Deterministic Nonmonotone Learning of Recurrent Networks for Temporal Sequence Processing

PhD Project
Chun-Cheng Peng
ccpeng@dcs.bbk.ac.uk

PhD Supervisor
Dr. George D. Magoulas
gmagoulas@dcs.bbk.ac.uk

Project Web Site
www.dcs.bbk.ac.uk/~ccpeng

Keywords
Temporal Sequence Processing, Nonmonotone Learning, Recurrent Neural Networks

Project Aims

Our goal is to develop new and efficient deterministic algorithms for recurrent neural networks (RNNs) to tackle temporal sequence processing problems.

Problem

Sequence processing involves several tasks such as clustering, classification, prediction, and transduction of sequential data, which can be symbolic, non-symbolic or mixed. If the content of a sequence is varying through different time steps, the sequence is called *temporal* or *time-series*. In general, a temporal sequence consists of nominal symbols from a particular alphabet, while a time-series sequence deals with continuous, real-valued elements. Processing both these sequences mainly consists of applying the current known patterns to produce or predict the future ones, while a major difficulty is that the range of data dependencies is usually unknown. Therefore, an intelligent system with memorising capability is crucial for effective sequence processing and modelling.

Nonmonotone Learning

In cognitive development, researchers have found that learning processes of humans occurs in a nonmonotone (NM) way, while in unconstrained optimisation nonmonotone strategies have provided several advantages, such as global and superlinear convergence, requiring fewer numbers of line searches and function evaluations, and proved to be effective for large-scale unconstrained problems.

In this work, we propose several NM algorithms for training RNNs on the temporal sequence processing problems that exploit nonmonotone conditions of the following form:

$$E\left(w_k+\beta_k\right) \le \max_{0\le j\le M}\left\{E\left(w_{k-j}\right)\right\} + \delta\cdot\Phi,$$

where E is the function error, w is the vector of weights and biased, β is the stepsize, δ is a constant, Φ is a forcing function.

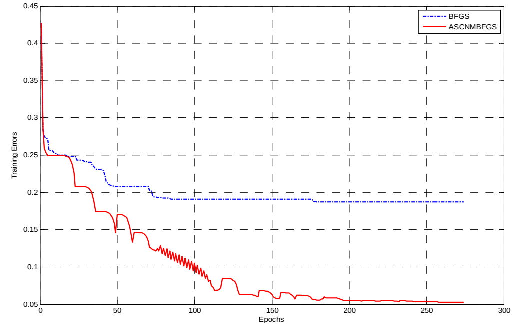

Figure 1. Learning Behaviours of BFGS Methods

When the monotone algorithm (blue-line) partially learns the training data (converges in a local minimum), our algorithm reaches the training goal.

Key Publications

1. C.-C. Peng and G.D. Magoulas, "Sequence Processing with Recurrent Neural Networks", *Encyclopedia of Artificial Intelligence*, forthcoming.
2. C.-C. Peng and G.D. Magoulas, "Adaptive Self-Scaling Non-Monotone BFGS Training Algorithm for Recurrent Neural Networks", *Proc. ICANN'07*, 9-13 September 2007, Porto, Portugal, pp. 259-268, 2007.
3. C.-C. Peng and G.D. Magoulas, "Advanced Adaptive Nonmonotone Conjugate Gradient Training Algorithm for Recurrent Neural Networks", International Journal of Artificial Intelligence Tools, forthcoming.