ISSN 2348 – 9928
Doi: 10.46532/ijaict-2020012 Published on 05 (4) 2020

# ARTIFICIAL NEURAL NETWORK ARCHITECTURES FOR SOLVING THE CONTRACT BRIDGE

**M. Dharmalingam**
**Assistant Professor**

**Department of Computer Science,**
**Bharathiar University Arts and Science College,**
**(Modakkurichi) Sivagiri, Bharathiar University,**
**Erode, Tamilnadu, India**

***Abstract— Contract Bridge is an intelligent game, which enhances the creativity with multiple 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 stage is purely based on the decision made on the immediate preceding stage. One among 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 is used to solve the Bridge problem by using Resilient Back-Propagation (R-prop) Algorithm and Work Point Count System.***

***Keywords— ANN, CCNN, ENN, R-prop, Sigmoid activation functions, Contract Bridge, DDBP, Bidding, Playing, WPCS.***

## I. 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 have been successfully applied to various recognition, classification problems [2] and games [3] [4], [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],[7],[8],[9],[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

ISSN 2348 – 9928
IJAICT Volume 7, Issue 4, April 2020 Doi: 10.46532/ijaict-2020012 Published on 05 (4) 2020

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 Dummy Bridge Problems in Contract Bridge. A Point Count method and Distributional Point methods are the two types of hand strength in human estimators. 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. 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 Neural Network Methodology. 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

Contract bridge, usually known simply as bridge, is a trick - taking card game. There are four players in two fixed partnerships (Pairs). Partners sit facing each other. It is established to refer to the players according to their position at the table as *North* (*N*), *East* (*E*), *South* (*S*) and *West* (*W*), so *N and S* are partners playing against *E* and *W*. Example shown in Fig. 1.

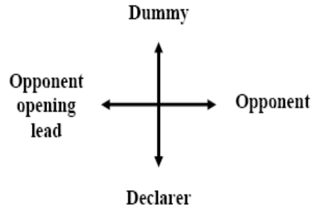


***Fig 1 : Game Disposition.***

A standard 52 card pack is used. The cards in each suit rank from the highest to the lowest as *Ace (A)*, *King (K)*, *Queen (Q)*, *Jack (J )*, 10, 9, 8, 7, 6, 5, 4, 3, 2. The dealer deals out all the cards one at a time so that each player receives 13 of them. The game then proceeds through a bidding and playing phase. The purpose of the biding 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 another level bid to decide who will be the declarer. A bid specify a number of tricks and a trump suit or no-trump. The side which bids highest will try to win at least that number of tricks bid, with the specified suit as trumps. There are 5 possible trump suits: *spades* (♠), *hearts* (♥), *diamonds* (♦), *clubs* (♣) and "*no-trump*" which is the term for contracts played without a trump. After three successive passes, the last bid becomes the contract. The team who made the final bid will at the moment try to make the contract. The first player of this group who mentioned the value of the contract becomes the declarer. The declarer's partner is well-known as the dummy shown in Fig. 2.

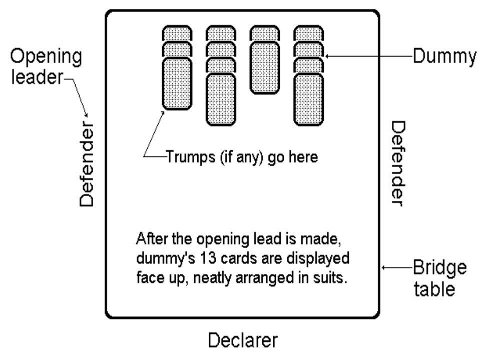


***Fig 2: Bridge Table***

The player to the left of the declarer leads to the first trick and instantly after this opening lead, the dummy's cards is showing. The aim of the declarer is to take at least the



***Corresponding Author: M.Dharmalingam, Bharathiar University Arts and Science College, Erode, India*** 

ISSN 2348 – 9928
Doi: 10.46532/ijaict-2020012 Published on 05 (4) 2020

number of tricks announced during the bidding phase. The players of the opposite pair try to prevent him from doing it [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].

### *2.1 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, since 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 approximation the predictable number of tricks to be taken in each case [18], [19].

### *2.2 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 aspect 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],[21], [22].

### *2.3 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].

### *2.4 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 similar suit as the one led. If a heart is lead, for instance, each player must play a heart if potential. Only if a player 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, even Aces are useless 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 behind suit, however, a trump can be played, and the trump is higher and more influential than any card in the suit led [18].

### *2.5 Work Point Count System*

The Work Point Count System (WPCS) which scores 4 points for Ace, 3 points for King, 2 points for Queen and 1point 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 LIBARARY

The data used in this game of DDBP was taken from the Ginsberg's Intelligent Bridge (GIB) Library. The data created

ISSN 2348 – 9928
Doi: 10.46532/ijaict-2020012 Published on 05 (4) 2020

by Ginsberg's Intelligent Bridge player [14]. In our research for implementing GIB library data we used MATLAB 2008a. 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].

### *5.1 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 counterpart, 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 Fig.3 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.

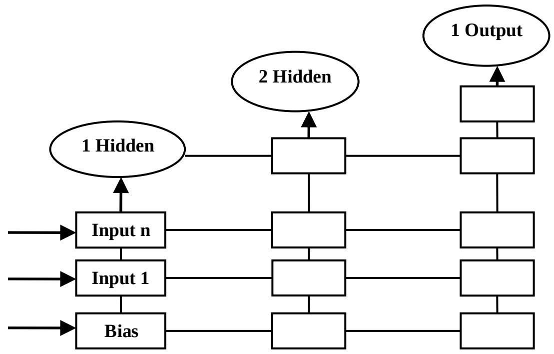


*Fig 3 : Architecture of Cascade-Correlation Neural Network*

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.

$$S = \sum_{0=1}^{O} \left| \sum_{p=1}^{P} (V_p - \bar{V})(E_{po} - \overline{E_o}) \right| \qquad (1)$$

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

ISSN 2348 – 9928
Doi: 10.46532/ijaict-2020012 Published on 05 (4) 2020

$E_{po}$ 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

$$\Delta_{w_i} = \sum_{0=1}^{O} \sum_{p=1}^{P} \sigma_o \left(E_{po} - \overline{E_o}\right) f_{p'}\, x_{ip} \qquad (2)$$

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.

### *5.2 Elman Neural Network Architecture*

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. An Elman neural network has four kinds of layers Input layer, hidden layer, context layer 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 sensitive to history of input data by this mechanism.

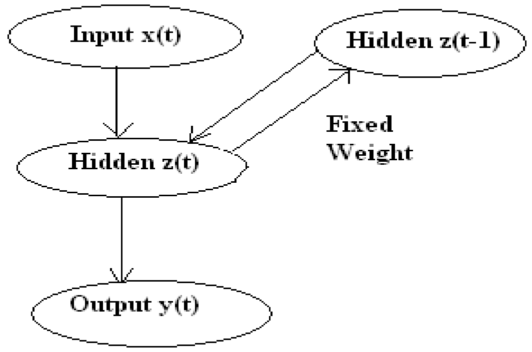


**Fig 4 : Architecture of Elman Neural Network**

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

### *5.3 The Resilient Back-Propagation (R-prop) Algorithm*

The algorithm R-prop is a local adaptive learning scheme, 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 consequence, only 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 change of the partial derivative of the total error function compared to the last iteration, the update value for that weight is multiplied by a factor η−, where 0 <η− < 1. If the last iteration produces the same sign, the update value is multiplied by a factor of η +, where η+ > 1. The update values are calculated for each weight in the above manner, and finally each weight is changed by its own update value, in the opposite direction 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:

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

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), that is if the derivative is positive, the weight is decreased by its update value, if the derivative is negative, the update-value is added.

ISSN 2348 – 9928
Doi: 10.46532/ijaict-2020012 Published on 05 (4) 2020

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

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

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

$$\Delta w_{ij}(t) = -w_{ij}(t-1),$$
$$if \frac{\partial E}{\partial w_{ij}}(t).\frac{\partial E}{\partial w_{ij}}(t-1) < 0 \quad (6)$$

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 adaptation of the update-value in the succeeding step. In practice this can be done by setting $\frac{\partial E}{\partial w_{ij}}(t-1) = 0$ in the $\Delta_{ij}$ update-rule above.
The equation (7), the partial derivative of the total error is given by the following formula:

$$\frac{\partial E}{\partial w_{ij}}(t) = \frac{1}{2}\sum_{p=1}^{p} \frac{\partial E_p}{\partial w_{ij}}(t) \quad (7)$$

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) architectures 52(13x4) for solving the DDBP in contract bridge.

### *6.1 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*♣ Fig. 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*.

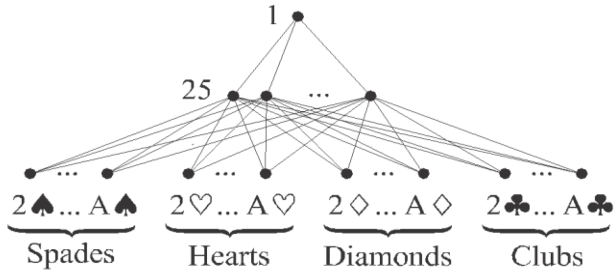


**Fig 5 :** Neural Network Architecture with 52 input neurons

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

## VII. IMPLEMENTATION

### *7.1 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.

### *7.2 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.

### *7.3 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 1 and Table 2 respectively.

IJAICT Volume 7, Issue 4, April 2020

ISSN 2348 – 9928
Doi: 10.46532/ijaict-2020012 Published on 05 (4) 2020

***Table 1 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 2 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 2011a. 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 Fig 6 and Fig 7 shown 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.

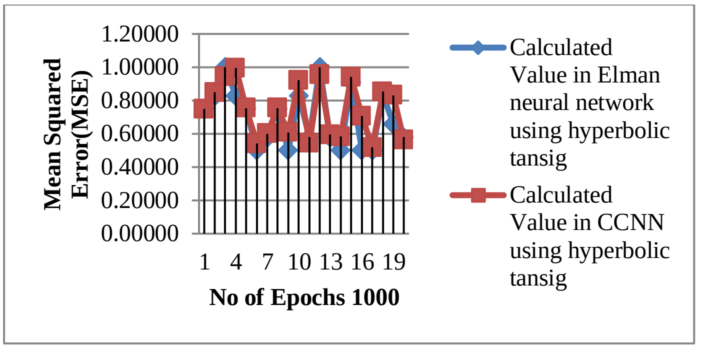


***Fig 6 : Elman and CCNNs using hyperbolic tansig sigmoid function training deal sampling 1000 epochs***

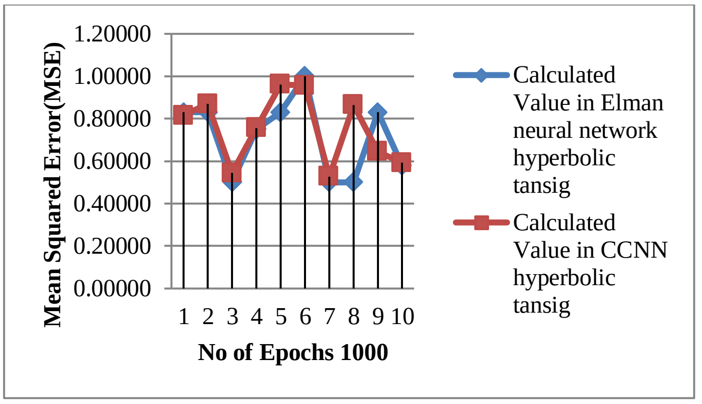


***Fig 7 : Both networks using hyperbolic tangent function testing deal sampling 1000 epochs***

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.

ISSN 2348 – 9928
Doi: 10.46532/ijaict-2020012 Published on 05 (4) 2020

## 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, is a good information system and it provides some new ideas to the bridge players and helpful for beginners and semi professional players too in improving their bridge skills. Furthermore we would enlarge the hybrid architecture and different algorithms to solve DDBP more professionally and fruitfully.

## References


[1] Jacek Mandziuk, “Knowledge-free and Learning – Based methods in Intelligent Game Playing” Springer, Chapter 5, pp 53 – 70, 2010.

[2] Jacek Mandziuk, “Computational Intelligence in Mind Games”, In: Studies in Computational Intelligence, vol. 63, Springer, pp. 407–442, 2007.

[3] I. Frank, D. A. Basin, “A Theoretical and Empirical Investigation of Search in Imperfect Information Game,” Theor.Comput.Sci.vol.252, no.1-2, pp 217-256, 2001.

[4] Jacek Mandziuk, “Some thoughts on using Computational Intelligence methods in classical mind board games”, In: Procedings of the 2008 International Joint Conference on Neural Networks (IJCNN 2008), Hong Kong, China, pp. 4001–4007, 2008.

[5] Jacek Mandziuk and Krzysztof Mossakowski, “Looking InsideNeuralNetworks Trained to Solve Double-Dummy Bridge Problems,” Int.Proceeding 5th Game – On Computer Games: Artif. Intell., U.K. pp. 182-186, 2004.

[6] S.N Sivanandam and S.N Deepa, “Principles of Soft Computing”, Chapter 3, First Edition, pp.59-82, 2007.

[7] Jacek Mandziuk and Krzysztof Mossakowski, “Neural networks compete with expert human players in solving the double dummy bridge problem” Proc. of 5th Int. Conf. on Computational Intelligence and games, pp. 117-124, 2009.

[8] [Krzysztof Mossakowski, and Jacek Mandziuk, “Artificial neural networks for solving double dummy bridge problems”, In: AI and Soft computing vol. 3070, Springer, pp. 915–921, 2004.

[9] M. Sarkar, B. Yegnanarayana, and D. Khemani, “Application of neural network in contract bridge bidding,” in Proc. of National Conf. on Neural Networks and Fuzzy Systems, Anna University, Madras, pp. 144-151, 1995.

[10] M. Dharmalingam and R.Amalraj, “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, 2013.

[11] I. Frank and D.A. Basin, “Optimal play against best Defence: Complexity and Heuristics” in Lecture Notes in Computer Science Germany: Springer - Verlag, Vol.1558, pp 50-73, 1999.

[12] S.J.J. Smith and D.S. Nau, “An Analysis of Forward Pruning” in Proce of the National Conference on AI, pp 1386-1391, 1994.

[13] S.J.J. Smith, D.S. Nau and T.A. Throop, “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, 1998.

[14] M.L. Ginsberg, “GIB: Imperfect Information in a Computationally Challenging Game” Journal of Artificial Intelligence Research, vol. 14, pp 303-358, 2001.

[15] H. Francis, A. Truscott, and D. Francis, The Official Encyclopedia of Bridge, 5th ed. Memphis, TN: American Contract Bridge League Inc, 1994.

[16] A. Roshini and H. Anandakumar, "Hierarchical cost effective leach for heterogeneous wireless sensor networks," Advanced Computing and Communication Systems, 2015 International Conference on, Coimbatore, 2015, pp. 1-7.doi: 10.1109/ICACCS.2015.7324082

[17] S. Divya, H. A. Kumar and A. Vishalakshi, "An improved spectral efficiency of WiMAX using 802.16G based technology," Advanced Computing and Communication Systems, 2015 International Conference on, Coimbatore, 2015, pp. 1-4.doi: 10.1109/ICACCS.2015.7324098

[18] M. Suganya and H. Anandakumar, "Handover based spectrum allocation in cognitive radio networks," Green Computing, Communication and Conservation of Energy (ICGCE), 2013 International Conference on, Chennai, 2013, pp. 215-219.doi: 10.1109/ICGCE.2013.6823431

[19] H. Anandakumar and K. Umamaheswari, Supervised machine learning techniques in cognitive radio networks during cooperative spectrum handovers, Cluster Computing (2017), 1–11. doi: 10.1007/s10586-017-0798-3

[20] A.Amit and S.Markovitch, “Learning to bid in bridge” Machine Learning, vol.63, no.3, pp 287-327, 2006.

[21] T. Ando and T. Uehara, “Reasoning by agents in computer bridge bidding,” in Computers and Games, vol. 2063, pp. 346–364, 2001.

[22] T. Ando, N. Kobayashi and T. Uehara, “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, 2003.

[23] D. Khemani, “Planning with thematic actions.” in AIPS, pp. 287–292, 1994.

ISSN 2348 – 9928
Doi: 10.46532/ijaict-2020012 Published on 05 (4) 2020

[24] I. Frank, and D. A. Basin, "Strategies explained" in Proceeding 5th Game programming Workshop in Japan. pp1-8, 1999.

[25] Ali Awada, May Dehayni and Antoun Yaacoub, "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, 2011.

[26] Krzysztof Mossakowski and Jacek Mandziuk, "Neural networks and the estimation of hands strength in contract bridge," in Artificial Intelligence and Soft Computing ICAISC vol. 4029. Springer, pp. 1189-1198, 2006.

[27] B. Yegnanarayana, "Artificial Neural Networks" Chapter 4, pp.88-141, 2010.

[28] B.Yegnanarayana, D. Khemani, and M. Sarkar, "Neural networks for contract bridge bidding," Sadhana, vol. 21, no. 3, pp. 395-413, 1996.

[29] S.N Sivanandam and M Paulraj, "Introduction Artificial Neural Networks", Chapter 5,pp.119-147, 2011.

[30] M. Dharmalingam and R.Amalraj, "Supervised Learning in Imperfect Information Game", International Journal of Advanced Research in Computer Science, vol. 4, no.2, pp. 195-200, 2013.

[31] M. Dharmalingam and R.Amalraj, "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, 2013.

[32] S.E Fahlman,V Lebiere, The cascade-correlation learning architecture. Advances in Neural Information Processing, pp. 524-532, 1990.

[33] Dharmalingam,M., & Amalraj,R. (2014a). 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, 3(3),189-195.

[34] J. Elman, Finding structure in time, cognitive science 14(3), pp. 179-211, 1990.

[35] Dharmalingam,M., & Amalraj,R. (2014b). Supervised Elman Neural Network Architecture for Solving the Double Dummy Bridge Problem in Contract Bridge, International Journal of Science and Research, 3(6),2745-2750.

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