



Artificial Neural Network Architectures for solving Double Dummy Bridge Problem in Imperfect Information Game

Dr.M Dharmalingam

Associate Professor

*Department of Computer Science
Nandha Arts and Science College, Erode
Bharathiar University Coimbatore
Tamil Nadu - 638316, India.
emdharma@gmail.com*

Dr.R.Amalraj

Associate Professor

*Department of Computer Science
SriVasavCollegeErode
BharathiarUniversity Coimbatore
Tamil Nadu - 638316, India.*

Abstract-Contract Bridge is an intelligent game, which enhances the creativity with various skills and quest to acquire the intricacies of the game, because no player knows exactly what moves other players are capable of during their turn. The Bridge being a game of imperfect information is to be equally well defined, since the outcome at any intermediate phase is purely based on the decision made on the immediate prior stage. One along with the architectures of Artificial Neural Networks (ANN) is applied by training on sample deals and used to estimate the number of tricks to be taken by one pair of bridge players is the key idea behind Double Dummy Bridge Problem (DDBP) implemented with the neural network paradigm. This study mainly focuses on Cascade-Correlation Neural Network (CCNN) and Elman Neural Network (ENN) which are used to solve the Bridge problem by using Resilient Back-Propagation (R-prop) algorithm. The Work Point Count System (WPCS) is an exclusive, most important and popular system which is used to bid a final contract in Bridge game.

Keywords-ANN, CCNN, ENN, R-prop algorithm, Sigmoid activation function, Contract Bridge, DDBP, Bidding, Playing, WPCS.

1. INTRODUCTION

The bridge is a game which requires some amount of intelligence and it increases the creativity of the human in decision making and there are extremely powerful Artificial Neural Network (ANN) approaches are available in which playing agents are equipped with carefully designed evaluation functions. In the game playing domain, the most popular Computational Intelligence (CI) disciplines are Neural Networks (NN), Evolutionary Methods (EM), and Supervised Learning (SL) [1]. ANN is a computational structure capable of processing information in order to finish a given task. A Neural Network is composed of many simple neurons each of which receives inputs from selected other neurons, and performs basic operations on these input information and sends its response to other neurons in the network. ANN models can therefore be regarded as roughly a simplification and abstraction of biological networks. ANN has been successfully applied to various recognition, classification problems [2] and games [3-5].

Artificial Neural Networks are classified under a broad spectrum of Artificial Intelligence (AI) that attempts to imitate the way a human brain works and the Cascade-Correlation Neural Network (CCNN) and Elman Neural Network (ENN) are most common types of neural networks in use and these are often trained by the way of supervised learning supported by Resilient Back-Propagation (R-prop) algorithm [6-10] and they have been formalized in a best defense model, which presents the strongest possible assumptions about the opponent. This is used by human players because modeling the strongest possible opponents provides a lower bound on the pay off that can be expected when the opponents are less informed. The new heuristics of beta-reduction and iterative biasing were introduced and represents the first general tree search algorithm capable of consistently performing at and above expert level in actual card play. The effectiveness of these heuristics, particularly when combined with payoff-reduction mini-maxing results in iprm-beta algorithm. The problems from the game of bridge, iprm-beta actually makes less errors than the human experts that produced the model solutions. It thus represents the first general search algorithm capable of consistently performing at and above expert level on a significant aspect of bridge card play [11].

Forward pruning techniques may produce reasonably accurate result in bridge game. Two different kinds of game trees viz., N-Game trees and N-Game like trees were used to inspect, how forward pruning affects the probability of choosing the correct move. The results revealed that, mini-maxing with forward pruning did better than ordinary mini-maxing, in cases where there was a high correlation among the mini-max values of sibling nodes in a game tree. The result suggested that forward pruning may possibly be a viable decision-making technique in bridge games [12]. The Bridge Baron is generally acknowledged to be the best available commercial program for the game of Contract Bridge. The Bridge Baron program was developed by using Domain Dependent Pattern-matching Techniques which has some limitations. Hence there was a need to develop more sophisticated AI techniques to improve the performance of the Bridge Baron which was supplemented by its previously existing routines for declarer play with routine based on

Hierarchical Task-Network (HTN) planning techniques. The HTN planning techniques used to develop game trees in which the number of branches at each node corresponds to the different strategies that a player might pursue rather than the different cards the player might be able to play [13].

GIB is a production program, expected to play bridge at human speeds. GIB used Monte Carlo methods exclusively to select an action based on the Double Dummy analysis. All other competitive bridge-playing programs have switched their card play to similar methods, although GIB’s double dummy analysis is substantially faster than most of the other programs and its play are correspondingly stronger. If the bidding simulation indicates that the opponents are about to achieve a result much inferior than what they might achieve if they saw each other’s cards, that is evidence that there may be a gap in the database. Unfortunately, it is also evidence that GIB is simply effectively troublesome its opponents efforts to bid accurately. GIB’s bidding is substantially better than that of earlier programs but not yet of expert caliber [14].

Among the various neural networks, in this paper we mainly focus Cascade-Correlation Neural Network (CCNN) and Elman Neural Network (ENN)for training and testing the data. Resilient Back-Propagation (R-prop) algorithm was used in the network to train the data for solving Double DummyBridge Problems in Contract Bridge. A Point Count method and Distributional Point methods are the two types of hand strength in human estimators. The structure of this paper is organized as follows. Section II and Section III gives a brief description of Contract Bridge Game and data representation respectively. Section IV discuss about Soft computing and Section V briefing ArtificialNeural Networks and R-prop algorithm. Our proposed Double Dummy Bridge Problem and problem Implementations are discussed in Section VI and VII. Section VIII gives about the results and discussion and Section IX discussed about the conclusion and future links of our research.

II. THE CONTRACT BRIDGE GAME

Contractbridge,usuallyknownsimplyasbridge,isatricks-takingcardgame. Therearefourplayersin two fixedpartnerships (Pairs). Partnerssitfacingeachother.Itisestablished torefertothe playersaccordingtotheirpositionatthetableas*North (N)*, *East (E)*,*South (S)*and*West (W)*,so*N andS*arepartners playingagainst*E andW*. Example shown in Figure. 1.

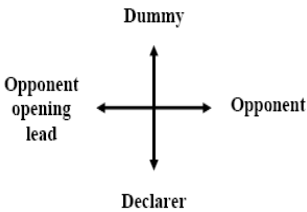


Figure.1.Game Disposition.

Astandard52cardpackisused.Thecardsineachsuit rankfrom the highestto the lowest as*Ace (A)*, *King (K)*,*Queen (Q)*, *Jack (J)*,10,9,8,7,6,5,4,3,2.Thedealerdealsoutallthe cardsoneatatimesothateachplayerreceives13ofthem. The game then proceeds through a bidding and playing phase. The purpose of the bidding phase is to identification of trumps and declarer of the contract. The playing phase consists of 13 tricks, with each player contributing one card to each trick in a clockwise fashion with anotherlevel bidto decidewhowillbethedeclarer.The side which bid highest will trytowinatleastthatnumber oftricksbid,withthespecified suitastrumps. Thereare5possibletrumpsuits:*spades (♠)*,*hearts (♥)*, *diamonds (♦)*,*clubs(♣)*and“no-trump”whichistheterm forcontractsplayedwithoutatrump. Afterthreesuccessivepasses,the lastbidbecomes the contract.The teamwhomadethefinalbidwillat the momenttryto makethecontract.Thefirstplayerofthisgroupwhomentioned thevalue ofthecontractbecomes thedeclarer.The declarer’s partneriswell-knownasthedummy shown in Figure. 2.

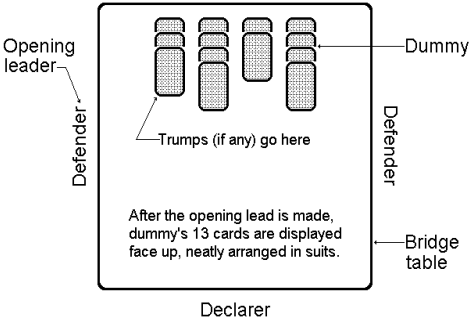


Figure. 2. Bridge Table

Theplayertotheleftofthedeclarerleadstothe first trick and instantly afterthisopeninglead,thedummy’s cardsis showing. Theaimofthedeclareristotakeatleastthenumberof tricksannouncedduringthebiddingphase. The players of theopposite pairtrytopreventhimfromdoingit [15,16]. In bridge, special focus in game representation is on the fact that players cooperate in pairs, thus sharing potentials of their hands [17].

A. Double Dummy Bridge Problem

To estimate the number of tricks to be taken by one pair of bridge players is the basis in Double Dummy Bridge Problem (DDBP). A bridge problem is presented for entertainment, in which the solver is presented with all four hands and is asked to determine the course of play that will achieve or defeat a particular contract. The partners of the declarer, whose cards are placed face up on the table and played by declarer. Dummy has few rights and may not participate in choices concerning the play of the hand. Estimating hands strength is a decisive aspect of the bidding phase of the game of bridge, because the contract bridge is a game with incomplete information and during the bidding phase. This incompleteness of information might allow for many variants of a deal in cards distribution. The player should take into account all these variants and quickly approximate the predictable number of tricks to be taken in each case [18,19].

B. The Bidding

The bidding phase is a conversation between two cooperating team members against an opposing partnership. It aims to decide who will be the declarer. Each partnership uses an established bidding system to exchange information and interpret the partner's bidding sequence. Each player has knowledge of his own hand and any previous bids only. A very interesting characteristic of the bidding phase is cooperation of players in a North with South and West with East. In each, player is modeled as an autonomous, active agent that takes part in the message process. The agent-based algorithm to use of achieve in appropriate learning, a bidding ability close to that of a human expert [20-22].

C. The Playing

In the game, the play phase seems to be much less interesting than the bidding phase. ANN approaches tried to imitate the human strategy of the play by using some tactics. The new system was able to find a strategy of play and additionally a human explanation of it [23]. The play proceeds clockwise and each of the other three players in turn must, if potential, play a card of the same suit that the person in charge played. A player with no card of the suit led may play any card of his selection. A trick consists of four cards, one from each player, and is winning by the maximum trump in it, or if no trumps were played by the maximum card of the suit led. The winner of a trick leads to the subsequently and may lead any card. Dummy takes no lively part in the play of the hand and is not permitted to offer any advice or observation on the play. At any time it is dummy's turn to play, the declarer should say which of dummy's cards is to be played, and dummy plays the card as inculcated. Finally, the scoring depends on the number of tricks taken by the declarer team and the contract [24,25].

D. No-trump & Trump-suit

A trick contains four cards one contributed by each player and the first player starts by most important card, placing it face up on the table. In a clockwise direction, each player has to track suit, by playing a card of the alike suit as the one led. If a heart is lead, for instance, each player must play a heart if possible. Only if a participant doesn't have a heart he can discard. The maximum card in the suit led wins the trick for the player who played it. This is called playing in no-trump. No-trump is the maximum ranking denomination in the bidding, in which the play earnings with no-trump suit. No-trump contracts seem to be potentially simpler than suit ones, because it is not possible to ruff a card of a high rank with a trump card. Though it simplifies the rules, it doesn't simplify the strategy as there is no guarantee that a card will take a trick, still Aces are ineffective in tricks of other suits in no-trump contracts. The success of a contract often lies in the hand making the opening lead. Hence even knowing the location of all cards may sometimes be not sufficient to indicate cards that will take tricks [17]. A card that belongs to the suit has been chosen to have the highest value in a particular game, since a trump can be any of the cards belonging to any one of the players in the pair. The rule of the game still necessitates that if a player can track suit, the player must do so, otherwise a player can no longer go at the rear suit, on the other hand, a trump can be played, and the trump is higher and more influential than any card in the suit led [18].

E. Work Point Count System

The Work Point Count System (WPCS) which scores 4 point for Ace, 3 point for King, 2 point for Queen and 1 point for a Jack is followed in which no points are counted for 10 and below. During the bidding phase of contract bridge, when a team reaches the combined score of 26 points, they should use WPCS for getting final contract and out of thirteen tricks in contract bridge, there is a possibility to make use of eight tricks by using WPCS.

III. THE DATA REPRESENTATION OF GIB LIBRARY

The data used in this game of DDBP was taken from the Ginsberg's Intelligent Bridge (GIB) Library. The data created by Ginsberg's Intelligent Bridge player [14]. The GIB library includes 7,00,000 deals and for each of them provides the number of tricks to be taken by N S pair for each combination of the trump suit and the hand which makes the opening lead. [26].

IV. SOFT COMPUTING

Presently the on-going development of computer technology, soft computing will considerably enhance traditional computation methods. The machine-intelligent behavior is determined by the flexibility of the architecture, the ability to recognize machine incorporations of human expertise, laws of inference procedure and the high speed of learning. All these titles are the main constituents of the research area named Soft Computing and it is a practical alternative for solving scientifically complex problems [27]. Soft computing involves partnership of several areas, the most important being Artificial Neural Networks (ANN), Fuzzy Logic (FL), Genetic Algorithm (GA) and Evolutionary Computations (EC) [6]. Among the above fields, Artificial Neural Networks used to solve the Double Dummy Bridge Problem in Contract Bridge.

V. ARTIFICIAL NEURAL NETWORK (ANN)

Artificial Neural Network consists of several processing units which are interconnected according to some topology to accomplish a pattern classification task. An Artificial Neural Network is configured for a precise application, such as pattern recognition or data classification through learning process. ANNs are non-linear information processing devices, which are built from organized elementary processing devices called neurons [28, 29]. In Artificial Neural Network following the supervised learning; each input vector requires a matching target vector, which represents the desired output. The input vector along with the target vector is called training couple. In supervised learning, a supervisor is necessary for error minimization. Consequently the network trained by this method is said to be using supervised learning methodology. In supervised learning, it is assumed that the correct target output values are known for each input pattern [30, 31].

A. Cascade-Correlation Neural Network Architecture

The cascade-correlation architecture was introduced by [32] starts with a one layer neural network and hidden neurons are added depends on the need. The Cascade-Correlation begins with a minimal network, then mechanically trains and adds new hidden units one by one, creating a multi-layer configuration. Once a new hidden unit has been added to the network, its input-side weights are frozen. The new hidden neuron is added in each training set and weights are adjusted to minimize the magnitude of the correlation between the new hidden neuron output and the residual error signal on the network output that has to be eliminated. The cascade-correlation architecture has many rewards over its complement, as it learns at a faster rate, the network determines its own dimension and topology, it retains the structures it had built, still if the preparation set changes, and it requires no back-propagation of error signals through the associations of the network. [33] During the learning progression, new neurons are added to the network one by one and each one of them is placed into a new hidden layer and connected to all the preceding input and hidden neurons. Once a neuron is finally further to the network and activated, its input connections become frozen and do not change anymore.

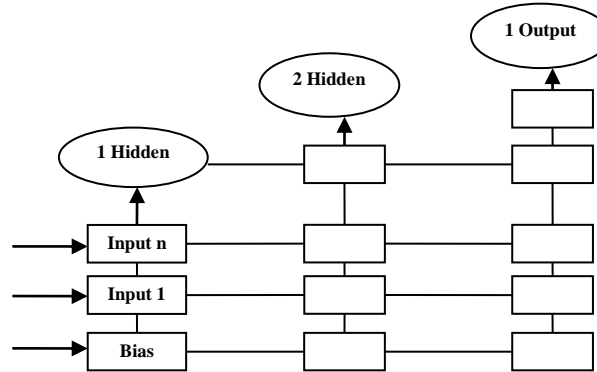


Figure. 3. Architecture of Cascade-Correlation Neural Network

The neuron to be added to the existing network can be made in the following two steps: (i) The candidate neuron is connected to all the input and hidden neurons by trainable input connections, but its output is not connected to the network. Then the weights of the candidate neuron can be trained while all the other weights in the network are frozen. (ii) The candidate is connected to the output neurons and then all the output connections are trained. The whole process is repeated until the desired network accuracy is obtained. The equation(1) correlation parameter 'S' defined as below is to be maximized.

$$S = \sum_{o=1}^O |\sum_{p=1}^P (V_p - \bar{V})(E_{po} - \bar{E}_o)| \quad (1)$$

where O is the number of network outputs, P is the number of training patterns, V_p is output on the new hidden neuron and E_{po} is the error on the network output. In the equation(2) the weight adjustment for the new neuron can be found by gradient descent rule as

$$\Delta w_{li} = \sum_{o=1}^O \sum_{p=1}^P \sigma_o (E_{po} - \bar{E}_o) f_p' x_{ip} \quad (2)$$

The output neurons are trained using the generalized delta learning rule for faster convergence in Back-Propagation algorithm. Each hidden neuron is trained just once and then its weights are frozen. The network learning building process is completed when satisfied results are obtained. The cascade-correlation architecture needs only a forward sweep to compute the network output and then this information can be used to train the candidate neurons.

B. Elman Neural Network Architecture

The Elman neural network is a partial recurrent neural network model first proposed by Elman [34] it is a special kind of feed-forward neural network, which has extra local memory neurons and feedback loop. The Elman neural network is capable of approximating a nonlinear system without an explicit physical model. Elman neural network has frequently separated into four kinds of layers input, hidden, context and output layer. The context layer is utilized to constitute the back-forward loop, from which the hidden layer selects input [35]. In comparison with other forms of feed forward neural network, the Elman neural network is responsive to history of input data by this mechanism.

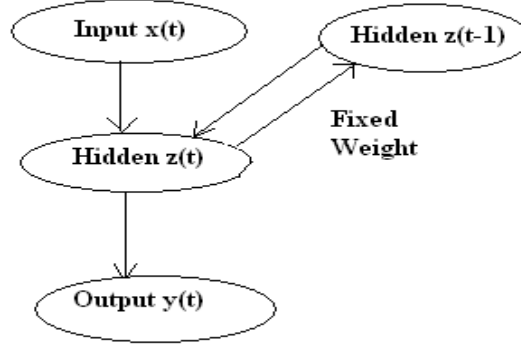


Figure.4 Architecture of Elman Neural Network

In Figure.4 shows architecture of an Elman neural network, with the addition of a set of context units in the input layer. There are links from the hidden layer to these context units permanent with a weight. At every one time step, the input is propagated in a standard feed-forward approach, and then a supervised learning rule is applied. The permanent back connections result in the context units always maintaining a copy of the previous values of the hidden units.

C. The Resilient Back-Propagation (R-prop) Algorithm

The algorithm R-prop is a local adaptive learning method, performing supervised batch learning in Cascade-Correlation Neural Network Architecture and Elman Neural Network Architecture. The basic principle of R-prop is to eliminate the harmful influence of the size of the partial derivative on the weight step. As a result, single the sign of the derivative is considered to indicate the direction of the weight update. The algorithm acts on each weight separately. The equation(3), for each weight, if there was a sign modify of the partial derivative of the total error function compared to the final iteration, the update value for that weight is multiplied by a factor η^- , where $0 < \eta^- < 1$. If the last iteration produces the similar sign, the renew value is multiplied by a factor of η^+ , where $\eta^+ > 1$. The renew values are calculated for each weight in the above manner, and lastly each weight is changed by its own renew value, in the opposite path of that weight's partial derivative. This is to minimize the total error function. η^+ is empirically set to 1.2 and η^- to 0.5.

To elaborate the above description mathematically we can start by introducing for each weight w_{ij} its individual update value $\Delta_{ij}(t)$, which exclusively determines the magnitude of the weight-update. This update value can be expressed mathematically according to the learning rule for each case based on the observed behavior of the partial derivative during two successive weight-steps by the following formula:

$$\Delta_{ij}(t) = \begin{cases} \eta^+ \cdot \Delta_{ij}(t-1), & \text{if } \frac{\partial E}{\partial w_{ij}}(t) \cdot \frac{\partial E}{\partial w_{ij}}(t-1) > 0 \\ \eta^- \cdot \Delta_{ij}(t-1), & \text{if } \frac{\partial E}{\partial w_{ij}}(t) \cdot \frac{\partial E}{\partial w_{ij}}(t-1) < 0 \\ \Delta_{ij}(t-1), & \text{else} \end{cases} \quad (3)$$

Where $0 < \eta^- < 1 < \eta^+$.

A clarification of the adaptation rule based on the above formula can be stated. The equation(4), it is evident that whenever the partial derivative of the equivalent weight w_{ij} varies its sign, which indicates that the last update was large in magnitude and the algorithm has skipped over a local minima, the update - value $\Delta_{ij}(t)$ is decreased by the factor η^- . If the derivative holds its sign, the update - value will to some extent increase in order to speed up the convergence in shallow areas. When the update-value for each weight is settled in, the Weight-update itself tracks a very simple rule. The equation(5), which is if the derivative is optimistic, the weight is decreased by its renewing value, if the derivative is harmful, the update-value is added.

$$\Delta w_{ij}(t) = \begin{cases} -\Delta_{ij}(t), & \text{if } \frac{\partial E}{\partial w_{ij}}(t) > 0 \\ \Delta_{ij}(t), & \text{if } \frac{\partial E}{\partial w_{ij}}(t) < 0 \\ 0, & \text{else} \end{cases} \quad (4)$$

$$w_{ij}(t+1) = w_{ij}(t) + \Delta w_{ij}(t) \quad (5)$$

However, there is one exception. The equation(6), if the partial derivative changes sign that is the previous step was too large and the minimum was missed, the previous weight-update is reverted

$$\Delta w_{ij}(t) = -w_{ij}(t-1), \quad (6)$$

$$\text{if } \frac{\partial E}{\partial w_{ij}}(t) \cdot \frac{\partial E}{\partial w_{ij}}(t-1) < 0 \quad (7)$$

Due to that ‘backtracking’ weight-step, the derivative is assumed to change its sign once again in the following step. In order to avoid a double penalty of the update-value, there should be no adjustment of the update-value in the following step. In perform this can be done by setting $\frac{\partial E}{\partial w_{ij}}(t-1) = 0$ in the Δ_{ij} update-rule above.

The equation(7), the partial derivative of the total error is given by the following formula:

$$\frac{\partial E}{\partial w_{ij}}(t) = \frac{1}{2} \sum_{p=1}^p \frac{\partial E_p}{\partial w_{ij}}(t) \quad (8)$$

Hence, the partial derivatives of the errors must be accumulated for all training patterns. This indicates that the weights are updated only after the presentation of all of the training patterns [36]. It is noticed that R-prop is much faster than the standard steepest descent algorithm.

VI. NEURAL NETWORK IN DOUBLE DUMMY BRIDGE PROBLEM

There are several Neural Network architectures have been used to solving the Double Dummy Bridge Problem. In this paper we focus Cascade-Correlation Neural Network (CCNN) and Elman Neural Network (ENN) architecture 52(13x4) for solving the DDBP in contract bridge.

A. 52 (13x4) Representation

In this architecture, positions of cards in the input layer were fixed, i.e. from the leftmost input neuron to the rightmost one the following cards were represented: 2♠, 3♠, ..., K♠, A♠, 2♥, ..., A♥, 2♦, ..., A♦, 2♣, ..., A♣. Figure.5. This way each of the 52 input neurons was assigned to a particular card from a deck and a value presented to this neuron determined the hand to which the respective card belonged, i.e. 1.0 for *North*, 0.8 for *South*, -1.0 for *West*, and -0.8 for *East*.

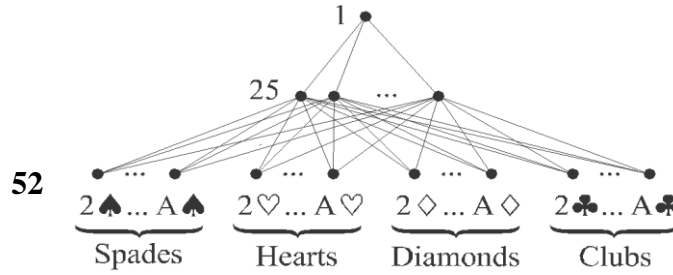


Figure.5. Neural Network Architecture with 52 input neurons

Layers were fully connected, i.e., in the 52 – 25 – 1 network all 52 input neurons were connected to all 25 hidden ones, and all hidden neurons were connected to a single output neuron.

VII. IMPLEMENTATION

There are several Neural Network architectures have been used to solving the Double Dummy Bridge Problem. In this paper we focus Cascade-Correlation Neural Network and Elman neural network architectures 52(13x4) for solving the DDBP in contract bridge. In our research for implementing GIB library data are used in MATLAB 2008a.

A. Input Layer

52 cards were used in input layer. Each member was received 13 cards. The card values are determined in rank card (2, 3, K, A) and suit card (♠ (S), ♥ (H), ♦ (D), ♣ (C)). The rank card is transformed using a uniform linear transformation to the range from 0.10 to 0.90.

The Smallest card value is 2(0.10) and highest card value is A (0.90). The suit cards are a real number of using the following mapping: Spades (0.3), Hearts (0.5), Diamonds (0.7) and Clubs (0.9). All combination cards value rank and suit cards represented by one hand.

B. Hidden Layer

There is a middle layered of hidden and internal representation 25 neuron were fully connected. The basically 4 suits, the power of trump suit, the weight of a rank card, the highest of *Ace* and lowest is *two*. The neuron representing a hand to which the card actually received input value equal 1.0. The other three neurons were assigned input values equal to 0.0.

C. Output Layer

In this layer only one output was received and getting the result, decision boundaries were defined within the range of (0.1 to 0.9). The results were defined a priori and target of ranges from 0 to 13 for all possible number of tricks was the use of a linear transformation. Gradient descent training function was used to train the data and gradient descent weight/bias learning function was used for learning the data. For training and learning the data, two networks viz., Elman neural network and CCNN were used in hyperbolic tangent sigmoid function. The results produced are represented in Table I and Table II respectively.

Table I Training deals sample 20

S.No	Actual value in GIB	Calculated value in Elman neural network hyperbolic tangent sigmoid function	Calculated value in CCNN using hyperbolic tangent sigmoid function
1	0.75000	0.50537	0.74336
2	0.83000	0.76809	0.82961
3	1.00000	0.83654	0.99209
4	0.83000	0.62388	0.82788
5	0.75000	0.54815	0.73020
6	0.50000	0.50021	0.50114
7	0.58000	0.50046	0.57565
8	0.75000	0.54373	0.74029
9	0.50000	0.50018	0.50012
10	0.83000	0.84651	0.84685
11	0.58000	0.50195	0.56703
12	1.00000	0.83235	0.99176
13	0.58000	0.61464	0.55733
14	0.50000	0.52667	0.50134
15	0.91000	0.58635	0.91803
16	0.50000	0.78009	0.50317
17	0.50000	0.50010	0.50110
18	0.83000	0.50799	0.82475
19	0.66000	0.50299	0.65817
20	0.58000	0.53902	0.58061

Table II Test deals sample 10(Even)

S.No	Actual value in GIB	Calculated value in Elman neural network hyperbolic tangent sigmoid function	Calculated value in CCNN using hyperbolic tangent sigmoid function
1	0.83000	0.81354	0.82995
2	0.83000	0.56256	0.83996
3	0.50000	0.70507	0.50778
4	0.75000	0.74712	0.75965
5	0.83000	0.64946	0.83962
6	1.00000	0.99577	0.99962
7	0.50000	0.50368	0.50053
8	0.50000	0.50346	0.50824
9	0.83000	0.80008	0.82525
10	0.58000	0.56167	0.58936

VIII. RESULTS AND DISCUSSION

In this paper sample deals data were used for training (20) and testing (10) in MATLAB 2008a. Together there are 20 numbers of each deal i.e. 5 trump suits by 4 sides. Here 5 trump suits are No-trumps, spades, Hearts, Diamonds and Clubs, No-trump which is the term for contracts played without trump. Four sides are West, North, East and South. So North and South are partners playing against East and West. The results presented in the Figure 6 and Figure 7 show that the comparison of target tricks along with Elman neural network and CCNN. While comparing the train and test data along with target data, the result indicated that, train and test data shown significantly better results in both networks, which minimized the total mean square.

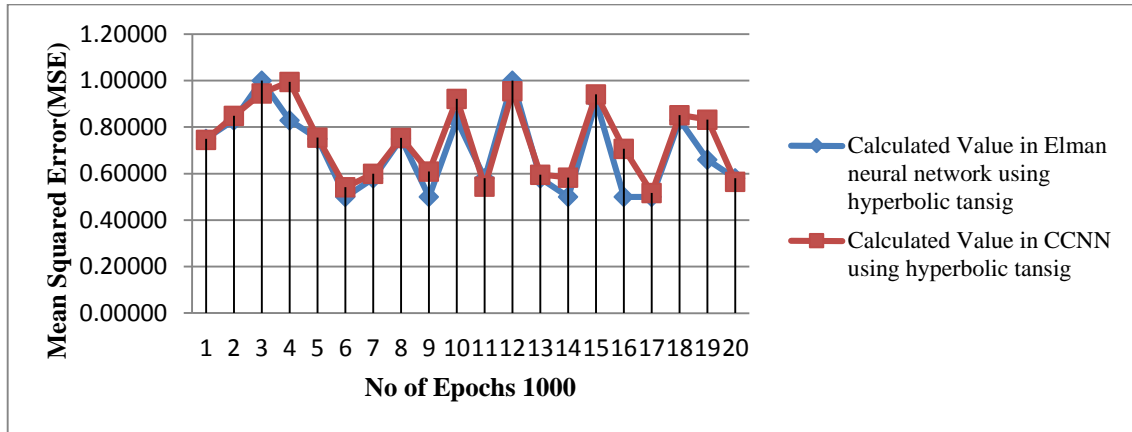


Figure.6. Elman and CCNNs using hyperbolic tangent sigmoid function in training deal sampling 1000 epochs

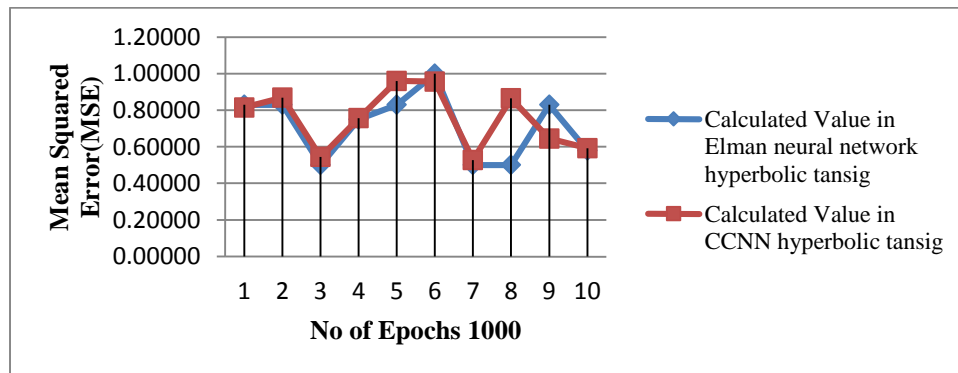


Figure.7 Both networks using hyperbolic tangent function in testing deal sampling 1000 epochs

The data trained and tested through this CCNN shows better performance and the time taken for training and testing the data were relatively minimum which also converged to the error steadily during the whole process. Elman neural network and CCNN were compared with each other and CCNN was given significantly superior results than Elman neural network. During bidding phase of contract bridge, Hyperbolic Tangent Sigmoid function was used in CCNN architecture in R-prop algorithm to take best WPCS for getting final contract.

IX. CONCLUSION AND FUTURE WORK

Artificial Neural Networks which were used to estimate the number tricks to be taken by one pair of players in the Double Dummy Bridge Problem in Contract Bridge. In Cascade-Correlation neural network, during training process new hidden nodes are added to the network one by one. For each new hidden node, the correlation magnitude between the new node output and the residual error signal is maximized. During the time when the node is being added to the network, the input weights of hidden nodes are frozen, and only the output connections are trained repeatedly. ENN has a superior performance, concerning the capability of ENN to obtain the parameter easier to follow the real and the future data enhanced to take only relatively less time in order to reach minimum value. Even though both the Cascade-Correlation neural network and Elman Neural Network produced better results, CCNN has given significantly superior result than ENN. The Work Point Count System used in R-prop algorithm which produced better results and used to bid a final contract. WPCS is a good information system and it provides some new ideas to the bridge players and helpful for beginners and semi professional players also in improving their bridge skills. Furthermore we would enlarge the hybrid architecture and different algorithms to solve DDBP more professionally and effectively.

REFERENCES

- [1] JacekMandziuk(2010) Knowledge-free and Learning – Based methods in Intelligent Game Playing, Springer.
- [2] JacekMandziuk(2007)Computational Intelligence in Mind Games, In: Studies in Computational Intelligence, Springer.
- [3] Frank I, Basin D A (2001) A Theoretical and EmpiricalInvestigation of Search in Imperfect Information Game, Theor.Comput.Sci.vol.no252, pp 217-256.
- [4] JacekMandziuk(2008) Some thoughts on using Computational Intelligence methods in classical mind board games, In: Proceedings of the 2008 International Joint Conference on Neural Networks (IJCNN 2008), Hong Kong, China, pp. 4001–4007.
- [5] JacekMandziuk, Krzysztof Mossakowski (2004) Looking InsideNeuralNetworks Trained to Solve Double-Dummy Bridge Problems, Int.Proceeding 5th Game – On Computer Games: Artif. Intell., U.K. pp. 182-186.
- [6] Sivanandam S N, Deepa S N(2007)Principles of Soft Computing, Wiley.
- [7] JacekMandziuk, Krzysztof Mossakowski(2009) Neural networks compete with expert human players in solving the double dummy bridge problemProc. of 5th Int. Conf. on Computational Intelligence and games, pp. 117-124.
- [8] Krzysztof Mossakowski,JacekMandziuk(2009) Artificial neural networks for solving double dummy bridge problems, In: AI and Soft computing vol.no 3070, Springer, pp. 915–921.
- [9] Sarkar M,Yegnanarayana B,Khemani D(1995)Applicationofneural networkincontractbridge bidding,inProc.ofNationalConf.onNeuralNetworksandFuzzySystems,AnnaUniversity,Madras, pp.144-151.
- [10] Dharmalingam M,Amalraj R(2013) Neural Network Architectures for Solving the Double Dummy Bridge Problem in Contract Bridge, in Proce of the PSG-ACM National Conference on Intelligent Computing,pp 31-37.
- [11] Frank I, Basin D A(1999) Optimal play against best Defence: Complexity and Heuristics in Lecture Notes in Computer Science Germany: Springer - Verlag, vol.no1558, pp 50-73.
- [12] Smith S J J,Nau D S(1994) An Analysis of Forward Pruning, in Proce of the National Conference on AI, pp 1386-1391.
- [13] Smith S J J, Nau D S, Throop T A(1998) Success in Spades: Using AI Planning Techniques to Win the World Championship of Computer Bridge, in Proce of the National Conference on AI, pp 1079-1086.
- [14] Ginsberg M L(2001) GIB: Imperfect Information in a Computationally Challenging Game Journal of Artificial Intelligence Research, vol. 14, pp 303-358.
- [15] Francis H,Truscott A,Francis D(1994)TheOfficial Encyclopediaof Bridge,5thed. Memphis,TN:AmericanContractBridgeLeague.
- [16] Root W H(1998) TheABCs ofBridge,Three RiversPress.
- [17] JacekMandziuk, Krzysztof Mossakowski (2007) Example – based estimation of hands strength in the game of bridge with or without using explicit human knowledge, In Proc. IEEE Symp. Comput.Intell.Data mining, pp.413-420.
- [18] Krzysztof Mossakowski,JacekMandziuk(2009) Learning without human expertise: A case study of Double Dummy Bridge Problem, IEEE Transactions on Neural Networks, vol.20, No.2, pp. 278-299.
- [19] Smith S J J,Nau D S, Throop T A(1998) Computer bridge - A big win for AI planning, AI Magazine. vol.19, No.2, pp. 93-106.
- [20] Amit A, MarkovitchS(2006) Learning to bid in bridgeMachine Learning, vol.63, no.3, pp 287-327.
- [21] Ando T,UeharaT(2001) Reasoning by agents in computer bridge bidding, in Computers and Games, vol. 2063, pp. 346–364.
- [22] Ando T, Kobayashi N,Uehara T(2003) Cooperation and competition of agents in the auction of computer bridge, Electronics and Communications in Japan, Part 3, vol. 86, no. 12, pp. 76–86.
- [23] KhemaniD(1994) Planning with thematic actions,in AIPS, pp. 287–292.
- [24] Frank I, Basin D A(1999) Strategies explained in Proceeding 5th Game programming Workshop in Japan. pp1-8.
- [25] Ali Awada, May Dehayni,AntounYaacoub(2011) An ATMS-Based Tool for Locating Honor Cards in Rubber Bridge,Journal of Emerging Trends in Computing and Information Sciences, vol. 2 no.5,pp.209-218.
- [26] Krzysztof Mossakowski,JacekMandziuk(2006) Neuralnetworksandtheestimationofhandsstrengthincontractbridge,inArtificialIntelligenceandSoftComputingICAISC vol.4029. Springer,pp.1189-1198.
- [27] YegnanarayanaB(2010)Artificial Neural Networks,Pretince Hall.
- [28] Yegnanarayana B,Khemani D,Sarkar M(1996)Neuralnetworksforcontractbridgebidding,Sadhana,vol.21,no.3,pp.395-413.
- [29] SivanandamS N,PaulrajM(2011) Introduction Artificial Neural Networks,Vikas.
- [30] DharmalingamM,Amalraj R(2013) Supervised Learning in Imperfect Information Game, International Journal of Advanced Research in Computer Science, vol. 4, no.2, pp. 195-200.
- [31] DharmalingamM, Amalraj R(2013) Artificial Neural Network Architecture for Solving the Double Dummy Bridge Problem in Contract Bridge, International Journal of Advanced Research in Computer and Communication Engineering, vol. 2, issue.12, pp. 4683-4691.
- [32] Fahlman S E, LebiereV(1990)The cascade-correlation learning architecture.Advances in Neural Information Processing, pp. 524-532.
- [33] DharmalingamM, AmalrajR(2014) Fast Supervised learning Architecture- a work point count system coupled with Resilient Back-propagation Algorithm for Solving the Double Dummy Bridge Problem, International Journal of Emerging Trends & Technology in Computer Science, vol. no3(3), pp. 189-195.
- [34] Elman J(1990)Finding structure in time, cognitive science 14(3), pp. 179-211.
- [35] DharmalingamM, AmalrajR(2014) Supervised Elman Neural Network Architecture for Solving the Double Dummy Bridge Problem in Contract Bridge, International Journal of Science and Research, vol.no 3(6),pp. 2745-2750.

- [36] Rumelhart D E, Hinton G E, Williams R J(1986) Learning internal representations by error back propagation,Parallel Distributed Processing: Ex-plorations in the Microstructure of Cognition, vol. 1, pp. 533-536.

Authors Biography



Dr. M Dharmalingam received his Under-Graduate, Post-Graduate and Master of Philosophy degrees from Bharathiar University, Coimbatore in the years 2000, 2004 and 2008 respectively. Currently he is working as an Assistant Professor in Computer Science, Nandha Arts and Science College, Erode and also pursuing part time PhD degree in Computer Science at Sri Vasavi College, Bharathiar University, Tamil Nadu, India. He has published several research papers in reputed National and International journals and the broad field of his research interest is Soft Computing.



Dr. R Amalraj is working as an Associate Professor in Computer Science, Sri Vasavi College, Erode, Tamil Nadu, India. He has obtained his PhD in Computer Science from PSG College of Technology, affiliated to Bharathiar University, Coimbatore in 2003. He is a Life Member of Computer Society of India. He has been a Principal Investigator for a second Minor Research Project sponsored by University Grants Commission, New Delhi. He has published several research papers in reputed National and International journals, with specific interest in Artificial Intelligence, Image Processing and Soft Computing.