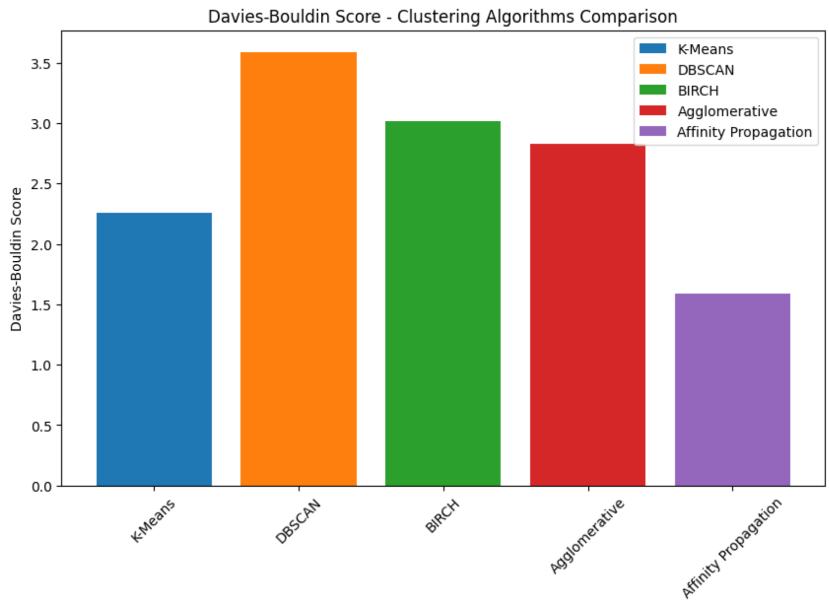

Figure 5.2: DB Index

the best solution. The algorithm is getting close to an optimal solution if the best fitness continuously rises or reaches a high plateau. On the other hand, if the fitness fluctuates or doesn't improve over time, there may be issues with the parameters or implementation of the method, or the result may not have converged yet.

- Speed of Convergence: The slope of the best fitness curve can be used to measure the pace of convergence of the algorithm. A sharp slope indicates rapid improvement, while a smooth slope indicates slower convergence. Unexpected increases or decreases in fitness may indicate changes in the population or search space, such as the emergence of new genetic operators or the discovery of more effective solutions.

- Stagnation or Plateauing: If the optimal fitness remains relatively constant over several generations, the algorithm might have reached a plateau or stagnation point. This may be caused by a number of things, such as insufficient population diversity, poor selection procedures, or reaching algorithmic limits on the maximum number of improved solutions that may be generated.

Figure 5.6 shows the execution time graph for GA. The performance and computing efficiency of a genetic algorithm can be inferred from the graph of the algorithm's execution time vs generation as it progresses through generations. A lot of information about the genetic algorithm may be deduced from it.

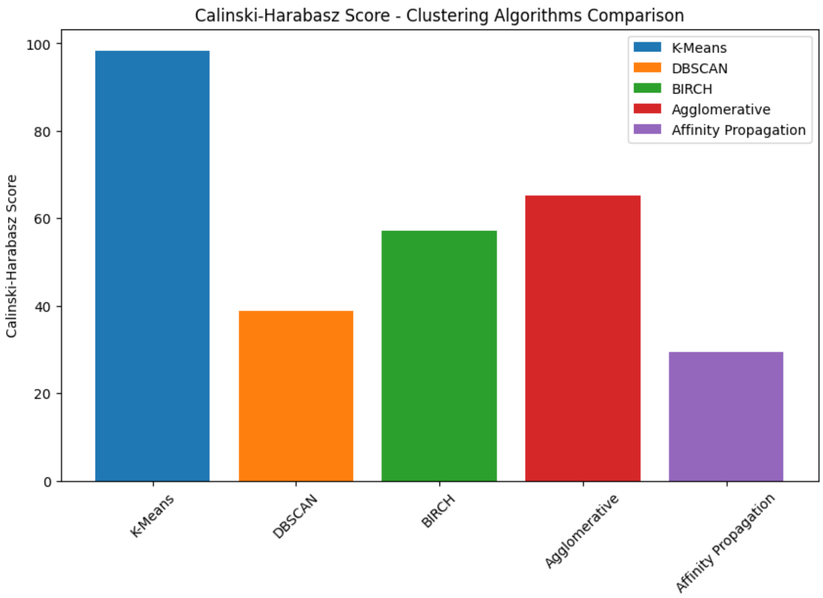

Figure 5.3: CH Score

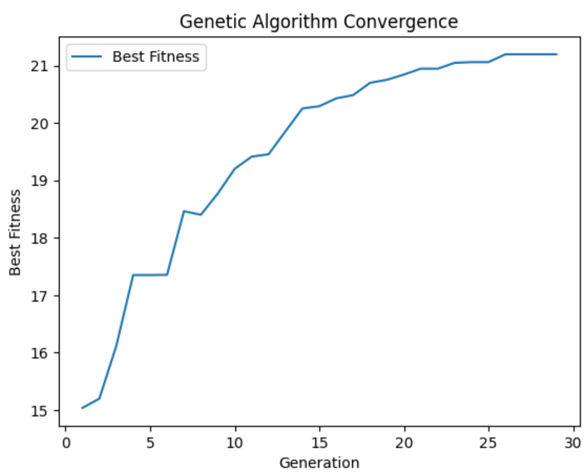

Figure 5.4: GA Convergence

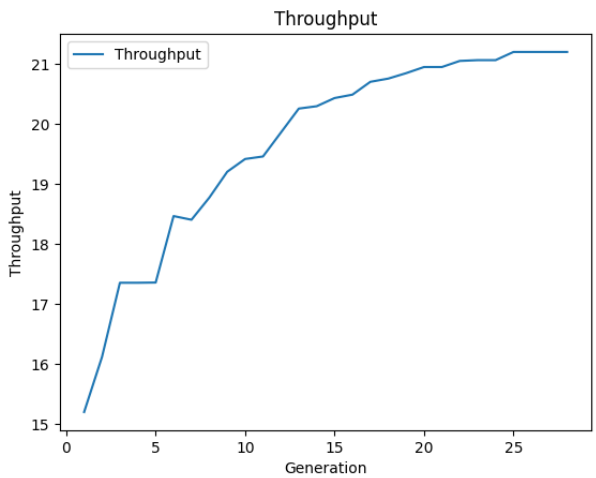

Figure 5.5: GA Throughput

- Computational Efficiency and Algorithm Performance: The form of the execution time curve could provide information about the performance features of the genetic algorithm. For example, a smooth, gradually decreasing slope indicates consistent progress and efficient search space exploration. On the other hand, sharp variations in algorithm performance, such as increased overhead from population management or genetic processes, may indicate computational challenges or variations in algorithm execution time.

- Convergence Rate: The algorithm is efficiently converging to a solution if its execution time decreases or remains steady over generations. Conversely, a significant increase or decrease in the execution time may indicate issues reaching convergence or sustaining advancement across several generations.

- Scalability: It can also provide insight into how effectively the evolutionary algorithm scales in relation to the magnitude of the issue or the power of the computing system. High scalability is indicated if the execution time increases or remains constant despite the problem's increasing complexity or the population's growth. On the other hand, it may indicate scalability problems that require attention if execution time rises swiftly in tandem with an increase in problem complexity or population number.

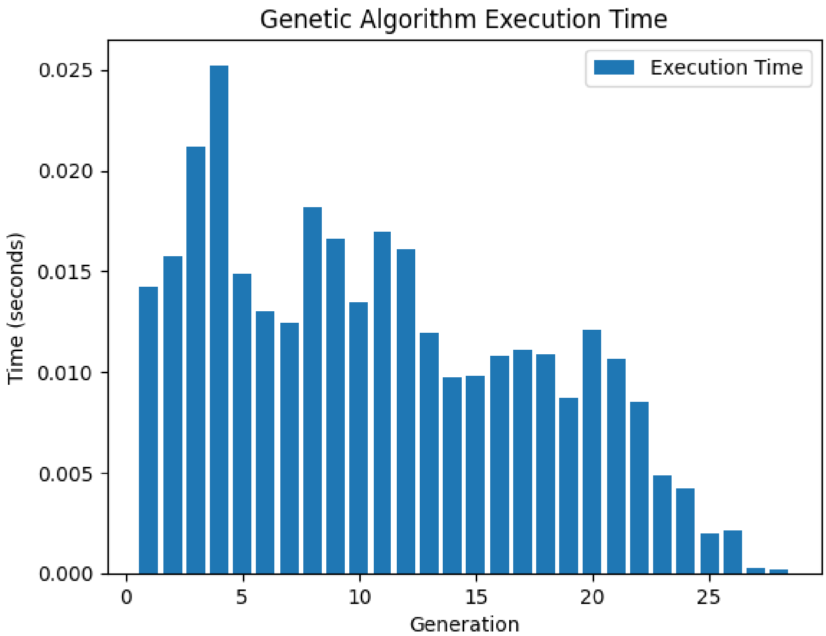

Figure 5.6: Genetic Algorithm Execution Time

Figure 5.5 shows the throughput for GA. Interference and throughput are inversely proportional. A higher throughput is a sign of reduced interference. The suggested framework converges to produce the best cluster of machines in an M2M network with the least interference.

From the graphs in Figure 5.4, Figure 5.6 and Figure 5.5, it is evident that the GA's execution time gradually drops over generations and the algorithm converges at a throughput of 21.2 bps. Therefore, we can conclude that the proposed framework's genetic algorithm operates effectively.

The comparison of throughput attained in each generation for a NOMA with SIC-

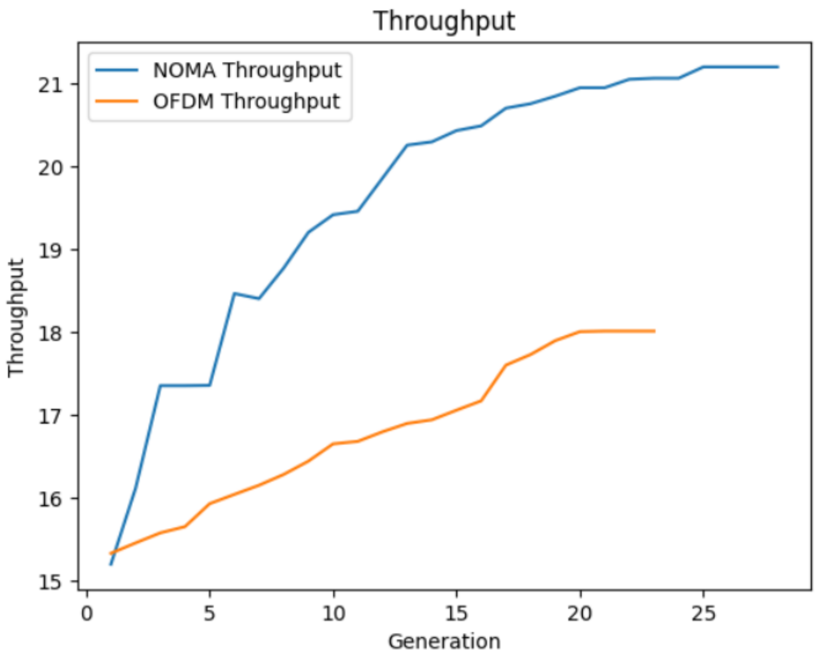

Figure 5.7: Throughput comparison of NOMA vs OFDM

based M2M network and an OFDM-based M2M network using the proposed framework can be observed in Figure 5.7.

Our proposed approach converges with a throughput of 18.01 bps when applied to a channel gain dataset of an OFDM-based communication system. This demonstrates very clearly how successfully interference is mitigated in both OFDM-based and NOMA-based communication systems by our proposed approach.

Chapter 6

Conclusion

This research presents a hybrid AI framework for interference mitigation in M2M networks. To accomplish that, a virtual M2M communication scenario with several machines is constructed inside the MATLAB program. The MATLAB simulation's dataset is extracted out and fed into the clustering method. DBSCAN is used as the clustering algorithm in the proposed framework. A number of alternative clustering techniques, including K-means, BIRCH, agglomerative clustering and affinity propagation, are compared to DBSCAN's performance. DBSCAN outperforms all other clustering algorithms. The clusters are fed into GA to find the optimal solution (cluster with the least interference). GA is an intelligent optimization technique. Along with being randomized and unbiased, it is highly scalable and robust. In the proposed framework, the GA is implemented to good effect, achieving convergence at throughputs of 21.2 bps and 18.01 bps on a NOMA with SIC-based and OFDM-based M2M network, respectively.

6.1 Future Work

To further enhance the optimality of the proposed framework, we shall replace GA with bio-inspired algorithms.

Bibliography

[1] B. Gu, C. Zhang, H. Wang, Y. Yao, and X. Tan, "Power control for cognitive m2m communications underlaying cellular with fairness concerns," *IEEE Access*, vol. 7, pp. 80789–80799, 2019.

[2] R. Yin, G. Yu, H. Zhang, Z. Zhang, and G. Y. Li, "Pricing-based interference coordination for d2d communications in cellular networks," *IEEE Transactions on Wireless Communications*, vol. 14, no. 3, pp. 1519–1532, 2015.

[3] F. de Oliveira Torres, D. L. Cardoso, and L. F. C. e Silva, "Interference mitigation in next generation networks using clustering and intelligence techniques," *Procedia Computer Science*, vol. 94, pp. 280–287, 2016. The 11th International Conference on Future Networks and Communications (FNC 2016) / The 13th International Conference on Mobile Systems and Pervasive Computing (MobiSPC 2016) / Affiliated Workshops.

[4] X. Gong, D. Plets, E. Tanghe, T. De Pessemier, L. Martens, and W. Joseph, "An efficient genetic algorithm for large-scale transmit power control of dense and robust wireless networks in harsh industrial environments," *Applied Soft Computing*, vol. 65, pp. 243–259, 2018.

[5] Z. Hou, Y. Huang, J. Chen, G. Li, X. Guan, Y. Xu, R. Chen, and Y. Xu, "Joint irs selection and passive beamforming in multiple irs-uav-enhanced anti-jamming d2d communication networks," *IEEE Internet of Things Journal*, vol. 10, no. 22, pp. 19558–19569, 2023.

[6] T. M. Inc., "Matlab version: 9.13.0 (r2022b)," 2023.

[7] S. K. Das and M. F. Hossain, "A location-aware power control mechanism for interference mitigation in m2m communications over cellular networks," *Computers Electrical Engineering*, vol. 88, p. 106867, 2020.

[8] B. Yan and X. Guo, "Successive interference cancellation algorithm based on coevolutionary power control," in *2020 5th International Conference on Information Science, Computer Technology and Transportation (ISCTT)*, pp. 242–246, 2020.

[9] M. A. Seimeni, P. Alevizaki, P. K. Gkonis, D. I. Kaklamani, and I. S. Venieris, "On resource scheduling and interference mitigation in distributed massive-mimo wireless orientations via noma transmission," *Physical Communication*, vol. 53, p. 101725, 2022.

[10] P. Dey, A. Masal, S. Kaimalettu, J. K. Milleth, and B. Ramamurthi, "Reference signal design to mitigate co-channel interference in 5g ofdm systems with multiple numerologies," *Physical Communication*, vol. 53, p. 101653, 2022.

[11] M. K. Hasan, S. Islam, T. R. Gadekallu, A. F. Ismail, S. Amanlou, and S. N. H. S. Abdullah, "Novel ebbdsa based resource allocation technique for interference mitigation in 5g heterogeneous network," *Computer Communications*, vol. 209, pp. 320–330, 2023.

[12] S. Arunprasath, A. Suresh, and J. Khakurel, *Interference Techniques Based on Deep Learning in Wireless Networks*, ch. 8, pp. 161–182. John Wiley Sons, Ltd, 2023.

[13] X. Cen, Y. Li, Z. Han, T. Gu, P. Zhang, and T. Cai, "Self-supervised learning method for sar multiinterference suppression," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 61, pp. 1–17, 2023.