2011 International Conference on Computer and Software Modeling
IPCSIT vol.14 (2011) © (2011) IACSIT Press, Singapore

Idea Generation Algorithm bound for Game Strategy Design

Akhil Mangla[a] and Omkar Padwalkar[b+]
Dept. of Computer Engg, Smt.Kashibai Navale College of Engg, University of Pune, India

Abstract. Thought process is one of the key procedure that distinctly identifies humans rather animals from machines. Machines proposed to be effective computers have failed when it has come in light to learn and produce new ideas. The algorithm uses parallel set of processors with a systolic array for data processing. The recognition of optimal output based on associated weights and feedback provided by to self-organizing neural network. The network provided with an unsupervised learning is proposed to provide the machine with its own set of ideas thus resulting in achievement of "thought process" in Machines. Any system with capability and use of idea generation algorithm shall be with inherent learning and intelligence. Algorithm stand at elevated point in being of exceptional use for Game Strategy design and thus build scenarios for games and its simulations.

Keywords: IGA (Idea Generation Algorithm); Game; Strategy; Cricket; neural networks; lateral thinking, thought process, ANN (Artificial Neural Network);

1. Introduction

The Machines have the ability to generate output set for given input set. Machines have now evolved with time from listeners and implementers to learners and selectors. Artificial intelligence [1] is advancing as an integral field in computer science. It revolutionizes on the machine's ability to serve its master. Machines which once performed the primitive tasks of movement in unique direction, to the age of computers, where they provided sources to automated labour work with satellites monitoring earth and robots landing on moon and collecting samples. Now comes the age where logic of permutation and combination can be applied for the machine to come out on a complete set of output, along with ability to perform the task of selection via trained networks, which learn and select on the laterally correct input to give a final selected output. Artificial intelligence is a mimic to human intelligence and the trained network is a mimic to biological neural network present inside the human brain. Logical selections and reasoning are the two firm bases in the trained networks. These networks follow supervised and unsupervised approaches to make selection a more definite and defined process. The learning rules are obtained from theoretical theories and practical experiences are accumulated over the time. The experience may be unique but may form one of the effective parts in making the network learn and act synchronized with human actions. In this paper, we propose interpretation of a text and symbol based language definition for gaming strategy development. Games defined as a contest or competition devoted to involvement of beings or subjects into object scenarios for gain over loss, no gain or gain to all objectives. Strategy is defined as a procedure or set of procedure with some logical background to amplify gain levels or increase opponent loss in an activity such as games. Paper shows how IGA can be a handy game strategizing tool. One may also state the game to be a competition in which one individual succeeds via making choices at the expense of its opponent response. Strategies are not simple moves but a collaboration of sequence of move with a targeted response from the opponent, expected to be a probabilistically determinant outcome. Games and game theory branch out of applied mathematics and bear their application to social science, economics, biology, engineering, others. In the following

+ [1]akhilmangla@gmail.com, [2]sonupadwalkar@gmail.com

sections we proceed by stating Idea Generation Algorithm followed with the support theories and lead to a cumulative discussion of IGA modelling to develop strategizing techniques for Game Strategies.

2. IGA Based System

Figure shows the proposed complete system diagram for Idea Generation Algorithm [2, 3]. The system divided into two sections Module 1 is input module and Module 2 is output module. Module 1 input process begins with input feed-in, other operations performed on input elements and ending at M3 memory segment referenced by parallel processors, which form the entry element to Module 2. Parallel processors in a systolic array generate permutations for the input structures read from M2 memory segment and store their output to M3 storage memory. This newly stored output from M3 storage memory is input to trained network, where final selection process takes place. Trained network [4] output is available to user via console and later stored into the Database.

3. Game Theory

Game theory [5] attempts to mathematically capture behaviour in strategic situations, or games, in which an individual's success in making choices depends on the choices of others

3.1 Representation of Games

The games studied in game theory are well-defined mathematical objects, can be represented in four different forms.

- Extensive form

The extensive form can be used to formalize games with some important order. Games here are often presented as trees (as pictured to the left). Here each vertex (or node) represents a point of choice for a player. The player is specified by a number listed by the vertex. The lines out of the vertex represent a possible action for that player. The payoffs are specified at the bottom of the tree.

- Normal form

The normal (or strategic form) game is usually represented by a matrix which shows the players, strategies, and payoffs (see the example to the right). More generally it can be represented by any function that associates a payoff for each player with every possible combination of actions.

- Characteristic function form

In cooperative games with transferable utility no individual payoffs are given. Instead, the characteristic function determines the payoff of each coalition. The standard assumption is that the empty coalition obtains a payoff of 0. The origin of this form is to be found in the seminal book of von Neumann and Morgenstern who, when studying coalitional normal form games, assumed that when a coalition C forms, it plays against the complementary coalition ($N \setminus C$) as if they were playing a 2-player game. The equilibrium payoff of C is characteristic. Now there are different models to derive coalitional values from normal form games, but not all games in characteristic function form can be derived from normal form games. Formally, a characteristic function form game (also known as a TU-game) is given as a pair (N,v), where N denotes a set of players and $v : 2^N \longrightarrow \mathbb{R}$ is a characteristic function. The characteristic function form has been generalized to games without the assumption of transferable utility.

- Partition function form

The characteristic function form ignores the possible externalities of coalition formation. In the partition function form the payoff of a coalition depends not only on its members, but also on the way the rest of the players are partitioned.

3.2 Types of Games

Cooperative or non-cooperative, Symmetric and asymmetric, Zero-sum and non-zero-sum, Simultaneous and sequential, Perfect information and imperfect information, Infinitely long games, Discrete and continuous games, One-player and many-player games and Meta-games.

4. Game Strategy Comprehensive Approach

Object oriented approach defines the world as a class of the general category. The world contains many prominent entities which have their defined conduct and behaviour, existing in it and adopting to all the applicable methods of the world class. Games are previously defined as to be entities of significance. The characteristic value of any game states its conduct in terms of permissible actions and outcomes, subject or player involvement, therefore a complete packaged entity. Thus, we could say it to belong to an object world of the general world class. To bring clarity to this issue we appropriate its behaviour by saying humans as entities of the real world and games as entities of the real world, depicted as Venn diagram [7] in figure 1. Thus all applicable methods to game world are valid to game entities.

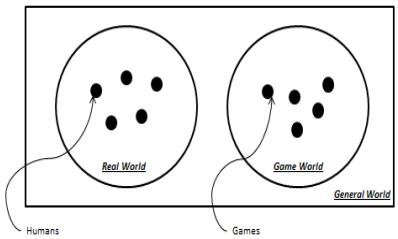

Figure 1. World Entity Venn Diagram

Computers are machines which understand binary language in hardware and then come other middle level and high level languages like C, Java, others which are boot strapped to provide a final apprehension. Real world entities use many languages for communication and expression namely English has been proposed and use as a universal language of communication. Nevertheless humans do communicate in sign language to communicate or express as in Handshake, salute, eye blink, and other physical notions. Game entities are physical in nature either a motion of its objects or an audio yield. Analogous to real world we defined the game world, using similar tactic we develop language and hierarchy or Game expression and statement. Symbols and objects have been used to put together statements to provide an abstraction and maintain good level privacy which comes with all it subclass properties of confidentiality and put together a spectrum or bunch of information together. This leads to strategy the way to go about a game logically and lead to a scenario for justification of actions. Game actions may or may not have justification all the time. They are perception of its players and referee. Guide and Coach also play a significant role in providing structures to these encapsulated languages. Thus, such development of results helps us in defining a boot strap to structure a language for game strategies for implementation in approaches discussed in following section. The method focuses onto use of Characteristic functions to represent game strategies as it helps in:-

- Ease of mathematical logic interpretation, Results highlighted
- Domain and Range specification provide clarity
- Effective relativity with the grammar used in logic implementation in Computers
- Idea Generation Algorithm support as structured approach and provides mapping functions.
- Effortless bootstrapping to implement rules in English language to a computer language like C, others

Thus, use of this methodology at the Input Generate Processor (IGP) shown in Figure 1 of IGA block diagram. This helps in providing combinatorics structure and expressing knowledge and learning rule for a cumulative combinatorics structure design.

5. Game Strategy Atop IGA

5.1. Strategy Protocol

Strategy involves a great deal of knowledge of the game and various factors involved in it. Thus, we classify strategic levels to which a game may be set up to. Players are crucial to any game, skillset, capacity and ability of a player form its profile which may again act as an advantageous tool to profile based strategies like Best First, Eliminate the weak, others. These levels are depicted in figure 5 and described as follows:-

- Base Strategy: Found upon the rules and use of information locally available. Does not involve player profiling. Here, one might even relatively use strategic data available for some primitive game

to set high end games by differentiating it with respect to the primitive game. Primitive game may be a board game like: Tic Tac Toe, Card Games, others.

- Agent Strategy: A strategic use of player profiles of both opponent and one's own team to have strategic gains. Intended to be of value once base strategy has been efficiently utilized.
- Component Strategy: Use of support material by allocation of resources and bringing the best to use. Enhances agent capacity (Player may be referred as game agent). This strategy requires bot base and agent strategy to have reached sufficient levels of outcome.

Strategy levels so discussed are sequential to each other but the intensity and dynamic nature of a game might lead to containment as shown in figure 2. Lead to simultaneous structure definition for deployment in game. We define strategy protocol from the point of view of machine thought process so an embryonic design should provide sufficient insight for virtual player or machine entity playing the game.

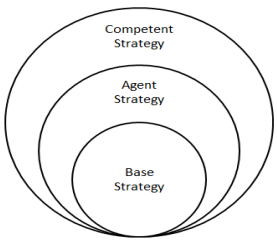

Figure 2. Layered Strategy level for Accuracy

5.2. Algorithm Adaption

In this sub section we directly shift our focus to define various steps to adapt the algorithm and deploy it for game strategizing. Thus instigating algorithm as follows:-

1. Start after obtaining initial observation data via sensors or manual layouts.
2. Feed the input to the IGA input storage memory segment M1
   i. Sport Type
   ii. Player Strength involved
   iii. Strategic level desired for accuracy
3. IGP fills internal M2 Memory segment with new structures of defined type as in knowledge base. Components necessary to be made available to IGP:-
   i. Learning Rule: Built up of Game Constraints Defined As Characteristic Function
   ii. Knowledge: about game to act as Support Data
   iii. Base strategy: From primitive/Static games Like Tic Tac toe, card Games, Others.
   iv. Cumulative Inference Mechanisms
4. Processing of data remains same as discussed in the actual algorithm in section II. Expected outcomes so far are:-
   i. Single, Dual or Multiple Strategy Combinatorics Structures.
   ii. Randomization Unit to maintain integrity of the algorithm.
5. The processors $P_1,P_2....P_n$ are a part of the systolic array and they operate to give patterns comprising of:-
   i. Logical strategy permutations of base strategy i.e. primitive games.
   ii. Player profile pattern and absent case consideration.
   iii. Component effect patterns on previous strategic results.
6. The ANN operates as a Self Organizing Map with a back propagation of error/frequency. The ANN is based upon the target data provided to the system in its initial stages.
7. The feedback generated helps in improving the output and by default feedback shall be unit delay or zero error for first output by the ANN
8. Display the output to the user for
   i. Selection by user increment the frequency count by 2
   ii. Rejection by user decrement the frequency count by 1
9. At the end, store the output into the database with its frequency update. This might act as a weighted unit to strengthen the system result.
10. Stop

Hence uniqueness and global acceptance of this algorithm is of high connotation. Base strategy involves primitive games and users of this system may is the same system to generate strategic data for the primitive games and thence simplify the results via data analysis and store at the knowledge base to be used for dynamic time games like cricket, football, others. Figure 6 summarizes the adapted algorithm and gives us the overall idea of system's flow. Hence we saw an evolutionary approach in algorithm design for game strategizing.

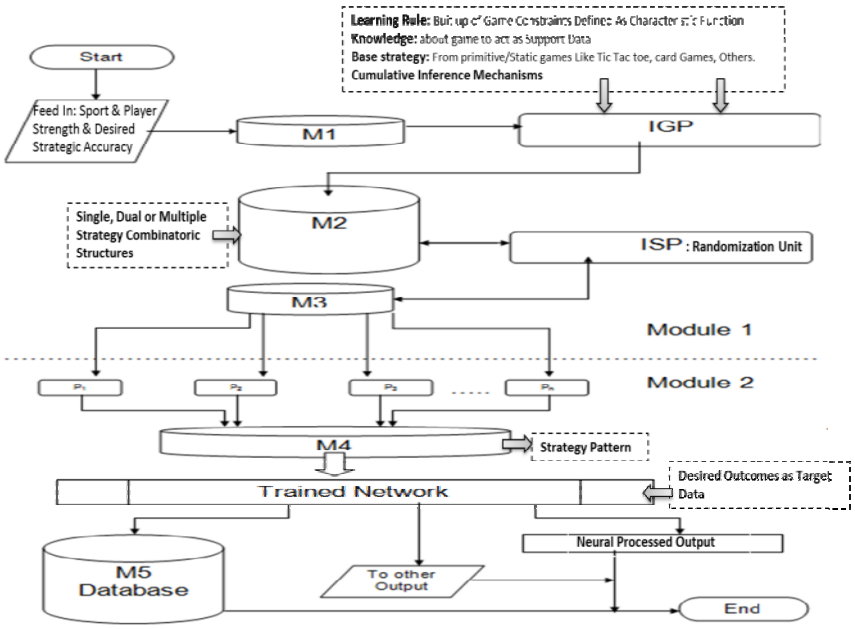

Figure 4. Game Strategy Flow atop Idea Generation Algorithm

6. Application Relevance

Any technology is an evolution of current techniques leading to definition of application domains to which it could be mapped to. Here briefly some application of the adapted algorithm is emphasized:

- Computer player strategy definition in both static and dynamic game play
- Testing lateral scenarios inclusive of all cases
- Creating simulation and test it on basis of new strategies building from different viewpoints.
- Helping player profile deployment strategizing.

This helps in identifying the applicability at deterministic levels in the hierarchy identified in the tree of players from real world to game world. The situation is softened by the fact that algorithm here is targeted to provide fresh approaches.

7. Case Study: Cricket

Cricket is a dynamic sport with time constraints or rule constraints. It's one of the most critical feature is fielding which involves an effective strategic approach. Thus we study fielding and find out that it's a structure plot of field where in players are to be placed around it centre where the pitch is built. Thus using Idea Generation Algorithm to strategize to support an effective fielding by a team. Cricket is a two team game with eleven players on each side. At the time of fielding only ten players field and one bowls. Out of ten one guards the bats man wicket. Applying IGA we intend to get a base strategy. Thus, using a primitive game would be IGA approach based on constraints and Cumulative mechanism. Even on user's suggestion a primitive game might be selected to be applicable in this scenario. Hence. Tic Tac Toe a game which may be solved using a min max search processing over a pattern of moves. The Artificial Neural Network might be handy in Min-Max Search and other algorithm modules to generate this pattern. Figure 5 helps us in visualizing the Min-Max tree for Tic Tac Toe. One may observe figure 5 and realize that a cricket field has four symmetry lines and a division of the ground into octants. These octants can be thought of as containers of a 3X3 matrix. This matrix is support matrix for tic tac toe.

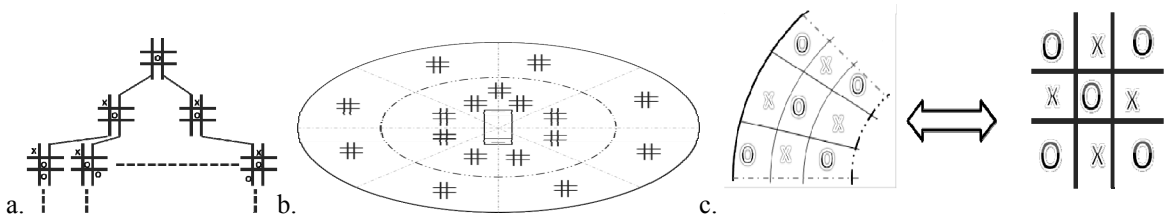

Figure 5. a.Min Max Search b.Cricket Ground Strategic Division for Base Strategy and Variable 3X3 Matrix realization c. Octant-TicTacToe

Hence a machine player might realize his placement octant as the area to field and prevent hits from any gain. In figure 5 the octant comparison is shown in comparison with Tic Tac Toe board. If X is a player's position and O's represent boll trajectory or drop point. Thus help the player to dynamically realize his position via a strategy generated for it using IGA. It provides a matching, Optimal and reliable solution in most cases. Thus a player might find optimal solutions with use of IGA or even develop unique scenarios to have gain or no gain scenarios. The dynamic support over static board games is a relative concept utilized in IGA. Similarly for player profiling IGA optimizes via use of Card Game for instance a player good in batting might be an Ace card and a player good bowling a King and a player good at wicket keeping a seven card. Thus as per a card game one might gamble with a hidden logic to rule out other player's strategy and this gamble scenario may be developed with the help of Idea Generation Algorithm.

8. Conclusions

The paper highlighted the IGA effectiveness in Game strategy development and design. There by opening new avenues to explore in machine thought process. We saw a case study where a dynamic sport like cricket is tackled for fielding using IGA's strategy protocol along with adaption algorithm basing onto the Tic Tac Toe board game to achieve the desired goals. IGA leads to a new beginning in achieving a task where in robots become more interactive with their masters and are able to work in remote location under an unsupervised environment. The future works aim at building simpler mathematical computational models and thereby redefining the solutions altogether. The future work aims at defining effective neural network based selection system with abstract layers for neural processing. An application based on testing scenarios simulation using Idea generation Algorithm is looked forward to. The algorithm is simplistic in design and computation.

9. References

[1] Russell Norvig. Artificial Intelligence. Second Edition, Pearson Prentice Hall.

[2] U. A. Jogalekar and A. Mangla. Idea Generation Algorithm Based System. ICCSIT 3rd IEEE Conference, Chengdu, China,2010

[3] U. A. Jogalekar and A. Mangla. Idea Generation Algorithm Based System. ICCRCrd IEEE Conference, Delhi, India,2011

[4] B. Karlik, Harnid Torpi and M. Alc. A Fuzzy-Neural Approach for the Characterisation of the Active Microwave Devices. 2002 12th Int. Crimean Conference "Microwave &Telecommunication Technology" (CriMiCo'2002). 9-13 Septcmber, Scvastopol. Crimea, Ukraine

[5] J. R. Green. Mas-Colell, Andreu; Whinston, Michael D. (1995), Microeconomic theory, Oxford University Press, ISBN 978-0-19-507340-9

[6] J. D. Williams, (1954) (PDF), The Compleat Strategyst: Being a Primer on the Theory of Games of Strategy, Santa Monica: RAND Corp., ISBN 9780833042224

[7] R. Pressman. Software Engineering – A Practitioners Approach. The 6th Edition (Paperback) ,Tata McGraw Hill Publications, 2004, ISBN 007-124083-7.