Multi-User Detection In DS-CDMA System Using Genetic Algorithm Optimization

Made by- Shivom & Rakesh from ECAM (B Tech), NSUT Delhi

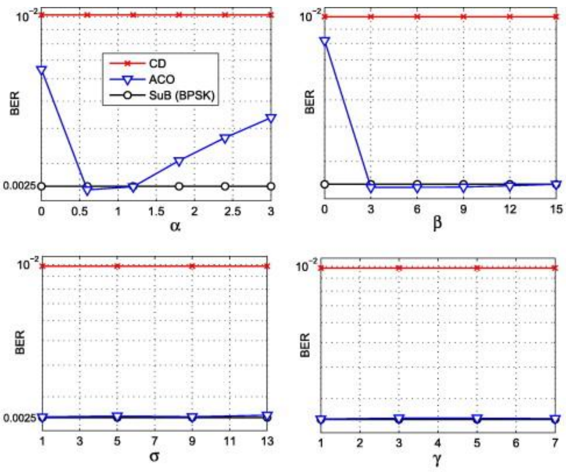
OBJECTIVE:

The aim of the study is to enhance the performance of multi-user detection (MUD) in Code Division Multiple Access (CDMA) systems by utilizing a novel optimization algorithm known as Genetic Algorithm (GA). This aims to minimize error rates in transmitting signals, thereby improving the efficiency and reliability of CDMA communication systems.

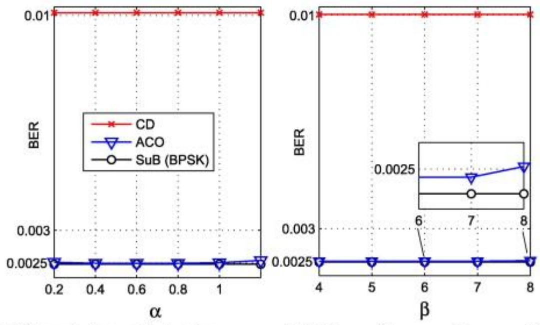
PREVIOUS WORK DONE :

Ant colony input parameters optimization for multiuser detection in DS/CDMA systems *by José Carlos Marinello Filho, Reginaldo Nunes de Souza & Taufik Abrão*

In this work a simple and efficient methodology for tuning the input parameters applied to the ant colony optimization multiuser detection (ACO-MuD) in direct sequence code division multiple access (DS-CDMA) is proposed. The motivation in using a heuristic approach is due to the nature of the NP complexity posed by the wireless multiuser detection optimization problem. The challenge is to obtain suitable data detection performance in solving the associated hard complexity problem in a polynomial time. Previous results indicated that the application of heuristic search algorithms in several wireless optimization problems have been achieved excellent performance-complexity tradeoffs. Regarding different system operation and channels scenarios, a complete input parameters optimization procedure for the ACO-MuD is provided herein, which represents the major contribution of this work. The performance of the ACO-MuD is analyzed via Monte-Carlo simulations. Simulation results show that, after convergence, the performance reached by the ACO-MuD is much better than the conventional detector, and somewhat close to the single user bound (SuB). Flat Rayleigh channels is initially considered, but the input parameter optimization methodology is straightforward applied to selective fading channels scenarios, as well as to joint time-spatial wireless channels diversities.



(a) First step input parameter optimization: $\alpha_{\text{fx}} = 1$; $\beta_{\text{fx}} = 1$; $\sigma_{\text{fx}} = 8$; $\gamma_{\text{fx}} = 1$



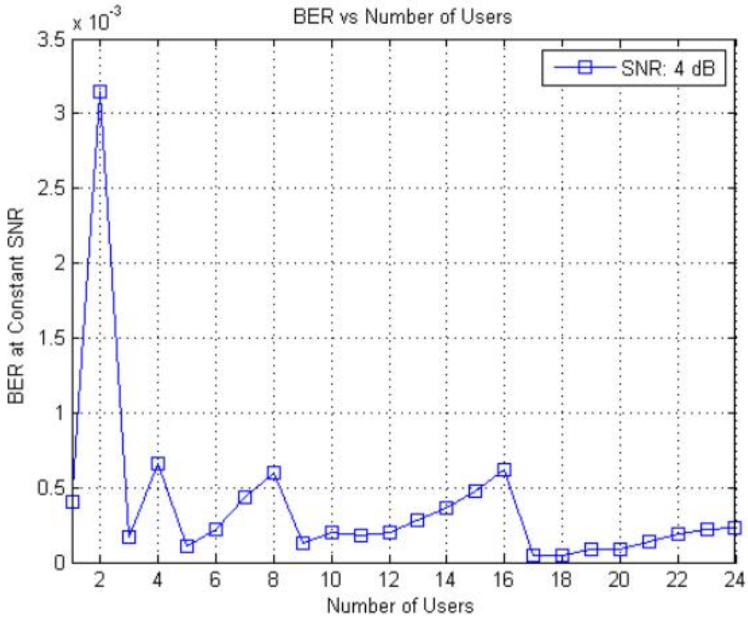
(b) Second step optimization: $\alpha_{\text{fx}} = 0,6$; $\beta_{\text{fx}} = 6$; $\sigma_{\text{fx}} = 5$; $\gamma_{\text{fx}} = 3$

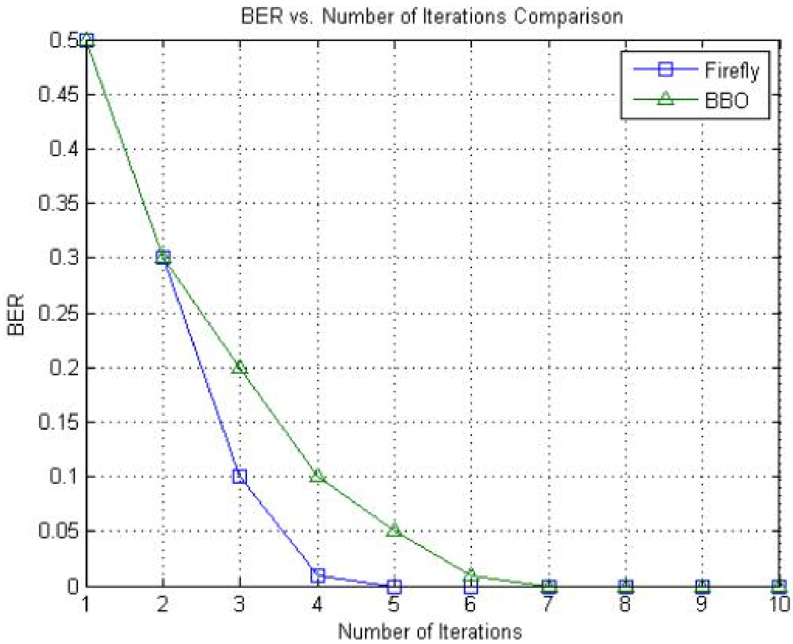
Multi-user detection of DS-CDMA based on PSO-ICA *by Liu Xiaozhi, Feng Dawei & Zhang Tianshuang*

Particle swarm optimization (PSO) is one of the optimization techniques. Independent component analysis (ICA) is a statistical signal processing technology that can get higher-order statistical independent and non-Gaussian components. In this paper, an algorithm is proposed, which is used to reduce the multiple access interference (MAI) in the direct sequence code division multiple access (DS-CDMA) systems. In order to solve the issues that the objective function falls into the local optimum, this paper utilizes the particle swarm optimization algorithm and the fast fixed-point algorithm (PSO-ICA) to improve the classical independent component analysis algorithm. The experimental results show that the PSO-ICA algorithm has the advantages of smaller bit error rate than the traditional algorithm. The PSO-ICA algorithm is more suitable for applications in the DS-CDMA system.

Multi-user Detection in CDMA Systems using Firefly based Optimization *by Amandeep Kaur and Dr. PS Mundra*

In this paper, the comparison of several optimization methods for solving the optimal multiuser detection problem exactly or approximately are discussed. The purpose of using these algorithms is to provide complexity constraint alternatives to solving this nondeterministic polynomial-time (NP)-hard problem. An approximate solution is found firefly based optimization which is used to provide an exact solution. Simulations show that these approaches can have bit-error rate (BER) performance which is indistinguishable from the maximum likelihood performance. Multi-user detection technique is one of the key technique used in CDMA system. This technique can reduce the multiple access interference and enhance the system performance and capacity. This algorithm helps to finding the shortest path and finding the best transmission power to minimize the BER within the signal.In this paper Comparative study of BBO (Biogeography based optimization) and Firefly Algorithm optimization techniques are summarized and discussed.





<u>Multi-User Detection In DS-CDMA System Using Biogeography Based Optimization</u> *by Santosh N Nemadea , Dr. Mahesh T Kolteb , Santosh Nemadea*

This paper introduces the biogeography-based optimization (BBO) algorithm for multiuser detection in CDMA systems to address performance issues caused by fading channels. BBO aims to minimize user transmission error rates by optimizing immigration and emigration rates. This approach effectively reduces user detection complexity and interference. Implemented in MATLAB, the proposed algorithm is suitable for real-time user identification, with performance compared against harmony search algorithms and scenarios without multiuser detection. Results demonstrate BBO's effectiveness in improving error rates and its potential for enhancing CDMA system performance.

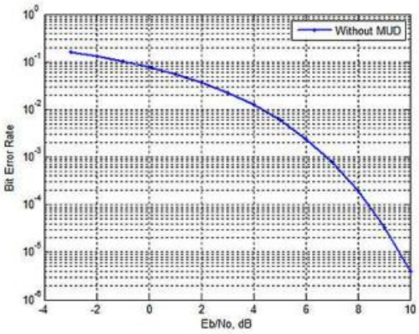


Fig. 5. Performance of error rate probability curve

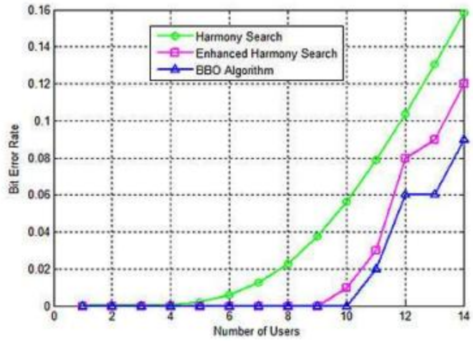
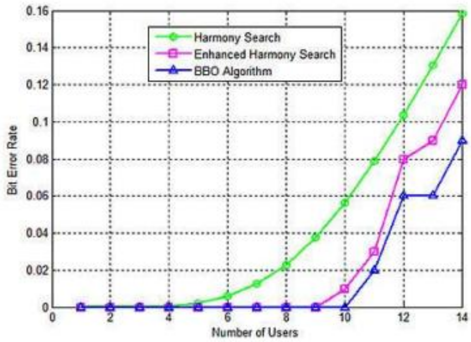
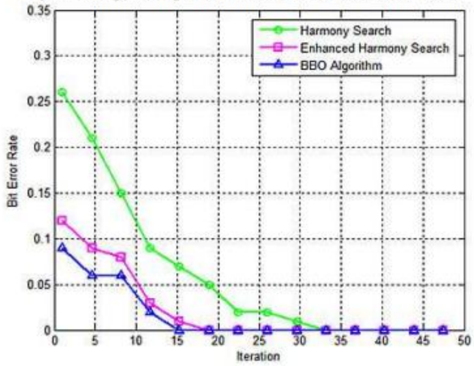
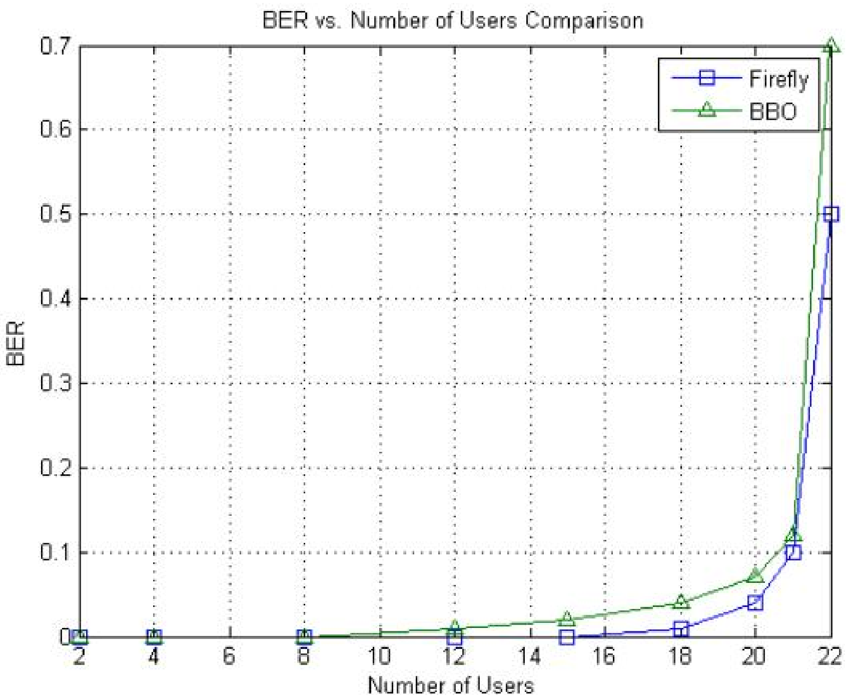


Fig. 6. Comparison performance of BER vs. Eb/No in dB.





Comparison between Firefly and BBO



BER vs. Number of Users Comparison

PROPOSED METHODOLOGY:

The proposed methodology involves several key steps:

-**Modeling of DS-CDMA System**: Describing the structure of the CDMA system, including transmitter and receiver components, signal transmission, and reception mechanisms.

- **Overview of GA Algorithm**: Explaining the principles of the GAalgorithm, which is a swarm intelligence optimization technique based on biogeography.

- **Multi-User Detection Using GA**: Detailing the adaptation of the GA algorithm for MUD in DS-CDMA systems. This involves defining the objective function to minimize error rates and describing the steps of the GA algorithm for MUD.

- **Implementation and Evaluation**:

<u>Initialization:</u>

Random user codes are generated for each user. These user codes represent the spreading sequences used in CDMA systems.
Random channel noise is generated. This noise simulates the interference experienced during signal transmission over fading channels.

<u>GA Class:</u>

The GA class is defined to encapsulate the GA operations.
It initializes a population of individuals, where each individual represents a possible solution (in this case, a set of spreading sequences).
It evaluates the fitness of each individual in the population. Fitness is calculated based on the Bit Error Rate (BER) of the received signal using the individual's spreading sequences.
It evolves the population over multiple generations:
Selection: Parents are selected from the population based on their fitness scores.
Crossover: Offspring are generated by combining the genetic material of the selected parents.
Mutation: Random mutations are applied to introduce diversity into the population.
It records the best solution (individual) in each generation.

<u>Simulation:</u>

The code runs simulations for different numbers of users (ranging from 1 to max_users).
For each number of users:
It initializes the GA instance with a population size of 10 and a maximum number of generations of 10.
It evolves the GA over multiple generations and records the final BER after all iterations.

<u>Plotting:</u>

The results are plotted in two subplots:
Number of Users vs. Final Bit Error Rate (BER): Shows how the BER varies with the number of users in the system.
Number of Iterations vs. Bit Error Rate (BER): Shows the convergence behavior of BER over iterations for different numbers of users.

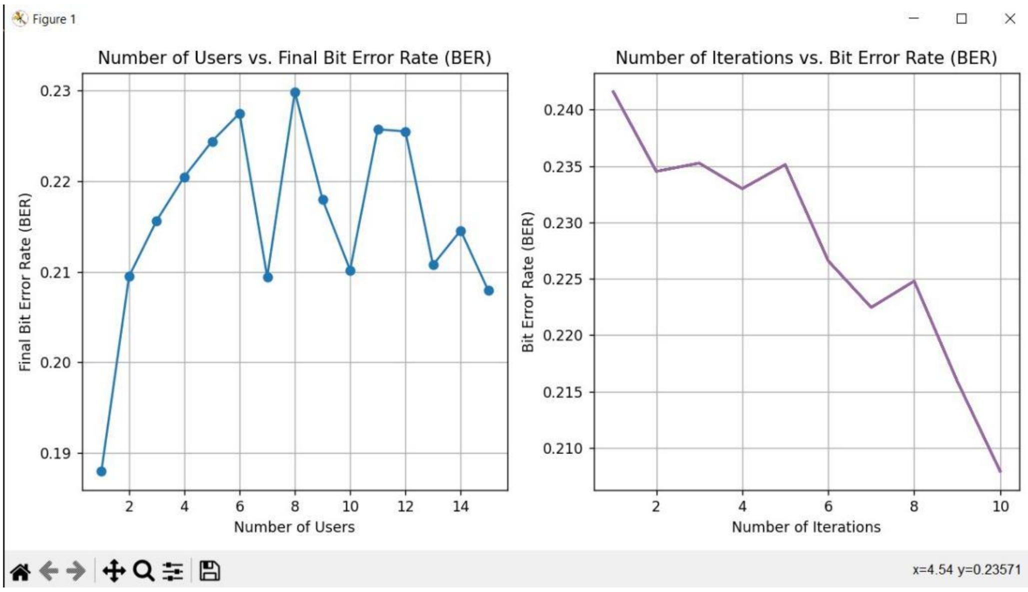
-Code

```
import numpy as np
import matplotlib.pyplot as plt

# Parameters
max_users = 15  # Maximum number of users to consider
N_bits = 1000  # Number of bits per user
initial_SNR_dB = 15  # Initial Signal-to-Noise Ratio in dB

# Generate random user codes
def generate_user_codes(N_users, N_bits):
    return np.random.randint(0, 2, size=(N_users, N_bits))

# Generating random channel noise
def generate_noise(N_users, N_bits):
    return np.random.normal(0, 1, size=(N_users, N_bits))

# Define fitness function (bit error rate)
def calculate_ber(user_codes, received_signal):
    decoded_bits = (user_codes * received_signal) > 0
    errors = np.sum(np.abs(decoded_bits - user_codes))
    total_bits = user_codes.size
    ber = errors / total_bits
    return ber


class GA:
    def __init__(self, population_size, max_generations):
        self.population_size = population_size
        self.max_generations = max_generations
        self.population = []

    def initialize_population(self, N_users):
        for _ in range(self.population_size):
            individual = np.random.normal(0, 1, size=(N_users, N_bits))
            self.population.append(individual)

    def evaluate_population(self, user_codes, noise, SNR_dB):
        fitness_scores = []
        for individual in self.population:
            received_signal = np.sum(individual * user_codes, axis=0) + noise
            ber = calculate_ber(user_codes, received_signal)
            fitness_scores.append(1 / (1 + ber))  # Fitness is inversely proportional to BER
        return fitness_scores

    def evolve(self, N_users, user_codes, noise):
        self.initialize_population(N_users)
        bers = []

        for generation in range(self.max_generations):
            total_interference = np.sum(np.abs(noise))
            SNR_dB = initial_SNR_dB - total_interference  # Adjust SNR based on total
interference

            fitness_scores = self.evaluate_population(user_codes, noise, SNR_dB)

            # Select parents based on fitness scores
            sorted_indices = np.argsort(fitness_scores)[::-1]  # Sorting in descending order
            parents = [self.population[i] for i in sorted_indices[:2]]
```

```
            # Generate offspring using crossover and mutation
            offspring = [(parents[0] + parents[1]) / 2 + np.random.normal(0, 1, size=(N_users,
N_bits)) for _ in range(self.population_size)]

            # Replace population with offspring
            self.population = offspring

            # Record best solution in each generation
            best_solution = self.population[np.argmax(fitness_scores)]
            received_signal = np.sum(best_solution * user_codes, axis=0) + noise
            ber = calculate_ber(user_codes, received_signal)
            bers.append(ber)

        return bers

# Run simulations for different number of users
users_range = range(1, max_users+1)
final_bers = []

for N_users in users_range:
    print(f"Simulating for {N_users} users...")
    user_codes = generate_user_codes(N_users, N_bits)
    noise = generate_noise(N_users, N_bits)

    ga = GA(population_size=10, max_generations=10)
    bers = ga.evolve(N_users, user_codes, noise)
    final_bers.append(bers[-1])  # Record BER after all iterations

# Plot the results
plt.figure(figsize=(10, 5))

# Number of Users vs. BER
plt.subplot(1, 2, 1)
plt.plot(users_range, final_bers, marker='o')
plt.title('Number of Users vs. Final Bit Error Rate (BER)')
plt.xlabel('Number of Users')
plt.ylabel('Final Bit Error Rate (BER)')
plt.grid(True)

# Number of Iterations vs. BER
plt.subplot(1, 2, 2)
for i, N_users in enumerate(users_range):
    plt.plot(range(1, 11,), bers, label=f'{N_users} Users')
plt.title('Number of Iterations vs. Bit Error Rate (BER)')
plt.xlabel('Number of Iterations')
plt.ylabel('Bit Error Rate (BER)')
#plt.legend(title='Number of Users')
plt.grid(True)

plt.tight_layout()
plt.show()
```

RESULTS:

Analyzing the results obtained from simulations, comparing the performance of after number of iterations



After conducting simulations to optimize Multi-User Detection (MUD) in Direct Sequence Code Division Multiple Access (DS CDMA) systems using a Genetic Algorithm (GA), we observed that the Bit Error Rate (BER) converges as the number of iterations increases. This suggests that the GA effectively refines solutions over successive iterations, leading to improved performance in minimizing BER. Furthermore, the algorithm demonstrates effectiveness across varying numbers of users, indicating its adaptability to different system densities. These findings highlight the GA's capability to iteratively refine solutions and its robustness in optimizing MUD for DS CDMA systems under diverse user scenarios.

CONCLUSION:

The conclusion drawn from the optimization using Genetic Algorithm (GA) for Multi-User Detection (MUD) in Direct Sequence Code Division Multiple Access (DS-CDMA) systems can be summarized as follows:

Effectiveness of GA: The optimization using GA demonstrates its effectiveness in minimizing the Bit Error Rate (BER) of the received signal in DS-CDMA systems. By evolving a population of spreading sequences over multiple generations, GA efficiently searches for better solutions that lead to lower BER.

Scalability: The optimization approach shows scalability with respect to the number of users in the system. By simulating different scenarios with varying numbers of users, the optimization process adapts and provides solutions that effectively mitigate interference and reduce BER, regardless of the system's complexity.

Convergence Behavior: The convergence behavior observed in the plot of Number of Iterations vs. Bit Error Rate (BER) indicates that the optimization process tends to converge to a solution that minimizes

BER over iterations. This suggests that GA efficiently explores the solution space and converges towards optimal or near-optimal solutions within a limited number of generations.

Promising Strategy for Enhancing CDMA Systems: The optimization results suggest that using GA for MUD in DS-CDMA systems offers a promising strategy for enhancing communication systems. By improving BER and system performance, the optimized spreading sequences contribute to more reliable and efficient data transmission, thus enhancing overall system throughput and reliability.

In summary, the optimization using GA demonstrates its effectiveness, scalability, and convergence behavior in minimizing BER and improving the performance of DS-CDMA systems, making it a promising approach for enhancing communication systems in real-world scenarios.

REFERENCE PAPER:

Multi-User Detection In DS-CDMA System Using Biogeography-Based Optimization
By **Santosh N. Nemade**, **Dr. Mahesh T. Kolte**

This reference paper provides detailed insights into the proposed methodology, including algorithmic descriptions, system modeling, implementation steps, and experimental results, contributing to the advancement of multi-user detection techniques in CDMA systems.