Performance Evaluation 68 (2011) 347–360

Contents lists available at ScienceDirect

Performance Evaluation

journal homepage: www.elsevier.com/locate/peva

# Evaluating perceived voice quality on packet networks using different random neural network architectures

Kapilan Radhakrishnan*, Hadi Larijani

*Department of Communication, Network and Electronic Engineering, School of Engineering and Computing, Glasgow Caledonian University, Glasgow, G4 0BA, UK*

ARTICLE INFO

*Article history:*
Available online 18 January 2011

*Keywords:*
Voice quality
Random neural networks
MOS
Non-intrusive
Feed-forward
Recurrent

ABSTRACT

Voice over Internet Protocol (VoIP) is one of the fastest growing technologies in the world. In VoIP speech signals are transmitted over the same network used for data communications. The internet is not a robust network and is subjected to delay, jitter, and packet loss. It is very important to measure and monitor the quality of service (QoS) the users experience in VoIP networks; this is not an easy task and usually requires subjective tests. In this paper we have analyzed three non-intrusive models to measure and monitor voice quality using Random Neural Networks (RNN). A RNN is an open queuing network with positive and negative signals. We have assessed the voice quality based on various parameters i.e. delay, jitter, packet loss, and codec. In our approach we have used the Mean Opinion Score (MOS) calculated using a Perceptual Evaluation of Speech Quality (PESQ) algorithm to generate data for training the RNN model. We have studied two feed-forward models and a recurrent architecture. We have found that the simple feed-forward architecture has produced the most accurate results compared to the other two architectures.

## 1. Introduction

Over the last few years, the Internet has grown enormously and has become a converged network of millions of heterogeneous networks. Many new technologies have been developed and among these, voice communication has gained popularity because of its many advanced features: i.e. low cost, easy local or long distance calls, mobility, caller ID, 3-way calling, virtual numbers, etc. The major problem VoIP faces is to provide the same "toll-quality" as traditional Public Switched Telephone Networks (PSTN). The Internet which is generally considered a "best-effort" network, was originally developed to carry only data traffic. This network is not robust and is subject to network delay, jitter, and packet loss. Voice is a real-time traffic type, its quality is affected by various factors of IP networks (delay, jitter, and packet loss) and other non-IP network parameters (codec, language, etc.) [1]. When transmitted over the internet the voice quality degrades significantly compared to data traffic. Delay, jitter, and packet loss (PL) are the three important parameters which can affect the voice quality over the internet. Reliability and availability are the other important parameters to be considered in VoIP networks, traditional PSTN provide service availability of 99.999% (five 9's) which is 5 min of downtime in a year [2].

The Internet is not as reliable as PSTNs because of complex design, multiple protocols, network management systems, vendors, etc. Transmitting voice over the Internet includes converting analog voice signals to digital signals, encoding, decoding, and converting back the digital signals to analog voice. This adds more delay to voice transmission. The best practice is to minimize this delay; towards this end it is better to encode near the speaker end and decode near the listener end. In addition to this, voice quality degrades if transcoding of speech is performed in the middle of transmission and with concatenation of low bit rate voice codecs [3].

* Corresponding author. Tel.: +44 141 331 8506; fax: +44 141 331 3690.
*E-mail addresses:* Kapilan.radhakrishnan@gcu.ac.uk (K. Radhakrishnan), H.larijani@gcu.ac.uk (H. Larijani).

0166-5316/$ – see front matter © 2011 Elsevier B.V. All rights reserved.
doi:10.1016/j.peva.2011.01.001

In voice communications, quality of service is measured between speaker and listener or simply stated as Mouth to Ear (M2E). Voice quality measurement methods are broadly classified into two types: subjective methods and objective methods. In subjective methods [4], human subjects are asked to listen to the degraded voice generated in a controlled environment and to use Mean Opinion Score (MOS) to rate the voice quality over the scale of 1–5 (1 being bad quality and 5 being excellent voice quality). Though subjective tests are time-consuming and very expensive they have remained a powerful method which are a preferred choice by international standard bodies for assessing voice quality. The limitations of subjective tests are 1. unsuitable for real time applications, 2. unsuitable for large volume of data, and 3. requires a controlled environment to conduct the tests. Objective methods [5] have been developed to overcome the limitations of subjective methods. Objective methods use subjective methods as benchmarks. Objective methods either use algorithms or mathematical calculations to predict voice quality. Objective methods have been further classified into either intrusive or non-intrusive methods. Intrusive methods [6] (e.g. ITU-T PESQ) use reference signals and distorted signals to predict voice quality. Intrusive methods are more accurate but unsuitable for monitoring live traffic. Non-Intrusive methods (e.g. ITU-T E-Model) are computational models which are suitable for live traffic, but have the disadvantage of complex calculations for R-factor and recalculations into MOS [7]. Measuring voice quality in real-time is a difficult task to do and previous research has shown many different methods to predict perceived voice quality [8–11] which falls into one of the categories discussed above. There is a need for new methods to measure voice quality non-intrusively and more importantly to perform it in real-time.

In recent years, the Neural Networks (NN) based models have been used in communication networks to predict voice quality [12,13]. The NN models have been very popular because of their success in a wide range of applications. The NN models are capable of learning the numerical or logical relationships between data given to them. Their parallel architecture gives them an advantage over other methods to overcome computational difficulties. They can adapt to any changes in the input data quickly by allowing training processes to continue while processing the new information [14]. They have their own limitations and the types of data used are limited as well. NN model computation time is less because of the use of nonlinear function calculations for each neuron. Random Neural networks are a recurrent neural network model developed by Gelenbe [15,16] in 1989. A RNN is an open queuing network of N fully connected neurons in which positive and negative impulse signals circulate. These signals are represented as excitatory (+1) and inhibitory (−1) impulses respectively. The combination of excitation and inhabitation impulse signals changes the potential of a receiving neuron. The neuron can fire only if its potential is positive. Some of the unique features of RNN models are:

- RNNs represent the signals transmitted in a network closer to a biological neuronal network than other networks (i.e. ANNs).
- The standard training algorithm (Gradient Descent) is less complex and has strong generalization capacity even for small training data sets.
- Although it is a spiked recurrent stochastic model [17], its steady state probability distribution is a simple analytical equation which can be computed easily and efficiently without the use of Monte Carlo methods.
- RNNs can be easily implemented in both hardware [18] and software [19] (its neuron can be represented by simple counters).
- The potential of each neuron can be represented as an integer rather than binary values.
- The excitatory and inhibitory signal behavior makes RNNs an excellent modeling tool for various applications [20].

The RNN based models have been used in modeling, optimization, hardware, image processing, communication systems, simulation, pattern recognition, and classification [20,14]. The use of RNNs has improved the modeling capabilities of communication networks. Recent research in the area of communication networks includes QoS of packet switching networks [21,22], detection of DoS attacks [23], and automatic quantification of the quality of service for real-time multimedia applications such as audio and video [24–27]. The subjective tests and pseudo-subjective quality assessment (PSQA) methods have been used to validate and to quantify multimedia traffic quality. Delay, jitter, PL, packetization interval, language, and codec have been used as quality affecting parameters.

Our main contribution in this paper has been the use of RNN models to measure perceived voice quality non-intrusively. We have used delay, jitter, packet loss, and codec as voice quality affecting parameters. We have added larger data sets and introduced a new codec (Speex) to our previous research [28]. We have also tested three different RNN architectures: simple feed-forward, 3-layer feed-forward, and recurrent architecture. This has allowed us to study the effect of voice quality parameters in more detail and to observe the behavior of different RNN architectures we have implemented.

The rest of the paper is divided into 5 sections: In Section 2, we discuss the related works. In Section 3, we discuss the RNN mathematical model we used in our method to predict voice quality. In Section 4, we discuss the various impairments used as voice quality affecting parameters in our experiments. In Section 5, we explain the simulation setup we used in this research. And in Section 6, we analyze the results we obtained in detail and is followed by the conclusions.

## 2. Related works

In [29], the effect of delay, jitter, and packet loss on voice quality was presented. The study was based on data collected using PESQ. In [24,27], a similar study was presented using data obtained from subjective tests. In [24], the authors used loss rate, loss distribution, codec, forward error correction, and packetization interval as voice quality affecting parameters.

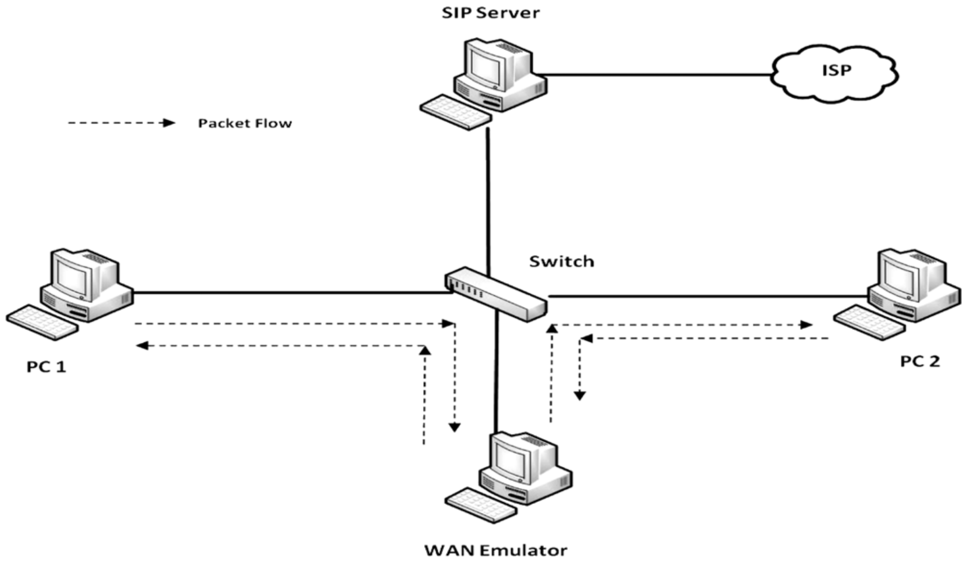
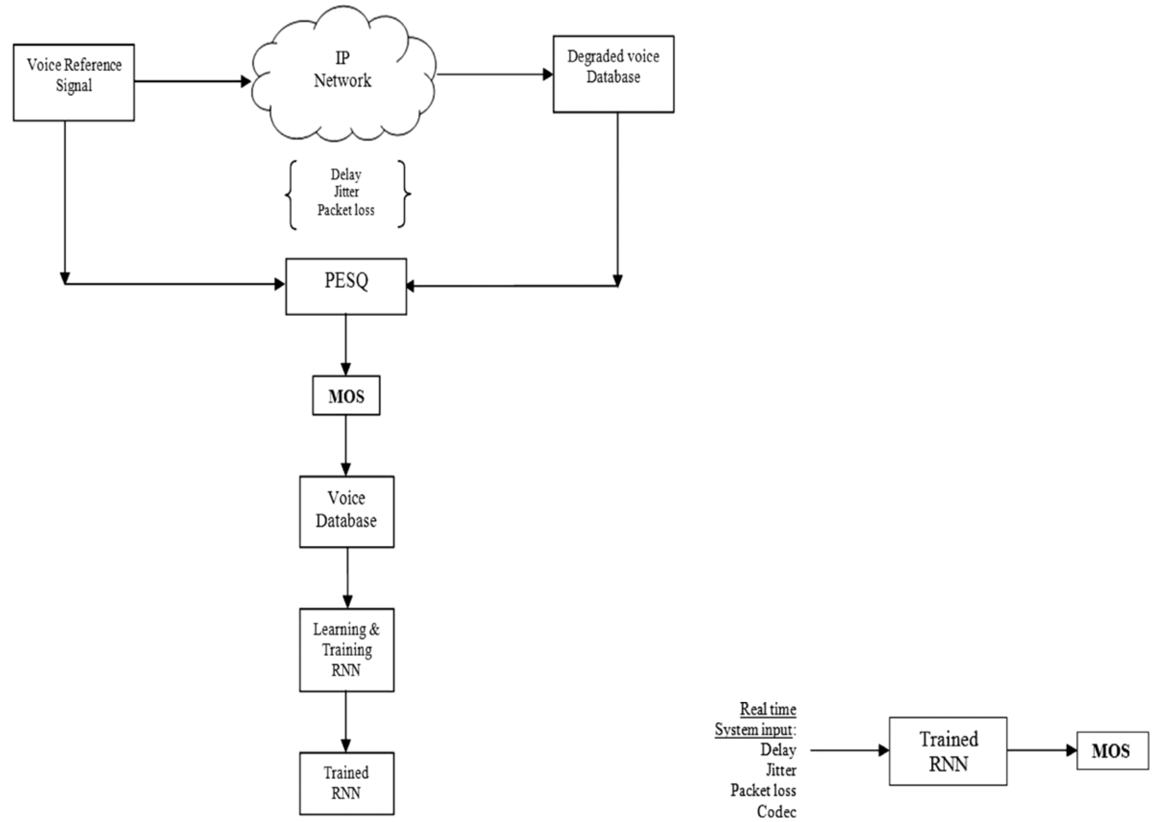


**Fig. 1.** Lab setup.

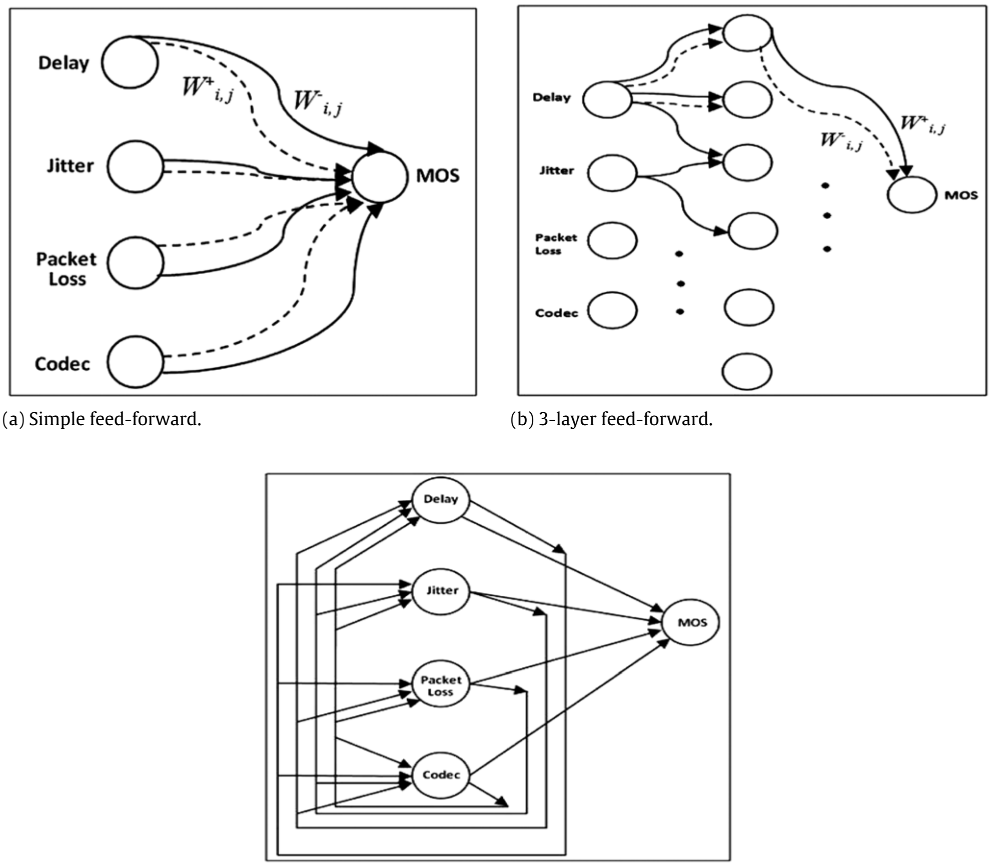


(a) Full training model.

(b) RNN experiment model.

**Fig. 2.** Schematic diagram of experiment setup.

The authors used subjective tests to create a degraded voice signal database to train and test the RNN model. The authors used 56 configurations for training the RNN, 56 configurations to test the RNN, and 20 configurations to validate it. In [27], the authors used a RNN model to study real-time video quality on packet networks not voice or audio (the MOS definitions and range are different). In both cases a RNN three-layer feed-forward architecture and subjective tests were used. In [12], a study based on ANN and PESQ was presented. ANNs can be easily over-trained and its processing time is greater than RNN



(a) Simple feed-forward.

(b) 3-layer feed-forward.

(c) Recurrent.

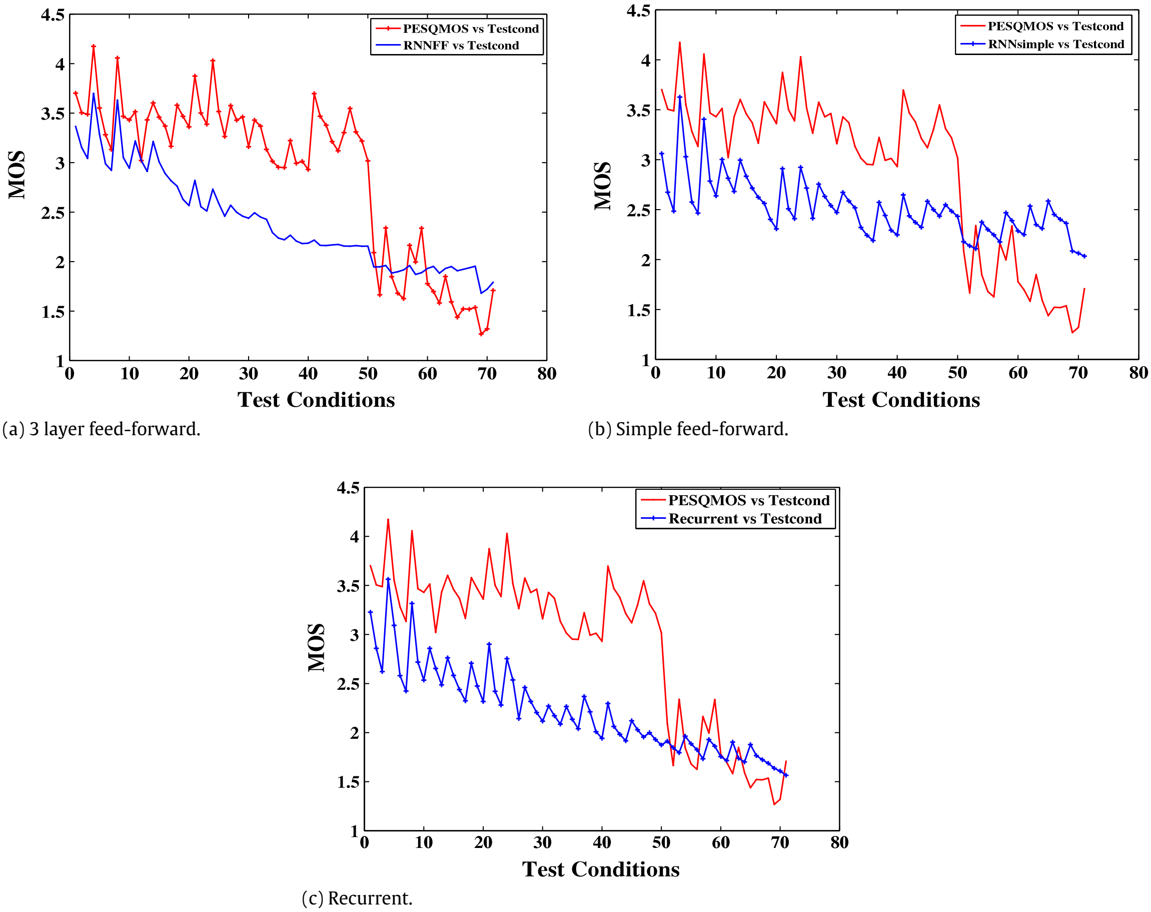
**Fig. 3.** Different RNN architectures implemented in our experiment.

models. In [30], we studied the effect of language on voice quality in addition to delay, jitter, and PL. We used French and Spanish samples to create a database of 125 test configurations for each language. In [28] delay, jitter, packet loss, and codecs were used as input parameters. Our findings showed that one-way transmission delay of more than 200 ms and packet loss of more 10% drastically reduces the voice quality compared to other parameters studied.

The effect of pure delay (without echo) on speech quality in telecommunication networks was presented in [31]. This main focus was on speech quality of user conversation and the authors concluded that a round-trip delay of 500ms makes it more difficult for the users to understand the conversation. The effect of packet loss on three codecs was presented in [32]. Three toll quality coders: Adaptive Differential Pulse Coded Modulation (ADPCM), Sub-band Coding (SBC), and Adaptive Predictive Coding (APC) were studied. Speech degradation due to packet loss on these three coders is more severe than the Pulse Code Modulation (PCM) coder. The authors concluded that the distortion caused by packet loss introduced gaps in the speech sequence and upsets the adaptation logic at the receiving end (at the decoder). The compensation techniques such as packet repetition, waveform substitution, and pitch replication aim to replace the missing segment with suitable data and restore the adaptation logic [32].

In [33], three different objective measures were used to accurately predict human perception under typical conditions of VoIP systems. The first type used perceptually weighted distortion measures, the second type used word-error rates output by a continuous speech recognizer, and the last method used was the ITU E-model. The results from [33] showed that the ITU E-model was the best to predict perceived voice quality out of all the three measures. The ITU E-Model is the only method which does not use the reference signal to calculate the voice quality and can be used for real-time applications. However the E-Model was designed as a transmission planning tool and not as a voice quality measurement tool. The E-Model identified several quality affecting parameters which were related to the signal processing field [33]. PL has only been added in the latest revision (2003) of the E-Model and only uniform loss distribution was considered. This does not provide an accurate model on packet loss. E-Model outputs are not as close to MOS outputs than other objective methods. The E-Model is an unsuitable method to calculate voice quality for IP networks and parameters considered are limited [13].

In [34] a software based method to achieve real-time adaptive video compression to maintain the video quality of decompressed images using the RNN was presented. The method proposed a simple motion detection method to determine whether a portion of the image needs to be transmitted. If the image needed to be transmitted, then a set of learning neural



(a) 3 layer feed-forward.

(b) Simple feed-forward.

(c) Recurrent.

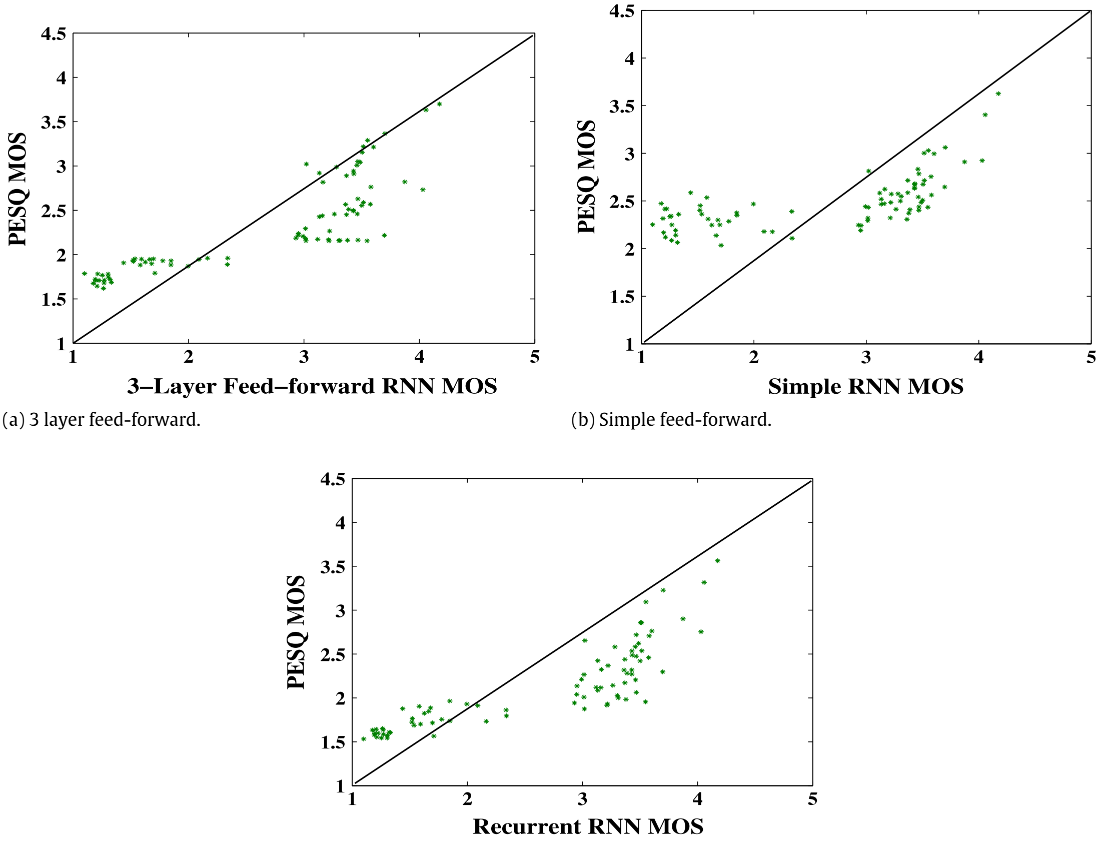
**Fig. 4.** Comparison of PESQ and RNN MOS for G711a codec.

networks were used to compress and decompress the image. In addition to cell loss rates, delay, and jitter, the adaptive compression algorithm also has an impact on the visual quality of the decompressed video sequence. The compression level was adaptively chosen to meet an image quality level $Q$ which was specified by the user. To modify compression levels and to achieve better image quality the sensitivity $d$ of the motion detector can also be varied. This method was very fast and implemented in real-time [34].

A similar study to [34] was presented in [35] the authors developed a RNN based method which used an adaptive algorithm approach based on user desired video quality $Q$, and for moving gray scale images it achieves a compression ratio of up to 500:1 based on combination of motion detection, compression, and temporal subsampling of frames. The compression is achieved by using a combination of motion detection, neural networks, and temporal subsampling of frames. The desired compression of each picture block was adaptively selected by a set of neural networks as a function of reconstruction quality [35]. In [36], the authors used a RNN model to extract precise morphometric information from Magnetic Resonance Imaging (MRI) scan of the human brain. The imaging methods investigated by the authors could lead to automated techniques to identify and quantify cortical regions of brain. Furthermore the authors concluded that the neural network based model could result in novel methods for processing and interpreting functional MRI which would lead to software tools which could be used by radiologists, clinicians, and medical researchers. This would help in understanding of the neuroanatomical substrate of important disorders, as well as of normal brain development [36].

In [37] the authors examined RNNs function approximation properties. The authors considered feed-forward Bipolar RNN (BRNN) [38] which had both positive and negative neurons in the output layer and proved that BRNN was a universal function approximator. The neural networks' ability to use the information to learn the similar but non-identical input–output mapping under novel circumstances makes it a good approximator [37].

From the literature, currently available objective voice quality measurement methods either need reference signals or complex computations to predict voice quality. This makes all the currently available methods unsuitable for real-time applications like VoIP. The properties of RNNs make our model computationally fast, accurate, and suitable for real-time applications to predict and measure voice quality.



(a) 3 layer feed-forward.

(b) Simple feed-forward.

(c) Recurrent.

**Fig. 5.** Comparison of PESQ and RNN MOS for G711a codec.

## 3. Random neural networks model

The RNN is a network of $N$ fully connected neurons. In a RNN, the signals between neurons circulate in the form of positive and negative impulses with unit amplitude. The positive signals represent excitation and the negative signals represent inhibition. At any time $t$, the potential of neuron $i$ is denoted by $k_i(t)$ which is a non-integer. The neuron $i$ may receive signals from another neuron within the network or from outside the network. The neuron $i$ may fire if its potential is positive, reduce its potential by $-1$, and have no effect if it is zero. The neuron receiving the signal adds $+1$ to its potential. The positive and negative external arrivals are poison processes with rates $\Lambda_i$ and $\lambda_i$ respectively. When a neuron's potential is positive it is referred as in an excited state. When a positive signal leaves neuron $i$ it heads to neuron $j$ with probability of $p_{ij}^{+}$, negative signals leave with probability of $p_{ij}^{-}$, and the signal departs from the network with probability $d_i$

$$d_i + \sum_{j=1}^{N} \left(p_{ij}^{+} + p_{ij}^{-}\right) = 1 \tag{1}$$

$$p_{ij} = p_{ij}^{+} + p_{ij}^{-}. \tag{2}$$

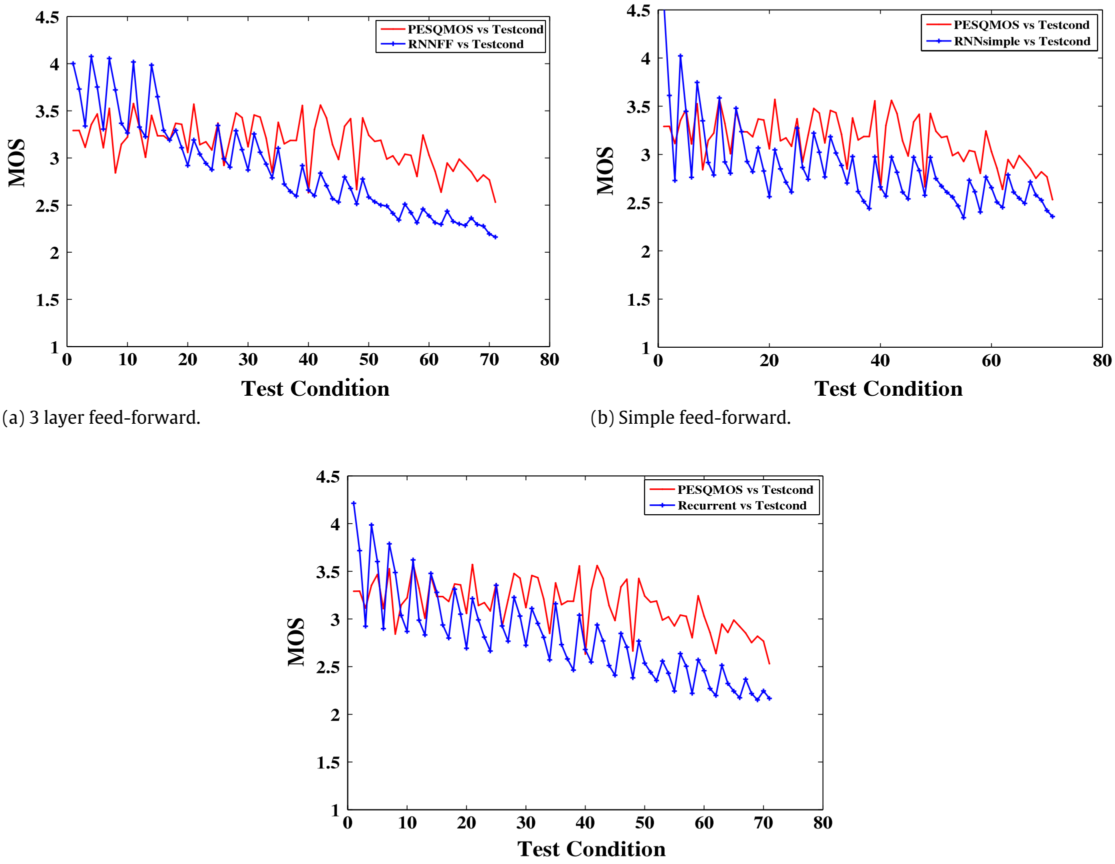
Signals leaving a neuron are not allowed to return directly back to same neuron $p_{ii} = 0$. The rates neuron $i$ fires positive and negative signals when it is excited are denoted as

$$w_{ij}^{+} = r_i p_{ij}^{+} \geq 0 \tag{3}$$

$$w_{ij}^{-} = r_i p_{ij}^{-} \geq 0. \tag{4}$$

The Poisson firing rate with exponential distributed interim pulse interval is

$$r_i = \sum_{j=1}^{N} \left(w_{ji}^{+} + w_{ji}^{-}\right). \tag{5}$$



(a) 3 layer feed-forward.

(b) Simple feed-forward.

(c) Recurrent.

**Fig. 6.** Comparison of PESQ and RNN MOS for iLBC codec.

### *3.1. Steady state network behavior*

At time $t$, the network state is described by the vector of $k(t) = k_1(t), \dots, k_N(t)$ the stationary probability distribution $p(k)$ is denoted by

$$p(k) = \lim_{t\to\infty} \text{prob}[k(t) = k]. \tag{6}$$

Using Theorem 1 in [15], the quantity $q_i$ can be given by

$$q_i = \frac{\lambda_i^+}{r_i + \lambda_i^-}. \tag{7}$$

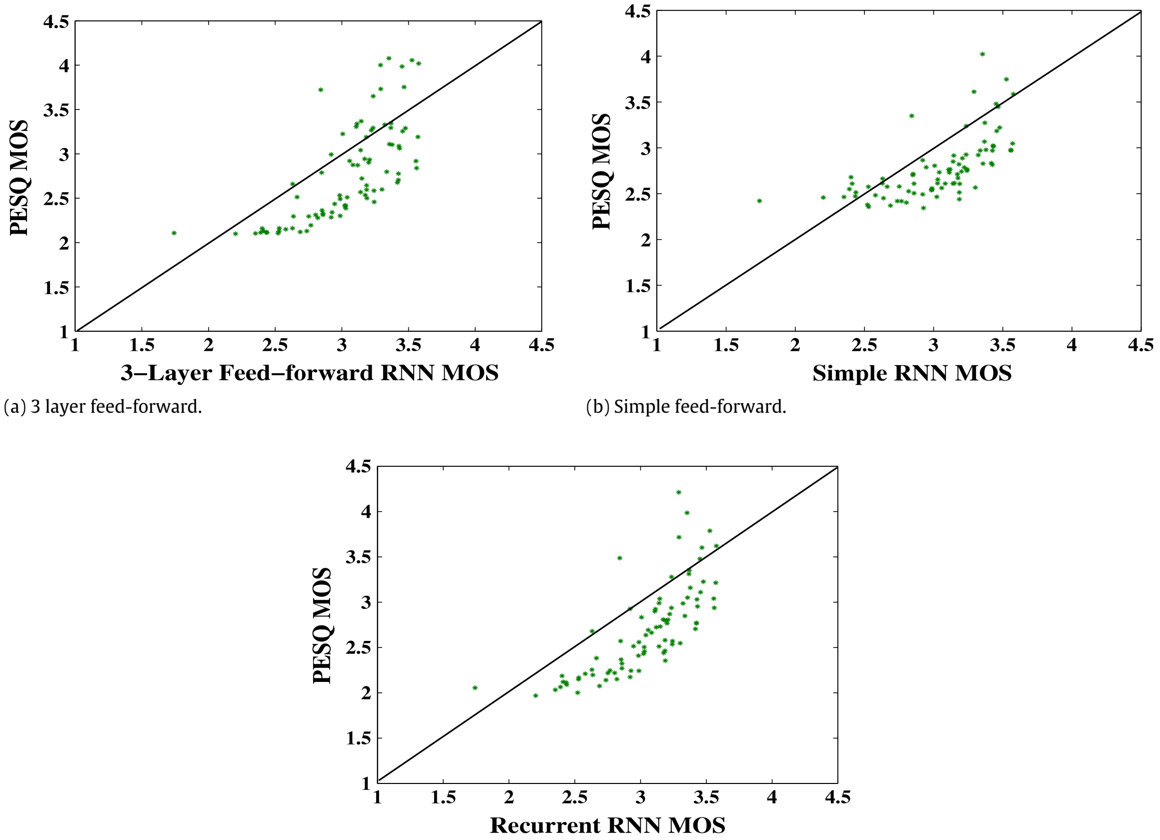
Where the positive $\lambda_i^+$ and negative $\lambda_i^-$ total arrival rates for $i = 1, \dots, N$ satisfy the system of nonlinear simultaneous equations:

$$\lambda_i^+ = \Lambda_i + \sum_{j=1}^{N} q_i r_i p_{ji}^+ \tag{8}$$

$$\lambda_i^- = \lambda_i + \sum_{j=1}^{N} q_i r_i p_{ji}^- . \tag{9}$$

Where,

$$0 \le q_i = \frac{\Lambda_i + \sum_{j=1}^{N} q_i r_i p_{ji}^+}{r_i + \lambda_i + \sum_{j=1}^{N} q_i r_i p_{ji}^-} \le 1. \tag{10}$$



(a) 3 layer feed-forward.

(b) Simple feed-forward.

(c) Recurrent.

**Fig. 7.** Scatter plot comparison of PESQ and RNN MOS for iLBC codec.

If a non-negative solution $\lambda_i^+$, $\lambda_i^-$ exists for the equations (Eqs. (8)–(10)) such that $q_i \leq 1$, then

$$p(k) = \prod_{i=1}^{N} [1 - q_i] q_i^{k_i}. \tag{11}$$

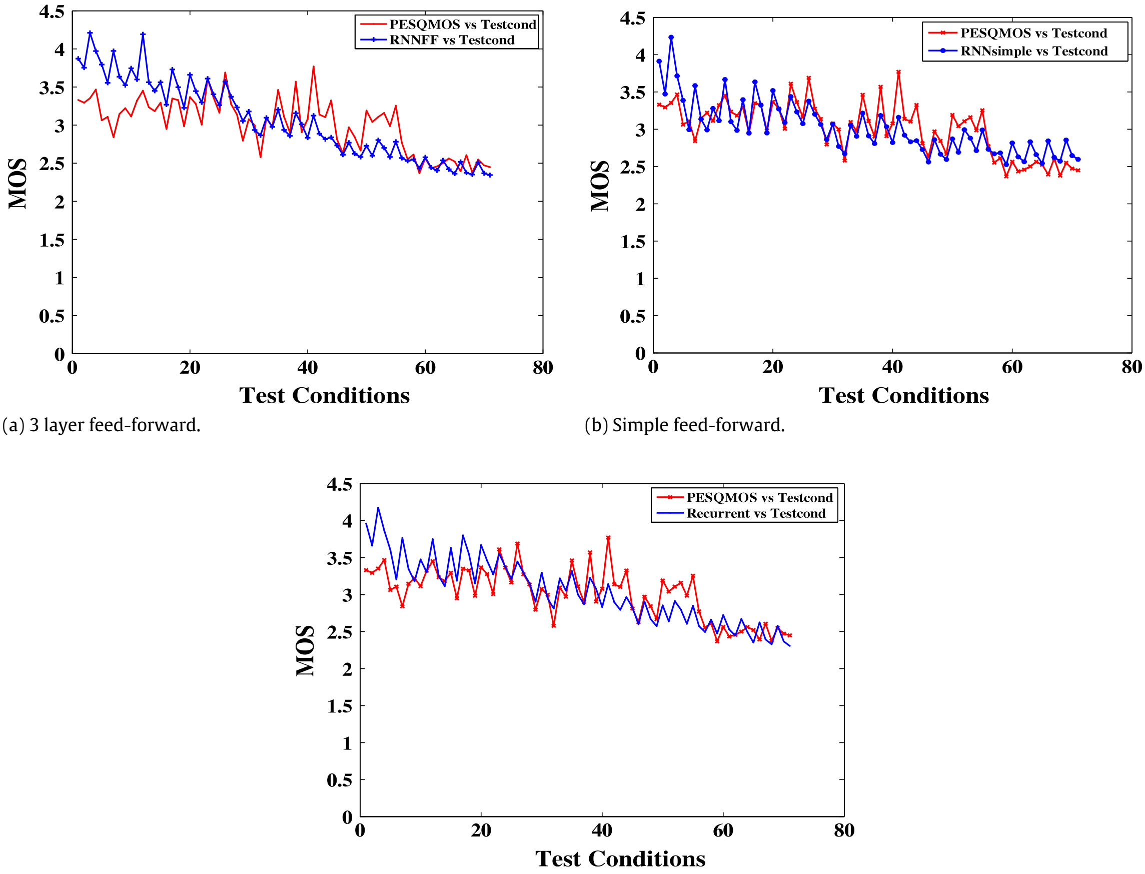
The network is stable which guarantees the excitation level of each neuron remains finite with probability 1. We can simply calculate the average potential at a neuron $i$ as $A_i = \frac{q_i}{(1-q_i)}$. If $q_i = \frac{\lambda_i^+}{r_i + \lambda_i^-} > 0$ this means neuron $i$ is saturated or unstable and the neuron $i$ is continuously firing in steady state.

## 4. Factors affecting voice quality in packet networks

Voice quality is affected by many parameters including delay, jitter, PL, and codec. Delay is the average time taken by a packet to reach its destination from the source. ITU-T G.114 [39] recommends 0–150 ms range for one way delay which is acceptable for most of the user applications. A delay of 150–400 ms is acceptable if the network administrators are aware of transmission time and the impact on quality of user applications. Delay in IP networks occurs due to many processes such as encoding, packet digitization, transmission delay, and playout delay at receiver end. Except transmission delay all other factors can be controlled at the user end [40]. We have used five different values in our experiments for delay: 0, 50, 100, 150, and 200 ms.

Jitter or delay variation can be created by queuing delays on the WAN links across the network. Jitter can also be caused by packets carrying voice signals from the same conversation taking different paths or different queues through the network. Jitter should be removed or reduced before replaying at the user end. De-jitter buffers can be used at the receiving end router or gateway. If delay variation exceeds the de-jitter buffer size, the packet has to wait too long which results in the same effect as packet loss. We have used five different values in our experiments for jitter: 0, 5, 7, 10, and 15 ms.

Packet loss is the percentage of undelivered packets in the network which severely degrades the voice quality. In IP networks PL occurs due to many reasons such as link failure, buffer overflow, high network traffic congestion, ethernet problems, and occasional misrouted packets. Use of Packet Loss Concealment (PLC) reduces the effect of packet loss or



(a) 3 layer feed-forward.

(b) Simple feed-forward.

(c) Recurrent.

**Fig. 8.** Comparison of PESQ and RNN MOS for Speex codec.

discarded packets. The packet loss characteristics of an IP network are bursty [41] and PLC would work effectively only for consecutive lost packets. In our experiments we have chosen 5 different values for packet loss: 0%, 3%, 5%, 7%, and 10%.

In VoIP, codecs are used for digitizing speech signals. For many years, ITU G.711 has been the standard for PSTN networks. In packet network communication, many new codecs such as G729, AMR, iLBC, and Speex have been developed which are currently being used.

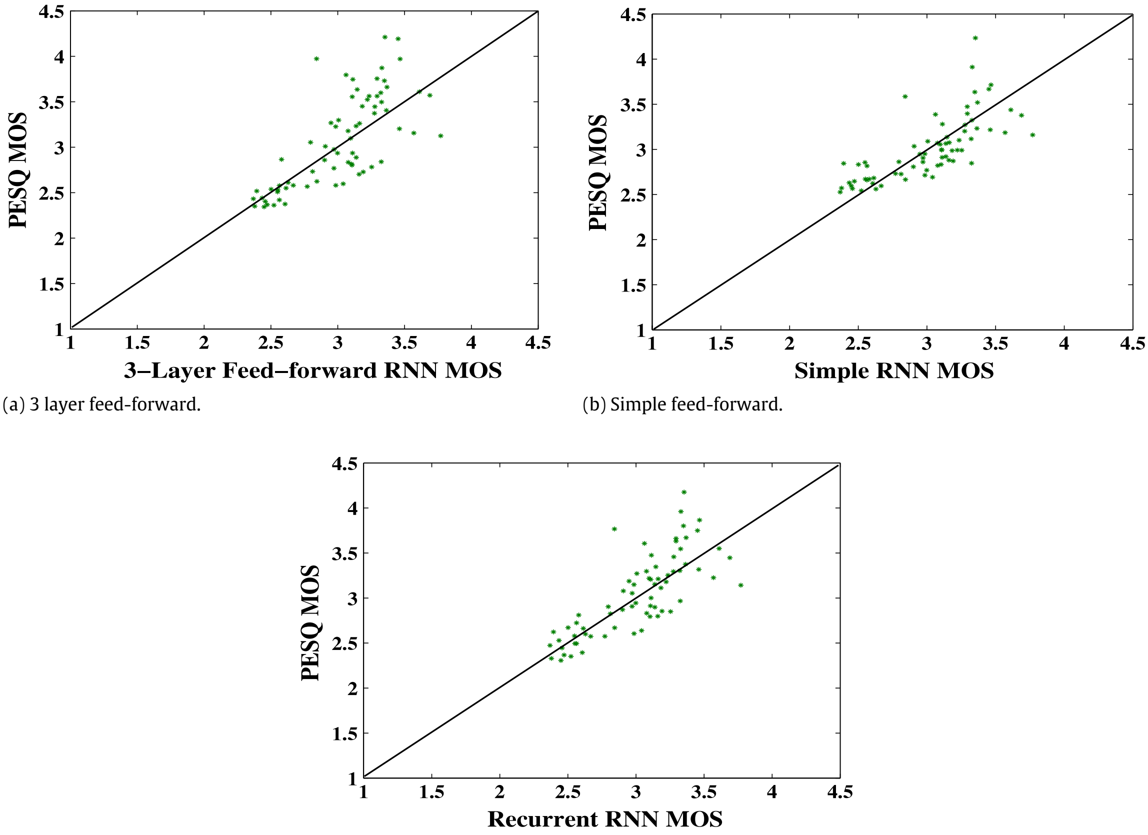
ITU G.711 (A-law) is the E1 standard used by most of the world (except North America and Japan) and is a high bit rate Pulse Code Modulation (PCM) codec which uses the sampling rate of 8 kbps. G711a algorithm uses logarithmic companding which can compress 16 bit audio samples to 8 bits. Logarithmic compression matches the audio quality exactly to the same way the human ear does. G.711 only loses information which cannot be processed by the human brain and gives good quality results for audio with uniform signal to noise ratio (SNR) [42]. G711 provides good voice quality as no compression is used and it is the same codec used in PSTN networks.

The iLBC is a narrow band speech codec which supports two bit rates. The first bit rate is 15.2 kbps and has an encoding frame length of 20 ms. The second bit rate is 13.3 kbps which has an encoding frame length of 30 ms. The iLBC uses block-independent linear predictive coding algorithm and is suitable for robust voice communications over packet networks [43].

The Speex codec is an open-source codec specially developed for packet networks and VoIP applications. The Speex codec was designed to be very flexible and robust to packet loss. Speex supports a wide range of speech quality and bit rates. The Speex codec has been based on Code Excited Linear Prediction (CELP) which has proven to be very reliable and can scale very well with both low bit-rates (4.8 kbps) and high bit-rates (16 kbps). Speex supports three different sampling rates narrowband (8 kHz), wideband (16 kHz), and ultra-wideband (32 kHz) [44].

## 5. Experiments

In this section we discuss the lab experiment setup we used in our study. Our experiments was divided two parts, the first one is to create a voice sample database. The second is to use the database to train RNNs and test the RNN models.



(a) 3 layer feed-forward.

(b) Simple feed-forward.

(c) Recurrent.

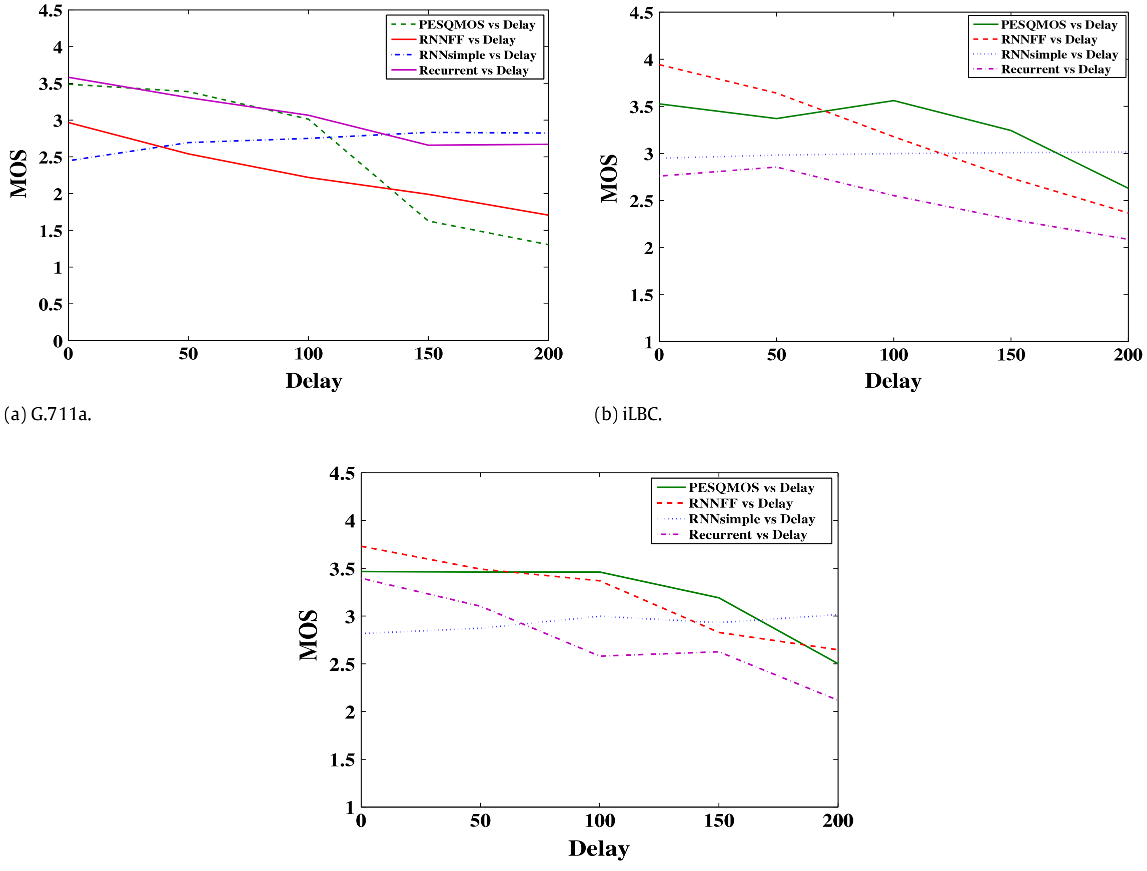
**Fig. 9.** Scatter plot comparison of PESQ and RNN MOS for Speex codec.

To create a degraded voice database, we used the lab setup shown in Fig. 1 and voice samples from ITU-T PESQ P.862 [6]. We have used WANem emulation software [45] to emulate the WAN traffic and X-Lite IP softphone [46] was used to establish VoIP calls. 3CX phone system [47] was used as the SIP server to provide VoIP services to the network.

We have used delay, jitter, and PL as IP network parameters with the values specified in Section 4. Codecs (G711a, iLBC and Speex) were used as non-IP network parameters. Using the lab setup we created original and degraded voice databases. PESQ developed in MATLAB [48] was used to calculate the MOS of the samples. One third of the database was used to train the RNN and rest was used to test our RNN models. Fig. 2 shows the schematic diagram of whole experiment process.

In our research we have tested three architectures: 1. three layer feed-forward, 2. simple feed-forward, and 3. recurrent architecture. Fig. 3(a) shows the simple feed-forward model which consists of an input layer with four neurons and output layer with one neuron. All input neurons are directly connected to an output neuron. Fig. 3(b) shows the three layer feed-forward architecture which consists of an input layer with 4 neurons, a hidden layer with 6 neurons, and an output layer with one neuron. The recurrent architecture implemented in our research is shown in Fig. 3(c) It consists of 5 neurons, all connected to each other.

We used RNN's with 4 input neurons (delay, jitter, PL, and codec) and one output neuron (MOS). The RNN training process is a very important and a complex process, under-trained or over-trained RNN network models will compute wrong results. To implement RNN we used RNNsimv.2 developed by Abdelbaki [19] in 1999 and the Gradient Descent training algorithm was used as the training algorithm. We created 375 data samples of test configurations to train and test the RNN models. We randomly choose one third of this configuration to train the RNNs and the rest was used to test and validate the RNN models. The MOS values calculated from PESQ algorithm were used to validate the RNN model output. The data samples were used to build a database of sample parameters (i.e. delay, jitter, PL, and codec) and MOS values. The original and distorted voice signals were also saved in the form of WAV files for reference. In the RNN simulator the input values (for the input neurons) were enumerated based on the number of different values possible (parameter values) and the MOS (output neuron) were assigned a value $\in (0, 1)$ which is then multiplied by 10 to give us the real MOS value $\in (0, 5)$ further details can be found in [19].



(a) G.711a.

(b) iLBC.

(c) Speex.

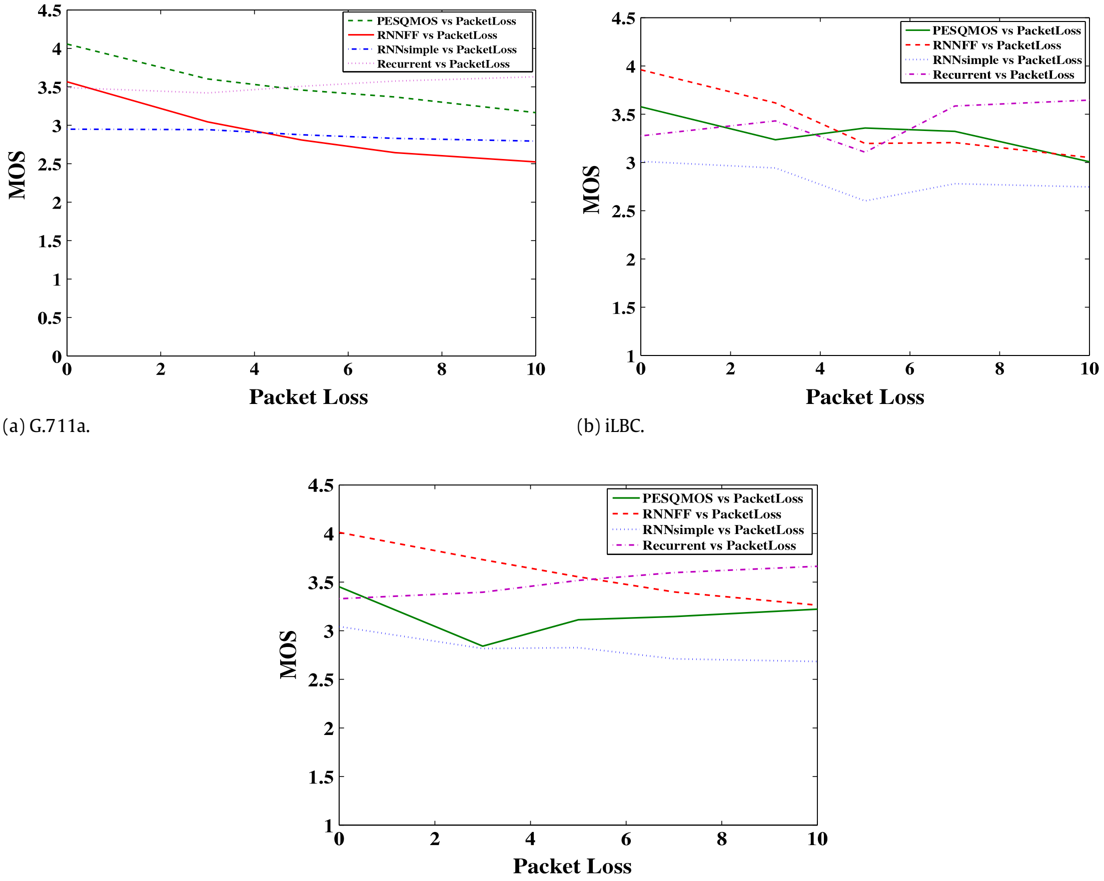
**Fig. 10.** Comparison of Delay and MOS.

## 6. Discussion

In this section, we will present a detailed analysis of results we obtained from our experiments. We studied the impact of delay, jitter, PL, and codecs on perceived voice quality. We compared the MOS results obtained from the objective method PESQ to our RNN models. The Figs. 6–9 show the comparison of outputs from all three RNN models we implemented to PESQ MOS. As we expected all three architectures we tested produced slightly different results. The Mean Square Error (MSE) of 0.0008 for the three layer feed-forward architecture is slightly better than 0.0024 for simple feed-forward and 0.0011 for the recurrent architecture model. On the other hand, the overall performance of the simple feed-forward architecture results are closer to PESQ MOS values compared to other architectures we tested. Another interesting finding in this research was the effect of a non-IP parameter (codec) on voice quality. G711a has been used as the standard codec for PSTN networks. Its performance in the IP networks was slightly different. G711a could not provide "toll-quality" even when the IP networks parameters were kept to a minimum (0). The other codecs we used in this experiment, iLBC and Speex were more robust to IP network changes and outperformed G711a. The introduction of codec as an input to the RNN models produced much closer results compared to our previous works [30,28] in terms of MSE.

Fig. 4 compares the results from our RNN models to PESQ MOS for codec G711a and Fig. 5 shows the scatter plot. The simple feed-forward architecture results are closer to the PESQ MOS compared to the other two architectures we tested. The RNN models underestimate for some values which can be seen in the graphs. This is due to the property of the gradient descent training algorithm used in our RNN models.

Fig. 6 shows the comparison of RNN results to PESQ MOS for iLBC codec and the scatter plot is shown Fig. 7. The results of Speex codec are shown in Figs. 8 and 9. From the results we found that using iLBC and Speex for VoIP networks improves the voice quality. In contrast to G711a, iLBC and Speex were more robust to variations of PL or network delay. This provided better voice quality even when the network parameters were kept at max (10% for PL and 150 ms for Delay). Our study also proved the importance of keeping the IP network parameter values under control (low) and using the right codec to provide better quality of experience to the users. The results showed that the RNN models had the ability to learn quickly from the changes in input data and proved to be more efficient and accurate.



(a) G.711a.

(b) iLBC.

(c) Speex.

**Fig. 11.** Comparison of Packet Loss and MOS.

We will look at the effect of delay and jitter now. Fig. 10 shows the effect of delay on perceived voice quality using different codecs. All codecs introduced some amount of delay in transmission due to frame processing. Our tests also proved that delay variation had very little or no effect on voice quality when delay was kept constant and for this reason we decided to neglect jitter as a main factor. For delay we have used five different values of 0, 50, 100, 150, and 200 ms. From the results we concluded that each codec responded differently to the test conditions we used. The use of different codecs in our research had more impact on voice quality, than we anticipated. The MOS for G711a codec dropped drastically below 2 when the delay was more than ITU-T recommended value of 150 ms. On the other hand iLBC and Speex were more robust to network delay and provided better voice quality at the user end. From our results it was evident that when the one-way delay exceeded its max ITU-T recommendation of 150 ms, the voice quality reduced below the acceptable level of 3.

Finally we will see how the PL affects voice quality. Fig. 11 shows the effect of PL on voice quality. Packet loss is unavoidable in packet networks. Delay plus jitter adds to PL, as the packets arriving too late will simply be dropped. We can see that the MOS drops when the packet loss increases. The overall voice quality for 0%–10% packet loss drops just 0.89 for G711a, 0.57 for iLBC, and 0.23 for Speex. G711a provided high voice quality when the network parameters were low and the voice quality degraded quickly when these were high. On the other hand, iLBC and Speex which were specifically designed for packet networks provided better voice quality compared to G711a. Both the codecs were robust to changes in IP network parameters. From the results it was evident that G711a performed poorly in VoIP networks. From our results, we concluded that by keeping the network delay below the ITU-T recommended value and keeping the packet loss constant better voice quality would be provided to the users.

## 7. Conclusions

In this paper we have presented our method to measure and monitor user perceived voice quality using Random Neural Network models. In this study, we have used both IP network parameters (delay, jitter, and packet loss) and a non-IP parameter (codec) as voice quality affecting parameters to test their impact on transmitted voice quality. We have tested and compared results from PESQ, simple feed-forward, three layer feed-forward with 6 hidden layer neurons, and a recurrent architecture. To validate our RNN model results we have used MOS output from an intrusive objective speech quality evaluation algorithm PESQ. The outputs from our RNN models and the PESQ MOS correlate well. This leads us to conclude with confidence that our model is accurate and useful for real-time measurements. Our method is a non-intrusive and different from existing QoS measuring methods, which can be used in real-time for both voice and video traffic. The perceived voice quality can be improved to match "toll-quality" if delay and packet loss are kept at low values with the right codec. Future research in this area may include adding some other voice quality affecting parameters and testing different training algorithms for the RNN model.

## References

[1] M. Manousos, S. Apostolacos, I. Grammatikakis, D. Mexis, D. Kagklis, E. Sykas, Voice-quality monitoring and control for VoIP, IEEE Internet Comput. 9 (4) (2005) 35–42.

[2] P. Louis, Voice over internet protocol (VoIP): the "killer" application?, 2004. www.mindcommerce.com.

[3] B. Goode, Voice over internet protocol, Proc. IEEE 90 (9) (2002) 1495–1517.

[4] P.800, Methods for subjective determination of transmission quality, ITU-T Recommendations P.800, 1996.

[5] P.861, Objective quality measurement of telephone-band (300–3400 Hz) speech codecs, ITU-T Recommendations P. 861, 1998.

[6] P. 862, Perceptual evaluation of speech quality (PESQ): an objective method for end-to-end speech quality assessment of narrow-band telephone networks and speech codecs, ITU-T Recommendations P. 862, 2001.

[7] G.107, The E-model, A computational model for use in transmission planning, ITU-T Recommendations, 2008.

[8] Y. Bai, M. Ito, A study for providing better quality of service to VoIP users, in: 20th International Conference on Advanced Information Networking and Applications — Vol. 1, AINA'06, Vienna, Austria, pp. 799–804.

[9] L. Sun, E. Ifeachor, New models for perceived voice quality prediction and their applications in playout buffer optimization for voip networks, in: IEEE International Conference on Communications, ICC'04, 3, Paris, France, pp. 1478–1483.

[10] D. Kim, A. Tarraf, Perceptual model for non-intrusive speech quality assessment, in: IEEE International Conference on Acoustics, Speech, and Signal Processing, ICASSP'04, Montreal, Canada, pp. 1060–1063.

[11] A. Markopoulou, F. Tobagi, M. Karam, Assessing the quality of voice communications over internet backbones, IEEE/ACM Trans. Netw. 11 (5) (2003) 747–760.

[12] L. Sun, Speech quality prediction for voice over internet Protocol networks, Ph.D. Thesis, Univ. of Plymouth, Plymouth, UK, 2004.

[13] S. Mohamed, F. Cervantes-Pérez, H. Afifi, Audio quality assessment in packet networks: an "inter-subjective" neural network model, in: 15th International Conference on Information Networking, Beppu, Japan, pp. 579–586.

[14] K. Bakircioğlu, T. Koçak, Survey of random neural network applications, Eur J. Oper. Res. 126 (2) (2000) 319–330.

[15] E. Gelenbe, Random neural networks with negative and positive signals and product form solution, Neural Comput. 1 (4) (1989) 502–510.

[16] E. Gelenbe, Stability of the random neural network model, Neural Comput. 2 (2) (1990) 239–247.

[17] E. Gelenbe, K. Hussain, Learning in the multiple class random neural network, IEEE Trans. Neural Netw. 13 (6) (2002) 1257–1267.

[18] H.M. Abdelbaki, E. Gelenbe, S.E. El-Khamy, Analog hardware implementation of the random neural network model, in: IEEE/INNS/ENNS International Joint Conference on Neural Networks, IJCNN'00, vol. 4, Como, Italy, pp. 197–201.

[19] H.M. Abdelbaki, Random neural network simulator for use with MATLAB, 1999. http://www.cs.ucf.edu/~ahossam/.

[20] S. Timotheou, The random neural network: A survey, Comp. J. 53 (3) (2010) 251–267.

[21] E. Gelenbe, Sensible decisions based on QoS, Comput. Manag. Sci. 1 (1) (2003) 1–14.

[22] E. Gelenbe, R. Lent, A. Nunez, Self-aware networks and QoS, Proc. IEEE 92 (9) (2004) 1478–1489.

[23] G. Öke, G. Loukas, A denial of service detector based on maximum likelihood detection and the random neural network, Comp. J. 50 (6) (2007) 717–727.

[24] S. Mohamed, G. Rubino, M. Varela, Performance evaluation of real-time speech through a packet network: a random neural networks-based approach, Perform. Eval. 57 (2) (2004) 141–161.

[25] G. Rubino, M. Varela, J. Bonnin, Controlling multimedia QoS in the future home network using the PSQA metric, Comp. J. 49 (2) (2006) 137–156.

[26] A. da Silva, M. Varela, E. de Souza e Silva, R. Leao, G. Rubino, Quality assessment of interactive voice applications, Comput. Netw. 52 (6) (2008) 1179–1192.

[27] S. Mohamed, G. Rubino, A study of real-time packet video quality using random neural networks, IEEE Trans. Circuits Syst. Video Technol. 12 (12) (2002) 1071–1083.

[28] K. Radhakrishnan, H. Larijani, T. Buggy, A non-intrusive method to assess voice quality over internet, in: 2010 International Symposium on Performance Evaluation on Computer and Telecommunication Systems, SPECTS, Ottawa, Canada, pp. 380–386.

[29] H. Larijani, K. Radhakrishnan, Voice quality in VoIP networks based on random neural networks, in: Ninth International Conference on Networks, ICN, 2010, Menuires, France, pp. 89–92.

[30] K. Radhakrishnan, H. Larijani, A study on QoS of VoIP networks: a random neural network (RNN) approach, in: Proceedings of the 2010 Spring Simulation Multiconference, SpringSim'10, Orlando, FL, USA, pp. 1–6.

[31] N. Kitawaki, K. Itoh, Pure delay effects on speech quality in telecommunications, IEEE J. Select. Areas Commun. 9 (4) (1991) 586–593.

[32] A. Choi, A. Constantinides, Effect of packet loss on 3 toll quality speech coders, in: Second IEE National Conference on Telecommunications, York, UK, pp. 380–385.

[33] T. Hall, Objective speech quality measures for internet telephony, in: Preceedings of SPIE Voice over IP VoIP Technology, Denver, CO, USA, pp. 128–136.

[34] E. Gelenbe, M. Sungur, C. Cramer, P. Gelenbe, Traffic and video quality in adaptive neural compression, Multimedia Syst. 4 (6) (1996) 357–369.

[35] C. Cramer, E. Gelenbe, H. Bakircioglu, Low bit rate video compression with neural networks and temporal subsampling, Proc. IEEE 84 (10) (1996) 1529–1543.

[36] E. Gelenbe, Y. Feng, K.R.R. Krishnan, Neural network methods for volumetric magnetic resonance imaging of the human brain, Proc. IEEE 84 (10) (1996) 1488–1496.

[37] E. Gelenbe, Z. Mao, Y. Li, Function approximation with spiked random networks, IEEE Trans. Neural Netw. 10 (1) (1999) 3–9.

[38] E. Gelenbe, A. Stafylopatis, A. Likas, Associative memory operations of the random neural network, in: Int. Conf. Artificial Neural Networks, Espoo, Finland, pp. 307–312.

[39] G.114, One-way transmission time, ITU-T Recommendations, 2003.

[40] Cisco, Understanding delay in packet voice networks, White paper, 2008.

[41] S. Jelassi, H. Youssef, G. Pujolle, Parametric speech quality models for measuring the perceptual effect of network delay jitter, in: IEEE 34th Conference on Local Computer Networks, Zrich, Switzerland, pp. 193–200.

[42] G.711, Pulse code modulation (PCM) of voice frequencies, ITU-T Recommendations, 1993.

[43] S. Andersen, A. Duric, H. Astrom, R. Hagen, W. Kleijn, J. Linden, Internet Low Bitrate Codec, RFC 3951, 2004.

[44] J. Valin, The Speex codec manual Version 1.2 Beta 3, Speex.org, 2007.

[45] TCS, Wanem v2.0, 2008. http://wanem.sourceforge.net/.

[46] Counterpath, X-lite, 2010. http://www.counterpath.com/x-lite.html.

[47] 3CX, Software based PBX, 2010. http://www.3cx.com/.

[48] P. Loizou, PESQ and other objective measures for evaluating quality of speech processed by noise suppression algorithms, 2007. www.utdallas.edu/~loizou/speech/software.htm.

**Kapilan Radhakrishnan** is a Ph.D. candidate with the school of engineering and computing at Glasgow Caledonian University, Glasgow, UK. He received his B.E in computer science and engineering from University of Madras, India in 2002 and M.Sc. in Advanced Computer Networking from Glasgow Caledonian University in 2005. His professional qualifications also include Cisco Certified Network Associate (CCNA) since 2003 and Cisco Certified Academy Instructor (CCAI) since 2007. His current research interest is focused on performance evaluation of VoIP networks and Routing in IP networks.

**Hadi Larijani** received his Ph.D. in computer Science from Heriot-Watt Univ. Edinburgh, UK in 2006. He is a senior lecturer in the Dept. of CNEE in the School of Engineering and Computing at Glasgow Caledonian University. He was the principal investigator of a major grant with IBM as an industrial partner developing Virtual Call Center agents. He has worked with 3Com Europe, is a Cisco Certified Network Professional academy Instructor. He has several patents pending and his research interests are: Performance evaluation of computer systems and networks, Computer network simulation, Software engineering, VoIP, Call Center Applications and Intelligent Software Agents.