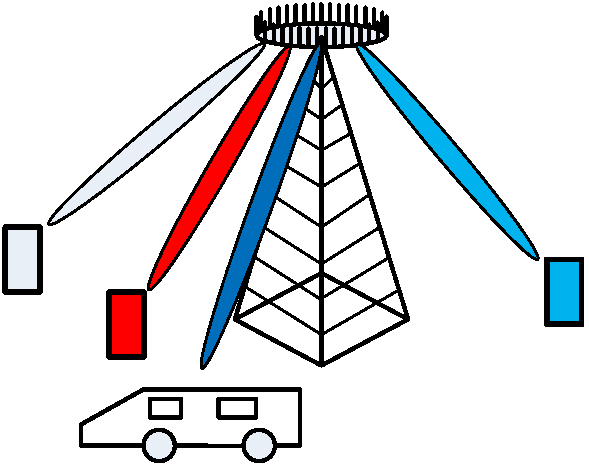
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| A black and white logo  Description automatically generated with low confidence | INTERNATIONAL TELECOMMUNICATION UNION  **TELECOMMUNICATION** **STANDARDIZATION SECTOR**  STUDY PERIOD 2022-2024 | | **Focus Group on AI Native Networks** | |
| **AINN-I-xx** | |
| **Original: English** | |
| **Question(s):** | | N/A | Virtual, TBD 2024 | |
| **INPUT DOCUMENT** | | | | |
| **Source:** | | *SSNCE-ECE* | | |
| **Title:** | | Dynamic Beamforming Optimization Using Genetic Algorithm for Mobile Devices, Fleets, and Drones in Multi-Antenna Systems | | |
| **Contact:** | | Srikanth S,  Syed Azim,  Prasanna Karthikeyan G | | E-mail:  [srikanth2110893@ssn.edu.in](mailto:srikanth2110893@ssn.edu.in),  [syed2110753@ssn.edu.in](mailto:syed2110753@ssn.edu.in),  [prasanna2110778@ssn.edu.in](mailto:Prasanna2110778@ssn.edu.in) |

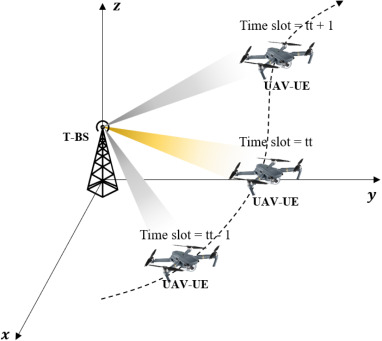
|  |  |
| --- | --- |
| **Abstract:** | This document presents a solution for dynamic beamforming in multi-antenna systems using a genetic algorithm, applicable not only to mobile devices but also to fleets and drones that require continuous monitoring. Conventional static or slowly adaptive beamforming mechanisms become inefficient when devices, fleets, or drones are in motion. This proposal addresses this issue by using historical antenna weight values and signal strength data to predict optimal weight adjustments in real time. A genetic algorithm explores a search space of potential weights, applying mutation and crossover techniques to maintain strong connectivity despite the uncertain movement of devices. This approach is designed to enhance wireless network performance in highly dynamic environments, such as transportation fleets, drone operations, and mobile communication systems*.* |

## Use case introduction:

In modern wireless systems, maintaining optimal beamforming is challenging when devices, vehicles, or aerial systems (such as drones) are moving. For example, in fleets or drone monitoring operations, real-time beamforming is crucial for maintaining uninterrupted connectivity. The proposed system dynamically adjusts antenna weight values to ensure consistent and strong signal strength, even as the monitored entities are in motion. By using a genetic algorithm, the system adapts to changes without relying on exact positional data, making it suitable for a variety of real-time, dynamic applications.

* **Phase 1:**  
  A device, fleet vehicle, or drone connects to the multi-antenna system. Initial antenna weight values are assigned, and signal strength is measured.
* **Phase 2**:  
  The connected entity (device, fleet, or drone) begins to move. The system detects a change in signal strength, indicating the need for dynamic weight adjustment.
* **Phase 3**:  
  The genetic algorithm is triggered. It initializes a population of weight values near the previous set, and signal strength is used to evaluate the fitness of each individual.
* **Phase 4**:  
  The algorithm applies mutation and crossover to iteratively search for better weight values, continuously optimizing beamforming as the entity moves.
* **Phase 5**:  
  Optimal weight values are found, signal strength is maximized, and the system maintains connectivity. Historical weight values are stored for future predictions.
* **Phase 6**:  
  If the entity moves again, the system uses the learned weight values to predict the next optimal beamforming configuration, repeating the process as needed.





**Use Case Requirements:**

* **Problem**: Current static beamforming methods struggle to maintain optimal connectivity in environments with moving devices, fleets, or drones.
* **Proposed Solution**: Use a genetic algorithm to dynamically adjust antenna weights based on historical weight values and signal strength data.
* **ML Concept**: Apply a genetic algorithm to optimize weight values in real time, ensuring connectivity for moving devices, fleets, and drones.

**Pipeline design (Genetic Algorithm Design)**

* **Initial Step**: When movement is detected (in devices, fleets, or drones), initialize a population of antenna weight values near the previous configuration.
* **Fitness Evaluation**: Signal strength serves as the fitness function to rank the effectiveness of each weight configuration.
* **Genetic Operations**: Crossover and Mutation are performed iteratively to explore better configurations.
* **Convergence**: The algorithm quickly identifies the optimal weight values to maintain signal strength.

**Relation to Standards**

* **5G/6G and Beyond**: The approach aligns with future telecommunications standards focusing on dynamic beamforming and multi-antenna systems.
* **Application to Fleets and Drones**: Extends the existing use cases of MIMO systems to fleet management and drone operations, where maintaining strong connectivity is crucial.

**Code submission**

Github link: <https://github.com/Srikanth-Drklrd/MIMO_Genetic_Algorithm.git>

Contains:

* Input Document
* Code Implementations in Jupyter notebook (.ipynb )
* Result in Jupyter notebook (.ipynb )

**Self-Testing results**

The signal strength using a dynamic genetic algorithm (blue-colored plot) is higher than that of having a static tracker without using any search algorithm (green-colored) plot. The red-colored plot is the signal strength transmitted by the transmitter, i.e. the maximum signal strength.

A graph showing a number of different colored lines

Description automatically generated with medium confidence

The error values of estimating the location of the transmitter is calculated by using Euclidean distance between the actual location of the transmitter and the estimated location of the transmitter. The Error is kept below 200 units by the algorithm despite the fast motion of the Ackley function (high speed of transmitter motion).

A graph showing a line graph

Description automatically generated

**Appendix A:**

**Genetic Algorithm Overview:**

In this section, we explain the Genetic Algorithm (GA) implemented in the Python notebook for dynamic beamforming in multi-antenna systems.

**1. Initialization:**

The GA starts by initializing a population of potential solutions (antenna weight values). These solutions are initialized near the previous optimal weight configuration to ensure the search starts from a reasonable point in the solution space. If the previous optimal weight configuration is unavailable, then the search begins at a random pooint.

**2. Fitness Function:**

The fitness of each solution is evaluated based on the signal strength obtained from the antenna configuration. Higher signal strength results in higher fitness scores. This metric ensures that the algorithm is always optimizing toward the best connectivity for mobile devices, fleets, and drones.

Here we use, dynamic time-varying Ackley function to effectively model the challenges faced by a drone's antenna system as it navigates through different spatial and temporal conditions. The dynamic Ackley function is often used in optimization problems due to its complex landscape, which includes multiple local minima and a single global minimum. When considering its movement in the x, y, and z directions over time, you can think of it as a three-dimensional surface that changes dynamically, creating a rich landscape that can simulate various conditions.

A graph of a function

Description automatically generated

**Reasons for Using Ackley function as fitness function and model for drone antenna beamforming**

1. **Global Minima and Antenna Orientation**: The global minima of the Ackley function can be likened to optimal antenna orientations. Just as the Ackley function seeks to minimize energy, an optimal polarization map indicates the best configuration for signal reception or transmission.
2. **Movement in Space**: As the Ackley function dynamically changes with respect to time, it can represent a drone moving through various environmental conditions. The changing surface might reflect varying signal strengths, obstacles, or atmospheric effects affecting the antenna's performance.
3. **Complex Landscape**: Both the Ackley function and the antenna map involve navigating complex landscapes. The Ackley function's rugged terrain represents challenges like interference or multipath propagation that a drone might encounter as it moves.
4. **Optimization**: The goal in both scenarios is optimization. For the Ackley function, it's finding the minimum, while for the drone, it's about finding the best antenna configuration for given conditions, possibly adapting in real-time as the environment changes.

**3. Selection:**

Selection involves choosing the fittest individuals from the current population for reproduction. In our implementation, individuals with better fitness (stronger signal strength) have a higher probability of being selected.

**4. Crossover (Recombination):**

Crossover is applied to the selected individuals to produce the next generation. In this step, portions of the antenna weight values from two parent solutions are combined to create a new solution. This encourages the exploration of new configurations while regaining the properties of the parent.

**5. Mutation:**

Mutation introduces small random changes to individual solutions. This step is crucial for maintaining diversity within the population and avoiding premature convergence on suboptimal solutions. Since vehicle motion can be abrupt and random, introducing some amount of randomness in the search process could prove to be beneficial.

**6. Adaptive Learning:**

As the GA operates, it learns from the historical weight values and signals strengths, enabling it to predict the next optimal weight configurations when the device, fleet, or drone moves again.

**7. Error function:**

The Euclidean distance between the estimated genetic algorithm and the actual point of maximum signal strength is used as an error function.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_