A Research of Heart Rate Prediction Model based on Evolutionary Neural Network

Feng Xiao, Ming Yuchi*, Mingyue Ding
Key Laboratory of Image Processing and Intelligent Control of Education Ministry of China
Life College of Science and Technology
Huazhong University of Science and Technology
Wuhan, China
m.yuchi@gmail.com

Jun Jo
School of Information and Communication Technology
Griffith University
Queensland, Australia
j.jo@griffith.edu.au

***Abstract*—Heart rate (HR) signal analysis is widely used in the medicine and medical research area. Physical activities (PA) are commonly recognized to greatly affect the changes of heart rate. A method of Evolutionary Neural Network -- Neuro-evolution of Augmenting Topologies (NEAT) is used to build a PA-based HR predictor model. Through special coding, crossover and mutation operator, NEAT can implement network topology and connectivity weights evolution simultaneously. The common problem in evolutionary neural network, like competing conventions, how to protect the new innovation are effectively solved. The experimental results demonstrated the application potential of the approach.**

Keywords-Heart Rate Prediction; Physical Activity; Neural Network; Neuro-evolution of Augmenting Topology

I. INTRODUCTION

Heart Rate (HR), refers to the number of heart beats per minute, is affected by many factors, including Physical Activities (PA), mental state and the surrounding environment. Among these factors, the greatest and most direct one is PA.

The researches on HR and PA were applied in many areas, for example, energy expenditure measurement[1,2], autonomic nervous system assessment[3-5] and sports research[6]. However, Few works have focused on how PA influenced HR: Pawar el al.[7] presented one body movement activity detection system which was based on ECG signal, but not HR. Meijer el al.[8]built a linear relationship between the HR and the body movements, but his experiments were implemented in specific conditions and the body movement was recorded as the counted number of activities, which could not appropriately reflect the actual PA.

HR prediction, is to estimate the HR at next time step based on the current HR. Effective HR prediction can predict and prevent parts of cardiovascular diseases. Currently, most HR prediction is based on the past HR changes, no external factors(such as PA) was taken into account.

The main purpose of this paper is to build a prediction model using the neural network to reflect the effects of PA on the HR. The model was based on the author's previous

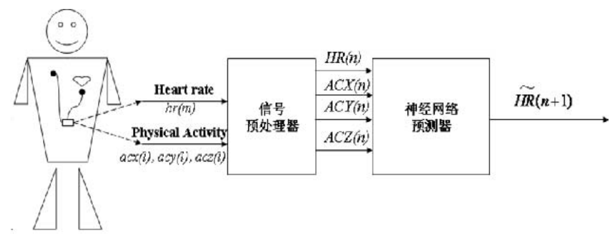


Fig.1 the block diagram of the whole system

work[9]- [11]. In the experiment, the subject was equipped with a portable HR and PA monitor, performing normal daily activities without any special routine or restriction. The recorded HR and PA signals were used as the input of predictor after preprocess, and the output of predictor is the HR at next time step. Feed-forward Neural Network (FFNN) is used as the mathematical model of the predictor. A neural network self-learning method, neuro-evolution of augmenting topology(NEAT) technique [12], [13] was used in the design of neural network predictor.

II. HR PREDICTOR MODE

To investigate the relationship between the HR and PA, all the signals need to be recorded simultaneously. A portable HR and PA monitor from Alive Technologies was used here.

The monitor measures and records the wearer's ECG and PA (3-D acceleration) signals and determines the HR from the ECG in real-time. The left part of Fig.1 shows the subject (user) wearing the monitor. The specification of the monitor will be described in Section. II-A.

The middle part of Fig.1 is the preprocessor which converts the HR signal (*hr(m)*) and acceleration signals *acx(l), acy(l), acz(l)* into usable format. The outputs of the preprocessor include four synchronized sequences: HR(n) and Acx(n), Acy(n), Acz(n), which are used in the FFNN as inputs. The output of the neural network is HR(n + 1), which is the predicted HR in next time step.

A. HR and PA Recorder

Many studies on HR are based on the experimental data gathered in specific conditions and/or environments, whereas, this research was conducted with the data

TABLE I. DATA SPECIFICATIONS OF ALIVE HEART MONITOR

Signal	*ECG*	*Accelerometer*
Channels/Axis	Single Channel	3 Axes
Resolution	8 bits	8 bits
Sampling Rate	1 samples/sec	75 samples/sec
Dynamic Range	$-2.66mV$ - $2.66mV$	$-2.7g$ - $2.7g$
Bandwidth	$0.5Hz$ - $90Hz$	$0Hz$ - $20Hz$

collected from normal daily activities, without any pre-planned outline. Consequently, a portable device is needed, which can monitor and record the HR and PA signals simultaneously for a period of time with relatively high accuracy. According to the device requirements, one commercial product Alive Heart Monitor(AHM) is chosen for our experiments. The collected data can be saved in an internal SD memory card or transmitted to PC, smart phone or PDA using Bluetooth in real time. The data specification of the AHM is shown in Table. I.

B. *Signal Preprocess*

The sampling rates of HR and acceleration are set differently in the AHM (1 samples/sec and 75 samples/sec, respectively) even though the inputs of the neural network are required to be sequences with same sampling rate. Here, $hr(m)$ and $acx(l)$, $acy(l)$, $acz(l)$ are converted into four synchronized sequences $HR(n)$ and $Acx(n)$, $Acy(n)$, $Acz(n)$ through a processing period τ.

Assume the whole recording period is T , the recorded data on each signal channel are evenly divided into N segments, each segment has the length of τ, When $\tau = 4s$, HR segment has 1 samples/s $\times 4s$ = 4 samples (N_{hr}), and each acceleration segment has 75 samples/s$\times 4s$ = 300 samples (N_{ac}). Then, the nth (n = 1, . . .,N) hr segment is converted (1) into $HR(n)$, and the nth acx, acy, acz segments are converted (2) into $Acx(n)$, $Acy(n)$, $Acz(n)$.

$HR(n)$ is the average(1) heart rate of nth segment. $Acx(n)$,$Acy(n)$,$Acz(n)$ are worked as average values(2) of the corresponding movements. However, instead of the HR signals being directly used, the absolute difference values of adjacent acceleration signals are adopted to calculate $Acx(n)$, $Axy(n)$, $Acz(n)$. This reflects the PA change between adjacent time steps.

$$HR(n) = \frac{\sum_{m=(n-1)*N_{hr}+1}^{n*N_{hr}} hr(m)}{N_{hr}}$$

$$Acx(n) = \frac{\sum_{l=(n-1)*N_{ac}+1}^{n*N_{ac}} \left| acx(l+1) - acx(l) \right|}{N_{ac}}$$

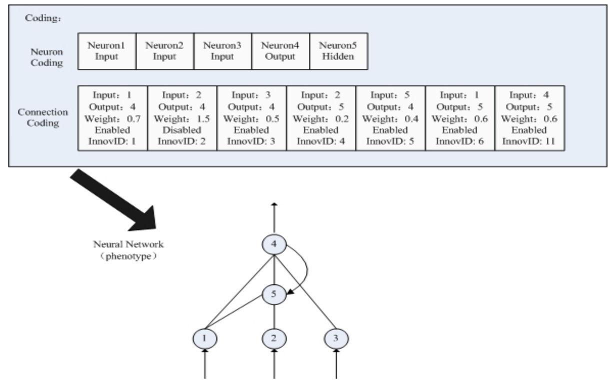


Fig.2 An example of NEAT coding and its phenotype

C. *Neural Network Predictor*

The mathematical model of predictor used in this paper is artificial neural network [14]. The design of the neural network is a complex problem [9], it mainly include two aspects:

(1) the type of neural network.

(2) the structure of the neural network, including the number of layers and the numbers of neurons in each layer.

The type of neural network is usually chose by the experts, but he structure of the neural network is hard to decide by human. To solve this problem, neural network self-learning based on evolutionary algorithm(Evolutionary Neural Network) provides a new and promising way. The method NEAT[12], [16] used in this work will be discussed in detail in next section.

III. NEAT METHOD

A. *Genetic Coding*

The Coding method used in NEAT is direct coding [16]. Each individual's genetic coding includes two parts: neuron coding (including all the neuron information) and connection coding (containing all the connection information among neurons). As shown in Fig. 2, neuron coding contains every neuron's ID and type. Connection coding between neurons includes: the start and end neuron ID of a connection, connection weights, connection state (enabled/disabled), the connection ID in the update table. Recording the state of connection is to enable or abandon a connection in mutation process. Recording the innovation ID in the update table is to track the genetic inheritance of the individuals, and avoid competition convention in crossover.

B. *Crossover*

A public update table [16] is used to record the evolution of neural networks (the growth process of neural networks), and it's also the basis of crossover. As shown in Figure 3, connection genes have the same innovation ID (ie. 1,2,3 in Fig. 3) denotes they have a common ancestor on the inheritance, one individual is chose from the two parents randomly. When the innovation ID does not match for the

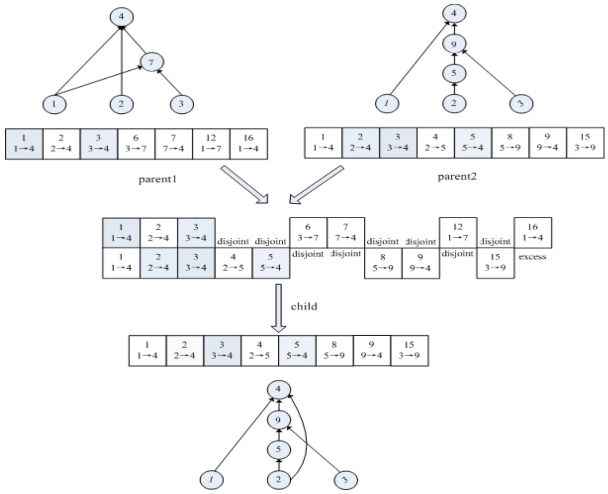


Fig.3 Crossover operator. Coding with light blue is the connection disabled. 'disjoint' means innovation ID don't match and locate in the middle of the gene sequence, 'excess' means innovation ID don't match and locate at the tail of the gene sequence.

connection genes, the individual with higher fitness is selected. (assumes parent 2 in Figure. 3 has a higher fitness.)

C. Mutation

The evolution strategy of neural network is to evolve from the simplest topology (only input and output layers), then add neurons or connections and define the neural network complexity penalty function to control the complexity of the neural network at the same time. Once add a new connection or a new connection, an update record is added into the update table to track the genetic inheritance of each individual. Mutation operator is used on the inner weights of neural network. Therefore, the mutation operator [16] in NEAT method can be divided into four categories:

(1) add a new connection(Fig. 4(a)).
(2) add a new neuron (at the same time, disable an connection, add two new connections, Fig. 4(b))
(3) Gaussian mutation on each connection weights.
(4) Gaussian mutation on the curvature of each neuron's activation function.

IV. EXPERIMENT

A. Experiment Specification

In this work, a 24-year-old healthy male with no history of heart disease was selected for the experiment subject. Recording time period is 50 minutes. During this continuous time, the subject wore AHM and did the following activities: sat at a computer to surf the Internet, played table tennis, rode bike and any other daily activities.

Recorded signals were evenly divided into two parts: The first 25 minutes as the training set was used for the FFNN evolutionary self-learning, and the remaining 25 minutes as the test set was used to verify that the trained neural network predictor. Preprocess parameter is set to 30 seconds and for all the training set and test set, N = 50.

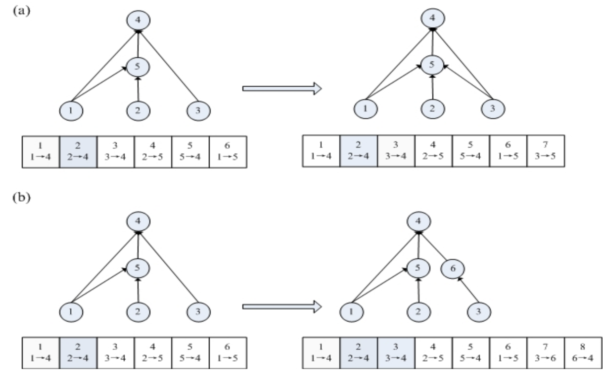
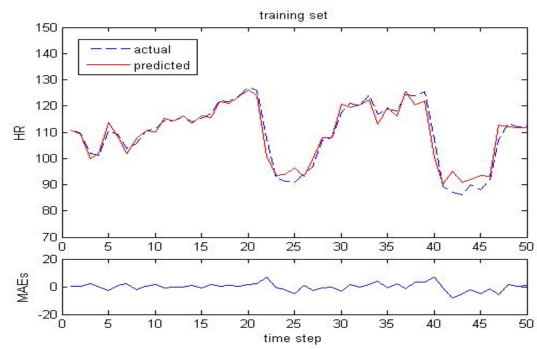
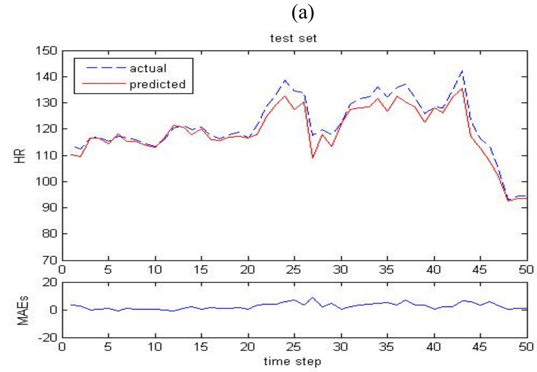


Fig.4 Mutation operator. (a) add a new connection, (b) add a new neuron. Marking with light blue background means connection disabled.



(a)



(b)

Fig.5 The performance of the proposed predictor.
(a) Training set; (b) Test set.

In evolution process, the population size was set to 50, crossover rate and mutation rate in individuals were 0.7 and 0.8 respectively. In mutation, the probability of adding a new neurons and a new connections were 0.03 and 0.07 respectively. The probability of gaussian mutation occurred in connection weights was 0.5, while the probability of Gaussian mutation occurred on curvature of each neuron's activation function was 0.1. FFNN was trained for 500 generations on the training set unless the train goal meets. Then it was tested on the test set.

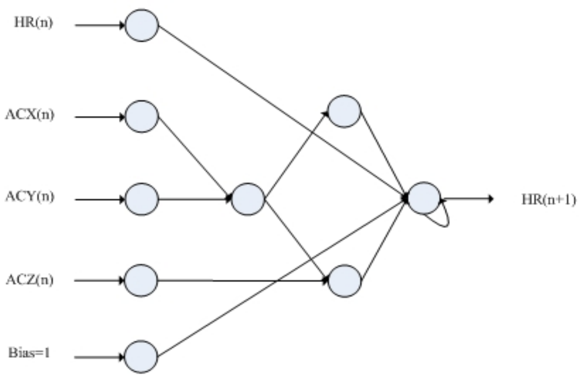


Fig. 6 The topology structure of NN predictor after evolutionary self-learning

TABLE II. MAEs BETWEEN ACTUAL AND PREDICTED DATA

MAEs(training/test)	*τ=4s*	*τ=30s*
NEAT	1.15/2.32	2.03/2.63
Method in [10]	1.89/2.18	3.42/3.88
BP with fixed topology	3.13/4.09	4.12/6.32

B. *Result and Discussion*

The performance of the neural network predictor on the training set and test set is shown in Fig. 5. To make a clear identification, the predicted HR(n + 1) is denoted with a red solid line, while the actual HR(n+1) is represented by a blue dashed line. The figure indicate that predicted HR(n + 1) can follow the variance of actual HR(n + 1) well on both the training set and test set after training.

The residual errors between the actual HR(n + 1) and the predicted HR(n + 1) are also shown in Fig. 5. The topology structure of neural network predictor obtained after self-learning is shown in Fig. 6. The corresponding mean absolute errors (MAEs) on training set and test set are 2.03 and 2.63. Considering experimental data is based on real-life and the prediction interval is 30 seconds, the MAEs for the training set and test set are all acceptable. Nevertheless, the variance of MAEs is relatively large: 7.9 and 8.7, respectively. As can be seen from Fig. 5, although most of the residuals are less than 5, there are some residual to about 8. Therefore, Reducing prediction error and instability is a major goal in the future research.

Comparison is implemented between NEAT method and any other NN method. It can be easily found from Table. II that the prediction accuracy of our method is much better. In addition, prediction interval is smaller, prediction accuracy is higher.

V. CONCLUSION

In this work, a PA-based predictor of HR is proposed. FFNN is used as the mathematical model of predictor, a new neural network self-learning method NEAT is used to design the predictor model. Experiment is based on a 50-minute set of real-time data, which is collected from a 24-year-old healthy men with a heart monitor AHM. Predictor performs prediction every 30 seconds. Compared Prediction data with actual data, the average absolute error can be limited to a smaller extent, but some of prediction errors is still very large. Prediction consistency should be improved in the future work.

REFERENCES

[1] K. Rennie, T. Rowsell, S.A. Jebb, D. Holburn, and N.J. Wareham, A combined heart rate and movement sensor: proof of concept and preliminary testing study, European Journal of Clinical Nutrition, vol.54: 409-414, 2000

[2] S. Brage, N. Brage, P. W. Franks, U. Ekelund, M.-Y. Wong, L.B. Andersen, K. Froberg, and N. J. Wareham, Branched equation modeling of simultaneous accelerometry and heart rate monitoring improves estimate of directly measured physical activity energy expenditure, Journal of Applied Physiology, vol.96(1): 343-351, 2004

[3] H.L. Chan, M.A. Lin, P.K. Chao and C.H. Lin, Correlates of the shift in heart rate variability with postures and walking by time frequency analysis, Computer Methods and Programs in Biomedicine, vol.86: 124-130, 2007

[4] H. L. Chan, S. C. Fang, Y. L. Ko, M. A. Lin, H. H. Huang and C. H. Lin, Correlates of the shift in heart rate variability with postures and walking by time frequency analysis, Computer Methods and Programs in Biomedicine, vol.86: 124-130, 2006

[5] L. Mourot, M. Bouhaddi, S. Perrey , J. D. Rouillon and J. Regnard, Quantitative Poincar plot analysis of heart rate variability: effect of endurance training, Eur. J. Appl. Physiol., vol.91(1): 79-87, 2004

[6] Y. H. Xiao, S. Z. Xiao and Z. H. Cao, A Study on phenocardiagram exercise test, IEEE Trans. on Medicine and biology Magazine, vol.18(4): 111-115, 1999

[7] T. Pawar, S. Chaudhuri and S.P. Duttagupta, Body movement activity recognition for ambulatory cardiac monitoring, IEEE Trans. on Biomed. Eng., vol.54: 874-882, 2007.

[8] G.A. Meijer, K.R. Websterp and H. Koper, Assessment of energy expenditure by recording heart rate and body acceleration, Medicine and Science in Sports and Exercise, vol.21: 343-347, 1989.

[9] M. Yuchi and J. Jo, Heart Rate Prediction Based on Physical Activity Using Feedforwad Neural Network, International Conference on Hybrid Information Technology: 344-350, 2008.

[10] F. Xiao, M. Yuchi, M. Y. Ding, J. Jo and W. G. Hou, A Research of Physical Activitys Influence on Heart Rate Using Feedforward Neural Network, International Symposium on Neural Network: 1089-1096, 2009.

[11] F. Xiao, M. Yuchi, M. Y. Ding, J. Jo and J. H. Kim, A Multi-step Heart Rate Prediction Method based on Physical Activity using Adams-Bashforth Technique, IEEE International Symposium on Computational Intelligence in Robotics and Automation: 355-359, 2009.

[12] K. O. Stanley and R. Miikkulainen , Evolving Neural Networks Through Augmenting Topologies, Evolutionary Computation, vol.10(2): 99-127, 2002

[13] K. O. Stanley and R. Miikkulainen, Efficient Reinforcement Learning through Evolving Neural Network Topologies, Proc. of the Genetic and Evolutionary Computation Conference, vol.10(2): 99-127, 2002

[14] S. Haykin, Neural Networks: A Comprehensive Foundation, Macmillan College Publication, NewYork, US, 1994.

[15] D. J. Montana and L. Davis, Training feedforward neural networks using genetic algorithms, Proc. of the Eleventh Int. Joint Conf. on Artificial Intelligence, vol.10(2): 762-767, 1989

[16] M. Buckland, Ai Techniques For Game Programming, Premier Press, 2002