STABLE SPEECH SYNTHESIS USING RECURRENT RADIAL BASIS FUNCTIONS

Iain Mann and Steve McLaughlin
Dept. of Electronics and Electrical Engineering,
University of Edinburgh, King’s Buildings,
Mayfield Road, Edinburgh. EH9 3JL. UK
Tel: +44 131 6505655; fax: +44 131 6506554
E-mail: inm@ee.ed.ac.uk, sml@ee.ed.ac.uk

ABSTRACT
The problem of stable vowel sound synthesis using a nonlinear free-running recurrent radial basis function (RBF) neural network is addressed. Voiced speech production is modelled as the output of a nonlinear dynamical system, rather than the conventional linear source–filter approach, which, given the nonlinear nature of speech, is expected to produce more natural-sounding synthetic speech. Our RBF network has the centre positions fixed on a hyper-lattice, so only the linear–in–the–parameters weights need to be learnt for each vowel realisation. This leads to greatly reduced complexity without degrading performance. The proposed structure, in which regularisation theory is used in learning the weights, is demonstrated to be stable when functioning in recurrent mode with no external input, correctly producing the desired vowel sound.

1 INTRODUCTION
In this paper we describe a neural network approach to the synthesis of voiced speech sounds. Our new technique uses a recurrent radial basis function with only a relatively small number of varying parameters and, once trained, is able to function as a stable nonlinear oscillator with no external input.

Nonlinear techniques, generally in the form of neural networks, have been applied to speech synthesis previously by several researchers. The interest in using a nonlinear method, as opposed to conventional linear ones, stems from the fact that speech has been demonstrated to be a nonlinear process [1]. Therefore a nonlinear technique should be better able to model the signal.

An RBF network for generating vowel sounds was proposed in [2]. It uses an extended Kalman filter approach to learn the weights, with centres that are chosen as a subset of the input data. This structure, although sometimes able to resynthesise the desired sound, is generally very unstable. Further work [3] solved this problem by blocking the input signal into frames and using all of the data within each frame as centres. This method is able to resynthesise on a frame–by–frame basis, but is unsatisfactory due to the very large number of varying parameters and the fact that it is not free–running synthesis per se. A different approach, using multi–layer perceptrons (MLP’s), rather than RBF’s, is taken in [4]. Only the Japanese vowel /a/ is examined, but it appears that stable resynthesis is achieved. However the learning process in MLP’s is difficult and costly, and there will be a large number of nonlinear parameters varying between different vowel realisations. At the same time, work on modelling radar back scatter from the sea has produced free–running RBF synthesis structures [5] which make use of regularisation theory to ensure stability. However it is not clear if this technique uses all of the training data as centres. Our new method draws on this application of regularisation theory to produce a stable free–running synthesiser of vowel sounds. It also uses only a small number of centres, whose positions are fixed, thus greatly reducing complexity.

We first present the structure of the recurrent RBF network and describe its operation. We then consider issues relating to the stability of the network, and show how regularisation theory allows us to create a completely stable structure. Following this some results of synthesised speech are presented, and we conclude with a discussion of the problems and possibilities found with our new technique.

2 RADIAL BASIS FUNCTION NETWORK
The radial basis function network is used to approximate the underlying nonlinear system producing a particular stationary voiced sound. Specifically, it is trained to perform the prediction

\[ y_{i+1} = \mathcal{F}\{y_i\} \] (1)

where \( y_i = \{ y_{i-\tau}, y_{i-(\tau-1)}, \ldots, y_{i-(m-1)\tau} \} \) is a vector of previous inputs spaced by some delay \( \tau \) samples, and \( \mathcal{F} \) is a nonlinear mapping function. Considering the problem from a nonlinear dynamical theory perspective, we can view this as a time delay embedding of the one dimensional speech signal into an \( m \)–dimensional state space, producing an attractor reconstruction of the underlying \( d \)–dimensional system. Takens theorem states that \( m \geq 2d + 1 \) for an adequate embedding [6], although in practice it is often possible to reduce \( m \). Appropriate choice of the non–trivial delay \( \tau \) opens up this reconstruction. It
has been shown that vowels are low dimensional, having a
dimension of around three [7], and that a value of \( \tau = 10 \)
samples is suitable for a sampling rate of 22.05 kHz [8].
In our RBF, \( m = 6 \) taps are used, which is sufficient to
give good results. The general structure is shown in Figure
1, for both training and synthesis modes of operation.
The centres are spread uniformly over an \( m \)-dimensional
hyper–lattice, rather than being chosen as a subset of the
input data as is more usually the case. The network output
(using \( P \) Gaussian centres) is given by
\[
\hat{y}_{i+1} = \sum_{j=0}^{P-1} \phi_{ij} w_j
\]  
(2)
where the output of each centre is
\[
\phi_{ij} = \exp \left( -\frac{||y_i - c_j||^2}{2\sigma^2} \right)
\]  
(3)
c_k being the position of the \( k \)-th centre on the hyper–
lattice, and all the centres having a bandwidth \( \sigma^2 \) (determined experimentally). Each \( k \)-th centre output is multi
plied by a weight \( w_k \). With the centre positions known,
the determination of the weights is linear in the parameters
and may be solved by standard matrix inversion techniques, subject to some stability issues which are discussed in the following section.

Once the weights have been determined, we can switch from training to synthesis mode: the input signal is removed and the delayed network output is fed back into the delay line to form a free–running recurrent RBF network, allowing any duration of signal to be synthesised. Since the centre positions are fixed on the hyper–lattice identically for all signals, this means that there are only \( P \) varying parameters in our system (i.e. the elements of \( w \)). This greatly reduces the complexity of the system without reducing performance. Figure 2 shows a comparison of mean square prediction error for a regularised RBF network trained with centres on a hyper–lattice and for one where the centres are chosen as a subset of the training data, over a database of four speakers (two male and two female). Very similar performance is observed for both methods, although in all cases our method gives on average approximately 1 dB better performance, clearly justifying the use of the hyper–lattice.

\begin{figure}[h]
\centering
\includegraphics[width=0.8\textwidth]{figure1.png}
\caption{Recurrent RBF network structure.}
\end{figure}

\begin{figure}[h]
\centering
\includegraphics[width=0.8\textwidth]{figure2.png}
\caption{MSE comparison of centre positioning strategies for 4 speakers, averaged over the 3 vowels /\textipa{a}/, /\textipa{æ}/ and /\textipa{u}/. The RBF network was trained with 800 samples, using 64 centres (either on a 6–dimensional hyper–lattice, or as a subset of the training data). The one–step–ahead MSE was then calculated on an unseen 800 sample validation set.}
\end{figure}

\section{3 Stability Issues}
In this section we will consider how the training technique for learning the weights must be modified in order to achieve stable resynthesis. Conventionally the weights are determined to minimise a sum of squares error function, \( E_s(\mathcal{F}) \), over \( N \) samples of training data:
\[
E_s(\mathcal{F}) = \frac{1}{2} \sum_{i=1}^{N-1} (\hat{y}_i - y_i)^2
\]  
(4)
where \( \hat{y}_i \) is the network approximation of the actual speech signal \( y_i \). Incorporating Equation 2 into the above and differentiating with respect to the weights, then setting the derivative equal to zero gives the least-squares problem [9]. This can be written in matrix form as
\[
(\Phi^T \Phi) \mathbf{w}^T = \Phi^T \mathbf{y}
\]  
(5)
where \( \Phi \) is an \( N \times P \) matrix of the outputs of the centres; \( \mathbf{y} \) is the target vector of length \( N \); and \( \mathbf{w} \) is the \( P \) length vector of weights. This is normally solved by finding the pseudo–inverse by singular value decomposition.

However, our experiments have shown that if the weights are determined in this manner it is not possible to guarantee stable resynthesis when the RBF network is operated in the free–running recurrent mode. In fact it is generally the case that the output is erroneous, with a correct output found only at certain training lengths for certain input signals. More over it does not seem possible to predict which signals will produce a correct output and which will not. Very similar observations are made by Birgmeier [2, 10]. Types of erroneous output encountered include periodic limit cycles not resembling the original speech signal; constant (sometimes zero) output; extremely large spikes at irregular intervals on otherwise correct waveforms. We note here that these observations
hold true irrespective of centre positioning (i.e. as a subset of the training data or as a hyper–lattice), bandwidth and variations of other network parameters.

This problem can be solved by the application of regularisation theory, first described as applied to RBF’s in [11]. In essence this means the application of some smoothing factor to the approximated function $\mathcal{F}$. Mathematically, we define a new cost function

$$\mathcal{E}(\mathcal{F}) = E_s(\mathcal{F}) + \lambda E_r(\mathcal{F}) = E_s(\mathcal{F}) + \frac{1}{2} \lambda \| P \mathcal{F} \|^2$$

(6)

where $E_s(\mathcal{F})$ is the sum of squares error defined previously, $E_r(\mathcal{F})$ is the regularising term and $\lambda$ is the real, positive regularisation parameter. A full derivation is beyond the scope of this paper (see the chapter on regularisation theory in [12]), but given the appropriate choice of the differential operator $P$, it is possible to write the solution for the weights for the specific case of all training data being used as centres ($P = N$) as

$$\mathbf{w} = (\mathbf{G} + \lambda \mathbf{I})^{-1} \mathbf{y}$$

(7)

where $\mathbf{I}$ is the $N \times N$ identity matrix and $\mathbf{G}$ is an $N \times P$ matrix of Green’s functions, which is equal to the $N \times P$ matrix $\Phi$.

This solution is only of limited interest, since we have to use all of the training data as centres, but it can be extended to the approximate regularised solution, where $P < N$ and the centres are not necessarily on the training data. In this case we obtain

$$\mathbf{w} = (\mathbf{G}^T \mathbf{G} + \lambda \mathbf{G}_0) \mathbf{y}$$

(8)

where $\mathbf{G}_0$ is a symmetric $P \times P$ matrix composed of elements $G(c_i; c_j)$:

$$G(c_i; c_j) = \exp \left( -\frac{\| c_i - c_j \|^2}{2\sigma^2} \right); \quad 0 \leq i, j \leq P - 1$$

(9)

When $\lambda = 0$ there is no regularisation, and as $\lambda$ is increased so greater smoothness is imposed onto $\mathcal{F}$. Our results have shown that by using this method, stable realistic resynthesis is possible for all the signals in our database.

### 4 VOWEL RESYNTHESIS

In order to test the ability of our RBF structure to synthesise vowels in a stable and realistic manner, we examined its performance on our specially recorded database of stationary vowels. From the 16 speakers in the database, we randomly chose two male and two female speakers. Their utterances of the three cardinal vowels /i/, /a/ and /u/ form the basis of our test. All signals were sampled at 22.05 kHz and have 16 bit resolution. The RBF network was trained using 800 samples of the original waveform, before being set into free–run synthesis mode to generate the desired length signal. The 6–dimensional hyper–lattice used has 64 centres, corresponding to a centre at each ‘corner’ of the hyper–lattice. The value of the regularisation parameter, $\lambda$, was set at $10^{-1}$. In all cases we were able to synthesise stable, speech–like waveforms. As an assessment of the quality of these synthesised waveforms, they were compared against the signal generated by the classic linear prediction synthesiser. In this case, the linear prediction (LP) filter coefficients were found from the original vowel sound (analogous to the training stage of the RBF network). Using the estimate $\{\tilde{F}_s, + 4\}$ [13], we calculated a model with 26 poles. Then, using the source–filter model, the linear prediction filter was excited by a dirac pulse train to produce the desired length LP synthesised signal. The distance between dirac pulses was set to be equal to the pitch period of the original signal. Figure 3 shows the time domain signals generated by the two techniques for an example of the vowel /u/ (speaker mc). The original vowel sound is also included for comparative purposes. Figure 4 shows the corresponding frequency domain plots of the three signals. For the sake of clarity the spectral envelope (calculated via the LP coefficients), rather than the full Fourier spectrum, is shown.

### 5 DISCUSSION

Figures 3 and 4 clearly demonstrate the different modelling strategies between the LP and RBF approaches. In the linear prediction case, the technique attempts to model the
spectral features of the original, which is done by assuming that the excitation signal can be separated out. Hence the reasonable match seen in the spectral envelope (although the high frequencies have been over-emphasised), but the lack of resemblance in the time domain. The RBF network, on the other hand, attempts to model the complete underlying dynamical system (Equation 1). Thus the RBF synthesised signal resembles the original in the time domain: the RBF network has successfully captured many of the original signal characteristics. However the spectral plots show that the higher frequencies have not been well modelled by this method. This is because the RBF network has missed some of the very fine variations of the original time domain waveform, due to the imposition of the regularisation parameter $\lambda$. This, as described previously, adds smoothness to the approximated function $F$ to ensure stability. When listening to the signals in an informal listening test, most listeners preferred the RBF generated vowel against the LP synthesis result. The RBF signal was generally thought to be perceptually closer to the original in terms of naturalness, although it was still noticeably different.

6 CONCLUSION

We have presented a recurrent radial basis function network for vowel synthesis. By applying regularisation theory we can ensure the stability of the network, which has previously been found to be a problem with this technique. In addition, through the use of a hyper-lattice, the centre positions can be fixed in state space, thus reducing the number of varying parameters to the set of linear weights. This structure has been compared to the well-known source-filter linear model, with an example of the vowel /u/ presented. Our technique produces a more accurate version of the time-domain waveform, but does not adequately model the higher frequencies. This is due to the use of regularisation: by forcing the approximated function to be smooth to give stability, we implicitly remove some higher frequency information. However the vowel generated by the RBF network is more natural-sounding than the LP synthesised sound, justifying the use of a nonlinear method.

ACKNOWLEDGEMENTS

The authors would like to thank B. Mulgrew of Edinburgh University for useful discussions about this work.

References


