Impact of Learning Algorithms on Random Neural Network based Optimization for LTE-UL Systems

Ahsan Adeel, Hadi Larijani, Abbas Javed, Ali Ahmadinia

School of Engineering & Built Environment

Glasgow Caledonian University

Email: Ahsan.Adeel@gcu.ac.uk, H.Larijani@gcu.ac.uk, Abbas.Javed@gcu.ac.uk, Ali.Ahmadinia@gcu.ac.uk

Abstract

This paper presents a context-aware decision making solution to the problem of radio resource management (RRM) and inter-cell interference coordination (ICIC) in long-term evolution-uplink (LTE-UL) system. The limitations of existing analytical, artificial intelligence (AI), and machine learning (ML) based approaches are highlighted and a novel integration of random neural network (RNN) based learning with genetic algorithm (GA) based reasoning is presented. Initially, three learning algorithms (gradient descent (GD), adaptive inertia weight particle swarm optimization (AIW-PSO), and differential evolution (DE)) are applied to RNN and two learning algorithms (GD and levenberg-marquardt (LM)) are applied to artificial neural network (ANN). In the second phase, the GA based reasoning is applied to the trained ANN and RNN models for performance optimization. Finally, the ANN and RNN based optimization results are compared with the state-of-the-art fractional power control (FPC) schemes in terms of user throughput and power consumption. The simulation results revealed that an RNN-DE based cognitive engine (CE) can provide up to 14% more cell capacity along with 6dBm and 9dBm less user power consumption as compared to RNN-GD and FPC methods respectively.

Keywords: Artificial neural network, fractional power control, genetic algorithm, inter-cell interference coordination, long-term evolution-uplink system, radio resource management, random neural network.

1. Introduction

Human beings are context-aware decision makers. In any rational decision making situation, human beings follow four fundamental processes i.e., observe, analyse, decide, and act. The continuity among them makes a cycle, which is commonly known as a cognitive cycle. An illustration of natural context-aware decision making is shown in Fig. 1, where the decision making starts by observing the context first, then analysing the possible actions based on their expected performance e.g., if we decide this action, what would be the possible result? Or what will happen if we select another action? Based on the analysis, the most optimal action is to be selected to act upon.

The quality of decision making significantly depends on the quality of learning i.e., how well we learn from our past experiences [1]. Therefore, learning is the core of any decision making framework. An illustration of simple artificial context-aware decision making is shown in Fig. 2, where ML is integrated with the reasoning process. ML provides the learning capability and the ability to characterize the performance of different possible actions. Reasoning process selects the best action in order to achieve single or multi-objective optimization.

The selection of an appropriate ML and reasoning technique in Fig. 2 depends on the application requirements such as learning time, prediction accuracy, decision making speed during run-time, generalization etc. In addition, the constraints such as the amount of knowledge available for learning and computational complexity play a vital role in the selection of any ML/reasoning algorithm. The artificial context-aware decision making framework with an appropriate ML/reasoning algorithm can be applied to solve wide-range of decision making problems in multiple disciplines, such as information and communication technology (ICT), customer relationship management, and online advertisement scheduling etc.

The problem we have focused on is an ICT related problem i.e., RRM and ICIC in LTE-uplink. Power control (PC) is one of the most researched topics in LTE-uplink optimization. An advance PC mechanism can address inter-cell interference (ICI), adjacent channel interference (ACI), and user power consumption challenges in a better way. RRM and ICIC have been extensively studied and researchers have proposed solutions with variations in approaches from statistical/analytical [2] to cognitive or self-organized approaches [3]. However, the cognitive approaches are more appealing for ICIC and RRM because the combined learning and reasoning processes are capable to deal with non-ideal communication behaviours in a competent way as compared to analytical modelling. Researchers have borrowed ideas from ML and AI techniques in order to acquire cognitive features or desired intelligence capabilities. However, most of the proposed AI/ML based solutions for cognitive radio (CR) in literature ([4], [5]) are deficient in achieving the basic CE design feature concurrently i.e., fast decision making, long-term learning, and less computational complexity. The concurrent achievement of these three features is challenging but necessary for the practical deployment of any real-time cognitive communication system. A brief review on the limitations of some of the AI/ML techniques for CR adaptation and

decision making in real-time is presented in Table 1 [4] [5] [6].

The abilities of neural networks (NNs) have not been fully utilized because of the issues such as their training problems, sensitivity to the number of hidden neurons, local-minima problem, generalization capability, available training samples, and computational complexity. These issues and genuine reservations about NNs have restricted their applications in CR adaptation and decision making problem and in many other real-time applications. Otherwise, NNs can be effective because of their superior ability to deal with non-linear problems, natural behaviour of learning, fast parallel processing, easy implementation, and to be used in problems where no other heuristic or rule-based solution fits.
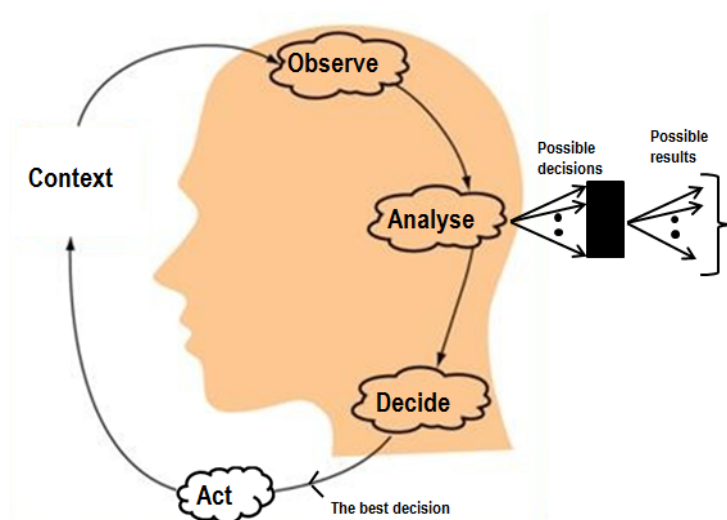

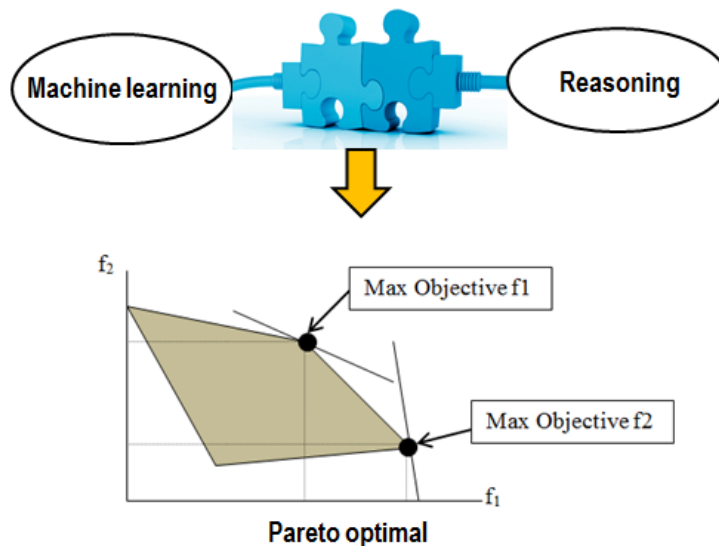Fig. 1: Natural Context-Aware Decision Making

Fig. 2: Artificial Context-Aware Decision Making

TABLE 1: AI/ML Techniques Limitations for Cognitive Radio Adaptation and Decision Making in Real-Time.

Algorithm	Limitations
Genetic algorithm and particle swarm optimization (when the priori characterizations of performance metrics is to be derived from the analytical modelling)	Limited learning, limited modelling assumptions, lack of ability to deal with non-ideal communication behaviours, and poor scalability.
Case-based-reasoning (CBR)	Performances are more related to the cases only stored in the case database and large number of cases may require large memory. In addition, not suitable to deal with unknown situations.
Rule-based system (RBS)	Performance depends on certain rules and cannot perform well in an unknown condition.
Hybrid approaches such as particle swarm optimization + case-based-systems	Memory problem, high computational complexity, limited learning, limited modelling assumptions, lack of ability to deal with non-ideal communication behaviours, and poor scalability.
Hidden Markov model (HMM)	Requires good training and computationally complex
Partially observable Markov decision processes (POMDPs)	Computationally very expensive and applicable to very simple problems.
Non-parametric learning: Drichlet process mixture model (DPMM)	Requires large number of iterations (slow convergence) and unsuited for real-time cases.
Support vector machines (SVM)	Choice of the best suited kernel function parameters, slow processing in testing phase, high algorithmic complexity, and extensive memory requirement for large-scale tasks
Artificial neural networks (ANNs)	Training and local-minima problems, limited generalization, and slow calculation rate during run-time.
Reinforcement learning (RL)	For realistic applications, the size of state-space could be so large that learning may take a long time and even become impossible in a reasonable time frame. As a result, the generalization over the state-space is necessary, which is insufficient in RL.

In our previous work [7], we presented an application of RNN to the problem of RRM and ICIC in LTE-UL. The presented RNN based CE model acquired features such as better generalization, fast calculation in run-time, and easy hardware/software implementation.

Furthermore, to achieve the best learning capability and avoid local-minima problem, four learning algorithms (GD, AIW-PSO, DE, and GA) were applied to RNN and critically analysed.

In this paper, we extended [7] by integrating a GA based reasoning with ANN and RNN based learning models, in order to optimize the performance of LTE-uplink system. The considered LTE-uplink system is the same except to inter-site distance and resource blocks (RBs) bandwidth configuration parameters. The slight change in the system configuration settings reflected considerably in training algorithm performances as compared to [7]. In [7], AIW-PSO performed the best, whereas GD, DE, and GA had low performances. In this paper, DE performed the best, whereas, GD and AIW-PSO had slightly lower performances. However, the focus of this work is to critically analyse the optimization performance in terms of throughput and power consumption.

In rest of the paper, Section 2 presents the related work in literature and their limitations. Section 3 presents an overview of RNN and training algorithms. Section 4, illustrates the proposed system model, modelling assumptions and calculations. Section 5 presents the proposed CE design. Section 6 illustrates the training and optimization results. Finally, Section 7 presents the conclusions.

2. RRM and ICIC Related Work

3rd generation partnership project (3GPP) defined FPC as a power control method for LTE-uplink system. FPC method aims to compensate the path-loss (PL) perceived by the users. However, in FPC method, the assumption that interference towards other cells is mostly because of the cell-edge users is not always true. The generated ICI in [8] negated this assumption, where the interior users contributed to the aggregated ICI as well. Therefore, the transmit power should be controlled in order to reduce ICI rather than the path-gain. In literature, different power controlling methods have been presented.

The authors in [9] presented six different scheduling algorithms for multi-cell wireless cellular systems. However, the inefficient use of spectrum and the requirements for backhaul communication and location information are some of the drawbacks in this approach. The authors in [8] proposed an interference based power control scheme which targets to achieve a fixed level of interference at every cell by dynamically adjusting the target signal-to-interference-noise ratio (SINR). However, the approach fails to concurrently improve both capacity and coverage and it is also unclear how the interference target could be dynamically set based on the traffic statistics. The authors in [10] extended the work of [2] in a self-organizing (SO) manner. The authors used support vector regression as a learning scheme in order to assist eNodeB in reallocating resources upon high traffic load. The approaches based on SVMs suffers from the problem of selecting kernel function and its parameters [11], high algorithmic complexity, training of multi-class SVM, and large memory requirement. An RL based approach for power control is presented in [12]. However, the approach suffers from less generalization capability and it is not suitable for

real-time cognitive communication system. Similarly, a game theory based power control framework is presented in [13]. This approach requires user-specific utility parameters which cannot always be acquired in many situations [4]. The authors in [14] integrated a Q-learning method with an ANN. However, both RL and ANN suffer from the same aforementioned less generalization problem; moreover, ANNs slow calculation rate at run-time, local minima, and over-fitting problems are some of the additional limitations.

To the best of our knowledge, limited work has been conducted to concurrently acquire CE design features. In contrast with existing AI/ML based approaches, RNNs inherent properties such as: (a) efficient computation (b) low complexity (c) energy-efficient hardware implementation (d) less dependency on network structure (e) strong generalization capability even with small training dataset, makes RNN a better choice for CE design [16]. Moreover, most of the presented RRM approaches in literature have failed to comply with LTEs full frequency spectrum usage requirement.

3. Random Neural Network

RNN, a machine learning technique, made up of highly interconnected processing elements called as neurons, processes the information by their state response and learns from examples. The main objective of the RNN model is to transform the inputs into meaningful outputs, learn the input-output relationship, and offer viable solutions to unseen problems (a generalization capability). RNNs were first developed by Gelenbe [17] as a new modified class of ANNs. RNN represents the transmission of signals in a very similar form to biological neural networks but offers more benefits and cope with the limitations of ANNs.

In RNN, the pulses travel in the form of impulses or exchange the signals in the form of spikes. The potential (k) of each neuron represents its state that increases/decreases with respect to an incoming signal. If a neuron receives an excitatory signal (+1), its potential increases and correspondingly decreases upon receiving inhibitory signal (-1) [17]. When the potential of neuron is equal to zero ($k_i = 0$), it is in idle state and when ($k_i > 0$), the neuron is excited. In the state of excitation, the neuron fires according to the Poisson process represented by the synaptic weights $w^+_{ij} = rP_{ij}^+$ and $w^-_{ij} = rP_{ij}^-$, where P_{ij}^+ and P_{ij}^- are the probabilities of excitatory and inhibitory signals and r is the spikes firing rate. The w^+_{ij} and w^-_{ij} can also be seen as the positive and negative rates of signal transmissions and these are typical interconnections weights of a neural network that RNN learns through the process of learning or training. The average rate of +ive signals at neuron i (λ_i^+), average rate of -ive signals at neuron i (λ_i^-), and the probability that neuron i is excited (q_i), are calculated using the following equations:

$$\lambda_i^+ = \Lambda_i + \sum_{j=1}^{n} q_j w_{ij}^+ \qquad (1)$$

$$\lambda_i^- = \lambda_i + \sum_{j=1}^{n} q_j w_{ij}^- \qquad (2)$$

$$q_i = \frac{\lambda_i^+}{r_i + \lambda_i^-} \qquad (3)$$

If $0 \leq q_i \leq 1$ for $i = 1, 2, 3..., n$ then the stationary joint probability of network $p(k, t) = p_r = [k(t) = k]$ can be written as:

$$p(k) = \prod_{i=1}^{n} (1 - q_i) q_i^{k} \qquad (4)$$

A general RNN model is depicted in Fig. 3. Further in-depth details are comprehensively presented in [18].
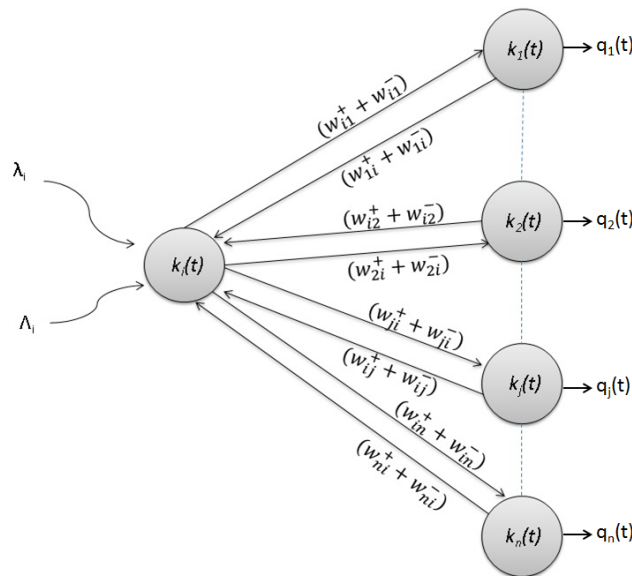

Fig. 3: Feed-forward random neural network architecture

3.1. RNN Training Algorithms

The goal of training is to learn desired system behaviour and adjust the network parameters (interconnections weights) to map (learn) the input-output relationship and minimize the mean square error (MSE). This paper has used GD, AIWPSO, DE and GA as learning algorithms. The GD algorithm calculates the gradient of the cost function with respect to all network synaptic weights and adjusts them in order to minimize the MSE. Further in-depth detail of RNN GD is presented in [17]. The procedure for AIW-PSO and DE learning algorithms is as follows:

3.1.1. AIW-PSO Learning Procedure

1. Initialize a population of S particles with random positions and velocities of d dimensions in the problem space. The position vector is an array of

interconnected weights of feed forward RNN having I Input nodes, H hidden nodes and O output nodes. The dimension of D is $2(I.H+H.O)$. The position vector is formulated as $X_{sd} = [w_{ih}^{+L1} \ w_{oh}^{+L2} w_{ih}^{-L1} \ w_{oh}^{-L2}]$, where $1 \leqslant i \leqslant I$, $1 \leqslant$ h $\leqslant$ H, $1 \leqslant o \leqslant O$. The weights are randomly distributed over the interval of [0; 1].

2. Each particle from a position in generation k moves to a new position $k+1$ by using PSO equation given in (1). The *c1* and *c2* constant values are set to 2.6 and 1.1 respectively.

$$V_{sd}^{k+1} = W V_{sd}^{k} + c_1 rand()(P_{bestsd}^{k} - x_{sd}^{k}) + c_2 rand()(G_{bestsd}^{k} - x_{sd}^{k}) \qquad (5)$$

Where P_{bestsd} and G_{bestsd} are the particle's local and global bests.

$$X_{sd}^{k+1} = X_{sd}^{k} + V_{sd}^{k+1} \qquad (6)$$

$$W_{sd}^{k+1} = 1 - \frac{1}{1 + \exp(-\alpha \, ISA_{sd}^{k})} \qquad (7)$$

Where α is a constant with values between 0 to 1 and ISA_{sd}^{k} is given as:

$$ISA = \frac{X_{sd}^{k} - P_{best}^{k}}{P_{bestsd}^{k} - G_{bestsd}^{k}} + \varepsilon \qquad (8)$$

Where ε is a small position constant

3. For each particle, evaluate the fitness function:

$$X_{sd}^{k+1} = \frac{1}{2} \sum_{p=1}^{N} \sum_{0=1}^{O} [q_o(p) - q_{des,o}] \qquad (9)$$

Where N is the number of patterns, O is the number of output, $q_{des,o}$ is the desired output in training pattern, *and qo* (p) is the output of RNN calculated by solving (1-3).

4. Compare particle fitness evaluation with particle local best (P_{best}). If current fitness evaluation value is less than P_{best}, then update P_{best} to current value and the P_{best} location equal to current location in D dimensional space.

5. Compare fitness evaluation with all P_{best} of population S. If P_{best} is less than global best (G_{best}) update G_{best} to the current particle array index.

6. For checking the convergence criteria, compute the average squared error. If the MSE is not less than threshold, go to Step-2. If the stopping criterion is met or maximum number of iterations is reached, learning is complete

3.1.2. DE Learning Procedure

The steps required for the implementation of DE training algorithm are as follows:

1. Initialize a population of *S* particles with random positions and velocities of *D* dimensions in the problem space. The position vector is an array of interconnected weights of feed forward RNN of *I* Input nodes, *H* hidden nodes and *O* output nodes. The dimension of *D* is *2(I.H+H.O)*. The position vector is formulated as $X_{sd} = [w_{ih}^{+L1} \; w_{oh}^{+L2} w_{ih}^{-L1} \; w_{oh}^{-L2}]$ where *1≤ i ≤I, 1≤ h ≤H, 1≤ o≤ O*. The weights are randomly distributed over the interval of [0; 1].

2. Randomly generate three integer numbers r_{1d}, r_2d, r_3d ϵ *[1; S]* where $r_{1d} \neq r_2d \neq r_3d \neq S$. Set the value of *F* and *CR* to 0.8 and 0.7 respectively.

3. Mutation operator is the prime operator of DE and it is the implementation of this operation that makes DE different from other Evolutionary algorithms. Mutate every particle of the population (*1 ≤ s ≤ S*) by applying the DE equation

$$Y_{sd}^{k+1} = X_{r1d}^{k} + F(X_{2d} - X_{r3d}) \tag{10}$$

The mutated s[th] particle at generation *k+1* is of dimension *D*. The mutated s[th] particle is sum of another particle at location r_1d and difference of particle values at location r_2d and r_3d. The contribution of difference of particles is controlled by parameter *F*.

4. Randomly generate one real number rand() ϵ [0; 1]. Cross over the mutated particle and the original particle using (11).

$$\begin{cases} U_{sd}^{k+1} = Y_{sd}^{k+1} if , rand() \leq CR \\ U_{sd}^{k+1} = X_{sd}^{k+1} if , rand() > CR \end{cases} \tag{11}$$

5. Evaluate the fitness function given in (12) for U_{sd}^{k+1}

$$E = \frac{1}{2}\sum_{p=1}^{N}\sum_{o=1}^{O}[q_o(p) - q_{des,o}]^2 \tag{12}$$

6. Same as Step 6 of AIW-PSO.

4. System Model

An OFDMA based RBs collision model is depicted in Fig. 4, where a power control strategy is employed to reduce the collision and ICI [9]. Fig. 5 shows a similar collision model, which has adopted 7-cell hexagonal layout (2 coexistent-10MHz Evolved Universal Terrestrial Radio Access frequency division duplexing systems) with omnidirectional

antennas at the centre of each cell. In the proposed system model, an RNN based CE is embedded inside the reference cognitive-eNodeB (C-eNodeB). The CE is responsible for monitoring and configuring the user equipment (UE) once it is attached and also the management of radio resources. In addition, the CE is responsible to suggest the optimal transmit power for UEs served by adjacent eNodeBs via X2-interface (a communication interface between eNodeBs).
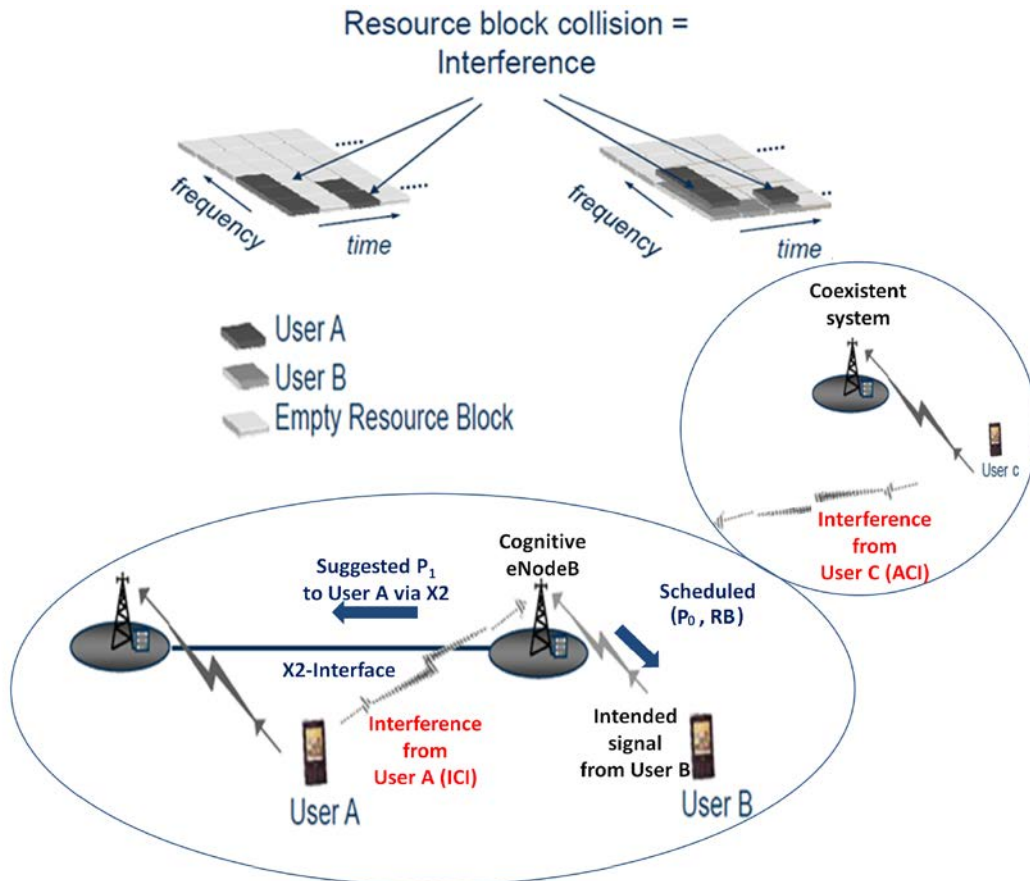

Fig. 4: ICI caused by collision between resource blocks which are used simultaneously by several users in different cells and corresponding scheduling based on the experienced channel quality information and interference in subsequent transmission time interval.

4.1. Modelling Assumptions and Calculations

The simulation is based on the assumptions and parameters of E-UTRA 10MHz macro cell system used in the Qualcomm simulator. The systems are 100% loaded with frequency reuse of 1/1. The UEs are deployed randomly according to a uniform geographical distribution in the whole network region. Moreover, wrap around technique is employed in order to remove the edge effects. The carrier frequencies of victim and external coexistent systems were set to 2000 MHz and 2010 MHz with inter-site distance of 433m. OFDMA urban macro propagation model is used. Base station (BS) and UE antennas gains were assumed to be 15 dBi and 0 dBi. 24 RBs per BS were assumed. In addition, log-normal Shadowing: 10 dB with correlation; minimum coupling loss (MCL): 70 dB; system

bandwidth: 10 MHz; bandwidth of RB: 375kHz; handover (HO) margin: 3 dB; thermal noise density: -174dBm/Hz; BS noise figure: 5 dB; UE min and max transmit power: -30 dBm to 24 dBm were the system settings. The FPC settings, OFDMA LTE link-to-system level mapping, adjacent channel leakage ratio/unwanted spectrum mask were the same as given in Qualcomm STG(08)13 and 3GPP technical specifications [24].
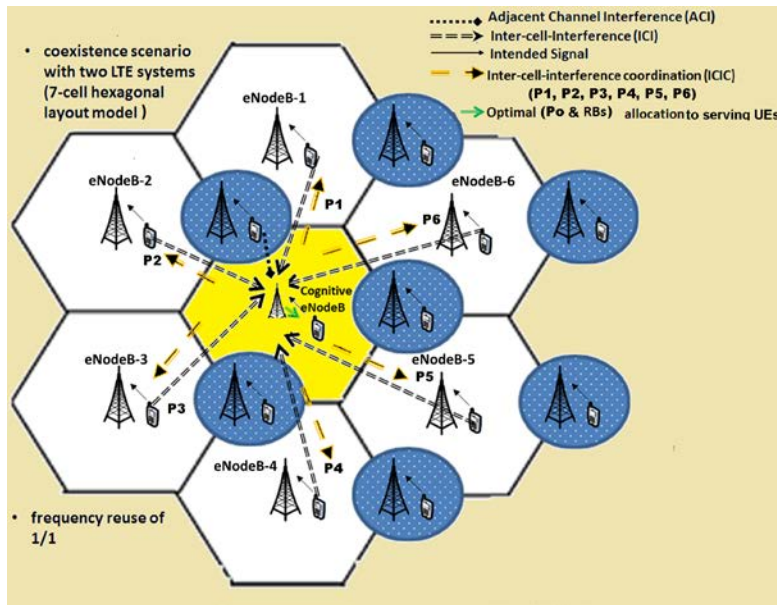

Fig. 5: Proposed system model

At the beginning of simulation (for every snapshot), UEs were randomly distributed throughout the system area and assigned a discrete speed value i.e. 0/3/30/100 km/hr with initial transmit power (maximum allowed power). The UEs get attached to the most appropriate BS depending on the handover margin, smallest PL, antenna gain, and lognormal fading. The connected UEs (active) were scheduled for every snapshot and allocated a certain amount of resources according to the quality-of-service (QoS) requirement. Every BS goes through with all mobile stations (MSs) on its served mobile list and tries adding their requested sub-carriers until all MSs are served or number of available sub-carriers exceeds the maximum limit. In the latter case, the BS discards the last MS and tries connecting the next mobile in line which may have less required number of sub-carriers. This is equivalent to modelling a round robin (RR) scheduler with a full buffer traffic model. The use of RR scheduling with power control achieves more gain as compared to channel aware scheduling such as proportional fair (PF) [9].

In [19], the UL power control is defined as follows:

$$P_t = P_{max} \times \min\left\{1, \max\left[R_{min}, \left(\frac{PL}{PL_{x-ile}}\right)^\lambda\right]\right\} \quad (13)$$

Where P_t is the transmit power of the UE in dBm, P_{max} is the maximum allowed transmit power in dBm, R_{min} is the minimum power reduction ratio to prevent UEs having good

channel condition to transmit at low power level i.e. $R_{min}=P_{min}/P_{max}$. PL is the path-loss in dB between UE and serving BS. $PL_{x\text{-ile}}$ is the x-percentile PL. With this PC scheme, the *x%* of UEs that have a PL $>PL_{x\text{-ile}}$ will transmit at P_{max}. However, $0<\gamma<=1$ is the balancing factor for UEs with bad/good channel. The value of γ plays a vital role for the trade-off between cell-edge UE performance and overall spectral efficiency. If $\gamma=0$, all UEs will be transmitting at P_{max}, resulting poor cell-edge performance. With $\gamma=1$, the equation reduces to traditional slow power control resulting poor spectral efficiency. The SINR for each UE with respect to link-to-system-level mapping is given as follows [20]:

$$SINR = \frac{c(j,k)}{I(j,k)} = \frac{P_t(j,k)*PL_{effective}(UE_{j,k},BS_j)}{I_{inter}(j,k)+I_{ext}(j,k)+N_t} \qquad (14)$$

Where C(j,k) is the received power at j^{th} serving eNodeB from the intended k^{th} UE, Pt(j,k) is the transmit power in dBm and $PL_{effective}$ is the effective path-loss which considered minimum coupling loss (MCL) as defined in [20].

The combined ICI and ACI at the victim reference cell are calculated as follows [20]:

$$I(j,k) = I_{inter}(j,k)+I_{ext}(j,k)+N_t \qquad (15)$$

Where $I_{inter}(j,k)$ is the ICI coming from the UEs of adjacent cells operating on the same frequency sub-carriers and is calculated as follows:

$$I_{inter} = \sum_{l=1,l\neq j}^{N_{cell}} p_t(l,k)*PL_{effective}(UE_{l,k},BS_j) \qquad (16)$$

$I_{ext}(j,k)$ is the ACI coming from the UEs on adjacent channels in coexistent LTE system. ACI is the combination of $I_{unwanted}$ (unwanted emission in adjacent band) and $I_{blocking}$ (blocking effect of receiver) and it is calculated as:

$$I_{ext} = \sum_{m=1}^{N_{ExtCell}} \sum_{v=1}^{k} iRss_{blocking}(UE_{m,v},BS_j)*iRss_{unwanted}(UE_{m,v},BS_j) \qquad (17)$$

The effect of $I_{ext}(j; k)$ onto the system performance is set to be equal to zero, in order to make better relationship between SINR and ICI for CE learning.

The achieved bit rate for all uplink users is calculated as follows [20]:

$$Bit-Rate = \frac{N_{SC_{per-UE}}}{N_{total_{sc}}}(x\frac{bps}{Hz})_{SINR}BW_{MHz} \qquad (18)$$

Where $N_{sc\text{-}per\text{-}UE}$ and $N_{total\text{-}sc}$ are the number of allocated sub-carriers to each UE and total number of sub-carriers available at each BS. The x_{bps} Hz is the spectral efficiency with respect to calculated SINR and BW_{MHz} is the bandwidth.

5. Cognitive Engine Design

The CE selects the best set of radio parameters (transmission power in our case) for a given channel state and application requirements. In CE, learning mechanisms such as ANN and RNN assist the reasoning process such as GA in deciding the radio parameters to be used in the next block of transmission. The reasoning algorithm uses the learning scheme to determine how closely a set of parameters can meet the desired goals. Afterwards, the reasoning algorithm correspondingly chooses the set of parameters in order to maximize the utility function. Fig.6 shows a general CE design for context-aware decision making, where the RNN based learning model is integrated with the GA based reasoning algorithm. The learning model is fed with the environmental measurements, configuration parameters, and performance metric. The aim of the learning model is to learn the relationship between the inputs and outputs i.e., how environmental measurements and configuration parameters are effecting the performance. Once the relationship is learnt, the reasoning process can tune the controlled parameters in order to maximize the system performance.

A simplified cognitive operation is illustrated in Fig.7, where the decision is to be made based on given environment conditions and current radio objectives. The learning module observes the channel X and estimates the performance S given radio configuration Y. The vectors X, Y, and S are the training parameters coming from the radio. The radio communicates with optimizer the required objectives and current channel quality information (CQI). The optimizer then queries the learning module with considered X and Y. The learning section returns the approximate performance of considered X and Y i.e. $P(S|X, Y)$. Based on this report, the optimization section decides the optimal parameters. In this particular case, the RNN based learning agent is able to characterize the performances of different possible powers and channels combinations for serving users and adjacent cell users.
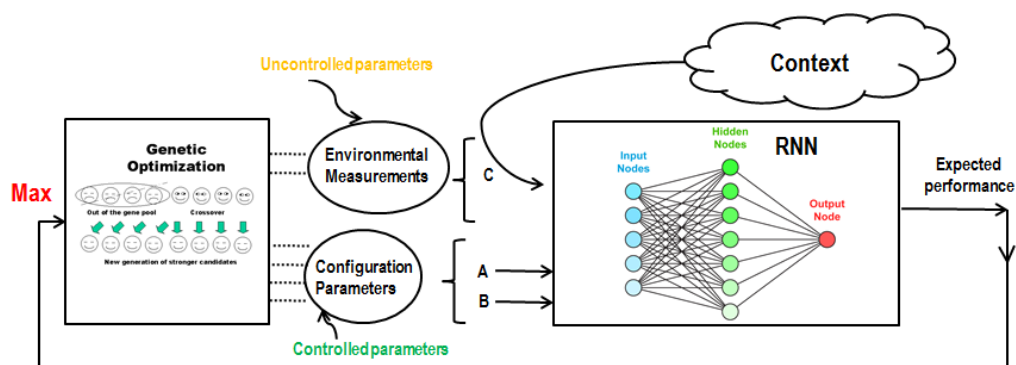

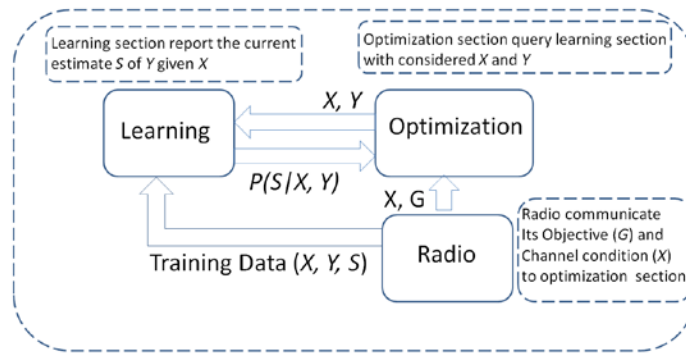Fig. 6: General CE design for context-aware decision making

Fig. 7: Cognitive engine operation

As stated in Section 1, the accurate learning is of excessive importance so that the controller can predict the performance with different combinations of configuration parameters. Prior to the above functioning, a process of training or learning is required, where some important input/output parameters of the system model should be selected. The information which is available to our cognitive controller can be classified into three different categories. One, environmental measurements (external factors effecting the reliability of communication), we have considered SINR and ICI as environmental measurements. Second, configuration parameters (tuning knobs which can be changed in an optimal way to achieve desired performance), considered parameters are available channels (RBs) and transmit power (P_0) of all UEs served by C-eNodeB (reference cell) and the transmit power (P_1, P_2, P_3, P_4, P_5, P_6) of all UEs served by adjacent eNodeBs, operating on the same frequency. Third, performance metric, expected throughput as a performance measure is considered.

The aforementioned environmental measurements and configuration parameters are the inputs of RNN black-box and the performance metric (throughput) is the output. With a feature set X, label set Y, and n training samples $T = ((x_1, y_1),\ldots,(x_n, y_n)) \in (X * Y)^n$, a ML algorithm creates a mapping A: X$\rightarrow$Y from features to labels and predicts the labels for new samples. Once the C-eNodeB scheduler selects the UE and assigns RBs for uplink transmission, the embedded CE selects the optimal power (P_0) for serving UE based on SINR and interference on scheduled RBs in subsequent transmission time interval (TTI), such that the target SINR is achieved. In addition, the CE suggests the optimal transmit powers of UEs served by adjacent 7 eNodeBs (P_1, P_2, P_3, P_4, P_5, P_6), operating on the same scheduled RBs via X2-interface. Note that the objective of embedded CE is to maximize the throughput of all attached UEs both interior/exterior under the constraints of a given interference threshold. The throughput maximization under interference limitation is achieved by putting constraints on ICI environmental measurement parameter during optimization.

6. Performance Evaluation

6.1. Training
6.1.1. Training Accuracy

The main aim of CE training is to achieve the least possible MSE in less training time, which represents how well the CE learnt the system behaviour. The CE was trained with ANN and RNN using the dataset obtained from the system model. In training, different numbers of neurons, hidden layers and epochs were tried. The best performed RNN and ANN structures were 1 hidden layer with 6 neurons and 1 hidden layer with 10 neurons respectively.

The least MSE achieved by ANN-GD was 2.88 E-03 in 10K iterations (33 s) and ANN-LM achieved the MSE of 1.37 E-06 in 313 iterations (57 s). RNNs detailed statistical results including mean of MSE (MMSE), standard deviation of MSE (SD-MSE), and the best MSE (BMSE) is given in Table 2. The MMSEs, SD, and BMSEs were calculated by running the algorithm 10 times for each iteration case (represented as iter in Table 2 which starts from 50 up to 300 iterations, in order to get more precise approximation of these quantities). When all the training algorithms were run for 50 iterations, the performance of GD was better than PSO and DE. However, to achieve the least possible MSE, the iterations were increased to 300, where the least MSE achieved by GD algorithm was 4.72 E-4, DE converged to 3.67 E-4, and PSO achieved the least MSE of 5.01 E-04. Similarly, the least MMSE for GD, DE, and AIW-PSO was 4.77 E-04, 4.08 E-04, and 5.63 E-04, where the better performance of DE and slightly lower performance of AIW-PSO are evident. The training performance of ANN-LM in terms of accuracy and training time was better than RNN but RNN outperformed ANN in performance optimization.

TABLE 2: Statistical Results- MMSE: Mean of Mean Squared Error, SD-MSE: Standard Deviation of Mean Squared Errors, BMSE: Best Mean Squared Error

Iter	GD			AIW-PSO			DE		
	MMSE	*SD-MSE*	*BMSE*	*MMSE*	*SD-MSE*	*BMSE*	*MMSE*	*SD-MSE*	*BMSE*
50	7.10 E-04	4.00 E-06	7.03 E-04	3.41 E-03	4.59 E-03	2.95 E-03	1.08 E-02	4.66 E-03	8.37 E-03
100	5.99 E-04	2.02 E-06	5.95 E-04	1.32 E-03	2.36 E-04	1.08 E-03	3.82 E-03	1.89 E-03	1.30 E-03
200	5.16 E-04	2.01 E-06	5.12 E-04	6.51 E-04	6.72 E-05	5.83 E-04	1.61 E-03	5.48 E-04	4.53 E-04
300	4.77 E-04	3.05 E-06	4.72 E-04	5.63 E-04	6.23 E-04	5.01 E-04	4.08 E-04	3.10 E-05	3.67 E-04

6.1.2. Computational Complexity and Calculation Rate:

There is a trade-off between accuracy, computational complexity, and calculation time. The computational complexity of GD ($O(N:I:H:O)$) is less than PSO/DE ($O(N:I^2:H)$) and GA. Correspondingly, the execution time of GD (0.89 sec/iteration) was less than AIW-PSO (6.47 sec/iteration) and DE (10.31 sec/iteration). Hence, GD outperformed all other algorithms in terms of training speed or computational complexity. However, in real-time cases (having training time constraints), the CEs are trained once to achieve least possible

MSE and the retraining is generally not appropriate. In that case, the calculation time is not of significance. Hence, DE is preferred over GD for CE training and to achieve more prediction accuracy.

6.2. Testing

6.2.1. FPC reference point selection:

3GPP defined γ over the range of [0, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9, and 1]. For a given FPC curve, moving from left to right, a trade-off between cell coverage and capacity exists. In two extreme cases, when $\gamma=0$ (no PL compensation), all MSs transmits at fixed power level and BS receives a wide range of power levels. This case leads to poor cell-edge user's throughput due to users low transmission power. However, with full compensation power control (FCPC), $\gamma=1$, BS receives signals with almost the same power level. Hence, this case improves cell-edge user's throughput at the cost of reduced system capacity. This is due to high cell-edge user's transmission power and corresponding increased interference to neighbouring cells. Therefore, it is difficult to ensure fairness between cell-centre and cell-edge users and indeed impossible to improve both capacity and coverage simultaneously. Consequently, a basic conclusion is the trade-off between the overall cell throughput and the outage cell throughput at different γ values.

Before the coverage-capacity gains comparison among ANN, RNN, and FPC, the best coverage-capacity trade-off case for FPC based method was determined. This was achieved by evaluating the FPC based power allocation over the range of different γ values. Fig. 8 shows the cumulative distribution function (CDF) plots of 5%-tile user throughput (coverage) and capacity. In literature, researchers have shown different optimal choices for various γ values such as 0.5, 0.6, 0.7, and 0.8 [21] [22] [8] [23]. In our case, we found the best coverage- capacity trade-off at $\gamma=0.8$ and taken it as a reference point for further investigation and comparisons.

6.2.2. FPC, ANN-LM, RNN-GD, and RNN-DE coverage/capacity comparison:

Table 3 illustrates the 5th, 25th, and 50th-%-tile user and cell throughput. Fig.9 and Fig.10 illustrate the CDF of instantaneous user and cell throughput respectively. The performance investigation has revealed that ANN-LM provided 28% more throughput to the cell-edge users while keeping the same cell capacity as compared to FPC. In contrast, the RNN-GD and RNN-DE based CEs provided 14% more cell capacity while penalising the cell-edge users of reference cell as compared to FPC. The ANN based CE allocated high transmission power to the cell-edge users in reference cell, which correspondingly increases the ICI in adjacent cells. In contrast, the RNN based CE allocated less power to connected cell-edge users and higher to users near centre in reference cell. It is to be noted that the variation in γ value has caused more changes in the coverage level as compared to the average cell capacity. This is due to the considered network layout and configuration. However, a different network layout/configuration setting may cause more deviation in average cell

capacity as well. Nevertheless, the aim here is to show the performance differences among FPC at γ=0.8 (optimal reference point for comparison), ANN-LM, RNN-GD, and RNN-DE.

6.2.3. FPC, ANN-LM, RNN-GD, and RNN-DE user power consumption comparison:

Table 4 illustrates the 5th, 25th, and 50th%-tile users' power consumption for the whole system. Fig.11 shows the CDF of users' power consumption for the whole system. The comparison among FPC, ANN-LM, RNNGD, and RNN-DE has revealed that the users connected to ANN-LM based CE have used almost the same power at the cell-edges and slightly lower power on average as compared to FPC. The users' connected to RNN-GD based CE have used approximately 8 dBm less power at cell-edges and 5dBm less power when they are in the cell centre, as compared to FPC. The RNN-DE based CE further reduced the power consumption, where the 50%-tile of users have used 10dBm, 8dBm, and 6dBm less power as compared to FPC, ANN-LM, and RNNGD respectively.

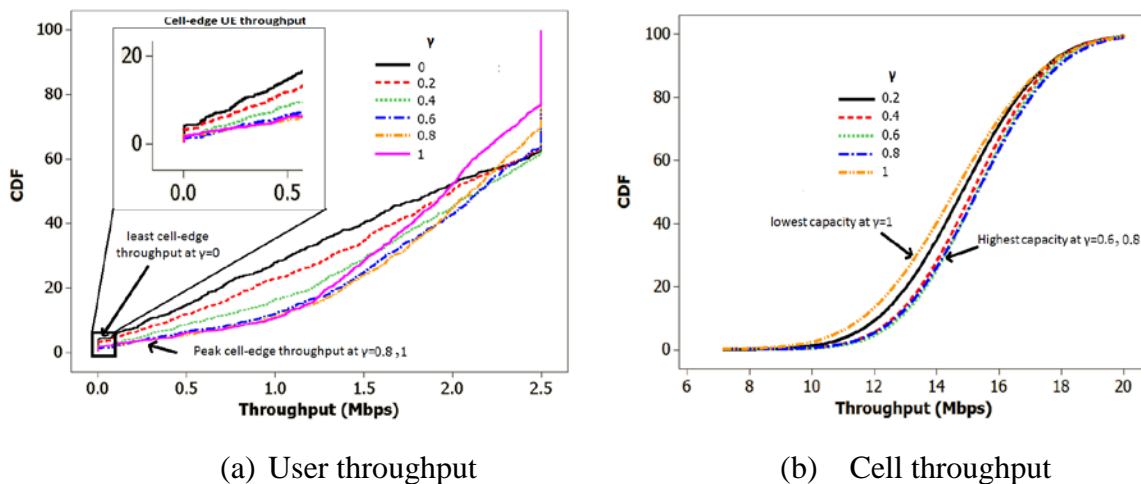(a) User throughput   (b)   Cell throughput

Fig. 8: FPC based power allocation. Note: how the PL compensation factor controls the trade-off between fairness in cell-edge performance (cell-edge users throughput) and sectors to accumulate throughput.

TABLE 3: 5%-tile, 25%-tile, and 50%-tile user throughput and average cell throughput (kbps)

Methods	5%-tile	25%-tile	50%-tile	Average cell (capacity)
FPC-0.4	270	1370	2140	14960
FPC-0.5	361	1412	2233	15152
FPC-0.6	385	1511	2150	15368
FPC-0.8	450	1548	2102	15028
FPC-1	420	1513	1920	14630
ANN-LM	540	1517	2225	14890
RNN-GD	360	1832	2500	16700
RNN-DE	366	1813	2500	16930

TABLE 4: 5%-tile, 25%-tile, and 50%-tile user power consumption (dBm)

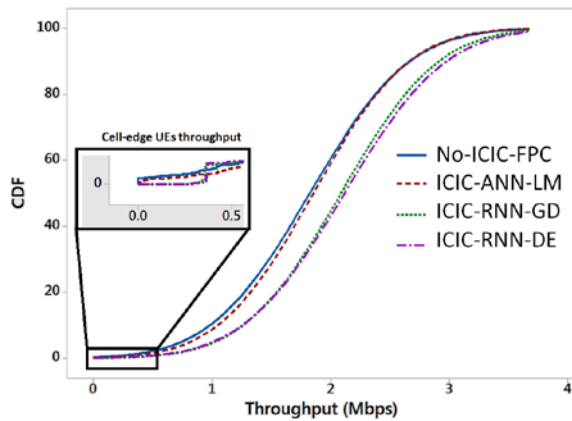| Methods	5%-tile	25%-tile	50%-tile
FPC-0.4	10.103	15.733	18.647
FPC-0.6	3.160	11.462	16.259
FPC-0.8	-4.671	6.809	13.409
FPC-1	-12.184	2.221	10.341
ANN-LM	-13.916	-1.098	8.664
RNN-GD	-22.527	-6.432	6.584
RNN-DE	-21.421	-12.518	0.5

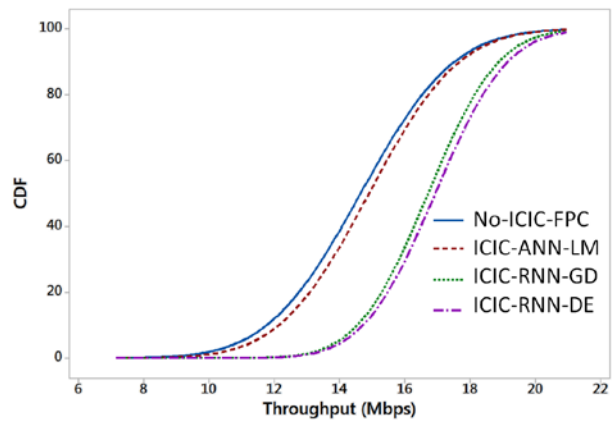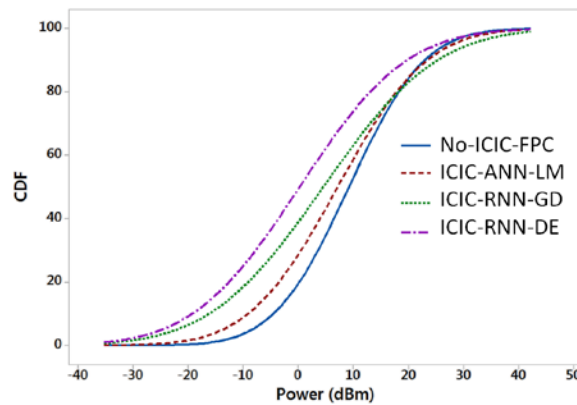Fig. 9: Instantaneous user throughput

Fig. 10: Cell throughput

Fig. 11: Instantaneous user power consumption

7. Conclusion and Future work

In this paper, we presented a context-aware decision making framework to address the problem of RRM and ICIC in LTE uplink. The proposed framework was built by combining RNN based learning and GA based reasoning algorithms. Firstly, the ANN and RNN based learning models were evaluated in terms of learning accuracy and then the optimization was applied to ANN-LM, RNN-GD, and RNN-DE based learning models. The performance

evaluation of learning process showed that DE was 22% better than GD and 26% better than AIW-PSO in terms of learning accuracy (based on BMSE). The performance evaluation of optimization process revealed that the ANN based CE can provide up-to 25% better cell-edge performance while keeping the same cell capacity and power consumption as compared to FPC. The RNN-GD based CE provided 14% more cell capacity with 8dBm and 5dBm less power consumption by cell-edge and centre UEs. The RNN-DE based CE provided 14% more cell capacity while further reducing the average UE power consumption up-to 6dBm and 9dBm as compared to RNN-GD and FPC respectively. Hence, the RNN-DE based CE remained the optimal choice. Table 5 explains the abbreviations used in this paper. In future, we intend to compare the performance of simulation with field test, which will better reflect the suitability of each algorithm in different scenarios.

TABLE 5: Abbreviations

3GPP	3rd Generation Partnership Project
AI	Artificial Intelligence
AIW-PSO	Adaptive Inertia-Weight Particle Swarm Optimization
ANN	Artificial Neural Network
ACI	Adjacent-Channel Interference
BS	Base Station
CR	Cognitive Radio
CE	Cognitive Engine
CQI	Channel Quality Information
CCO	Capacity and Coverage Optimization
CBS	Case-Based System
CLPC	Closed Loop Power Control
CDF	Cumulative Distribution Function
DE	Differential Evolution
EUTRA	Evolved Universal Terrestrial Radio Access
FPC	Fractional Power Control
GA	Genetic Algorithm
GD	Gradient Descent
HMM	Hidden Markov Model
ICI	Inter-Cell-Interference
ICIC	Inter-Cell-Interference-Coordination
LTE	Long Term Evolution
LM	Levenberg-Marquardt
ML	Machine Learning
MCL	Minimum Coupling Loss
MSE	Mean Square Error
MS	Mobile Station
NN	Neural Network

OFDMA	Orthogonal Frequency Division Multiplexing Access
PC	Power Control
PL	Path-Loss
PSO	Particle Swarm Optimization
PF	Proportional Fair
QoS	Quality-of-Service
RL	Reinforcement Learning
RNC	Radio Network Controller
RBS	Rule-Based System
RB	Resource Block
RRM	Radio Resource Management
RNN	Random Neural Network
RR	Round Robin
SINR	Signal-to-Interference-Noise-Ratio
SVM	Support Vector Machine
SON	Self-Organizing-Network
SO	Self-Organizing
SD	Standard Deviation
TTI	Transmission Time Interval
UL	Up-Link
UE	User Equipment

References

[1] J. Johnson, Designing with the Mind in Mind: Simple Guide to Understanding User Interface Design Guidelines, Elsevier, 2013.   http://dx.doi.org/10.1145/2020976.2021005

[2] A. L. Stolyar and H. Viswanathan, "Self-organizing dynamic fractional frequency reuse for best-effort traffic through distributed inter-cell coordination," in INFOCOM 2009, IEEE. IEEE, 2009, pp. 1287–1295. http://dx.doi.org/10.1109/INFCOM.2009.5062043

[3] A. Attar, V. Krishnamurthy, and O. N. Gharehshiran, "Interference management using cognitive base-stations for umts lte," Communications Magazine, IEEE, vol. 49, no. 8, pp. 152–159, 2011. http://dx.doi.org/10.1109/MCOM.2011.5978429

[4] M. Bkassiny, Y. Li, and S. K. Jayaweera, "A survey on machine-learning techniques in cognitive radios," Communications Surveys & Tutorials, IEEE, vol. 15, no. 3, pp. 1136–1159, 2013. http://dx.doi.org/10.1109/SURV.2012.100412.00017

[5] A. He, K. K. Bae, T. R. Newman, J. Gaeddert, K. Kim, R. Menon, L. Morales-Tirado, J. J. Neel, Y. Zhao, and J. H. Reed, "A survey of artificial intelligence for cognitive radios," Vehicular Technology, IEEE Transactions on, vol. 59, no. 4, pp. 1578–1592, 2010. http://dx.doi.org/10.1109/TVT.2010.2043968

[6] C. J. Burges, "A tutorial on support vector machines for pattern recognition," Data mining

and knowledge discovery, vol. 2, no. 2, pp. 121–167, 1998. http://dx.doi.org/10.1023/A:1009715923555

[7] Adeel, Ahsan, et al. "Random neural network based power controller for inter-cell interference coordination in LTE-UL." Communication Workshop (ICCW), 2015 IEEE International Conference on. IEEE, 2015. http://dx.doi.org/10.1109/ICCW.2015.7247202

[8] M. Boussif, N. Quintero, F. D. Calabrese, C. Rosa, and J. Wigard, "Interference based power control performance in lte uplink," in Wireless Communication Systems, 2008. ISWCS'08. IEEE International Symposium on, IEEE, 2008, pp. 698–702. http://dx.doi.org/10.1109/ISWCS.2008.4726146

[9] G. Fodor, C. Koutsimanis, A. R´acz, N. Reider, A. Simonsson, and W. M¨uller, "Intercell interference coordination in ofdma networks and in the 3gpp long term evolution system," Journal of Communications, vol. 4, no. 7, pp. 445–453, 2009. http://dx.doi.org/10.4304/jcm.4.7.445-453

[10] M. Brehm and R. Prakash, "Proactive resource allocation optimization in lte with inter-cell interference coordination," Wireless networks, vol. 20, no. 5, pp. 945–960, 2014. http://dx.doi.org/ 10.1007/s11276-013-0657-y

[11] J. Shawe-Taylor and N. Cristianini, Kernel methods for pattern analysis, Cambridge university press, 2004.

[12] M. Simsek, M. Bennis, and I. Guvenc, "Learning based frequencyand time-domain inter-cell interference coordination in hetnets," IEEE Transaction, arXiv preprint arXiv: 1411.5548, 2014. http://dx.doi.org/10.1109/TVT.2014.2374237

[13] V. Poulkov, P. Koleva, O. Asenov, and G. Iliev, "Combined power and inter-cell interference control for lte based on role game approach," Telecommunication Systems, vol. 55, no. 4, pp. 481–489, 2014. http://dx.doi.org/10.1007/s11235-013-9803-1

[14] A. Galindo-Serrano and L. Giupponi, "Distributed q-learning for aggregated interference control in cognitive radio networks," Vehicular Technology, IEEE Transactions on, vol. 59, no. 4, pp. 1823–1834, 2010. http://dx.doi.org/10.1109/TVT.2010.2043124

[15] S. Deb and P. Monogioudis, "Learning based uplink interference management in 4g lte cellular systems," IEEE Transaction, arXiv preprint arXiv: 1309.2543, 2013. http://dx.doi.org/10.1109/TNET.2014.2300448

[16] S. Timotheou, "The random neural network: a survey," The computer journal, vol. 53, no. 3, pp. 251–267, 2010. http://dx.doi.org/10.1093/comjnl/bxp032

[17] E. Gelenbe, "Random neural networks with negative and positive signals and product form solution," Neural computation, vol. 1, no. 4, pp. 502– 510, 1989. http://dx.doi.org/10.1162/neco.1989.1.4.502

[18] E. Gelenbe, "Stability of the random neural network model," Neural computation, vol. 2, no. 2, pp. 239–247, 1990. http://dx.doi.org/10.1162/neco.1990.2.2.239

[19] 3GPP. (2009) Technical specification group radio access network, eutra, rf system scenarios (release 9). [Online]. Available: http://www.etsi.org/deliver/etsi tr/136900 136999/136942/08.02.00 60/tr 136942v080200p.pdf

[20] E. C. O. E. Handbook, SEAMCAT, "January 2010," 2010. [Online]. Available:http://www.cept.org/files/1050/documents/SEAMCAT%20Handbook%20January %202010.pdf

[21] W. Xiao, R. Ratasuk, A. Ghosh, R. Love, Y. Sun, and R. Nory, "Uplink power control, interference coordination and resource allocation for 3gpp e-utra," in Vehicular Technology Conference, 2006. VTC-2006 Fall, 2006 IEEE 64th, IEEE, 2006, pp. 1–5. http://dx.doi.org/10.1109/VTCF.2006.208

[22] R. Mullner, C. F. Ball, K. Ivanov, J. Lienhart, and P. Hric, "Contrasting open-loop and closed-loop power control performance in utran lte uplink by ue trace analysis," in Communications, 2009. ICC'09. IEEE International Conference on, IEEE, 2009, pp. 1–6. http://dx.doi.org/10.1109/ICC.2009.5198853

Copyright Disclaimer