Ant Lion Optimization based inertia weight Optimized KGMO for Mobility Management in Heterogeneous LTE Cellular Networks

Venkata Narasimha Reddy G.K (gkvnreddy@gmail.com)
Venkata Naga Jayudu T
Janardhan Komarolu
Rajesh Nichenametla
Narayana Reddy

Research Article

Keywords: Ant Lion Optimization, Kinetic Gas Molecular Optimization, Inertia Weight, Mobility Management, Ultra-dense heterogeneous network

Posted Date: April 8th, 2022

DOI: https://doi.org/10.21203/rs.3.rs-1528610/v1

Abstract

A Network densification is understood as a critical process in the development of cellular networks because to the exponentially rising demands of mobile data traffic. One possible technology to fulfil the requirements of exploding data traffic in 5G is the Ultra-dense heterogeneous network (UDHN). 5G networks of the future will feature dense networks that can deliver high data rates in crowded urban areas. As a result, in heterogeneous networks, the amount of interference and the number of handovers (HOs) increases as more small cells are deployed. In this case, mobility management is critical to ensure that the user's communication is uninterrupted while they travel between cells. Kinetic Gas Molecular Optimization (KGMO) and Ant Lion Optimizatio (ALO) are presented in this study as a hybrid optimization-based mobility management strategy for enhancing network performance Initially, the particles of the KGMO such as position and velocity of the gas molecule are calculated based on the kinetic energy. However, KGMO offers faster convergence in large dimensional space and this leads to the hybridization of KGMO. To overcome this issues of KGMO, ALO is used by modifying the inertia weight of KGMO. The simulation setup of proposed KGMO-ALO is done in MATLAB environment and validated with other popular meta-heuristic techniques such as Seagull and monarch butterfly optimizations in terms of throughput, Bit error rate, end-to-end delay and HO.

Introduction

In current years, heterogeneous networks (HetNets) have received a lot of interest. There are many different types of cells in these networks to fulfil the needs of different users. Increasing capacity and coverage are two of HetNet's primary goals as specified by the Third Generation Partnership Project (3GPP) [1]. For better user experience and augmented coverage in cellular networks, ultra-dense small cells with multiple small cells overlapping with macro cells are being implemented [2]. To address the ever-increasing demand for digital facilities in the everyday lives of citizens, wireless connection is essential. The ultra-dense network (UDN) architecture requires cells of various sizes and features to cohabit and collaborate in order to achieve the objective of ubiquitous connectivity and to deliver services with varying restrictions [4]. UDN is the greatest approaches to meet user expectations and promote the development of wireless services in the future. [5] When customers switch from one base station (BS) to another in a 5G that intersections with the 4G network, problems arise [6, 7]. It is necessary for users to conduct the handover (HO) process as they move from one BS to another while managing mobility in HetNets because of the variety of inter-frequency technologies involved. HO Ping-Pong (HOPP) and HO failure concerns need to be addressed and overcome before mobility robustness optimization (MRO) can recover user in 4G networks.

According on user feedback, it is possible to perform an improvement by adjusting the HO control parameters (HCPs). To evade service deterioration owing to high taxes of HO, HOPP, and radio connection failure, mobility management must be adequately managed (RLF). To guarantee that the 5G of cellular networks can deliver smooth communiqué during user migration between different placement scenarios, this issue must be handled. A number of services, including MRO and load balancing optimization, are

introduced in self-optimization networks. Optimizing the HCP values during the user's mobility is the goal of both routines, and both strive to tackle a variety of HO issues. Automated HCP optimization, for example, is performed by the MRO function, which automatically modifies HCP parameters to maintain system quality and does so with minimal human intervention. The HO margin (HOM) and the while to trigger are the two primary parameters of the HCP (TTT). The service quality is improved by modifying these strictures to the right levels through user movement in cell attention, resulting in lower rates of HOPP and HOF.

5G and beyond will have to achieve a wide range of objectives, including increased throughputs, reduced latency and generalised attention with a steady user knowledge for any conceivable of devices speed, unparalleled spectrum and energy efficiency levels, among others. Developing a generic optimization rule is impossible because of the interdependence of these multiple objectives [11]. For challenges like mobility management (MM), which is based on obstacles or occurrences, Evolutionary Algorithms are commonly utilised [12]. The MM model's outcome is mostly determined by how well the parameters are tuned. In order to better address the problems, the MMP model goes through a number of steps. Furthermore, parameter adjustment necessitates a deeper understanding of the methods used to solve the problems [13]. Consequently, any differences in the properties of the problems may have an impact on the algorithm's performance.

For some time now, scientists have been trying to come up with an optimization method that can tackle the challenges listed above with high accuracy and little convergence time toward an ideal solution. A population-based metaheuristic algorithm such as PSO, KGMO, or GWO has two basic phases: exploration (diversification) and exploitation (intensification) [14]. [15, 16]. The algorithm's goal in exploration is to locate intriguing locations that may contain the global optima by scouring the whole search space. But in exploitation, the algorithm seeks to find solutions in the vicinity of those found during exploration. It's also critical to focus on exploration rather than exploitation in the early phases of optimization because doing so increases your chances of finding better solutions that are similar to those found earlier. The algorithm's performance is greatly impacted by the algorithm's ability to strike a balance between examination and exploitation.

The gas molecules in KGMO [15–16] search the entire search space to find the lowest temperature. Each particle's velocity and position are generated using random vectors that fall within the respective ranges. For example, consider w as weight, which mimics gas molecules' inertia to impede their movement. Inertia weight The KGMO's convergence behaviour is determined by its inertia weight. Large inertia weight causes a slow convergence of the optimization, while low inertia weight results in trapping in the immediate area. This means that the inertia weight should be chosen for a better search-use tradeoff. This flaw must be fixed, in particular by boosting the rate of convergence of KGMO. To achieve this, new inertia weight change, which is briefly detailed in the suggested section, can be implemented.

The residual part of the research work is considered as shadows: Section 2 delivers the related works of mobility management issues with inertia weight of optimization techniques. The brief explanation of

proposed methodology (KGMO-ALO) is given in Section 3. The validation of proposed procedure with existing practises in terms of throughput, delay, BER and HO is presented in the Section 4. Finally, the scientific research contribution of proposed method with future work is given in Section 5.

Literature Review

HetNets' HO problem has been the subject of numerous studies, all of which offer different techniques for resolving it. Based on user equipment (UE) speed and HO type, Ni et al. [17] have optimised HCPs utilising mobility state estimate (MSE). TTT was only changed in this study based on user velocity and only a small number of TTT values were modified. Meanwhile the hole among these updated values is relatively large and only three secure standards are picked in agreement with the UE velocity, this strategy did not fully optimise the HO performance. To reduce unwanted HOs (UHOs) and service failures in HetNets, Tiwari & Deshmukh [18] have proposed a HO choice approach and MSE scheme. Sojourn time and the number of homes visited are used to estimate UE velocity. The suggested MSE model minimises the amount of UHOs and service disappointments, according to simulation findings. However, the performance of additional HO indicators, such as RLF, HOPP and latency, have not been explored.

Additionally, [19] have devised an procedure that adapts to the various HOM and load balancing requirements of each UE in a HetNet. SINR, rather than conventional signal strength, is used as an indicator for determining the HO level in the proposed approach, rather than relying solely on the HOM's received signal strength. In an effort to improve carrier aggregation by HO optimization, a team led by Shayea et al. [20] developed a weighted function. Automatic adjustment of the HOM values according to SINR, traffic load and velocity is part of the algorithm provided here. By enhancing both spectral efficiency and outage probability near cells, the suggested approach has been shown to improve system performance. However, cellular network packet delivery delays were not taken into account by this strategy.

Using fuzzy logic, Saeed et al. [21] have created a methodology to optimise HOM for HetNets. Call drop and load balancing index are the inputs to the fuzzy logic, which adjusts the HOM for large and small cells. A HO in the network is detected using the reinforcement learning principle in another study [22]. Session HOs that are well-executed lead to low drop-call rates and lower HOF and HOPP rates. Only a maximum speed of 120 km/h is supported by this method. To address the concern of mobility management in various mobile speed settings, multiple algorithms have been presented [23–25]. Too early HO, too late HO, and HO to the incorrect cells are all sorts of HO that can be used to adjust HCPs. According to the data, the customised HCPs lessen the frequencies of HOP, HOPP and RLF. Mobility robustness in uneven topologies could not be handled by this strategy.

In order to deal with the problem of mobility management, Dahi et al. [26] developed a low- complexity adaptive cellular genetic (CGA). The bi-dimensional neighborhood's interactions were regulated by a torus-like structured population. Using two light operators, the algorithm's population was periodically regenerated when its settings were adjusted. With fewer algorithmic parameters, the procedure was more

effective and easy. There were several flaws in the strategy because it did not take into account the extent of resources and how to handle interference in the network. Swayamsiddha et al. [28] employed the normalised percentile dwell time distribution integrated with bio-inspired optimization practices as GA and Binary particle swarm optimization (BPSO) [27]. As a result of applying optimization approaches to deal with mobility management and HO concerns, the PSO hurts from early convergence, which means that when penetrating for the global solution, and it can be easily caught in a specific solution, thus producing non-optimal results. As a result, numerous studies have attempted to tailor optimization strategies to address the aforementioned problems.

Basic Teaching learning-based (TLB), which does not take inertia into account, was first presented by Rao et al. The I-TLBO method, designed by the authors to improve exploration and exploitation, incorporates aspects including the number of teachers, the adaptive teaching and energetic learning. [30]. It was put through its paces using a variety of unrestricted benchmark routines. It is just like other population-based methods for addressing non-linear complicated glitches with several locally optimal solutions that TLBO gets stuck in a local optimum as well. On the Yang et al. [31] proposed a unique and computationally intelligent (CI) method named CI-TLBO to solve a complex numerical issue.

Raja et al. [32] studied the self-learning idea into the TLBO, which increased the solution excellence as well as accuracy, to avoid premature convergence. According to Pareto front analysis, a 99.40 percent lessening in total pressure has been observed at the cost of a 97.90 percent reduction in regenerator efficacy. A similar idea was put forth by Shukla and colleagues [33] in the TLBO algorithm, which is based on neighbor-based TLBO and differential mutations.

Higher-dimensional issues can't be solved using typical BA methods [34]. Due to a lack of searching capability, the classic BA has issues with local optimums and premature convergence. Based on the covariance adaptive process, a new BAT variant that is more capable of exploring is proposed [35]. Information in the proposed manner increases the search procedure by varying the search direction and population sampling distribution. An algorithm called IBA was created in 2013 to address the shortcomings of traditional BA. [36] An algorithm with adaptive step switch and mutation devices was designed by the authors to address the issue of local optimums [37]. They examined SABA's parameters to make sure they were consistent. Gaussian Distribution Random Walk (BAGD) was created in 2016 by Nazri et al. [38] to reduce the step size.

After each time step, the velocity of each particle in PSO changes in the direction of its gbest and pbest locations. Inertia weight can be used as a solution to their dilemma. Primary characteristic of PSO is apathy weight, which plays a significant role in balancing exploration and exploitation. As a result of the apathy weight, numerous PSO algorithms have been proposed to achieve the right balance between exploring and exploiting. Single- and double-diode solar cell limits have been extracted using a chaotic inertia weight PSO (CIWPSO) in [39]. To avoid premature convergence, the inertia weight follows a chaotic map logic in this variant, which results in an ideal solution. PV cell and module parameters can be estimated using a GCPSO (Guaranteed Convergence PSO) variation in [40]. Success and failure rates

were proposed in this form, together with a scale factor, which ensures the algorithm's convergence toward an optimal solution. PSO inertia weight has been applied in several fields [39, 40], but no researchers have concentrated on altering the inertia weight of PSO or other optimization strategies for mobility management and HO concerns [39, 40]. Researchers didn't focus on the KGMO optimization technique in mobility management, although it is based on several other areas [41–43]. This research study's primary goal is to alter the KGMO's inertia weight to address mobility management and HO concerns.

Proposed Methodology

3.1. System Model

There are three small cells in each macro BS of the HetNet architecture, which is composed of many four-gigabit-per-second (Mbps) 5G small-cell base stations (BCSs). All three sectors in the macro-BS are independent cells, but the smaller cells are all omnidirectional single-sector cells. Example of HetNet placement with three small cells on a single macro cell is shown in Fig. 1. The macro and small cell radii are R and r, respectively.

They work in frequency bands, with receptiveness of one, with the reuse frequency factor equal to one. Nk = 1,..., Nm, and Nl = 1,..., Nn define the sets of macro and tiny cells, respectively. Using a chance mobility model, the network's user set is designated as Nu with U = 1,..., U, where U is arbitrarily dispersed throughout the network. Either macro or small cells deliver requested traffic to the UEs. A dispersed self-organizing network accumulates HO information and optimises HCPs in each small and macro cell. Any time an end-user device (UE) switches from one network node to another, whether inside the same one or in another, the HO operation is initiated. The serving cell decides whether or not to begin the HO procedure to a board cell depending on the UE's measurement report (MR). Eq. [1] expresses the route loss model for various urban bands between a BS and the user as follows:

$$PL_{u,k,l} = 20log_{10}\left(\frac{4\pi r_0}{\lambda_l}\right) + 20log_{10}\left(\frac{d_{u,k}}{d_0}\right) + X$$

1

where:

$$BS = \begin{cases} smallcellifl = 1 \\ macrocellotherwise \end{cases}$$

2

There is a reference distance d 0 and a distance d (u,k,l) between the user U and the base station k, which is expected to be 50 m (d (u,k,l) - d 0). The carrier frequency's wavelength is ll (c,l). c has a variance of 2 and a standard deviation of zero.

The RLF is reduced to a minimum by the highest possible quality of service (QoS) requirements. For QoS to be satisfied, the performance of both UE u must fulfil the smallest data rate criterion. Figure 3 shows how SINR can be modelled for the UE as a function of its location in the channel:

$$SINR_{u,k,l} = \frac{\rho_{u,k,l} g_{u,k,l} b_{i,j}}{\sum_{i \in K\{k\}} \sum_{j \in u\{u\}} p_{i,j} g_{u,k,l} + P_{AWGN}'}$$

3

the received signal power at $p(u, k, l)$ and its channel gain are shown below. B ij = 1 indicates that user u is related with one BS, and bij is the binary association pointer for that user. Other than that, b ij is zero. As a result of the UE's interference, pij is a measure of the received signal power. P AWGN is the power of white Gaussian noise that is added to a signal.

3.2. Functioning

The details about the different heterogeneous network information are gathered for pre-processing from the core network. The collected data is considered for pre-processing such as normalization. The normalized data is sent to KGMO to estimate the initial Kinetic Energy (KE) and initial velocity of users. Based on the initial velocity of the users a random particle is generated to calculate the fitness function for the evaluation of the technique. The entire process to yield appropriate fitness function is depicted in Fig. 2.

Once fitness function is calculated, the best fitness is selected to update the KE and velocity, and consequently generate a new random particle. This process is repeated till the best fitness is obtained. This is performed by KGMO algorithm. Once the best fitness is obtained, the resultant data is given to the data collection unit to function the network more accurately as shown in Fig. 3. The network performance is calculated in terms of throughput, BER and latency based on the location of user and mobility management.

The chief goal of this work is to solve the fast convergence issues of KGMO, this paper proposes the ALO technique for modifying the inertia weight. Initially, the equatic explanation for KGMO is given as follows:

3.3. KGMO – The proposed algorithm

w A new algorithm, called KGMO, uses kinetic energy as a measure of performance because gas molecules are the agents in a search area. Until the container's lowest temperature and energy are reached, the gas molecules will continue to travel in the container. Using Van der Waal forces, gas molecules have been shown to attract one another. The molecules' positive and negative charges cause

the electrical pressure. Each gas molecule in the KGMO has four properties: velocity, and mass. Each gas molecule's kinetic energy influences its speed and location. A gas molecule's goal is to reach the coldest possible position in the algorithm by travelling to every corner of the search space.

After that, imagine a network with N agents (gas molecules). The ith agent's role is outlined in these terms:

$$X_i = \left(X_i^1, . . , X_i^d, ...X_i^n\right), for(i = 1,2, ..., N)$$

4

where X_i^d Signifies the location of the ith agent in the dth dimension.

The velocity of the ith agent is obtainable by

$$V_i = (v_i^1, ...v_i^d, ...v_i^n), for(i = 1,2, ..., N)$$

5

in which v id is the ith agent's speed in the dth dimension.

An exponential relationship exists between the kinetic energy of gas molecules and their velocity, which can be calculated using Boltzmann distributions. Movement in the environment is referred to as kinetic energy.

$$k_i^d(t) = \frac{3}{2}NbT_i^d(t), K_i = \left(k_i^1, . . k_i^d, ...k_i^n\right), for(i = 1,2. . N)$$

6

When the Boltzmann constant is b and the temperature of the ith proxy in the dth dimension at time t is T id.

The molecule's speed is changed every time the
$v_i^d(t+1) = T_i^d(t)wv_i^d(t) + C_1 rand_i(t)(gbest^d - X_i^t(t) + C_2 rand_i(t)(pbest_i^d(t) - X_i^d(t)))$ (7)

As time goes on, T I d decreases exponentially for the converging molecules and is calculated as

$$T_i^d(t) = 0.95 \times T_i^d(t-1)$$

8

Ith gas molecule's best prior location is represented by the vector (pbest)=(pbest)(i), and gbest = (g)(n) is the finest preceding position of all gas molecules in the container, as shown by the vector (g)best =

(gbest)(i), and pbest = I as shown by the vector I The starting velocity and position of each particle are determined by random vectors. In this example, [v min;v max] is used as the upper and lower limits of the velocity of the gas molecules. In other words, if the inertia weight of the gas molecule is more than w, then |v i |=vmax. rand i (t) is a random variable with a uniform distribution at time t in the interval [0,1], which gives the search method a random quality. [0,1]. Acceleration quantities C 1 and C 2

As long as the container contains only one type of gas, the gas molecules' mass m is randomly generated from a range of 0 m 6 1; once detected, this value remains constant throughout the algorithm's execution. Using a random integer, the technique is able to mimic a variety of gases.

The molecule's position can be determined from the equations of motion used in physics.

$$X_{t+1}^{i} = \frac{1}{2} a_i^d(t+1)\, t^2 + v_i^d(t+1)\, t + X_i^d(t)$$

9

where a id denotes the ith agent's acceleration in the dth-dimensional space.

Using the acceleration formula, we can derive

$$a_i^d = \frac{\left(dv_i^d\right)}{dt}$$

10

According to Eq. (9) of the gas molecule rules, we can also conclude that

$$dk_d^i = \frac{1}{2} m \left(dv_i^d\right)^2 \Rightarrow dv_i^d = \sqrt{\frac{2\left(dk_i^d\right)}{m}}$$

11

So, from Eqs. (10) and (11), the acceleration is distinct as

$$a_d^i = \frac{\sqrt{\frac{2\left(dk_i^d\right)}{m}}}{dt}$$

12

In the time intermission Dt, Eq. (12) can be as

$$a_d^i = \frac{\sqrt{\frac{2\left(\Delta k_i^d\right)}{m}}}{\Delta t}$$

13

As a result, the rate of acceleration is

$$a_d^i = \sqrt{\frac{2\left(dk_i^d\right)}{m}}$$

14

Then, from Eqs. (9) and (14), the place of the molecule is intended by

$$X_{t+1}^i = \frac{1}{2} a_i^d(t+1)\Delta t^2 + v_i^d(t+1)\Delta t + X_i^d(t) \Longrightarrow$$

$$X_{t+1}^i = \frac{1}{2}\sqrt{\frac{2\left(\Delta k_i^d\right)}{m}}(t+1)\Delta t^2 + v_i^d(t+1)\Delta t + X_i^d(t)$$

15

In order to keep things simple, the molecule is randomly generated in each run of the algorithm but is the similar for all the molecules in performance.

$$X_{t+1}^i = \sqrt{\frac{2\left(\Delta k_i^d\right)}{m}}(t+1) + v_i^d(t+1) + X_i^d(t)$$

16

The minimum fitness is originate by using

$$pbset_i = f\left(X_i\right), iff\left(X_i\right) < f\left(pbset_i\right)$$

$$gbest = f\left(X_i\right), iff\left(X_i\right) < f(gbest)$$

17

Eq. (8) for inertia weight change is obtained by using ALO algorithm's best fitness function. The ALO mathematical equations are summarised as such::

3.3.1. Operators of the ALO procedure

The ALO algorithm simulates the behaviour of antlions and ants in a trap. An ant colony and antlions can be simulated by allowing them to hunt each other in the search area and gain strength by utilising traps. Because ants' foraging behaviour is stochastically distributed in nature, the following random walk is used to mimic their movement:

$$X(t) = [0, cumsum(2r\left(t_1 - 1\right)), cumsum(2r\left(t_2 - 1\right)), \ldots cumsum(2r(t_n - 1))]$$

18

which adds the increasing sum, n is the amount of iterations, t represents the random walk step, and r(t) is the stochastic function distinct as follows::

Here, rand is a random number made in the range of [0,1] for each iteration of the random walk.

$$r(t) = \begin{cases} 1 if rand > 0.5 \\ 0 if rand \leq 0.5 \end{cases}$$

19

In the following matrix, the position of ants is kept and used during optimization:

$$M_{Ant} = \begin{bmatrix} A_{1,1} & A_{1,2} & \cdots & A_{1,d} \\ A_{21} & A_{2,2} & \cdots & A_{2,d} \\ \vdots & \vdots & \cdots & \vdots \\ A_{n1} & A_{n,2} & \cdots & A_{n,d} \end{bmatrix}$$

20

in which each ant's position is stored in M Ant as an array, Ai,j displays the value of the i-th variable (dimension) in M Ant, and the total number of variables in M Ant is equal to n x d.

A comparison between ants and particles or persons in GA is warranted. The ant's position is a good indicator of a solution's parameters. Matrix All ants' positions (variables from all solutions) will be saved by M Ant during optimization.

Optimization is used to evaluate each ant's fitness (objective function), which is stored in the subsequent matrix:

$$M_{oA} = \begin{bmatrix} f([A_{1,1} & A_{1,2} & \cdots & A_{1,d}]) \\ f([A_{21} & A_{2,2} & \cdots & A_{2,d})] \\ \vdots & \vdots & \cdots & \vdots \\ f([A_{n1} & A_{n,2} & \cdots & A_{n,d})] \end{bmatrix}$$

21

where n is the amount of ants, A (i,j) shows the value of the jth dimension of the ith ant, and f is the objective function, M oA is the matrix for preserving each ant's fitness.

We assume that the antlions are also lurking someplace in the search area, as well. The following matrices are used to keep track of their positions and fitness values:

$$M_{Antlion} = \begin{bmatrix} AL_{1,1} & AL_{1,2} & \cdots & AL_{1,d} \\ AL_{21} & AL_{2,2} & \cdots & AL_{2,d} \\ \vdots & \vdots & \cdots & \vdots \\ AL_{n1} & AL_{n,2} & \cdots & AL_{n,d} \end{bmatrix}$$

22

A matrix called M Antlion keeps track of the positions of all antlions, and a value called AL (i,j) displays the value of the number of dimensions in which an antlion appears in the i-th dimension (dimension).

$$M_{oAL} = \begin{bmatrix} f([AL_{1,1} & AL_{1,2} & \cdots & AL_{1,d}]) \\ f([AL_{21} & AL_{2,2} & \cdots & AL_{2,d})] \\ \vdots & \vdots & \cdots & \vdots \\ f([AL_{n1} & AL_{n,2} & \cdots & AL_{n,d})] \end{bmatrix}$$

23

For each antlion, there is a matrix called M oAL that is used to store the antlion's fitness, and AL (i,j) (n,f) is the goal function.

For optimization, certain rules are followed:

- Ants use a variety of random walks to explore the search area.

- Insects travel in random directions in all of their dimensions.
- The antlions' traps have an effect on random walks.
- As their fitness increases, antlions can build larger and larger trenches (the higher fitness, the larger pit).
- As a general rule, antlions with larger pits are more likely to trap ants.
- In each repetition, an antlion can capture every ant, and the elite (fittest antlion).
- As the ants slide towards the antlions, the random walk's range of motion is reduced.
- This means that when it gets stronger than the antlion, the antlion will grab the ant and drag it under the sand.
- In order to increase its chances of catching another victim after every hunt, an antlion moves closer to the prey it has just caught and constructs a pit.

3.3.1.1. Random walks of ants

The Eq. is the basis for all random walks (18). Random walk is used at every phase of optimization by ants. It is not possible to directly use Eq. (18) to update the position of ants since every search space has a border. The random walks are adjusted using the following equation to keep them within the search space.

$$X_i^t = \frac{(X_i^t - a_i) \times (d_i - C_i^t)}{(d_i^t - a_i)} + C_i$$

24

where ai represents the minimum possible random walk in the i-th variable, bi represents the largest possible random walk in the i-th variable, ct I represents the smallest possible random walk in the i-th variable at iteration t, and dt I represents the largest possible random walk in the i-th variable at iteration t.

To ensure that random walks occur within the search space, Eq. (24) should be used in each iteration.

3.3.1.2. Trapping in antlion’s pits

antlion traps have an effect on ants' random movements, as was previously addressed in detail. The following equations are presented to mathematically describe this assumption:

$$C_i^t = Antlion_j^t + C^t$$

25

$$d_i^t = Antlion_j^t + d^t$$

26

Where Ct is the minimum of altogether variables at a given iteration, and C I t is the smallest of altogether variables for an individual ant, and d i is the supreme of altogether variables for an individual ant, and Antlionj t is the position of the picked j-th antlion at that iteration.

3.3.1.3. Building trap

A roulette wheel is used to simulate the antlions' hunting abilities. During optimization, the ALO procedure must use a roulette wheel operator to select antlions based on fitness. Fitter antlions have a better chance of taking down ants thanks to this process.

3.3.1.4. Sliding ants towards antlion

Since antlions can build traps that are proportionate to their fitness, the techniques presented so far need ants to travel at random. Antlions, on the other hand, shoot sands out of the pit's centre once they detect an ant. The ant that is annoying to escape the trap is being dragged down by this behaviour. Ants' hyper-sphere radius is compact adaptively to model this behaviour in mathematics. In light of this, the following equations have been proposed:

$$C^t = \frac{C^t}{I}$$

27

$$d^t = \frac{d^t}{I}$$

28

dt is the vector containing all variables at t-th iteration, where I is the ratio, C is the minimum, and t is the iteration numeral.

I = 14 10w t/T in Equations (27) and (28) When $t > 0.1T,\ w = 2;$ when it is > 0.5T, $w = 3$; when it is > 0.75T, it is w = 4; and w = 6 when it is > 0.95T, t is the current iteration. T is the maximum amount of iterations. The constant w can be used to fine-tune the level of exploitation accuracy. Using these equations, ants' positions are updated with a smaller radius and approximate the sliding motion of an ant within the hole. So that the available research area can be used to its full potential.

3.3.1.5. Catching prey and re-building the pit

Hunting concludes when an ant falls to its doom and is swallowed whole by the antlion. After this, the antlion eats the ant's body by snatching it from the ground and burying it. Predation occurs when an ant gets more physically fit than its counterpart, such as by entering the sand, to emulate this process. To upsurge its odds of infectious new prey, an antlion must then update its location to reflect the most recent location of the chased ant. In this regard, the following equation has been proposed:

$$Antlion_j^t = Ant_i^t iff\left(Ant_i^t\right) > f\left(Antlion_j^t\right)$$

29

Here, Antliont j specifies which one of the three iterations is currently running, Antt I tells which one of those iterations is currently running, and the current iteration is t

.

3.3.1.6. Elitism

As a result of elitism, evolutionary algorithms can preserve the best solution(s) at any point in the optimization procedure. The most successful antlion from each iteration is kept and treated as an elite antlion in this investigation. Due to its superior physical condition, it should be in a position to influence all ants over subsequent iterations. A random tour around one of the roulette wheel-selected antlions and the elite is expected to occur simultaneously:

$$Ant_i^t = \frac{R_A^t + R_E^t}{2}$$

30

where R At is the roulette wheel's random walk about the antlion at tth iteration, R Et is the elite's random walk at tth iteration, and Antt I is the i-th ant's location at tth iteration's random walk around the elite.

3.3.1.7. ALO algorithm

The global optimum for optimization difficulties, the following three-tuple function is used: The ALO algorithm:

$$ALO(A, B, C)$$

31

Assuming that C is satisfied, then A returns true; otherwise, it returns false; and if B modifies A's initial population in any way, then C returns true. Following are the meanings of the functions A, B, and C:

$$\emptyset \underset{\rightarrow}{A} \{M_{Ant}, M_{OA}, M_{Antlion}, M_{OAL}\}$$

32

$$\left\{M_{Ant}, M_{Antlion}\right\} \underset{\rightarrow}{B} \{M_{Ant}, M_{Antlion}\}$$

33

$$\left\{{M}_{Ant},{M}_{Antlion}\right\}\underrightarrow{C}\{true, false\} \quad (34)$$

where M Ant is the ant position matrix, M Antlion contains the antlion position matrix, M OA contains the ant fitness matrix, and M OAL contains the antlion fitness matrix. M OAL values are the input value for Eq. (8) in order to tackle the problem of rapid KGMO convergence. After that, we'll compare the proposed method's results to those of the currently used procedures.

Results And Discussion

KGMO-ALO is tested and implemented on a Windows 10 laptop with 8 GB of RAM, an Intel Core I7 processor, and an NVIDIA GeForce card. As the principal implementation tool, Matlab (version 2019a) is employed. The number of base stations and users are changed and performance evaluation is carried out. The network architecture is developed based on the LTE network standard to meet the real-time communication network criteria. The network throughput, delay, BER, Number of Hand overs are the performance parameters for optimization of the algorithm and have been evaluated to analyze the overall performance of the network. The proposed system KGMO inertia weight has been optimized with the various optimization techniques such as KGMO-PSO, KGMO-MBO, KGMO-SOA and Proposed KGMO-ALO. And also the proposed KGMO is compared PSO optimization. The execution results are defined in the below sections.

4.1. Network setting

Network with heterogeneous model is represented in the Fig. 4. The position of every user is identified from the x and y coordinates of the graph. Every network consists of the CH and each of the CH is connected to the base station. The users in the network are marked as small circles and these are connected to the cluster head, where each clusters have approximately 10 users that is shown in Table 1. It shows the specification of network which is used to simulate the proposed technique. Number of users is represented in N and cluster head as CH, Bandwidth as BW and Signal to Noise Ratio (SNR) from − 10 to 10dB.

Table 1
Specification of network

1	**No of users (N)**	**50–100**
2	No of CH (NC)	5–10
3	Coverage Area	50mx50m
4	Bandwidth (BW)	20MHz
5	Cost Function	Minimum Distance
6	Optimization	KGMO
7	Inertia weight	PSO, Monarch Butterfly Optimization (MBO), Seagull Optimization Algorithm (SOA) and ALO
8	SNR Range	-10 to 10dB

4.2. Through put analysis

In this proposed system two different kind of throughput analysis are evaluated such as normal throughput and Cumulative throughput. The Normal throughput calculation is done using the Eq. (35). Cumulative throughput is the cumulative addition of throughput value with respect to time for all stake holders of the network by taking the mean value.

TH=\frac{MSS}{RTT}* \frac{1}{sqrt\left(p\right)}
35

MSS – Maximum Segment Size, RTT – Round Trip Time, p – Packet Loss.

From the above figure, it is clearly proves that the proposed KGMO-ALO has higher throughput than existing techniques such as PSO and KGMO. In this research work, the inertia weight of KGMO is modified by using PSO, MBO and SO, however, they shows less performance than KGMO-ALO algorithm. For better understanding, Table 2 shows the mean throughput for all techniques.

Table.2. Mean throughtput.

Technique	Mean throughput
PSO	1324782.105
KGMO	1350648.5169
KGMO-PSO	1367404.45
KGMO-MBO	1330541.5408
KGMO-SOA	1287274.109
KGMO-ALO	1444873.9291

The above table shows that KGMO-SOA provides poor performance in mean throughput, where PSO provides only 1324782.105 mean throughput. However, the proposed method KGMO-ALO achieved high mean throughput (i.e. 1444873.9291) than all other techniques. The following Fig. 6 shows the comparison for cumulative throughput.

From the above figure, it is proved that KGMO-PSO provides less performance than all other techniques such as KGMO-SOA, KGMO-MBO and KGMO-ALO. When the time increases, the KGMO-SOA provides less presentation in terms of cumulative throughput, the reason is that according to the fittest seagull, other seagulls can update their initial positions. Initially, the KGMO-ALO achieved less performance than KGMO, however, when the time increases, the performance of proposed KGMO-ALO is also increased, due to the modification of inertia weight of KGMO. The following Fig. 7 shows the graphical representation of proposed process in terms of throughput for number of clusters.

The throughput of KGMO-ALO achieved high performance, only when the number of cluster reaches the value 7. While comparing with all techniques, KGMO-PSO provides better performance than other existing such as SOA and MBO. Without modifying the inertia weight, KGMO and PSO provides very poor performance, the reason is that they attained the fast convergence at early stages. Here, when the amount of clusters upsurges, the presentation of KGMO-ALO starts to decreases, which will represents in the above figure and the KGMO-PSO achieved better performance than proposed KGMO-ALO at the number of cluster is 10. This needs to be further enhanced by modifying the proposed method in future work. Figure 8 shows the comparative examination of proposed KGMO-ALO in terms of throughput with number of users.

When the amount of users upsurges, the performance of KGMO-ALO is also increased, however, when the analysis reaches the number of users at 100. The existing techniques such as PSO, MBO and SOA provides higher performance only when the number of user is high. The reason for poor performance of ALO is that the performance of ALO in each round of process is matched with the finest antlion which guaranteed an outstanding solution during during the optimization processes. Figure 9 shows the end-to-end delay of proposed KGMO-ALO with existing techniques.

When the delay is less, the performance of the techniques is good at the number of users. This proves that the PSO has higher delay than all other techniques. But, when PSO is implemented with KGMO, it has less delay than KGMO and the delay is varied with number of users increases. The existing techniques such as KGMO-PSO, KGMO-SOA and KGMO-MBO has delay variations as the number of users increases. But, the proposed method KGMO-ALO is less than all techniques, whenever the number of users increases from 50 to 100. Figure 10 provides the graphical representation of KGMO-ALO with other techniques in terms of BER.

As like delay, the proposed KGMO-ALO achieved better performance than all existing techniques. However, the BER of PSO, KGMO-PSO and KGMO-SOA is high when compared with KGMO and KGMO-MBO. The reason for poor performance of PSO. But, the proposed KGMO solves the issues of local optimum and low convergence rate by using ALO and achieves better performance than all other techniques. Figure 11 shows the graphical representation of proposed KGMO-ALO in terms of HO.

The performance of proposed scheme is high than all existing techniques, when the number of user reaches the value 80. Number of handovers is one of the major factors which effects the performance of the network. For good performance, the number of handovers should be constant. From the Fig. 11 it is observed that the amount of handovers is relatively constant in the KGMO-ALO algorithm as compated to other techniques.

Conclusion

Wireless access skills are improving to deliver mobile users with extraordinary data rates and allow new requests that machine-type communications, as the number of mobile users continues to rise rapidly. Due to the cumulative complexity of network topology in 5G HetNets with the integration of many various base station types, mobility management in 5G construction faces several obstacles such as HO failures and delays and mobility management concerns. The KGMO algorithm is proposed as a solution to mobility management difficulties in this study. The inertia weight of KGMO is modified by using ALO to solve the fast convergence rate of KGMO algorithm. In this regard, some actual solutions have been demonstrated to meet the obligation of 5G mobility organisation. The experiments are carried out by using MATLAB and validated KGMO-ALO with other heuristic algorithms such as PSO, SOA and MBO in terms of throughput, end-to-end delay, BER and number of HO. From the result, it is proved that the proposed KGMO-ALO achieved better performance in delay, BER and HO, where the throughput is less than existing techniques, when it reaches the high number of users or clusters. ALO matches the best antlion in every rounding process for better solution and this needs to be enhanced by modifying the proposed method in future work.

References

1. Universal Mobile Telecommunications System (UMTS): Mobility Enhancementsin Heterogeneous Networks; 3GPP TR 36.839, Tech. Report, 3GPP. Valbonne, France (2012)

2. Ding, M., Lopez-Perez, D., Claussen, H., Kaafar, M.A.: On the fundamental characteristics of ultra-dense small cell networks. IEEE Netw. **32**, 92–100 (2018)
3. Chen, Y., Liu, W., Niu, Z., Feng, Z., Hu, Q., Jiang, T.: Pervasive intelligent endogenous 6G wireless systems: Prospects, theories and key technologies. Digit. Commun. Netw. **6**, 312–320 (2020)
4. Ge, X., Tu, S., Mao, G., Wang, C.X., Han, T.: 5G Ultra-Dense Cellular Networks. IEEE Wirel. Commun. **23**, 72–79 (2016)
5. López-Pérez, D., Ding, M., Claussen, H., Jafari, A.H.: Towards 1 Gbps/UE in Cellular Systems: Understanding Ultra-Dense Small Cell Deployments. IEEE Commun. Surv. Tutor. **17**, 2078–2101 (2015)
6. User Equipment (UE): Radio Transmission and Reception; Part 1: Range 1 Standalone; 3GPP TR 38.101, Tech. Report, 3GPP. Valbonne, France (2018)
7. Ericsson, H.E., Modified RRH Arrangement for HST SFN; 3GPP: Valbonne, France, 2019
8. Castro-Hernandez, D., Paranjape, R.: Optimization of Handover Parameters for LTE/LTE-A In-building Systems. IEEE Trans. Veh. Technol. **67**, 5260–5273 (2017)
9. Zhang, H., Huang, W., Liu, Y.: Handover Probability Analysis of Anchor-Based Multi-Connectivity in 5G User-Centric Network. IEEE Wirel. Commun. Lett. **8**, 396–399 (2018)
10. Evolved Universal Terrestrial Radio Access (e-utra): User Equipment (ue) Procedures in Idle Mode; release 14 (Tech. Rep. No. TS36.304); 3GPP. Valbonne, France (2018)
11. Bjornson, E., Jorswieck, E., Debbah, M., Ottensen, B.: Multiobjective signal processing optimization: The way to balance conflicting metrics in 5g systems. IEEE Signal. Process. Mag **31**, 14–23 (2014)
12. Shelke, P.M., Prasad, R.S.: (2019). DBFS: Dragonfy Bayes Fusion System to detect the tampered JPEG image for forensic analysis. Evolutionary Intelligence, 1–17
13. Dahi, Z.A., Mezioud, C., Alba, E.: (2016). A novel adaptive genetic algorithm for mobility management in cellular networks. In Proceedings of the 11th international conference on hybrid artifcial intelligent systems, (HAIS) (pp. 225–237)
14. El-Ghazali, T.: Metaheuristics: from design to implementation, vol. 74. John Wiley & Sons (2009)
15. Hopwood, S.J., Jeans, J.: 2009. An Introduction to the Kinetic Theory of Gases
16. Loeb, L.B.: The kinetic theory of gases. Courier Corporation (2004)
17. Nie, S., Wu, D., Zhao, M., Gu, X., Zhang, L., Lu, L.: An Enhanced Mobility State Estimation Based Handover Optimization Algorithm in LTE-A Self-organizing Network. Procedia Comput. Sci. **52**, 270–277 (2016)
18. Tiwari, R., Deshmukh, S.: Analysis and Design of an Efficient Handoff Management Strategy via Velocity Estimation in HetNets. Trans. Emerg. Telecommun. Technol. 2019, e3642
19. Ray, R.P., Tang, L.: Hysteresis Margin and Load Balancing for Handover in Heterogeneous Network. Int. J. Future Comput. Commun. **4**, 231 (2016)
20. Shayea, I., Ismail, M., Nordin, R., Ergen, M., Ahmad, N., Abdullah, N.F., Alhammadi, A., Mohamad, H.: New weight function for adapting handover margin level over contiguous carrier aggregation

deployment scenarios in LTE-advanced system. Wirel. Pers. Commun. **108**, 1179–1199 (2019)

21. Saeed, M., Kamal, H., El-Ghoneimy, M.: Novel Type-2 Fuzzy Logic Technique for Handover Problems in a Heterogeneous Network. Eng. Optim. **50**, 1533–1543 (2018)
22. Chaudhuri, S., Baig, I., Das, D.: Self-organizing method for handover performance optimization in LTE-advanced network. Comput. Commun. **110**, 151–163 (2017)
23. Alhammadi, A., Roslee, M., Alias, M.Y., Shayea, I., Alriah, S., Abas, A.B. Advanced Handover Self-optimization Approach for 4G/5G HetNets Using Weighted Fuzzy Logic Control. In Proceedings of the 2019 15th International Conference on Telecommunications (ConTEL), Graz, Austria, 3–5 July: 2019; pp. 1–6
24. Alhammadi, A., Roslee, M., Alias, M.Y., Shayea, I., Alraih, S. Dynamic Handover Control Parameters for LTE-A/5G Mobile Communications. In Proceedings of the 2018 Advances in Wireless and Optical Communications (RTUWO), Riga, Latvia, 15–16 November: 2018; pp. 39–44
25. Abdulraqeb, A., Mardeni, R., Yusoff, A.M., Ibraheem, S., Saddam, A.: Self-optimization of Handover Control Parameters for Mobility Management in 4G/5G Heterogeneous Networks. Autom. Control Comput. Sci. **53**, 441–451 (2019)
26. Dahi, Z.A., Alba, E., Draa, A.: A stop-and-start adaptive cellular genetic algorithm for mobility management of GSM-LTE cellular network users. Expert Syst. Appl. **106**, 290–304 (2018)
27. Parija, S., Singh, S.S., Swayamsiddha, S.: Particle swarm optimization for cost reduction in mobile location management using reporting cell planning approach. In: Recent Developments in Intelligent Nature-Inspired Computing, pp. 171–189. IGI Global (2017)
28. Swayamsiddha, S., Parija, S., Sahu, P.K., Singh, S.S.: Optimal reporting cell planning with binary differential evolution algorithm for location management problem. Int. J. Intel Syst. Appl. **9**(4), 23–31 (2017)
29. Rao, R.V., Savsani, V.J., Vakharia, D.P.: Teaching learning-based optimization: a novel method for constrained mechanical design optimization problems. Comput. Des. **43**, 303–315 (2011)
30. Rao, R.V., Kalyankar, V.D., Waghmare, G.: Parameters optimization of selected casting processes using teaching-learning-based optimization algorithm. Appl. Math. Model. **38**, 5592–5608 (2014)
31. Yang, C., Cheng, Y., Chuang, L.: A natural PCR-RFLP primer design for SNP genotyping using teaching-learning-based optimization with elite strategy. IEEE Trans. Nanobiosci. **15**, 657–665 (2016)
32. Raja, B.D., Jhala, R.L., Patel, V.: Multi-objective optimization of a rotary regenerator using tutorial training and self-learning inspired teaching-learning based optimization algorithm (TS-TLBO). Appl. Therm. Eng. **93**, 456–467 (2016)
33. Shukla, A.K., Singh, P., Vardhan, M.: Neighbour teaching learning based optimization for global optimization problems. J. Intell. Fuzzy Syst. **34**, 1583–1594 (2018)
34. Cui, Z., Li, F., Zhang, W.: Bat algorithm with principal component analysis, Int. J. Mach. Learn. Cybern. (2018) 1–20

35. Shan, X., Cheng, H.: Modified bat algorithm based on covariance adaptive evolution for global optimization problems. Soft Comput. **22**(16), 5215–5230 (2018)
36. Yilmaz, S., Kucuksille, E.U.: Improved bat algorithm (IBA) on continuous optimization problems. Lect. Notes Softw. Eng. **1**(3), 279 (2013)
37. Lyu, S., Li, Z., Huang, Y., Wang, J., Hu, J.: Improved self-adaptive bat algorithm with step-control and mutation mechanisms. J. Comput. Sci. **30**, 65–78 (2019)
38. Nawi, N.M., Rehman, M., Khan, A., Chiroma, H., Herawan, T.: A modified bat algorithm based on Gaussian distribution for solving optimization problem. J. Comput. Theor. Nanosci. **13**(1), 706–714 (2016)
39. Kiani, A.T., Faisal Nadeem, M., Ahmed, A., Sajjad, I.A., Raza, A., Khan, I.A. Chaotic Inertia Weight Particle Swarm Optimization (CIWPSO): An Efficient Technique for Solar Cell Parameter Estimation. In Proceedings of the 3rd International conference on Computing, Mathematics and Engineering Technologies Idea to Innovsource Building a Knowledge Economy iCoMET 2020, Kahului, HI, USA, 29–30 January: 2020; pp. 1–6
40. Nunes, H.G.G., Pombo, J.A.N., Mariano, S.J.P.S., Calado, M.R.A., Felippe, de Souza: J.A.M. A new high performance method for determining the parameters of PV cells and modules based on guaranteed convergence particle swarm optimization. Appl. Energy **211**, 774–791 (2018)
41. Panthagani, P., Rao, R.S.: 2017, March. KGMO for multi-objective optimal allocation of SVC and reactive power dispatch. In 2017 International Conference on Power and Embedded Drive Control (ICPEDC) (pp. 365–369). IEEE
42. Lakshminarayana, P., Kumar, T.V.: 2020, December. Kinetic Gas Molecular Optimized (KGMO) Artificial Neural Network (ANN) Based Software Reliability Prediction for Banking Applications. In International Conference on Information Systems and Management Science (pp. 160–170). Springer, Cham
43. Shabana Sulthana, S.L., Sucharitha, M.: Kinetic Gas Molecule Optimization (KGMO)-Based Speckle Noise Reduction in Ultrasound Images. In: Soft Computing and Signal Processing, pp. 447–455. Singapore, Springer (2022)

Figures

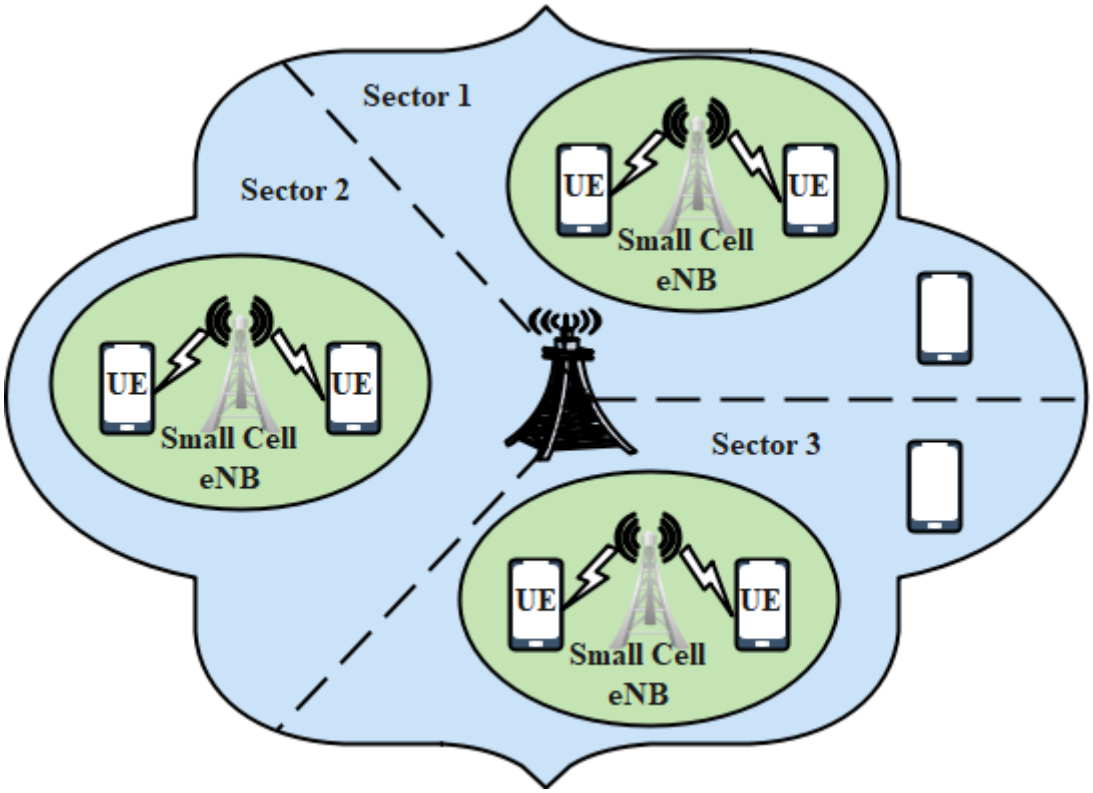

Figure 1

System model for HetNets.

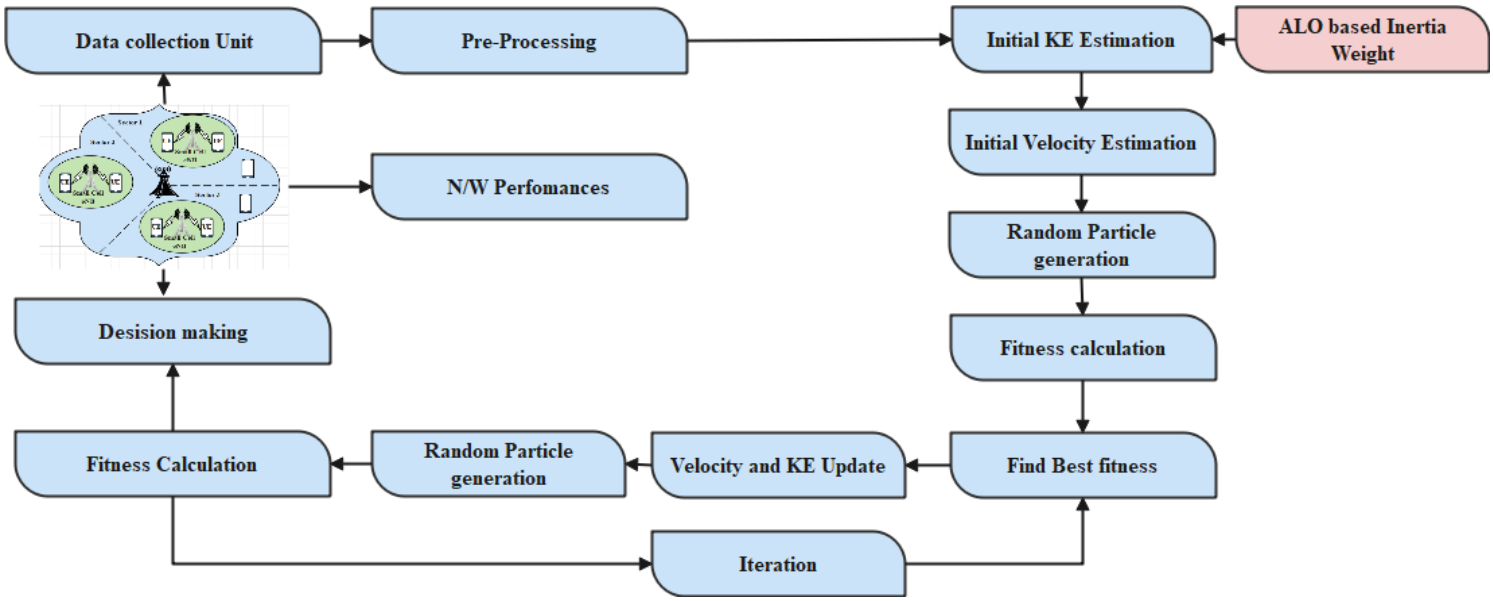

Figure 2

Block diagram for optimization technique

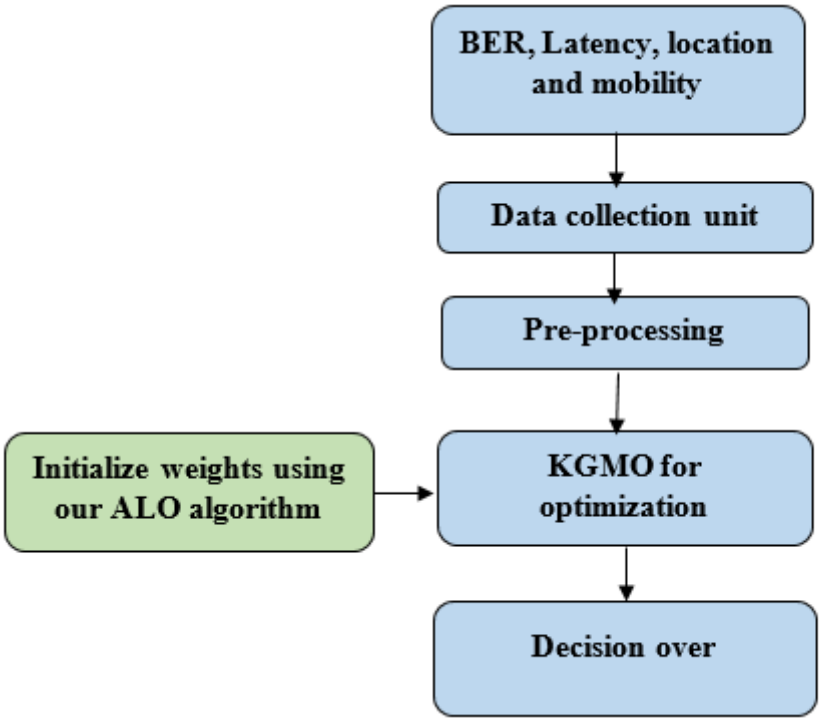

Figure 3

KGMO optimization in Heterogeneous networks

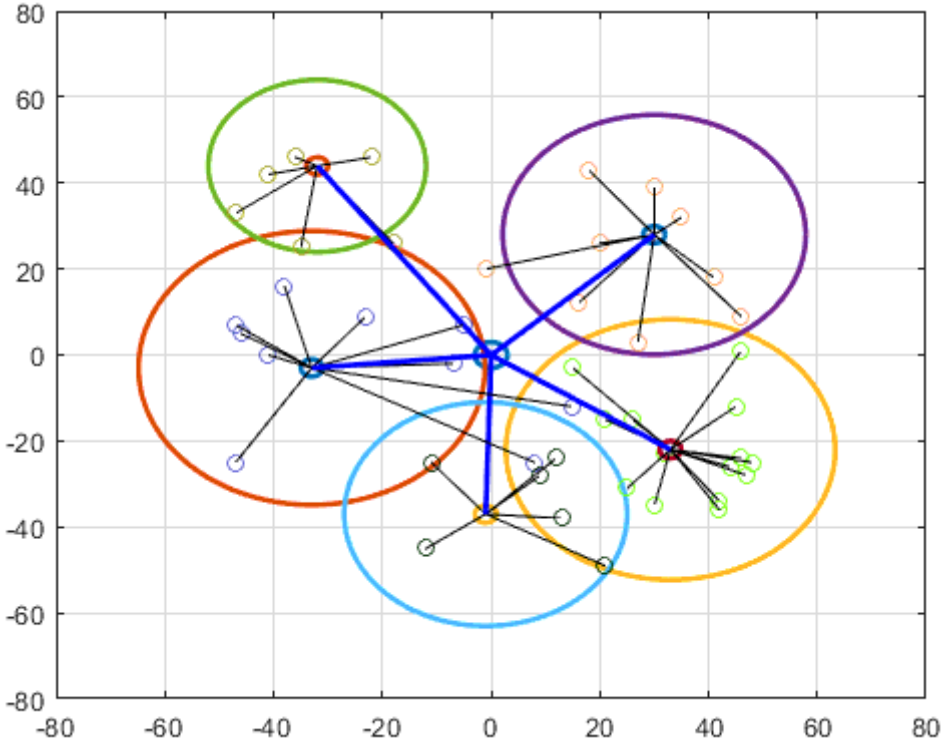

Figure 4

Proposed Network Implementation model.

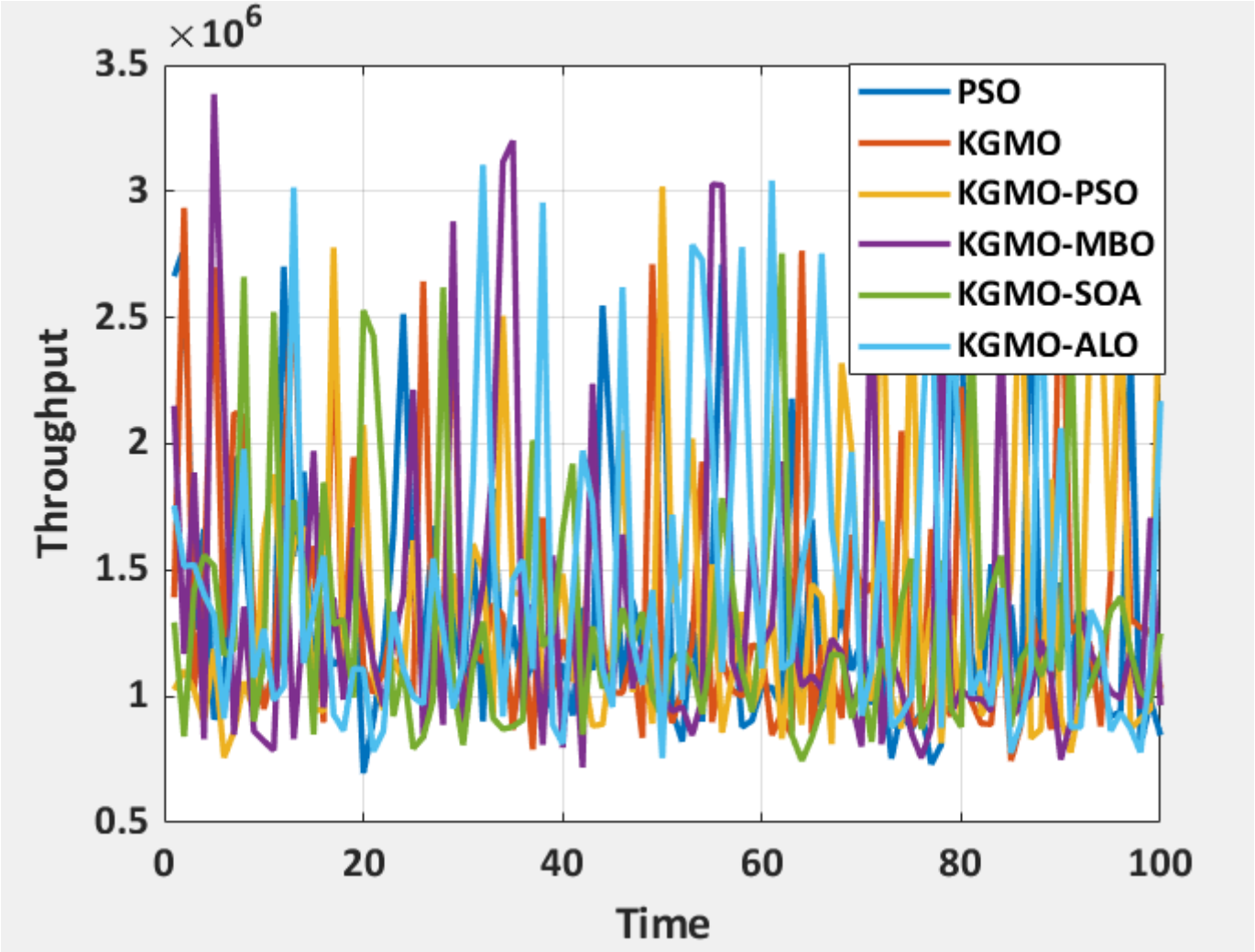

Figure 5

Validation of Proposed System in terms of Throughput.

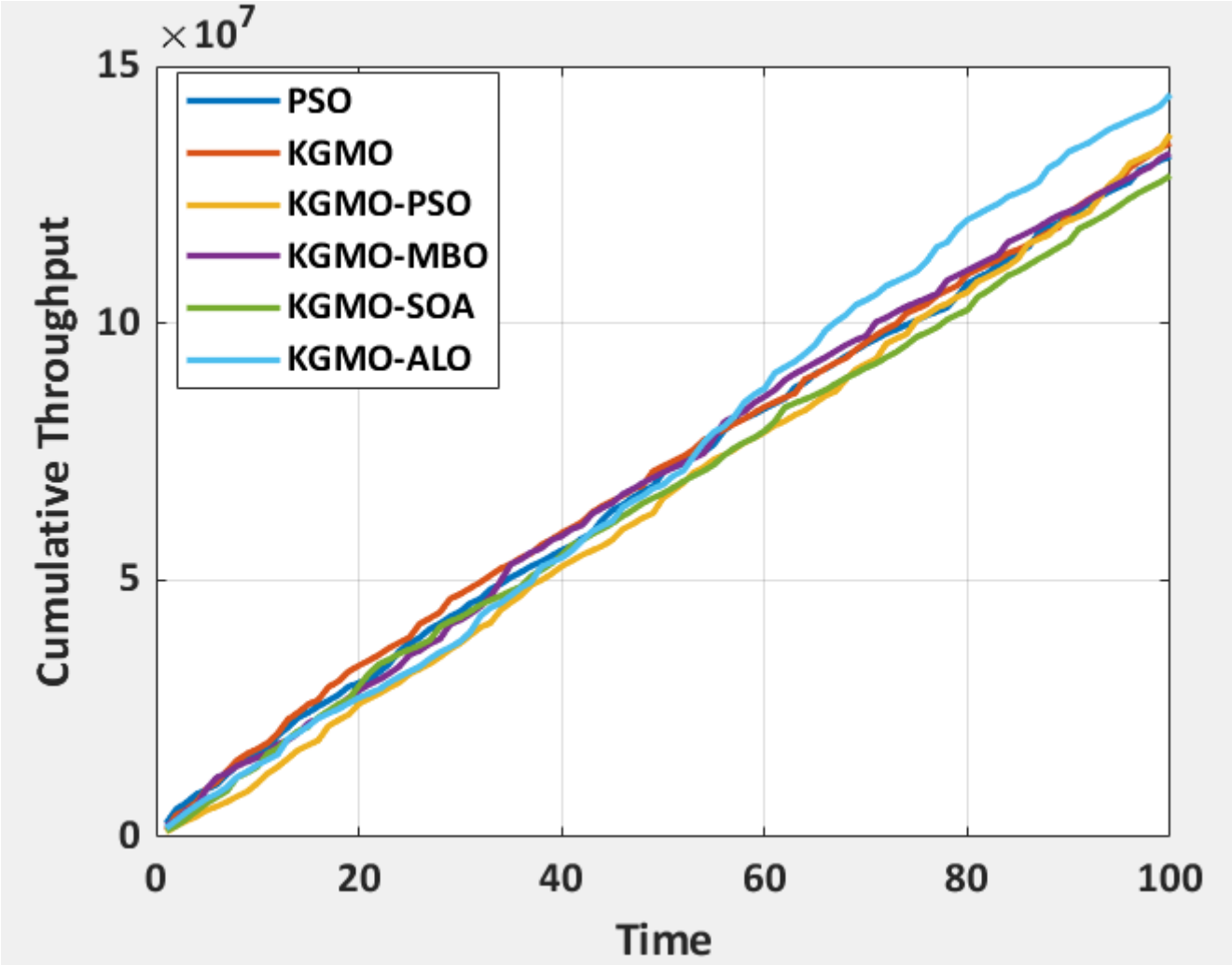

Figure 6

Validation of Proposed Method in terms of Cumulative throughput with respect to time

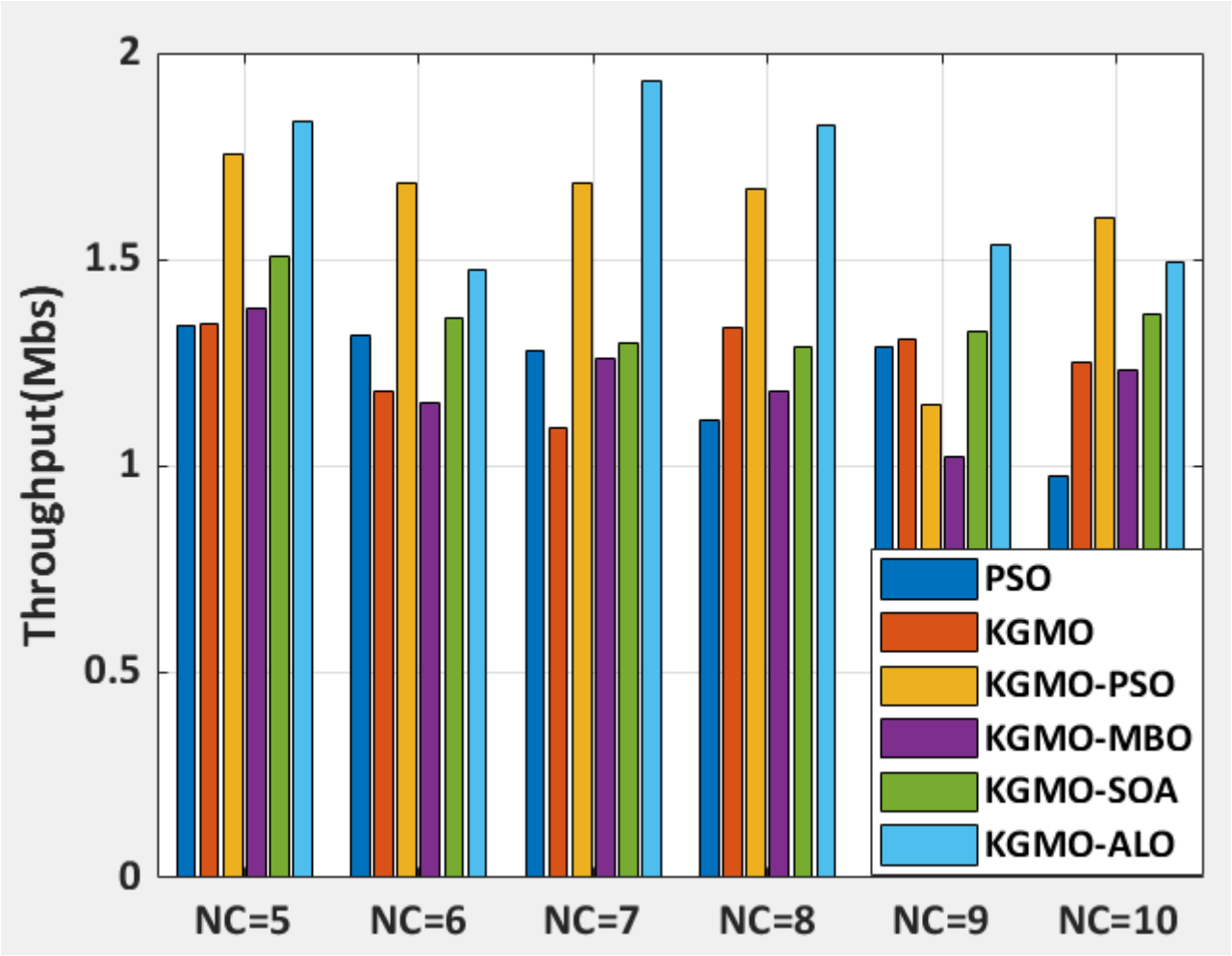

Figure 7

Validation of proposed method in terms of throughput by modifying the amount of clusters

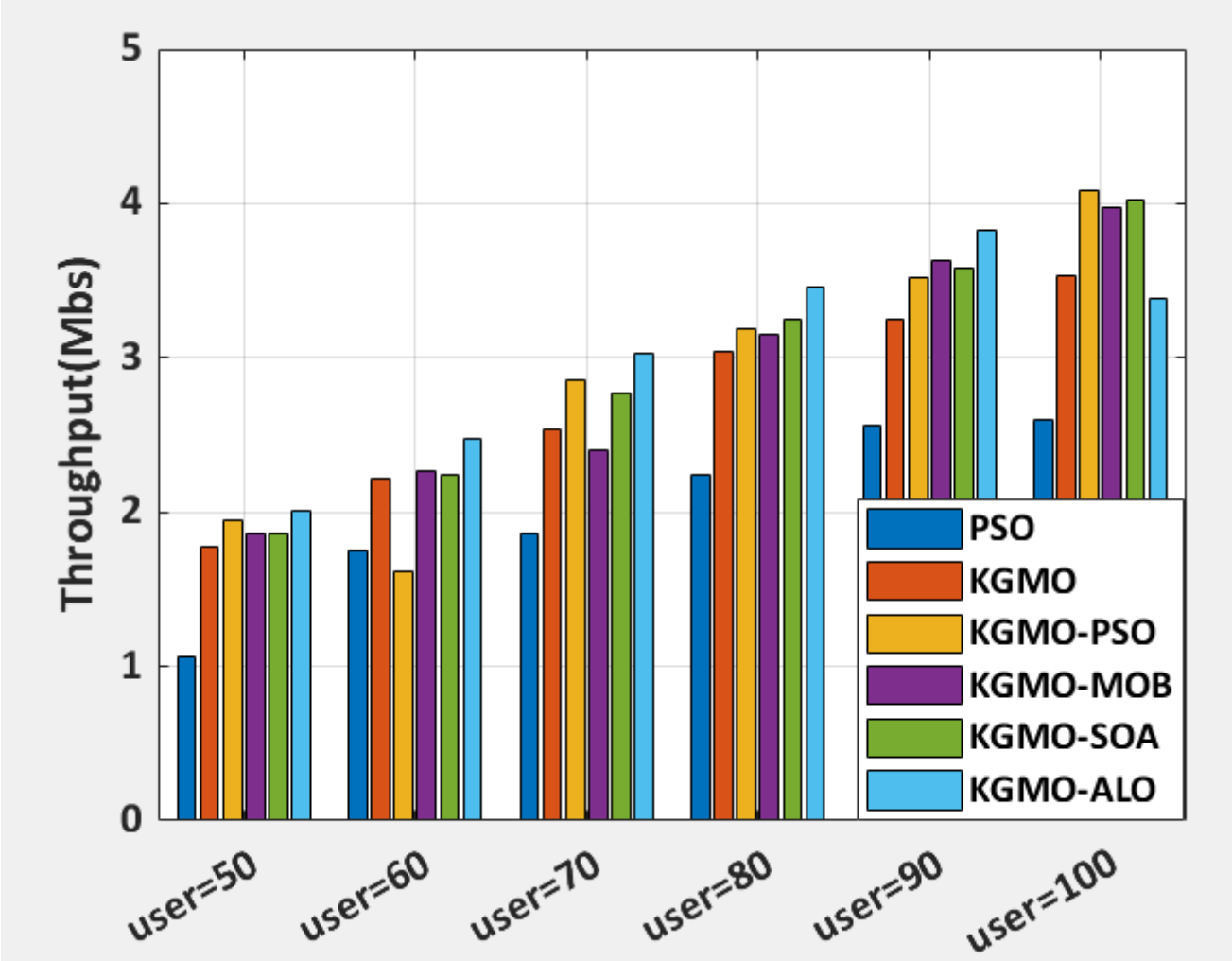

Figure 8

Validation of proposed process in terms of throughput by modifying the amount of users

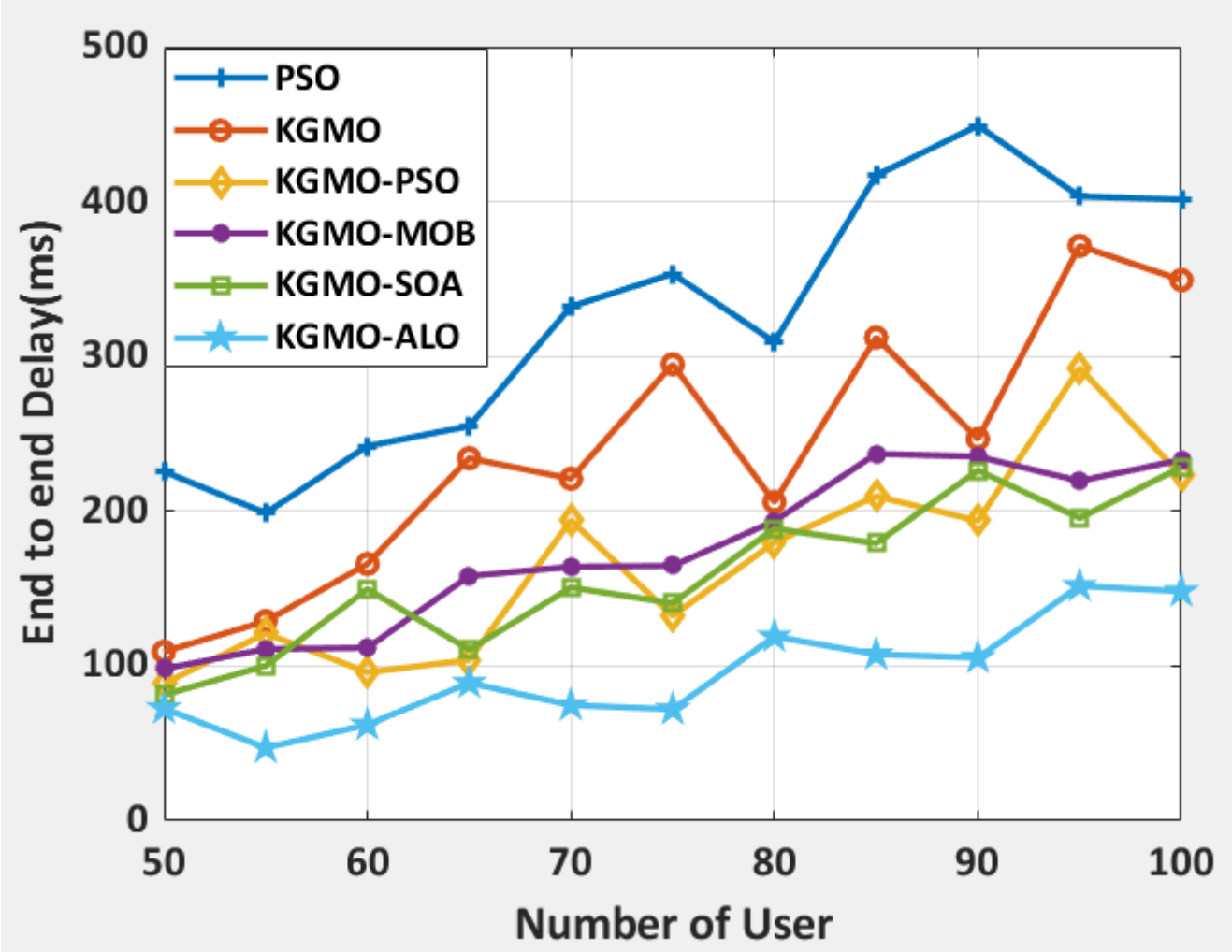

Figure 9

Validation of proposed method in terms of end to end delay with respect to number of users

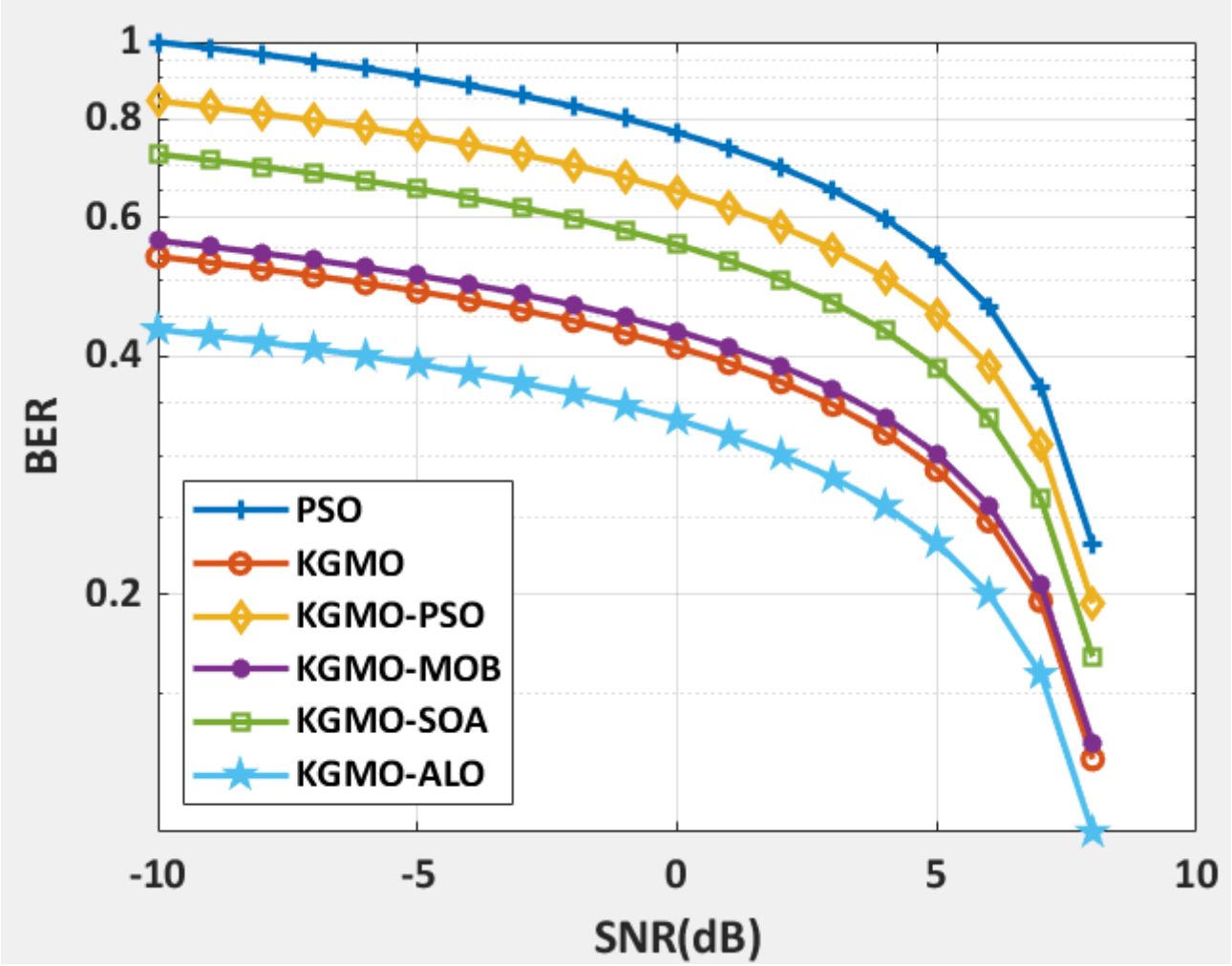

Figure 10

Validation of proposed method in terms of BER with respect to SNR

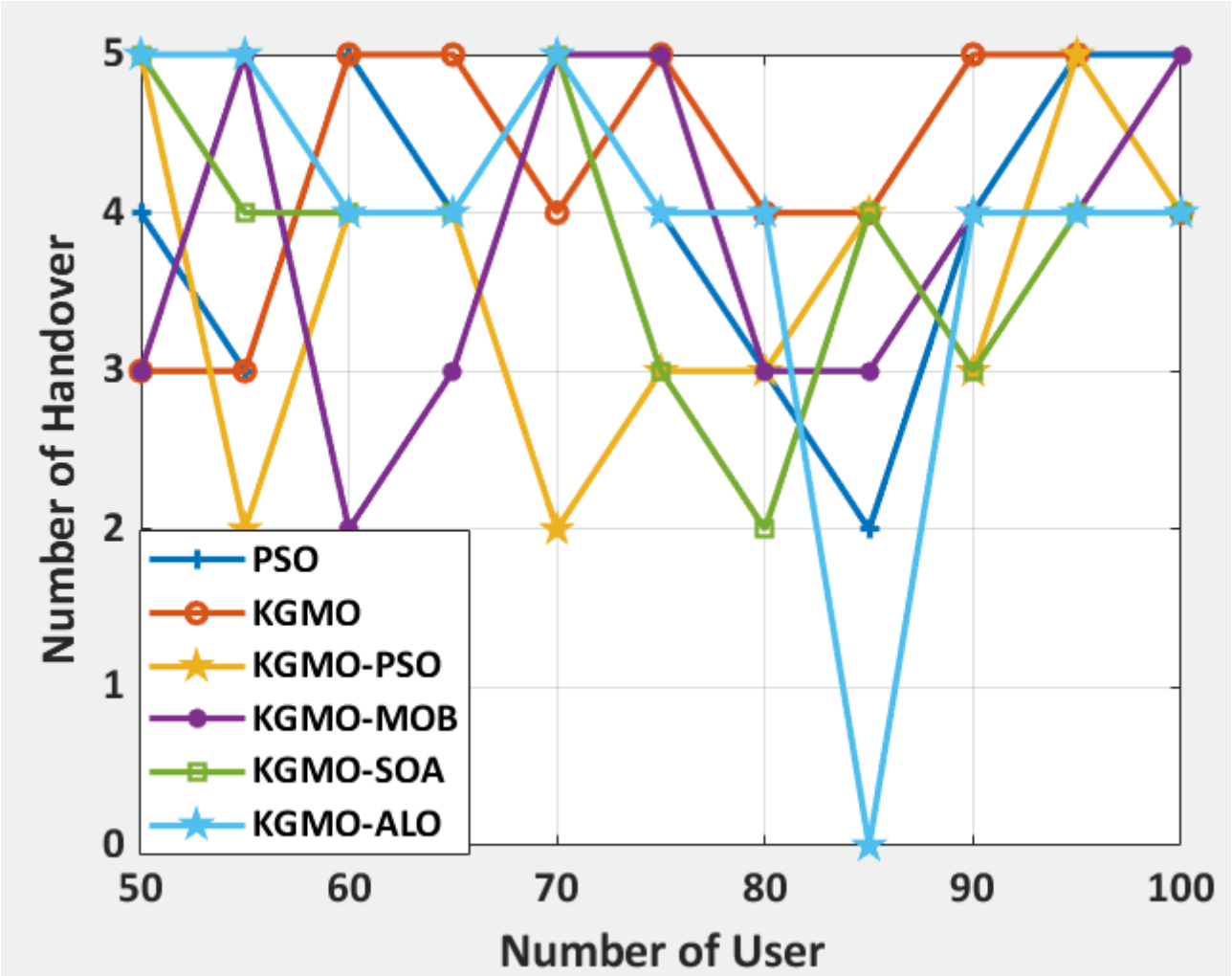

Figure 11

Validation of proposed method in terms of Handover with respect to amount of users