# Evolutionary Algorithms in High-dimensional Radio Access Network Optimization

Dmitriy Semenchikov
Saint Petersburg State University
Huawei, SPb Research Center
dmitriy.semenchikov@huawei.com

Anna Filippova
Huawei, SPb Research Center
filippova.anna@huawei.com

Dmitriy Volf
Saint Petersburg State University
Huawei, SPb Research Center
volf.dmitriy@huawei.com

Nikolai Kovrizhnykh
Saint Petersburg State University
Huawei, SPb Research Center

Maxim Mironov
Moscow Institute of Physics and Technology
Huawei, SPb Research Center

Zou Jinying
Saint Petersburg State University
Huawei, SPb Research Center

Luo Ronghui
Huawei, China

Zhu Yuanming
Huawei, China

Huang Wei
Huawei, SPb Research Center

Chai Dapeng
Huawei, SPb Research Center

## ABSTRACT

This article describes the project result of modeling and optimizing Radio Access Network. We have proposed a solution for controlling a large number of antennas in the conditions of engineering constraints and a large search space dimension. For estimating the performance, a virtual environment has been developed, that allows changing the parameters of Radio Access antennas to control the coverage and signal quality for all User Equipments. To optimize the Radio Access network, we have analyzed DE, CMA-ES, MOS, self-adaptive surrogate CMA-ES, lq-CMA-ES, BIPOP CMA-ES, sep-CMA-ES, lm-CMA-ES, HMO-CMA-ES, JADE, PSO, which have been adapted to the constraints of the task. To reduce dimension, graph clustering methods - Spectral clustering, Label propagation, Markov Clustering - are compared in dividing the network into groups. The experiments illustrate the efficiency of optimizing a large Radio Access network by the cluster approach.

## KEYWORDS

Telecommunications, Evolution strategies, Parameter control, Multi-objective optimization, Empirical study

**ACM Reference Format:**
Dmitriy Semenchikov, Anna Filippova, Dmitriy Volf, Nikolai Kovrizhnykh, Maxim Mironov, Zou Jinying, Luo Ronghui, Zhu Yuanming, Huang Wei, and Chai Dapeng. 2021. Evolutionary Algorithms in High-dimensional Radio Access Network Optimization. In *2021 Genetic and Evolutionary Computation Conference Companion (GECCO '21 Companion), July 10–14, 2021, Lille, France.* ACM, New York, NY, USA, 8 pages. https://doi.org/10.1145/3449726.3463216


*GECCO '21 Companion, July 10–14, 2021, Lille, France*

ACM ISBN 978-1-4503-8351-6/21/07…$15.00
https://doi.org/10.1145/3449726.3463216

## 1 INTRODUCTION

Radio Access networks provide access to wide spectrum and multi-gigabit data rates for User Equipments (UE) [5]. One of the key features of these systems is the phased array antennas that are able to focus the signal at a desirable direction. That helps to enhance the power of usable signal and to decrease the cross-cell interference.

The most important problem when working at the High-Band spectrum is the limited coverage area and invalid dead zones. The physics of propagation of a radio signal at mmWave frequencies is different from low or medium bands. Thus, an important task is to improve the coverage and signal quality for the UEs of the network when operating Radio Access networks.

This problem involves an uncertain environment in which it isn't trivial to obtain a physical model of propagation and attenuation of signals for using the derivative-based methods. Thus, it was suggested to apply the Black-Box optimization approach [16], which doesn't have this drawback and could be adapted to different signal simulation systems.

In general, this problem is a multi-objective optimization problem involving multiple performances of the network signal. Evolutionary algorithms have been successfully proven themselves as reliable solutions for Radio Access networks in the problems of Energy consumption [6], Resource Allocation [4, 22], and Pre-coding [7].

As noted earlier, deployment and maintenance of networks involve optimization of service coverage and radio base station antenna configuration where was suggested several the solution approaches based on local search methods, e.g., simulated annealing [23], or iterative optimization procedure based on Taguchi's method [2]. The solution approach this article adopts is an Evolutionary algorithm and consider business scenario on the angle parameters.

The main contributions of this article are:

- Evolutionary algorithms' analysis for Radio Access Network Optimization in low and high dimension scenarios,
- Comparison of network clustering methods.

*In Section 2*, we define the optimization problem, describe the Radio Access network Model for signal simulation. *In Section 3*, we explain the main ideas of researched optimization algorithms. *In Section 4*, we describe graph clustering methods for reducing the problem dimension by dividing the network into groups. *In Section 5*, we introduce experiment results on high-dimension and low-dimension problems.

## 2 PROBLEM FORMULATION

We consider the model of Radio Access Network. The target function is a hybrid quality indicator:

$$\begin{gathered}\min_x \left(Coverage(x), Interference(x)\right),\\ x = [h_i, v_i, t_i, a_i, \ldots, h_N, v_N, t_N, a_N],\end{gathered} \tag{1}$$

where:

- Coverage (Cov) is the network coverage indicator,
- Interference (Inf) is the signal quality indicator,
- $h_i$ – $i$-th cell's horizontal width of beam (HBW),
- $v_i$ – $i$-th cell's vertical width of beam (VBW),
- $t_i$ – $i$-th cell's antenna tilt,
- $a_i$ – $i$-th cell's antenna azimuth,
- $N$ – number of cells.

In the following subsections, we describe the mathematical model of signal propagation in a radio access network.

### 2.1 Radio Access Network Model

Radio Access Network consists of:

- Grid of $N$ cells. Each cell has a set of static parameters: coordinates, height, transmitted power. Controllable parameters are: HBW, VBW, azimuth and tilt. Each cell has 8 Synchronization Signal Block beams (SSB beams).
- Grid of $M$ User Equipments. Each UE is a static object and has only fixed coordinates.

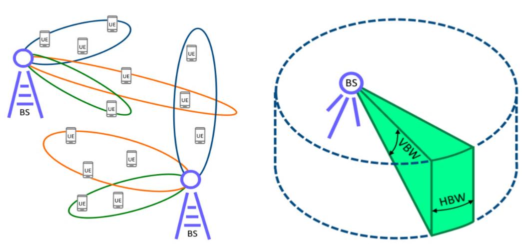


**Figure 1: Left. Covering users with multiple beams. Right. HBW and VBW parameters of cell beam**

To calculate the performance of the network, each cell's control parameters are applied. Next, the simulation system performs the following operations:

(1) Calculation of the Reference Signal Received Power (RSRP) from the cells for each UE.
(2) Assignment of the serving cell and SSB beam for each UE.
(3) Calculation of the Signal to Interference+Noise Ratio (SINR) for each UE.

#### 2.1.1 RSRP.

The power of the signal received by UE $u$ from the SSB beam $b$ of cell $c$:

$$\mathrm{RSRP}_{c,b,u} = \mathrm{Tx}_c - \mathrm{PL}_{c,b,u}, \tag{2}$$

where $\mathrm{Tx}_c$ – cell's transmit power, $\mathrm{PL}_{c,b,u}$ – propagation loss.

Propagation loss $\mathrm{PL}_{c,b,u}$ is a sum of simplified Log-distance path loss $\mathrm{PL}^D_{c,u}$ and attenuation caused by beamforming $\mathrm{PL}^B_{c,b,u}$. Simplified Log-distance path loss model represents logarithmic power loss [17]:

$$\mathrm{PL}^D_{c,u} = \begin{cases} L_0 + 10n\log_{10}\frac{d_{c,u}}{d_0}, & \text{if } d_{c,u} > d_0,\\ 0, & \text{otherwise,}\end{cases} \tag{3}$$

where $d_0 = 0.001$ is reference distance, $L_0 = 46.677$ is path loss at $d_0$, $n = 3$.

Attenuation caused by beamforming depends on angular difference between SSB beam direction and direction to UE [18]:

$$\mathrm{PL}^B_{c,b,u} = 10 \cdot 1.2\left(\left(\frac{\Delta a_{c,b,u}}{h_c}\right)^2 + \left(\frac{\Delta t_{c,b,u}}{v_c}\right)^2\right), \tag{4}$$

where $\Delta a_{c,b,u}$ and $\Delta t_{c,b,u}$ are angular differences in azimuth and tilt, respectively.

#### 2.1.2 Assigning serving cells.

In our model, the Radio Access network is static at each timestamp, no handovers happen. Thus, UE-to-beams serving map $\mathrm{Serv}(u)$ is defined as follows:

$$\mathrm{Serv}(u) = \arg\max_{c,b}\left(\mathrm{RSRP}_{c,b,u}\right). \tag{5}$$

#### 2.1.3 Coverage.

WC is a key performance indicator that shows the amount of UEs with received signal lower than a threshold value $Thd_{WC}$ [9].

$$\mathrm{WC} = \frac{1}{M}\sum_i^M \left[\mathrm{RSRP}_{\mathrm{Serv}(i),i} < Thd_{WC}\right], \tag{6}$$

where, depending on the selected threshold, the quality of the network coverage can be determined. The threshold value is usually taken in the range from -90 to -80 (in dBm).

#### 2.1.4 SINR.

SINR is power of usable signal from UE's serving cell, divided by total power of signals from neighboring cells and noise [24].

$$\mathrm{SINR}_u = \frac{10^{0.1\cdot\mathrm{RSRP}_{\mathrm{Serv}(u),u}}}{\sum_{\mathfrak{B}} 10^{0.1\cdot\mathrm{RSRP}_{c,b,u}} + \eta_u}, \tag{7}$$

where $\mathfrak{B} = \{(c,b)|(c,b) \neq \mathrm{Serv}(u), b = \mathrm{Serv}(u)_b\}$ – set of SSB beams sharing the same time slot, $\eta_u$ – background noise on the user side (e.g., $-97$dBm).

#### 2.1.5 *Interference.*

INF is a key performance indicator that shows the amount of UEs with SINR value lower than threshold value $Thd_{INF}$.

$$\text{INF} = \frac{1}{M}\sum_{i}^{M}\left[\text{SINR}_i < Thd_{INF}\right], \tag{8}$$

The threshold value is usually taken in the range from 3 to 10 (in dB).

# 3 OPTIMIZATION ALGORITHMS

A brief description of each researched optimization algorithms are presented in this section. In a view of the problem, two main classes of algorithms can be considered: single-objective, using the linear scalarization method,

$$\min_{x}\left(wCoverage(x) + (1-w)Interference(x)\right), \tag{9}$$

and multi-objective, where the performances are optimized separately (1).

Based on the related works [3, 25], it was assumed that Covariance Matrix Adaptation Evolution Strategy (CMA-ES) or its modifications outperform basic algorithms such as Particle Swarm Optimization (PSO), Differential Evolution (DE). Thus, we gave due attention to the variations of CMA-ES, including techniques for: reducing the complexity of the covariance matrix, controlling the step-size, restarting and approximating the target function (surrogate concept). For multi-objective we considered a special modification of the CMA-ES – Hybrid Multi-Objective CMA-ES (HMO-CMA-ES).

## 3.1 Particle Swarm Optimization

Particle Swarm Optimization (PSO) [10] is a metaheuristic population-based algorithm based on the bird's swarm concept, where a swarm is a special group with interaction and behavior rules. The main property of this concept is swarm intelligence which means the ability to communicate between each member and to use the previous experience.

In fact, PSO particles based on the two vectors: position $X_i^t = [x_{i1}, x_{i2}, x_{i3}, \ldots, x_{in}] \in \mathbb{R}^n$ and velocity $V_i^t = [v_{i1}, v_{i2}, v_{i3}, \ldots, v_{in}] \in \mathbb{R}^n$ at each iteration $i$. In order to estimate the fitness value of each particle, it is necessary to apply the position vector to the objective function. After that, an iterative process is started based on:

(1) Velocity update, based on best position of each particle and the whole particle in the swarm.
(2) Position update, based on the previous position and current velocity

## 3.2 Differential Evolution

The Differential Evolution algorithm [19] is a population-based stochastic optimizer based on the mechanism of natural selection. This algorithm consists of three operations: mutation, crossover, and selection. For a new population at each iteration, mutation and crossover are applied to individuals. Then, in a selection phase, each individual of a new population is compared to the old ones, and in the next iteration, only the best individuals take a part.

## 3.3 JADE

JADE is a modification of DE based on the implementation of a new mutation strategy "Current-to-$\rho$best" with Optional Archive and adaptive generation of parameters for crossover and mutation operations [28], which cease to be a constant and are generated in each iteration according to the Gaussian distribution and the Cauchy distribution respectively.

## 3.4 Covariance Matrix Adaptation Evolution Strategy

The CMA-ES [12] is a state-of-art population-based evolution algorithm for black-box optimization problems in continuous domain, which has proven to be an efficient algorithm for high-dimension, noisy and multimodal benchmarks of objective functions.

#### 3.4.1 *Main idea.*

CMA-ES algorithm can take the results of each generation, and adaptive increase or decrease the search space for the next step. Each iteration of the algorithm consists of 4 main operation:

(1) *Sampling and evaluating.* New candidates of each generation are samples normally distributed
(2) *Selection and Recombination.* Based on the individuals' fitnesses, only the best individuals from the population survive and become parents. Only fitness ranks are used.
(3) *Mutation.* This step includes covariance matrix adaptation, where the new one consists of information about: the previous state, correlations between generations (particularly important in small populations) and the entire population (important in large populations). The covariance matrix adaptation increases or decreases the scale only in a single direction for each selected step. That is why a step-size is needed in addition.

#### 3.4.2 *Reducing the covariance matrix complexity.*

The original CMA-ES algorithm has a computational complexity of $O(n^2)$ for each iteration [21]. For high-dimensional applications, CMA-ES can be a resource-intensive and time-consuming solution. To reduce complexity, there are the following approaches:

- **Cholesky-CMA-ES** [12]. Modification is based on the Cholesky decomposition for the covariance matrix.
- **lm-CMA-ES** [12]. Consists of 3 changes: reconstruction of the Cholesky factors of a covariance matrix using stored direction vectors and limited memory for their storage, and a non-standard technique for step-size adaptation.
- **sep-CMA-ES** [21]. Based on update of the covariance matrix diagonal elements only and the learning rate increase.

#### 3.4.3 *Step-size controlling.*

- **Two-Point Adaptation (TPA)**. In Two-Point Step-Size Adaptation [1], the first two individuals in the population are sampled along the shift vector from the previous solution, $X_{t-1}$, to the current solution $X_t$, as a mirrored pair, symmetric to $X_t$. If $X_t^1$ is better than $X_t^2$, $\sigma_t$ is increased as this indicates that there are better solutions in the direction of the latest solution shift. Otherwise, it is decreased.
- **Median Success Rule (MSR)**. The Median Success Rule Step-Size Adaptation [1] is defined as the median individual

of the current population, $X_t^{m(\lambda)}$, being better than the $j$-th best individual of the previous population, $X_{t-1}^{j:\lambda}$. The idea is then to increase the step-size if $X_t^{m(\lambda)}$ is fitter than $X_{t-1}^{j:\lambda}$ and decrease it otherwise.
- **TPA/MSR with CSA**. It was observed that TPA and MSR have a good starting speed at the first iterations, while Cumulative Step-size Adaptation (CSA), the standard step-size control method of CMA-ES, eventually converges to a higher value of the objective function. Thus, the sequential use of two methods – TPA or MSR with CSA – outperforms their result separately.

*3.4.4 Restart strategy.* Restart strategies have performed well on multimodal and noisy optimization benchmarks [14]. **BIPOP-CMA-ES** [8] is one of these concepts for CMA-ES. At each restart one of two scenarios are selected by estimating the function evaluation for each of them. The first scenario restarts with doubled population size and fixed step-size, the second one – with some small population size and small step-size.

*3.4.5 Target function approximation.* Our product environment is a very expensive system in terms of calculating the target function. If there are time constraints, it becomes impossible to obtain an optimal solution. Thus, we decided to consider the idea of a **surrogate CMA-ES** [15], in which new individuals can be sent partially or completely to a model trained on the original target function.

As a surrogate, any machine learning model to build predictions of the target function can be used. To determine the accuracy of this model, it is important to use the ranking metric of the obtained results. Depending on the value of the metric, the number of individuals or the number of generations to evaluate the objective function by surrogate model is determined.

*3.4.6 Multiple Offspring Sampling framework (MOS).* **MOS** [11] is a framework that allows to combine algorithms by controlling their participation in each new generation, depending on their performance at the previous stage. MOS allows to combine any metaheuristics optimization algorithms. Our choice was a combination of DE and CMA-ES.

## 3.5 Hybrid Multi-Objective CMA-ES

HMO-CMA-ES is a multi-objective optimization algorithm, which combines four sequentially operating algorithm [13]. Three of them is a variation of CMA-ES, supplemented by BOBYQA as a warm start. The order of the algorithms is as follows:

(1) Bound Optimization BY Quadratic Approximation (BOBYQA)
Optimize the weighted target function (9). Trust-region method BOBYQA is used to quickly approximate the Pareto front. A solution set of all runs are collected for the starting population of the following algorithm.
(2) Steady-state MO-CMA-ES (ss-MO-CMA-ES)
Optimize the multi-objective target function (1). This step is needed to quickly refine the Pareto front.
(3) Increasing population size CMA-ES (IPOP-MO-CMA-ES)
Optimize multi-objective target function (1). This is the Multi-Objective CMA-ES, which increases population size in twice every k iterations.
(4) Restart CMA-ES
Optimize the weighted target function (9). The algorithm is consistently restarted from the different starting points with different coefficients $w$.

# 4 GROUPED APPROACH

The networks considered in this paper assume more than 1000 cells, and since 4 parameters (HBW, VBW, azimuth, tilt) are optimized for each station, the search space becomes larger than 4000, which makes the problem of optimizing a radio access network high-dimensional. In this regard, the network can be divided into clusters and optimized separately, thereby reducing the dimension of the search space for the optimal solution.

When the network is divided into clusters, the interference between the UEs and the BSs is lost at cluster borders. This is due to the loss of information about the closeness of the BS during the optimization of each group. Thus, the signal from the border UE is overestimated, since the noisy signal from the BSs of neighboring clusters is no longer taken into account.

However, this approach has several advantages. The total time required to optimize the entire network across clusters is significantly less, due to the reduction of the optimization dimension of problem on each cluster.

In addition, our experiments show that the network final performance after group optimization are higher than after network optimization without clustering.

A method for constructing a graph representation of a radio access network and methods for dividing graphs into clusters (graph partitioning/graph clustering) is below.

## 4.1 Graph model

A Radio Access network can be interpreted as a graph that represents the relationships between each Base Station. Define a graph $G = (V, E)$ as a set of vertices together with a set of edges. The set of vertices $V = \{BS_1, BS_2, .., BS_N\}$ includes all BSs of the radio access network. Initially, all vertices are connected by a set of edges $E$, which characterizes the neighborhood of base stations. It is intuitively clear that all base stations are not equivalent neighbors to each other. Thus, it is possible to reflect their relationship through the function of weights for each edge. As a relationship between two Base Stations, let's consider:

- Euclidean Distance,
- Interference, assuming UE stands in the half of distance between them.

In order to convert into an affinity matrix, for example, built on the Euclidean distance between BSs, let's use the following equation:

$$A(i, j) = \exp\left(-\frac{\|x_i - x_j\|^2}{2\sigma^2}\right). \quad (10)$$

As a result, we get a weighted graph $G$ that reflects the initial relationships of stations in the network. Now we can apply graph partitioning/graph clustering algorithms to it.

### 4.2 Spectral Clustering

One of the most popular clustering methods in machine learning, computer vision, and speech processing is the spectral clustering algorithm. The spectral clustering algorithm [27] belongs to a class of methods that use the eigenvalues of the similarity matrix to combine similar objects into a single cluster. After the Spectral Clustering has received the affinity matrix as input, the following steps are taken:

(1) Calculate the graph Laplacian matrix by the formula:

$$L = D - A, \tag{11}$$

where $D$ is the degree matrix, which is a diagonal matrix with values that characterize the degrees of the vertices, and $A$ is the adjacency matrix.

(2) Use spectral (eigen) gap to select value of $k$ parameter.

(3) Form a new matrix from the eigenvectors of $L$ that match to the $k$ the largest eigenvalues.

(4) Cluster matrix by k-Means.

### 4.3 Label Propagation

Label Propagation [20] (LPA) is a semi-supervised machine learning algorithm based on the distribution of information in the network to detect the communities.

The main idea is to spread the labels over the vertices. Initially, each vertex has its own unique label. At each iteration of the propagation, a random vertex assigns itself a label that is most common among its neighbors. The algorithm converges if no label updates occurred during the iteration.

### 4.4 Markov Clustering

Markov Clustering (MCL), proposed by Stijn van Dongen [26], is an elegant graph clustering algorithm based on simulation of stochastic flows between vertices of the graph. MCL algorithm is robust to noise. MCL consists of two main operations: expansion and inflation, which are applied alternately.

The expansion helps in making the farther nodes or neighbors reachable. This is achieved mathematically by taking the e-th power of the matrix:

$$A_{expand} = A^e. \tag{12}$$

The inflation helps in making the strong neighbor values are strengthened and large neighbor values are demoted. This is can be achieved by raising the column value to non-negative power and then re-normalizing:

$$A_{inflate} = \frac{A(i,j)^r}{\sum_k A(k,j)^r}. \tag{13}$$

## 5 EXPERIMENTS

Experimental procedures and results are presented and discussed in this section. Comparison of algorithms performance based on the (9) objective function where $w = 0.5$.

Each experiment uses a start point $x_i$ from (9) for optimization based on the current configuration in the Network. It should be mentioned that, as these optimization algorithms have stochastic search which would lead to some changing results in different iterations to some degree, we make 15 runs for each experiment and obtain the mean of performance.

All experiments are performed using PC with Intel® Xeon® CPU E5-2690 v4, 2.60GHz, 64GB RAM, but should note that we didn't use parallel computing. The operating system is Ubuntu 16.04.6.

### 5.1 Virtual Environment

According to the Radio Access Network model described in Section 2, a virtual environment was developed. It allows to evaluate the performance of the network based on an applied configuration of control parameters.

The network input parameters are the number of Cells and their initial configurations, consisting of: coordinates, height, transmitted power, HBW, VBW, azimuth and tilt. Also, the location of User Equipment on the network is set.

As an example, we could consider 9 cells and place UEs around them on the grid. Thus, we build a map of values by calculating the received signal for each user, taking into account the path losses. (Fig. 2)

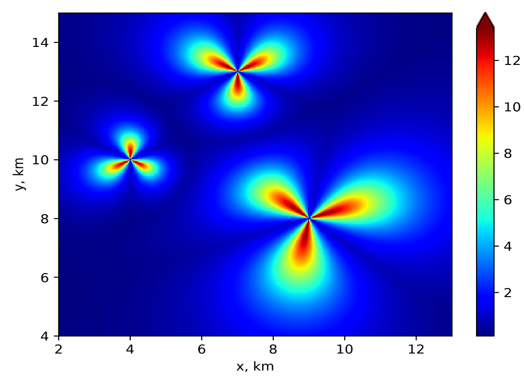


**Figure 2: Example of signal propagation of 9 cells**

### 5.2 Simulation results

*5.2.1* ***40 Dimension Benchmark.*** Experiments were conducted in 40D for several algorithms, where 10 cells were optimized. (Fig. 4) 40D was chosen as the most popular size of the cluster at Grouped Approach.

As surrogates for the CMA-ES algorithm, we use the following models on fig. 3: Gradient Boosting, XGBoost, Random Forest, Decision Tree, and NuSVR. Depending on the quality of the model, at each iteration, the algorithm decides whether to use a surrogate for evaluating the objective function or the original function. We selected the following threshold values 0.2, 0.12, 0.07 of accuracy according to which 3, 4, or 5 iterations were allocated. The results show that the Gradient Boosting and XGBoost models have better accuracy compared to other models. Due to their similar results, we chose XGBoost as a surrogate for the CMA-ES algorithm.

The results on figures 4 present the performance of objective function (9) where $\omega = 0.5$ improvement for each algorithm. The HMO-CMA-ES algorithm work with original KPIs Coverage, Interference and combine them by using weight equal to 0.5 to compare with single-optimization algorithms. Furthermore, the staircase

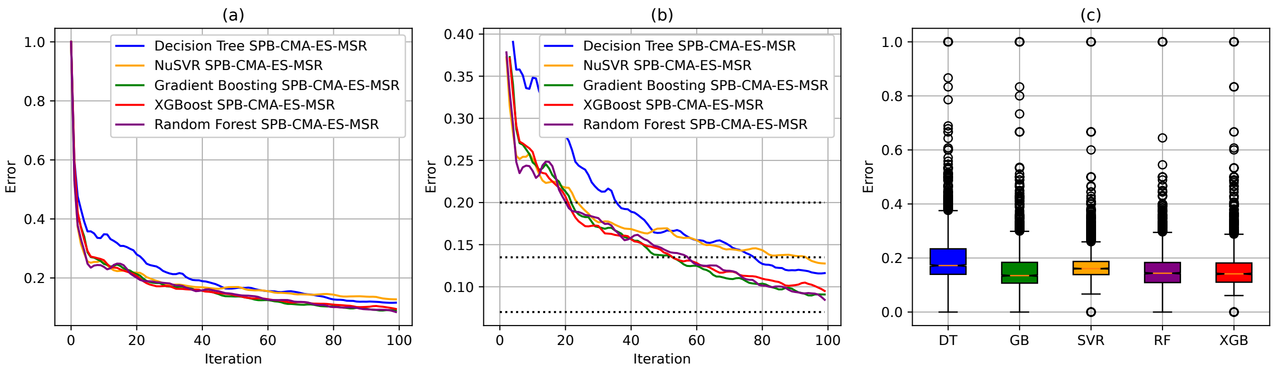


**Figure 3: Surrogate models training for the 40D problem.**
**a. Rank-error on each iteration, b. Zoomed rank-errors on each iteration, where the dashed line are thresholds, c. Box-plot of rank-errors.**

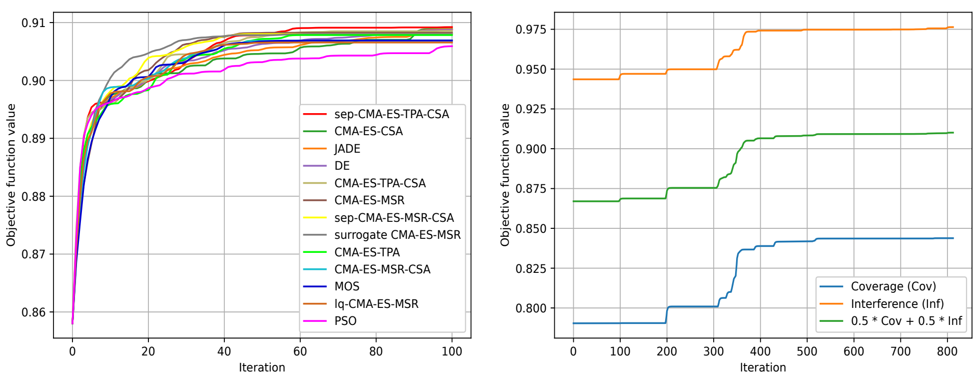


**Figure 4: Left. Single-objective optimization algorithms performance on 40D. Right. Multi-objective HMO-CMA-ES performance on 40D.**

pattern in figure 4 based on the ss-MO-CMA-ES algorithm performance which start optimization on the non-dominated set results from BOBYQA as initial points.

| Algorithm | Fitness | Time, sec |
|---|---|---|
| **sep-CMA-ES-TPA-CSA** | **0.9092** | 34.85 |
| CMA-ES-CSA | 0.9089 | 41.65 |
| JADE | 0.9088 | 35.72 |
| DE | 0.9087 | 33.38 |
| CMA-ES-TPA-CSA | 0.9086 | 40.96 |
| CMA-ES-MSR | 0.9082 | 41.41 |
| sep-CMA-ES-MSR-CSA | 0.9080 | 34.74 |
| **surrogate CMA-ES-MSR** | 0.9080 | **24.92** |
| CMA-ES-TPA | 0.9078 | 40.80 |
| CMA-ES-MSR-CSA | 0.9069 | 40.79 |
| MOS | 0.9069 | 47.48 |
| lq-CMA-ES-MSR | 0.9066 | 55.39 |
| PSO | 0.9059 | 35.69 |
| **HMO-CMA-ES** | **0.9101** | 88.93 |

**Table 1: Optimization results for 40D.**

On the low-dimension (40D) all algorithms has comparable performance with difference in the 3 decimal after the point. Thus, the best algorithm has the worst time, because it restarts with bigger population size, while a surrogate with bad performance reduces the time. (Tab. 1)

*5.2.2* ***High-dimension scenario.*** The study of optimization algorithms in high-dimension problems is one of the main goals of this paper. Radio Access network includes a large number of control parameters, which makes the task time-consuming. As Telecommunication technologies are developing globally and will be supplemented in the future, the dimension of the task will only increase.

In order to evaluate the performance of algorithms in high-dimensions, consider the following settings: 482 cells with 4 parameters for each, UE grid at a distance of nearly 10 meters from each other, $Thd_{WC} = -90dBm$, $Thd_{INF} = 3dB$ and population size – 10.

The settings of surrogate algorithm for high-dimension are the same with setting from 5.2.1. It could be noted that the prediction error of the model is worse on the high-dimension problem. Also, we use XGBoost model as surrogate for CMA-ES.

The results are shown in the table 2. According to the final performance values, the CMA-ES-TPA-CSA outperforms other algorithms. At the same time, the surrogate CMA-ES significantly reduces the time for optimization, getting a not excellent final result.

TPA, MSR methods adapt step-size based on the values of the current population evaluations as opposed to CSA, hence its results.

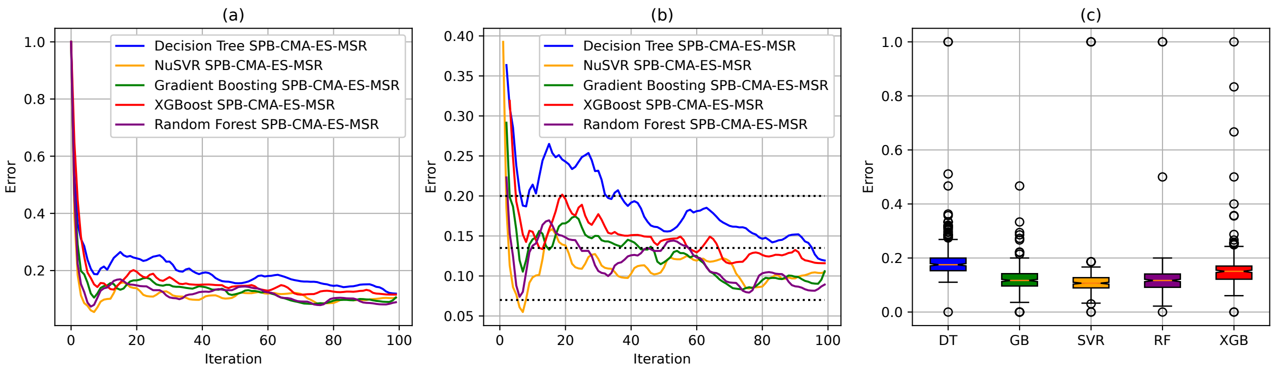


**Figure 5: Surrogate models training for the high-dimension problem.**
**a. Rank-error on each iteration, b. Zoomed rank-errors on each iteration, where the dashed line are thresholds, c. Box-plot of rank-errors.**

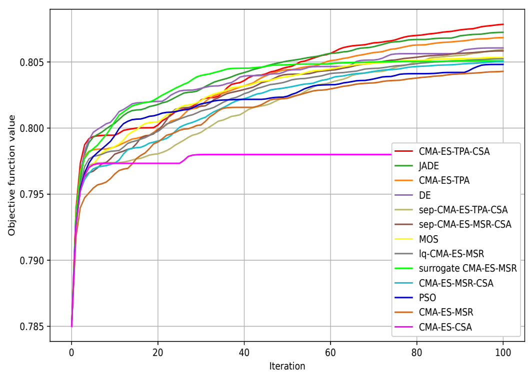


**Figure 6: Optimization algorithms performance on high-dimension.**

| Algorithm | Fitness | Time, h |
|---|---|---|
| **CMA-ES-TPA-CSA** | **0.8078** | 2.01 |
| JADE | 0.8072 | 2.07 |
| CMA-ES-TPA | 0.8068 | 2.01 |
| DE | 0.8061 | 1.99 |
| sep-CMA-ES-TPA-CSA | 0.8059 | 2.04 |
| sep-CMA-ES-MSR-CSA | 0.8058 | 2.02 |
| MOS | 0.8053 | 3.06 |
| lq-CMA-ES-MSR | 0.8052 | 2.16 |
| **surrogate CMA-ES-MSR** | 0.8051 | **1.15** |
| CMA-ES-MSR-CSA | 0.8050 | 2.01 |
| PSO | 0.8048 | 2.02 |
| CMA-ES-MSR | 0.8043 | 2.02 |
| CMA-ES-CSA | 0.7980 | 1.99 |

**Table 2: Optimization results for high-dimension (482 cells * 4 parameters).**

*5.2.3* ***Grouped Approach***. Experiments were conducted with each clustering algorithm. Each cluster was optimized separately with BS initial configurations as control parameters initial values. After optimization, the result was combined to the one vector and applied to the virtual environment for estimating the KPIs final values. The combination of the final values shows the signal interference between the clusters.

| | Markov | | Spectral | | Label | |
|---|---|---|---|---|---|---|
| Algorithm | Fitness | Time, h | Fitness | Time, h | Fitness | Time, h |
| BIPOP-CMA-ES | 0.813 | 1.63 | 0.810 | 0.809 | 0.812 | 1.42 |
| CMA-ES-TPA-CSA | 0.809 | 0.96 | 0.805 | 1.63 | 0.808 | 0.88 |
| JADE | 0.816 | 0.93 | 0.816 | 1.69 | 0.817 | 0.86 |
| sep-CMA-ES-MSR | 0.814 | 0.91 | 0.814 | 1.73 | 0.814 | 0.83 |
| DE | 0.817 | 0.9 | 0.816 | 1.65 | 0.817 | 0.82 |
| CMA-ES-TPA | 0.815 | 0.95 | 0.814 | 1.68 | 0.816 | 0.88 |
| MOS | 0.814 | 1.3 | – | – | 0.815 | 1.16 |
| **surrogate CMA-ES-MSR** | **0.816** | **0.59** | **0.814** | **1.01** | **0.816** | **0.56** |
| CMA-ES-MSR-CSA | 0.807 | 0.93 | 0.805 | 1.62 | 0.808 | 0.82 |
| PSO | 0.814 | 0.94 | 0.812 | 1.66 | 0.814 | 0.85 |
| CMA-ES-MSR | 0.814 | 0.93 | 0.813 | 1.67 | 0.815 | 0.83 |
| lq-CMA-ES-MSR | 0.815 | 1.06 | 0.814 | 1.71 | 0.814 | 0.93 |
| CMA-ES-CSA | 0.815 | 0.97 | 0.811 | 1.68 | 0.815 | 0.88 |

**Table 3: Optimization results for Grouped Approach.**

The Network was modeled like undirected weighted graph $G = (V, E)$ where $V = \{BS_1, BS_2, ..., BS_n\}$, $|V| - number\ of\ BSs$; $E = \{(v_i, v_j) \mid BS_i, BS_j - \text{neighbor BaseStations}\}$; weight function based on the Euclidean metric of BS locations.

Hyper-parameter settings for each cluster algorithm:

- For MCL – default values for expansion is 2, for inflation – 2. The number of clusters – 94.
- For LPA – since LPA involves a certain amount of randomization, we execute each test 10 times and use the means to obtained the performance. The number of clusters – 113.
- For Spectral clustering – the Elbow method estimates optimal number of clusters. The number of clusters – 24.

The table 3 shows the total time (clusters were optimized sequentially) and the final performance for each optimization method. On Markov clusters all algorithms achieve a higher final KPI for the entire network. Surrogate CMA-ES is the most efficient algorithm among others both in time and value: getting a high KPI and significantly reducing the time for optimization.

Analysis of the results of the BIPOP restart strategies showed that the limit of 1000 function evaluations is not enough to apply it on 40D and higher dimensions. While for Grouped approach the dimension is reduced to 10D and in such dimensions BIPOP shows a comparable result with other algorithms.

## 6 CONCLUSION

The purpose of this paper was to explore and analyze the performance of evolutionary algorithms for optimization of Radio Access Network in the high-dimension scenario. The results show that in most cases they obtained a good performance of signal KPIs. The CMA-ES algorithm is one of the powerful methods for optimization which allows to adapt it to many constraints of the problem to achieve a better performance. Furthermore, these results could be improved by using the Grouped approach for dividing the Network into clusters.

The investigation results were used in the production system for Radio Access Network System which significantly improved the efficiency of tuning Base Station parameters depend on load hours. The applied algorithm reduced the time by 60% and improved the performance by 2.5 times for 2000 dimension.

Radio Access network requires real-time controlling in a high-dimension environment. Deep Neuroevolution approach, which combines Evolutionary Algorithms and Neural Networks, is a potentially suitable concept for future research in this area as an alternative to the Reinforcement Learning approach. Such an idea could be a potential direction for further work. However, all of these concepts require an accurate simulation environment to train the model.

## REFERENCES


[1] Asma Atamna. 2015. Benchmarking IPOP-CMA-ES-TPA and IPOP-CMA-ES-MSR on the BBOB Noiseless Testbed. In *Proceedings of the Companion Publication of the 2015 Annual Conference on Genetic and Evolutionary Computation (GECCO Companion '15)*. Association for Computing Machinery, 1135–1142. https://doi.org/10.1145/2739482.2768467

[2] Ahmad Awada, Bernhard Wegmann, Ingo Viering, and Anja Klein. 2011. Optimizing the Radio Network Parameters of the Long Term Evolution System Using Taguchi's Method. *IEEE Transactions on Vehicular Technology* 60, 8 (2011), 3825–3839. https://doi.org/10.1109/TVT.2011.2163326

[3] V. Berthier. 2015. Experiments on the CEC 2015 expensive optimization testbed. *2015 IEEE Congress on Evolutionary Computation (CEC)* (2015), 1059–1066. https://doi.org/10.1109/CEC.2015.7257007

[4] Pedro M. de Sant Ana and Nikolaj Marchenko. 2020. Radio Access Scheduling using CMA-ES for Optimized QoS in Wireless Networks. In *2020 IEEE Globecom Workshops (GC Wkshps*. 1–6. https://doi.org/10.1109/GCWkshps50303.2020.9367458

[5] V. Gjokaj, J. Doroshewitz, J. Nanzer, and P. Chahal. 2017. A Design Study of 5G Antennas Optimized Using Genetic Algorithms. In *2017 IEEE 67th Electronic Components and Technology Conference (ECTC)*. 2086–2091. https://doi.org/10.1109/ECTC.2017.286

[6] Sotirios K. Goudos, Margot Deruyck, David Plets, Luc Martens, Kostas E. Psannis, Panagiotis Sarigiannidis, and Wout Joseph. 2019. A Novel Design Approach for 5G Massive MIMO and NB-IoT Green Networks Using a Hybrid Jaya-Differential Evolution Algorithm. *IEEE Access* 7 (2019), 105687–105700. https://doi.org/10.1109/ACCESS.2019.2932042

[7] S. K. Goudos, M. Deruyck, D. Plets, L. Martens, K. E. Psannis, P. Sarigiannidis, and W. Joseph. 2019. A Novel Design Approach for 5G Massive MIMO and NB-IoT Green Networks Using a Hybrid Jaya-Differential Evolution Algorithm. *IEEE Access* 7 (2019), 105687–105700. https://doi.org/10.1109/ACCESS.2019.2932042

[8] Nikolaus Hansen. 2009. Benchmarking a BI-Population CMA-ES on the BBOB-2009 Function Testbed. In *Proceedings of the 11th Annual Conference Companion on Genetic and Evolutionary Computation Conference: Late Breaking Papers (GECCO '09)*. Association for Computing Machinery, 2389–2396. https://doi.org/10.1145/1570256.1570333

[9] Joseph Isabona. 2014. Maximising Coverage and Capacity with QOS Guarantee in GSM Network by Means of Cell Cluster Optimization. *International Journal of Advanced Research in Physical Science, ISSN 2349-7874* 1 (10 2014), 44–55.

[10] J. Kennedy and R. C. Eberhart. 1995. Particle swarm optimization. *IEEE Int. Conf. Neural Networks* (1995), 1942–1948.

[11] Antonio Latorre, Santiago Muelas, and José Peña. 2013. Evaluating the Multiple Offspring Sampling Framework on Complex Continuous Optimization Functions. *Memetic Computing* (12 2013). https://doi.org/10.1007/s12293-013-0120-8

[12] Ilya Loshchilov. 2014. A Computationally Efficient Limited Memory CMA-ES for Large Scale Optimization. In *Proceedings of the 2014 Annual Conference on Genetic and Evolutionary Computation (GECCO '14)*. Association for Computing Machinery, 397–404. https://doi.org/10.1145/2576768.2598294

[13] Ilya Loshchilov and Tobias Glasmachers. 2016. Anytime Bi-Objective Optimization with a Hybrid Multi-Objective CMA-ES (HMO-CMA-ES). In *Proceedings of the 2016 on Genetic and Evolutionary Computation Conference Companion (GECCO '16 Companion)*. Association for Computing Machinery, 1169–1176. https://doi.org/10.1145/2908961.2931698

[14] Ilya Loshchilov, Marc Schoenauer, and Michele Sebag. 2012. Black-Box Optimization Benchmarking of NIPOP-ACMA-ES and NBIPOP-ACMA-ES on the BBOB-2012 Noiseless Testbed. In *Proceedings of the 14th Annual Conference Companion on Genetic and Evolutionary Computation (GECCO '12)*. Association for Computing Machinery, 269–276. https://doi.org/10.1145/2330784.2330823

[15] Ilya Loshchilov, Marc Schoenauer, and Michele Sebag. 2012. Self-Adaptive Surrogate-Assisted Covariance Matrix Adaptation Evolution Strategy. In *Proceedings of the 14th Annual Conference on Genetic and Evolutionary Computation (GECCO '12)*. Association for Computing Machinery, 321–328. https://doi.org/10.1145/2330163.2330210

[16] David Lynch, Takfarinas Saber, Stepan Kucera, Holger Claussen, and Michael O'Neill. 2019. Evolutionary Learning of Link Allocation Algorithms for 5G Heterogeneous Wireless Communications Networks. In *Proceedings of the Genetic and Evolutionary Computation Conference (GECCO '19)*. Association for Computing Machinery, 1258–1265. https://doi.org/10.1145/3321707.3321853

[17] NSNAM. 2019. ns-3 Model Library. Propagation. (2019). Retrieved February 2, 2021 from https://www.nsnam.org/docs/models/html/propagation.html

[18] B. Partov, D. J. Leith, and R. Razavi. 2015. Utility Fair Optimization of Antenna Tilt Angles in LTE Networks. *IEEE/ACM Transactions on Networking* 23, 1 (2015), 175–185. https://doi.org/10.1109/TNET.2013.2294965

[19] A. K. Qin, V. L. Huang, and P. N. Suganthan. 2009. Differential Evolution Algorithm With Strategy Adaptation for Global Numerical Optimization. *IEEE Transactions on Evolutionary Computation* 13, 2 (2009), 398–417. https://doi.org/10.1109/TEVC.2008.927706

[20] Nandini Raghavan, Réka Albert, and Soundar Kumara. 2007. Near Linear Time Algorithm to Detect Community Structures in Large-Scale Networks. *Physical review. E, Statistical, nonlinear, and soft matter physics* 76 (10 2007), 036106. https://doi.org/10.1103/PhysRevE.76.036106

[21] Raymond Ros and Nikolaus Hansen. 2008. A Simple Modification in CMA-ES Achieving Linear Time and Space Complexity. In *Proceedings of the 10th International Conference on Parallel Problem Solving from Nature — PPSN X - Volume 5199*. Springer-Verlag, 296–305.

[22] Siddhartha Shakya, Beum Seuk Lee, and Gilbert Owusu. 2019. Optimizing Field Productivity by Mobile Warehouse Deployment Using Evolutionary Algorithms. In *2019 IEEE Symposium Series on Computational Intelligence (SSCI)*. 1652–1659. https://doi.org/10.1109/SSCI44817.2019.9003172

[23] Iana Siomina, Peter Varbrand, and Di Yuan. 2006. Automated optimization of service coverage and base station antenna configuration in UMTS networks. *IEEE Wireless Communications* 13, 6 (2006), 16–25. https://doi.org/10.1109/MWC.2006.275194

[24] N. Trabelsi, C. S. Chen, R. El Azouzi, L. Roullet, and E. Altman. 2017. User Association and Resource Allocation Optimization in LTE Cellular Networks. *IEEE Transactions on Network and Service Management* 14, 2 (2017), 429–440. https://doi.org/10.1109/TNSM.2017.2677778

[25] P. Korošec U. Škvorc, T. Eftimov. 2019. GECCO Black-Box Optimization Competitions: Progress from 2009 to 2018. https://doi.org/10.1145/3319619.3321996

[26] Stijn van Dongen. 2000. *Graph Clustering by Flow Simulation*. Ph.D. Dissertation. University of Utrecht, Utrecht, Netherlands.

[27] U. von Luxburg. 2007. A tutorial on spectral clustering. *Statistics and Computing* 17 (2007), 395–416. https://doi.org/10.1007/s11222-007-9033-z

[28] J. Zhang and A. C. Sanderson. 2009. JADE: Adaptive Differential Evolution With Optional External Archive. *IEEE Transactions on Evolutionary Computation* 13, 5 (2009), 945–958. https://doi.org/10.1109/TEVC.2009.2014613