www.myreaders.info/ , RC Chakraborty, e-mail rcchak@gmail.com , June 01, 2010

www.myreaders.info/html/artificial_intelligence.html

Game Playing

Artificial Intelligence

Game Playing, topics : Overview, definition of game, game theory, relevance of game theory and game plying, Glossary of terms – game, player, strategy, zero-Sum game, constant-sum game, nonzero-sum game, Prisoner's dilemma, N-Person game, utility function, mixed strategies, expected payoff, Mini-Max theorem, saddle point; taxonomy of games. Mini-Max search procedure : formalizing game - general and a Tic-Tac-Toe game, evaluation function ; MINI-MAX technique : game trees, Mini-Max algorithm. Game Playing with Mini-Max : example of Tic-Tac-Toe - moves, static evaluation, back-up the evaluations and evaluation obtained. Alpha-Beta Pruning : Alpha-cutoff, Beta-cutoff.

Game Playing

Artificial Intelligence

Topics

(Lectures 29, 30, 2 hours)

Game Playing

What is Game ?

- The term *Game* means a sort of *conflict* in which **n** individuals or groups (known as players) participate.
- Game theory denotes games of *strategy*.
- John von Neumann is acknowledged as father of game theory. Neumann defined Game theory in 1928 and 1937 and established the mathematical framework for all subsequent theoretical developments.
- Game theory allows decision-makers (players) to cope with other decision-makers (players) who have different purposes in mind. In other words, players determine their own strategies in terms of the strategies and goals of their opponent.
- Games are integral attribute of human beings.
  Games engage the *intellectual faculties* of humans.
- If computers are to mimic people they should be able to play games.

1. Over View

Game playing, besides the topic of attraction to the people, has close relation to "intelligence", and its well-defined states and rules.

The most commonly used AI technique in game is **"Search".**

A "Two-person zero-sum game" is most studied game where the two players have exactly opposite goals. Besides there are "Perfect information games" (such as chess and Go) and "Imperfect information games" (such as bridge and games where a dice is used).

Given sufficient time and space, usually an optimum solution can be obtained for the former by exhaustive search, though not for the latter. However, for many interesting games, such a solution is usually too inefficient to be practically used.

Applications of game theory are wide-ranging. Von Neumann and Morgenstern indicated the utility of game theory by linking with economic behavior.

- **Economic models** : For markets of various commodities with differing numbers of buyers and sellers, fluctuating values of supply and demand, seasonal and cyclical variations, analysis of conflicts of interest in maximizing profits and promoting the widest distribution of goods and services.
- **Social sciences** : The n-person game theory has interesting uses in studying the distribution of power in legislative procedures, problems of majority rule, individual and group decision making.
- **Epidemiologists** : Make use of game theory, with respect to immunization procedures and methods of testing a vaccine or other medication.
- **Military strategists** : Turn to game theory to study conflicts of interest resolved through "battles" where the outcome or payoff of a war game is either victory or defeat.

1.1 Definition of Game

- A game has at least *two players*.

  Solitaire is not considered a game by game theory.

  The term 'solitaire' is used for single-player games of concentration.

- An instance of a game begins with a player choosing from a set of specified *(game rules)* alternatives. This choice is called a *move*.

- After first move, the new situation determines which player to make next move and alternatives available to that player.
  - In many board games, the next move is by other player.
  - In many multi-player card games, the player making next move depends on who dealt, who took last trick, won last hand, etc.

- The moves made by a player may or may not be known to other players. Games in which all moves of all players are known to everyone are called games of *perfect information*.
  - Most board games are games of perfect information.
  - Most card games are not games of perfect information.

- Every instance of the game must end.

- When an instance of a game ends, each player receives a payoff.

  A *payoff* is a value associated with each player's final situation.

  A *zero-sum game* is one in which elements of payoff matrix sum to zero.

  In a typical zero-sum game :
  - win = 1 point,
  - draw = 0 points, and
  - loss = -1 points.

1.2 Game Theory

Game theory does not prescribe a way or say how to play a game.
Game theory is a set of ideas and techniques for analyzing conflict situations between two or more parties. The outcomes are determined by their decisions.

General game theorem : In every two player, zero sum, non-random, perfect knowledge game, there exists a perfect strategy guaranteed to at least result in a tie game.

The frequently used terms :

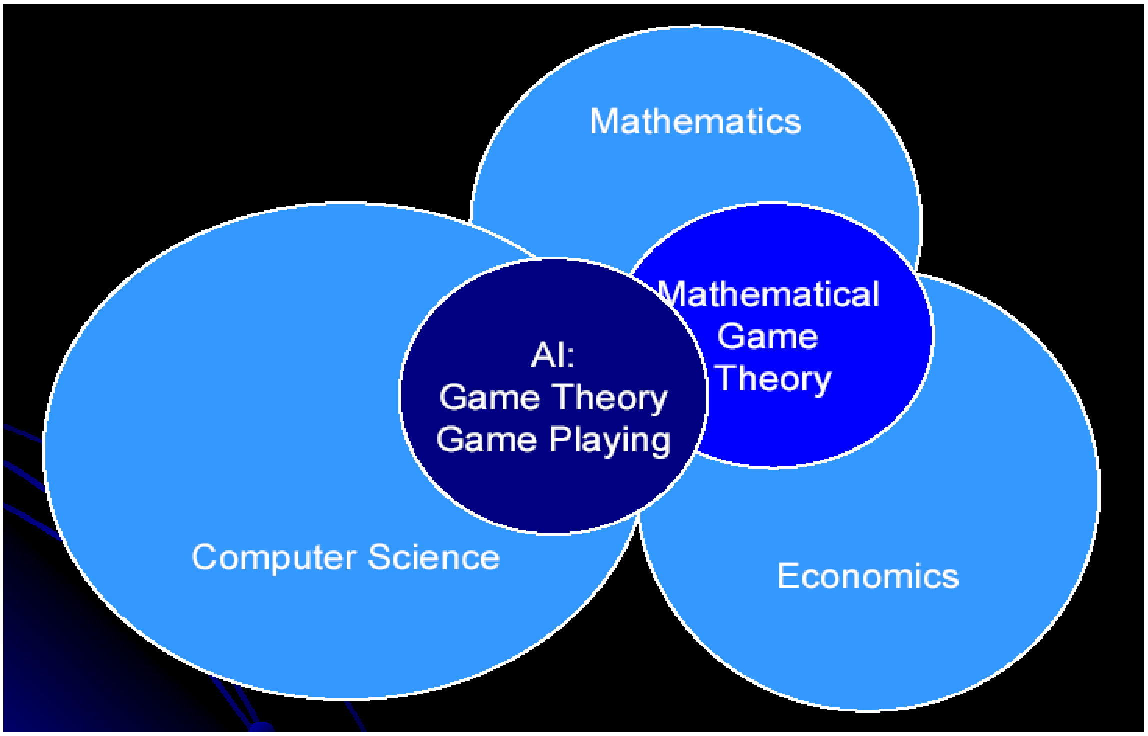
- The term **"game"** means a sort of conflict in which **n** individuals or groups (known as players) participate.
- A list of **"rules"** stipulates the conditions under which the game begins.
- A game is said to have **"perfect information"** if all moves are known to each of the players involved.
- A **"strategy"** is a list of the optimal choices for each player at every stage of a given game.
- A **"move"** is the way in which game progresses from one stage to another, beginning with an initial state of the game to the final state.
- The total **number of moves** constitute the entirety of the game.
- The **payoff or outcome**, refers to what happens at the end of a game.
- **Minimax** - The least good of all good outcomes.
- **Maximin** - The least bad of all bad outcomes.

The primary game theory is the **Mini-Max Theorem**. This theorem says :

"If a Minimax of one player corresponds to a Maximin of the other player, then that outcome is the best both players can hope for."

1.3 Relevance Game Theory and Game Plying

How relevant the Game theory is to Mathematics, Computer science and Economics is shown in the Fig below.



Game Playing

- Games can be Deterministic or non-deterministic.
- Games can have perfect information or imperfect information.

Games	Deterministic	Non- Deterministic
Perfect information	Chess, Checkers, Go, Othello, Tic-tac-toe	Backgammon, Monopoly
Imperfect information	Navigating a maze	Bridge, Poker, Scrabble

1.4 Glossary of terms in the context of Game Theory

- **Game**

  Denotes games of strategy. It allows decision-makers (players) to cope with other decision-makers (players) who have different purposes in mind. In other words, players determine their own strategies in terms of the strategies and goals of their opponent.

- **Player**

  Could be one person, two persons or a group of people who share identical interests with respect to the game.

- **Strategy**

  A player's strategy in a game is a complete plan of action for whatever situation might arise. It is the complete description of how one will behave under every possible circumstance. You need to analyze the game mathematically and create a table with **"outcomes"** listed for each strategy.

A two player strategy table

Players Strategies	Player A Strategy 1	Player A Strategy 2	Player A Strategy 3	etc
Player B Strategy 1	Tie	A wins	B wins	. . .
Player B Strategy 2	B wins	Tie	A wins	. . .
Player B Strategy 3	A wins	B wins	Tie	. . .
etc	. . .	. . .	. . .	. . .

- **Zero-Sum Game**

It is the game where the interests of the players are diametrically opposed. Regardless of the outcome of the game, the winnings of the player(s) are exactly balanced by the losses of the other(s).
No wealth is created or destroyed.

There are two types of zero-sum games:

- Perfect information zero-sum games
- General zero-sum games

The difference is the amount of information available to the players.

Perfect Information Games :

Here all moves of all players are known to everyone.
e.g., Chess and Go;

General Zero-Sum Games :

Players must choose their strategies simultaneously, neither knowing what the other player is going to do.
e.g., If you play a single game of chess with someone, one person will lose and one person will win. The win (+1) added to the loss (-1) equals zero.

- **Constant-Sum Game**

  Here the algebraic sum of the outcomes are always constant, though not necessarily zero.

  It is strategically equivalent to zero-sum games.

- **Nonzero-Sum Game**

  Here the algebraic sum of the outcomes are not constant. In these games, the sum of the pay offs are not the same for all outcomes.

  They are not always completely solvable but provide insights into important areas of inter-dependent choice.

  In these games, one player's losses do not always equal another player's gains.

  The nonzero-sum games are of two types:

  Negative Sum Games (Competitive) : Here nobody really wins, rather, everybody loses. Example - a war or a strike.

  Positive Sum Games (Cooperative) : Here all players have one goal that they contribute together. Example - an educational game, building blocks, or a science exhibit.

■ Prisoner's Dilemma

It is a two-person nonzero-sum game. It is a non cooperative game because the players can not communicate their intentions.

Example : The two players are partners in a crime who have been captured by the police. Each suspect is placed in a separate cell and offered the opportunity to confess to the crime.

Now set up the payoff matrix. The entries in the matrix are two numbers representing the payoff to the first and second player respectively.

Players	2nd player Not Confess	2nd player Confess
1st player Not Confess	5 , 5	0 , 10
1st player Confess	10 , 0	1 , 1

- If neither suspect confesses, they go free, and split the proceeds of their crime, represented by **5** units of payoff for each suspect.
- If one prisoner confesses and the other does not, the prisoner who confesses testifies against the other in exchange for going free and gets the entire **10** units of payoff, while the prisoner who did not confess goes to prison and gets nothing.
- If both prisoners confess, then both are convicted but given a reduced term, represented by **1** unit of payoff : it is better than having just the other prisoner confess, but not so good as going free.

This game represents many important aspects of game theory.

No matter what a suspect believes his partner is going to do, it is always best to confess.

- If the partner in the other cell is not confessing, it is possible to get **10** instead of **5** as a payoff.
- If the partner in the other cell is confessing, it is possible to get **1** instead of **0** as a payoff.
- Thus the pursuit of individually sensible behavior results in each player getting only **1** as a payoff, much less than the **5** which they would get if neither confessed.

This conflict between the pursuit of individual goals and the common good is at the heart of many game theoretic problems.

- **N-Person Game**

  Involve more than two players.

  Analysis of such games is more complex than zero-sum games. Conflicts of interest are less obvious.

  Here, what is good for player-1 may be bad for player-2 but good for player-3. In such a situation **coalitions** may form and change a game radically. The obvious questions are :

  - how would the coalition form ?
  - who will form coalitions ?
  - would the weak gang up against the strong ? or
    would weak players make an alliance with a strong ?

  In either case it could be looked at as two coalitions, and so effectively a two person game.

- **Utility Function**

  It is quantification of person's preferences respect to certain objects. In any game, utility represents the motivations of players. A utility function for a given player assigns a number for every possible outcome of the game with the property that a higher number implies that the outcome is more preferred.

- **Mixed Strategies**

A player's strategy in a game is a complete plan of action for whatever situation might arise. It is a complete algorithm for playing the game, telling a player what to do for every possible situation throughout the game.

A **Pure strategy** provides a complete definition of how a player will play a game. In particular, it determines the move a player will make for any situation they could face. A player's strategy set is the set of pure strategies available to that player.

A **Mixed strategy** is an assignment of a probability to each pure strategy. This allows for a player to randomly select a pure strategy. Since probabilities are continuous, there are infinitely many mixed strategies available to a player, even if their strategy set is finite.

A mixed strategy for a player is a probability distribution on the set of his pure strategies.

Suppose a player has only a finite number, m, of pure strategies, then a mixed strategy reduces to an m-vector, $X = (x_1, \dots, x_m)$, satisfying

$$x_i \geq 0 \; , \; \sum_{i=1}^{m} x_i = 1$$

Now denote the set of all mixed strategies for player-1 by X, and the set of all mixed strategies for player-2 by Y.

$$X = \{ x = (x_i , \dots , x_m) : x_i \geq 0 \; , \; \sum_{i=1}^{m} x_i = 1$$

$$Y = \{ y = (y_i , \dots , y_n) : y_i \geq 0 \; , \; \sum_{i=1}^{n} y_i = 1$$

[continued in the next slide]

[continuing from previous slide- mixed strategy]

Expected Payoff

Suppose that player1 and player2 are playing the matrix game **A**.

If player1 chooses the mixed strategy x_i, and

player2 chooses the mixed strategy y_j,

then the expected payoff a_{ij} will be computed by

$$\begin{array}{c} \\ x_1 \\ \\ \\ x_m \end{array} \begin{array}{cccc} y_1 & \cdot\cdot & \cdot\cdot & y_n \\ \end{array} \left\{ \begin{array}{cccc} a_{11} & \cdot\cdot & \cdot\cdot & a_{1n} \\ \cdot & \cdot\cdot & \cdot\cdot & \cdot \\ \cdot & \cdot\cdot & \cdot\cdot & \cdot \\ a_{m1} & \cdot\cdot & \cdot\cdot & a_{1n} \end{array} \right\} \rightarrow \left\{ \begin{array}{ccc} x_1 a_{11} y_1 & \cdot\cdot\cdot\cdot & x_1 a_{1n} y_n \\ \cdot & \cdot\cdot\cdot\cdot & \cdot \\ \cdot & \cdot\cdot\cdot\cdot & \cdot \\ x_m a_{m1} y_1 & \cdot\cdot\cdot\cdot\cdot & x_m a_{mn} y_n \end{array} \right\}$$

That is $$A(x, y) = \sum_{i=1}^{m} \sum_{i=1}^{n} x_i \, a_{ij} \, y_j$$

or in matrix form $$A(x, y) = x \, A \, y^T$$

This can be thought as a weighted average of the expected payoffs.

Player1's Maximin Strategy :

Assume that player1 uses x, and player2 chooses y to minimize $A(x, y)$; player1's expected gain will be

$$V(x) = \min_j x \; A_j$$ where A_j is j^{th} column of the matrix A

Player1 should choose x so as to maximize $V(x)$:

$$V_1 = \max_{x \in X} \min_j \; x \; A_j$$

Such a strategy x is player1's maximin strategy.

Player2's Minimax Strategy :

Player2's expected loss ceiling will be

$$V(y) = \max_i A_i \, y^T$$ where A_i is i^{th} row of the matrix A

Player2 should choose y so as to minimize $V(y)$:

$$V_2 = \min_{y \in Y} \max_i \; A_i \, y^T$$

Such a strategy y is player2's minimax strategy.

Thus we obtain the two numbers V_1 and V_2. These numbers are called the values of the game to player1 and player2, respectively.

- **Mini-Max Theorem** *[Ref. previous slide]*

  It is a concept of Games theory.

  Players adopt those strategies which will maximize their gains, while minimizing their losses. Therefore the solution is, the best each player can do for him/herself in the face of opposition of the other player.

  In the previous slide, the two numbers V_1 and V_2 are the values of the game to player1 and player2, respectively.

  Mini-Max Theorem : $V_1 = V_2$

  The minimax theorem states that for every two-person, zero-sum game, there always exists a mixed strategy for each player such that the expected payoff for one player is same as the expected cost for the other.

  This theorem is most important in game theory. It says that every two-person zero-sum game will have optimal strategies.

■ Saddle Point

An element a_{ij} of a matrix is described as a saddle point if it equals both the *minimum of row i* and the *maximum of column j*.

A **Saddle Point** is a payoff that is simultaneously a *row minimum* and a *column maximum*. In a game matrix, if the element a_{ij} corresponds to a saddle point then it is the largest in its column and the smallest in its row (LCSR).

The value a_{ij} is called the **optimal payoff**.

Examples : Three game payoff matrixes

A (rows i, columns j)

5	1	3
3	2	4
-3	0	1

one saddle point $a_{22} = 2$

B (rows i, columns j)

4	3	5
0	1	0
6	3	9

two saddle points $a_{12} = 3$, $a_{32} = 3$

C (rows i, columns j)

-1	1
1	-1

no saddle point

- The game payoff matrix ***B*** shows that saddle point may not be unique, but the optimal payoff is unique.
- A payoff matrix needn't have a saddle point, but if it does, then the usually minimax theorem is easily shown to hold true.

The Zero Sum Games modeled in matrices are solved by finding the saddle point solution.

[continued in the next slide]

[continuing from the previous slide – Saddle point]

How to find Saddle Point ?

Below shown two examples : A and B

A

	j			min
i	5	1	3	1
	3	2	4	2
	-3	0	1	-3
Max	5	2	4	

one saddle point
$a_{22} = 2$

B

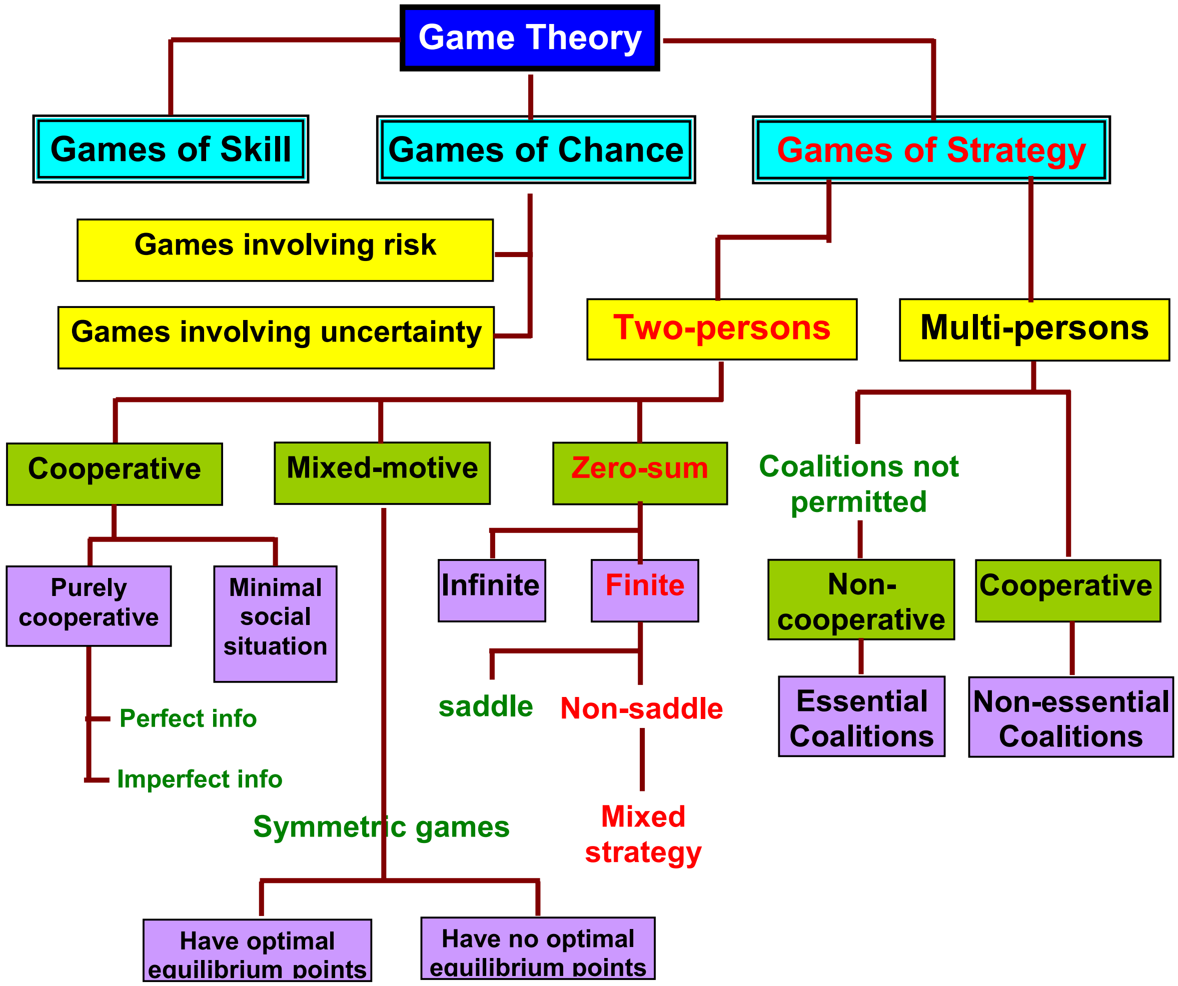
	j			min
i	4	3	5	3
	0	1	0	0
	6	3	9	3
Max	6	3	9	

two saddle points

1. Find your min payoff : Label each row at its end with its minimum payoff. This way you'll define your worst case scenarios when choosing a strategy.
2. Find your opponent's min payoff : Label each column at its bottom with its maximum payoff. This will show the worst case scenarios for your opponent.
3. Find out which is the highest value in the series of minimum values. It is at one place as **2** in matrix A and at two places as **3** in matrix B.
4. Then find out which is the lowest value in the series of maximum values. It is as 2 in matrix A and as 3 in matrix B.
5. Find out if there is a minimax solution : If these two values match, then you have found the saddle point cell.
6. If there is a minimax solution, then it is possible that both agents choose the corresponding strategies. You and your opponent are maximizing the gain that the worst possible scenario can drive.

1.5 Taxonomy of Games

All that are explained in previous section are summarized below.



2. The Mini-Max Search Procedure

Consider two players, zero sum, non-random, perfect knowledge games.
Examples: Tic-Tac-Toe, Checkers, Chess, Go, Nim, and Othello.

2.1 Formalizing Game

A general and a Tic-Tac-Toe game in particular.

- **Consider 2-Person, Zero-Sum, Perfect Information**
  - ‡ Both players have access to complete information about the state of the game.
  - ‡ No information is hidden from either player.
  - ‡ Players alternately move.
- **Apply Iterative methods**
  - ‡ Required because search space may be large for the games to search for a solution.
  - ‡ Do search before each move to select the next best move.

- **Adversary Methods**
  - ‡ Required because alternate moves are made by an opponent , who is trying to win, are not controllable.

- **Static Evaluation Function f(n)**
  - ‡ Used to evaluate the "goodness" of a configuration of the game.
  - ‡ It estimates board quality leading to win for one player.
  - ‡ **Example:** Let the board associated with node **n** then
    - ◇ **If f(n) = large +ve value**
      means the board is good for me and bad for opponent.
    - ◇ **If f(n) = large -ve value**
      means the board is bad for me and good for opponent.
    - ◇ **If f(n) near 0**
      Means the board is a neutral position.
    - ◇ **If f(n) = +infinity**
      means a winning position for me.
    - ◇ **If f(n) = -infinity**
      means a winning position for opponent.

- **Zero-Sum Assumption**

  ‡ One player's loss is the other player's gain.

  ‡ Do not know how our opponent plays ?;
  So use a single evaluation function to describe the goodness of a board with respect to both players.

  ‡ **Example :** Evaluation Function for the game **Tic-Tac-Toe** :

  f(n) = [number of 3-lengths open for me] − [number of 3-lengths open for opponent]

  where a 3-length is a complete row, column, or diagonal.

2.2 MINI-MAX Technique

For Two-Agent , Zero-Sum , Perfect-Information Game.

The Mini-Max procedure can solve the problem if sufficient computational resources are available.

- **Elements of Mini-Max technique**
  - ‡ Game tree (search tree)
  - ‡ Static evaluation,
    e.g., +ve for a win, -ve for a lose and 0 for a draw or neutral.
  - ‡ Backing up the evaluations, level by level, on the basis of opponent's turn.

- **Game Trees :** description

  ‡ **Root node** represents board configuration and decision required as to what is the best single next move.

  If my turn to move, then the root is labeled a **MAX node** indicating it is my turn;
  otherwise it is labeled a **MIN node** to indicate it is my opponent's turn.

  ‡ **Arcs** represent the possible legal moves for the player that the arcs emanate from.

  ‡ At each level, the tree has nodes that are all MAX or all MIN;

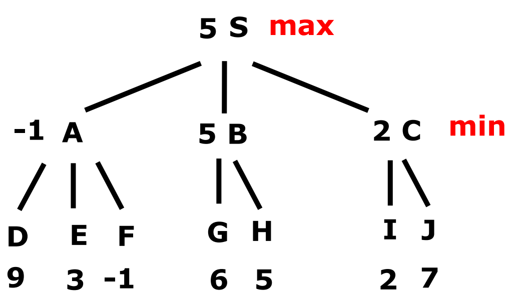
  ‡ Since moves alternate, the nodes at **level** i are of opposite kind from those at **level** $i+1$.

■ Mini-Max Algorithm

- Searching Game Tree using the Mini-Max Algorithm
- Steps used in picking the next move:
  - Since it's my turn to move, the start node is MAX node with current board configuration.
  - Expand nodes down (play) to some depth of look-ahead in the game.
  - Apply evaluation function at each of the leaf nodes
  - "Back up" values for each non-leaf nodes until computed for the root node.
  - At MIN nodes, the backed up value is the minimum of the values associated with its children.
  - At MAX nodes, the backed up value is the maximum of the values associated with its children.

Note: The process of "backing up" values gives the optimal strategy, that is, both players assuming that your opponent is using the same static evaluation function as you are.

- **Example :** Mini-Max Algorithm



- The MAX player considers all three possible moves.
- The opponent MIN player also considers all possible moves.
- The evaluation function is applied to leaf level only.

- **Apply Evaluation function :**
  - ‡ Apply static evaluation function at leaf nodes & begin backing up.
  - ‡ First compute backed-up values at the parents of the leaves.
    - ◇ Node **A** is a MIN node ie it is the opponent's turn to move.
    - ◇ **A's** backed-up value is **-1** ie min of **(9, 3, -1)**, meaning if opponent ever reaches this node, then it will pick the move associated with the arc from **A** to **F**.
    - ◇ Similarly, **B's** backed-up value is **5** and **C's** backed-up value is **2**.
  - ‡ Next, backup values to next higher level,
    - ◇ Node **S** is a MAX node ie it's our turn to move.
    - ◇ look best on backed-up values at each of **S's** children.
    - ◇ the best child is **B** since value is **5** ie max of **(-1, 5, 2)**.
    - ◇ So the minimax value for the root node **S** is **5**, and the move selected is associated with the arc from **S** to **B**.

3. Game Playing with Mini-Max - Tic-Tac-Toe

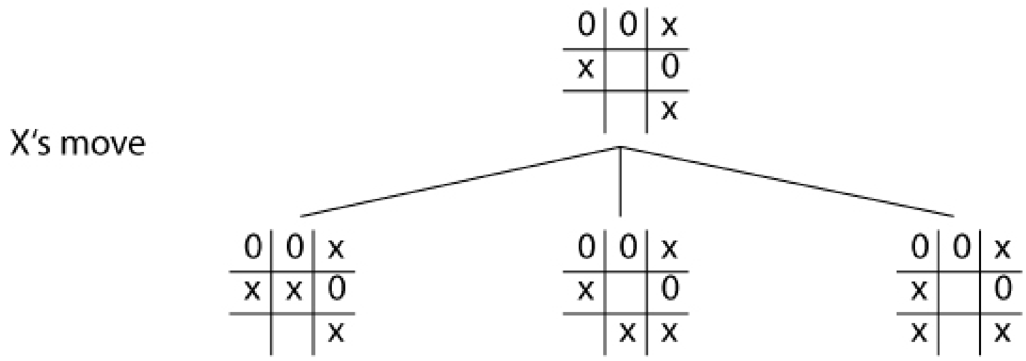
Here, Minimax Game Tree is used that can program computers to play games.

There are two players taking turns to play moves.

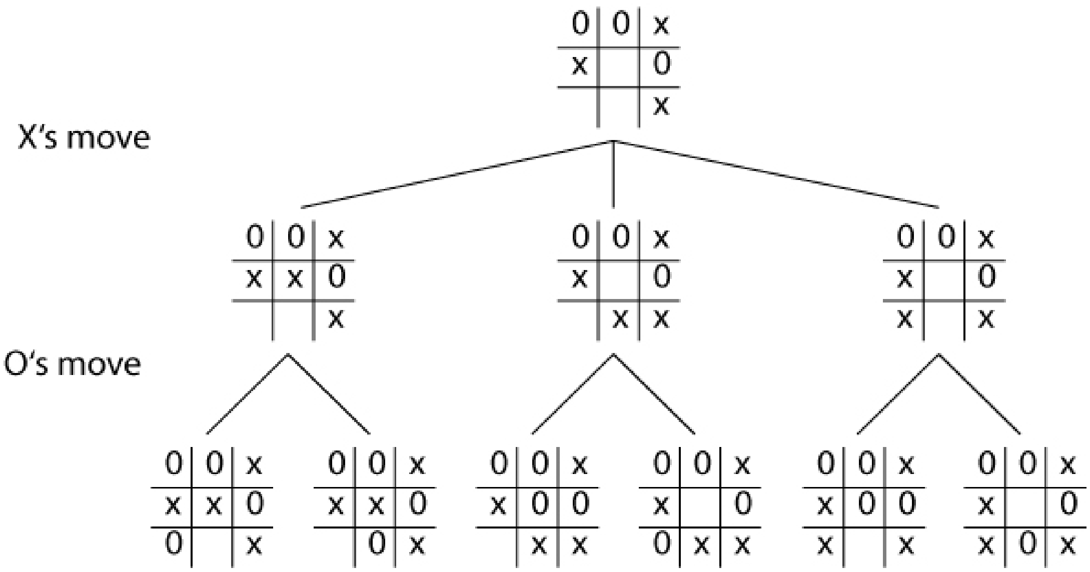
Physically, it is just a tree of all possible moves.

3.1 Moves

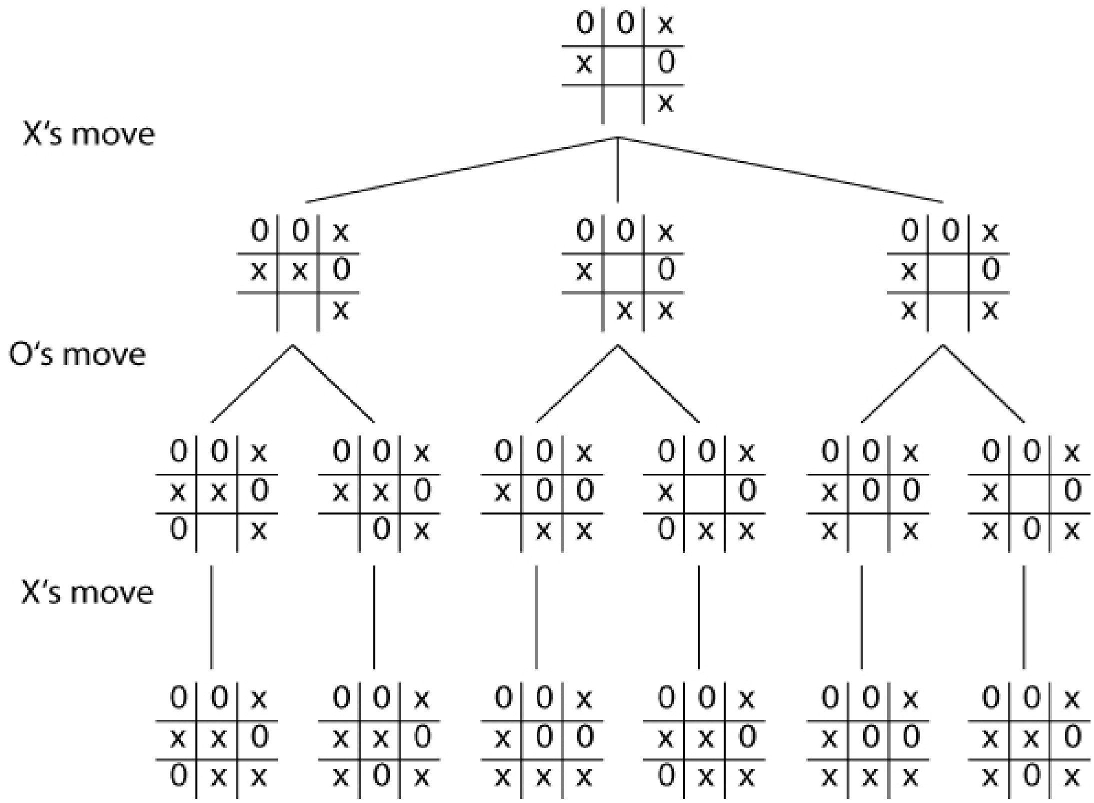
- **Start: X's Moves**



- **Next: O's Moves**



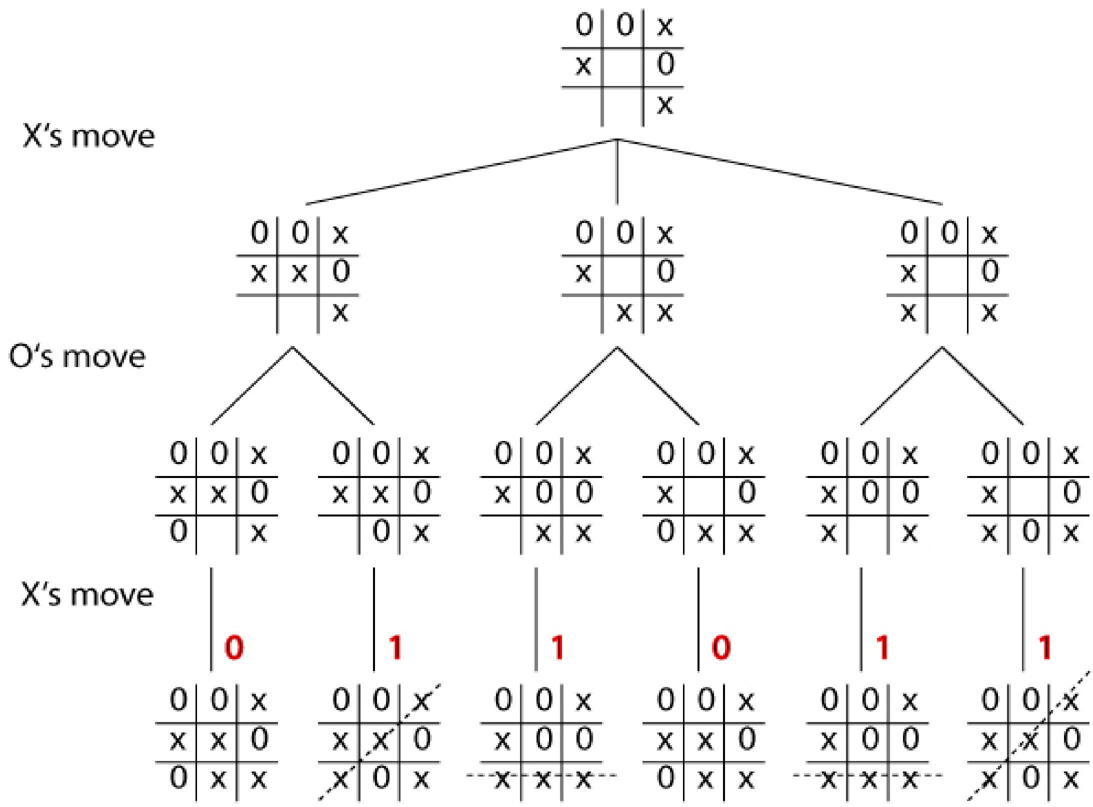
- **Again: X's moves**



3.2 Static Evaluation:

'+1' for a win, '0' for a draw

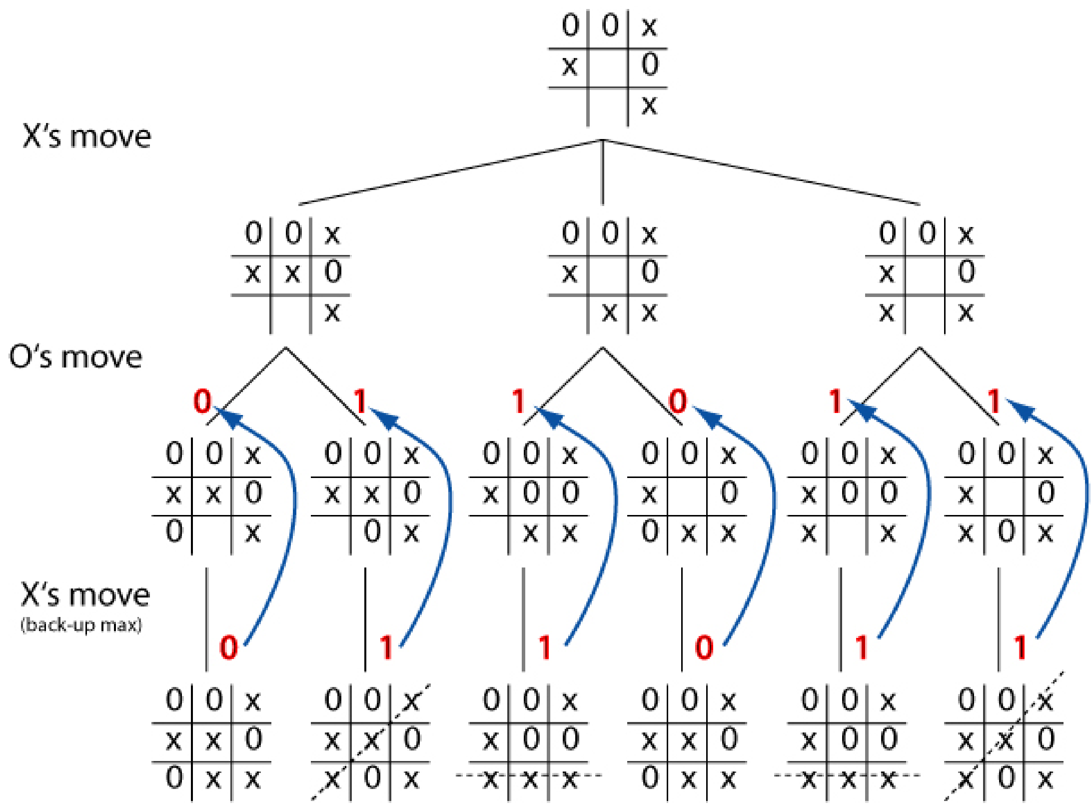
- **Criteria '+1' for a Win, '0' for a Draw**



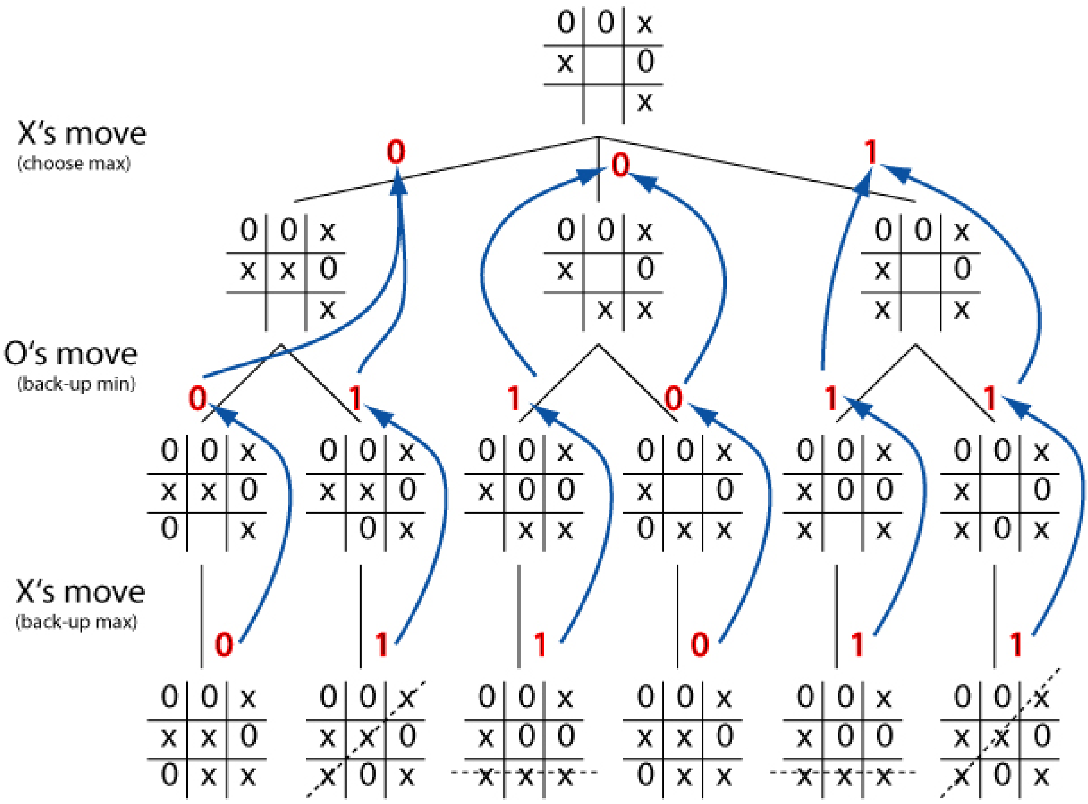
3.3 Back-up the Evaluations:

Level by level, on the basis of opponent's turn

- **Up : One Level**

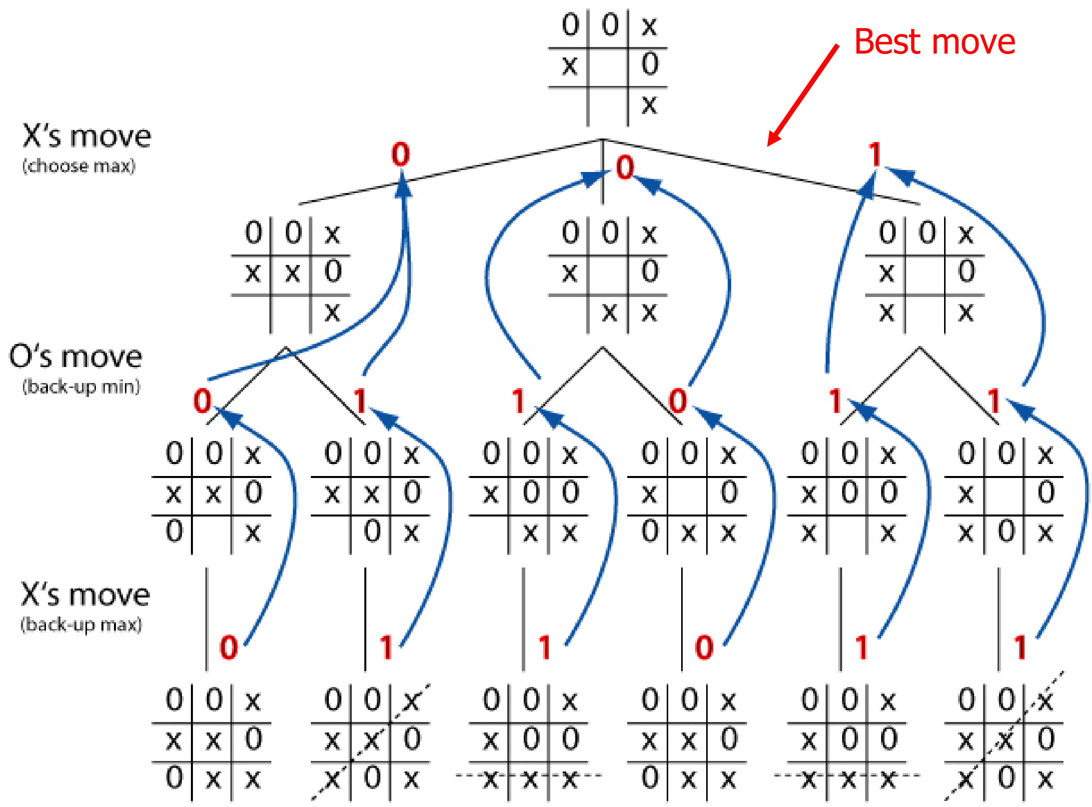


- **Up : Two Levels**



3.4 Evaluation obtained :

Choose best move which is maximum



4. Alpha-Beta Pruning

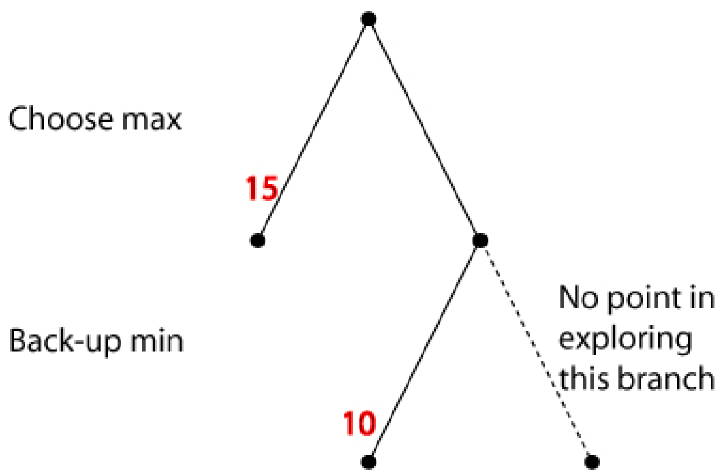
The problem with Mini-Max algorithm is that the number of game states it has to examine is exponential in the number of Moves.

The Alpha-Beta Pruning helps to arrive at correct Min-Max algorithm decision without looking at every node of the game tree.
While using Mini-Max, some situations may arise when search of a particular branch can be safely terminated. So, while doing search, figure out those nodes that do not require to be expanded. The method is explained below :

- Max-player cuts off search when he knows Min-player can force a provably bad outcome.
- Min player cuts of search when he knows Max-player can force provably good (for max) outcome
- Applying an **alpha-cutoff** means we stop search of a particular branch because we see that we already have a better opportunity elsewhere.
- Applying a **beta-cutoff** means we stop search of a particular branch because we see that the opponent already has a better opportunity elsewhere.
- Applying both forms is alpha-beta pruning.

5.1 Alpha-Cutoff

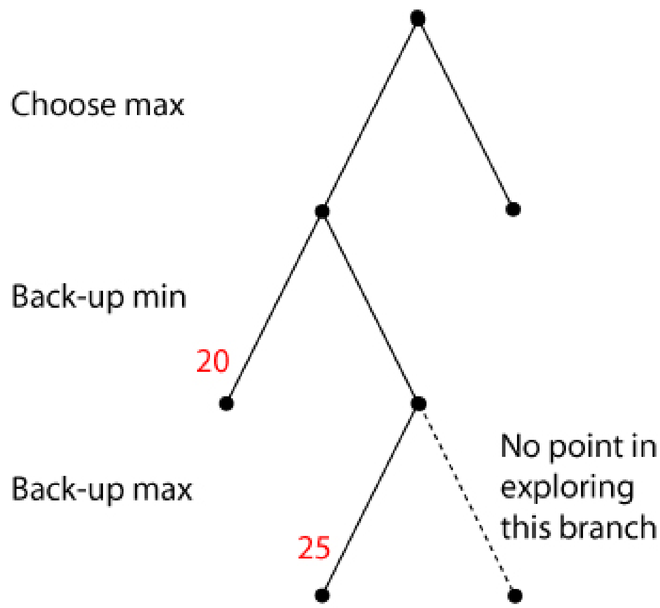
It may be found that, in the current branch, the opponent can achieve a state with a lower value for us than one achievable in another branch. So the current branch is one that we will certainly not move the game to. Search of this branch can be safely terminated.



5.2 Beta-Cutoff

It is just the reverse of Alpha-Cutoff.

It may also be found, that in the current branch, we would be able to achieve a state which has a higher value for us than one the opponent can hold us to in another branch. The current branch can be identified as one that the opponent will certainly not move the game to. Search in this branch can be safely terminated.



6. References : Textbooks

1. *"Artificial Intelligence", by Elaine Rich and Kevin Knight, (2006), McGraw Hill companies Inc., Chapter 12, page 305-326.*

2. *"Artificial Intelligence: A Modern Approach" by Stuart Russell and Peter Norvig, (2002), Prentice Hall, Chapter 6, page 161-189.*

3. *"Computational Intelligence: A Logical Approach", by David Poole, Alan Mackworth, and Randy Goebel, (1998), Oxford University Press, Chapter 4, page 113-163.*

4. *"AI: A New Synthesis", by Nils J. Nilsson, (1998), Morgan Kaufmann Inc., Chapter 12, Page 195-213.*

5. *Related documents from open source, mainly internet. An exhaustive list is being prepared for inclusion at a later date.*