EBIS2013 L4

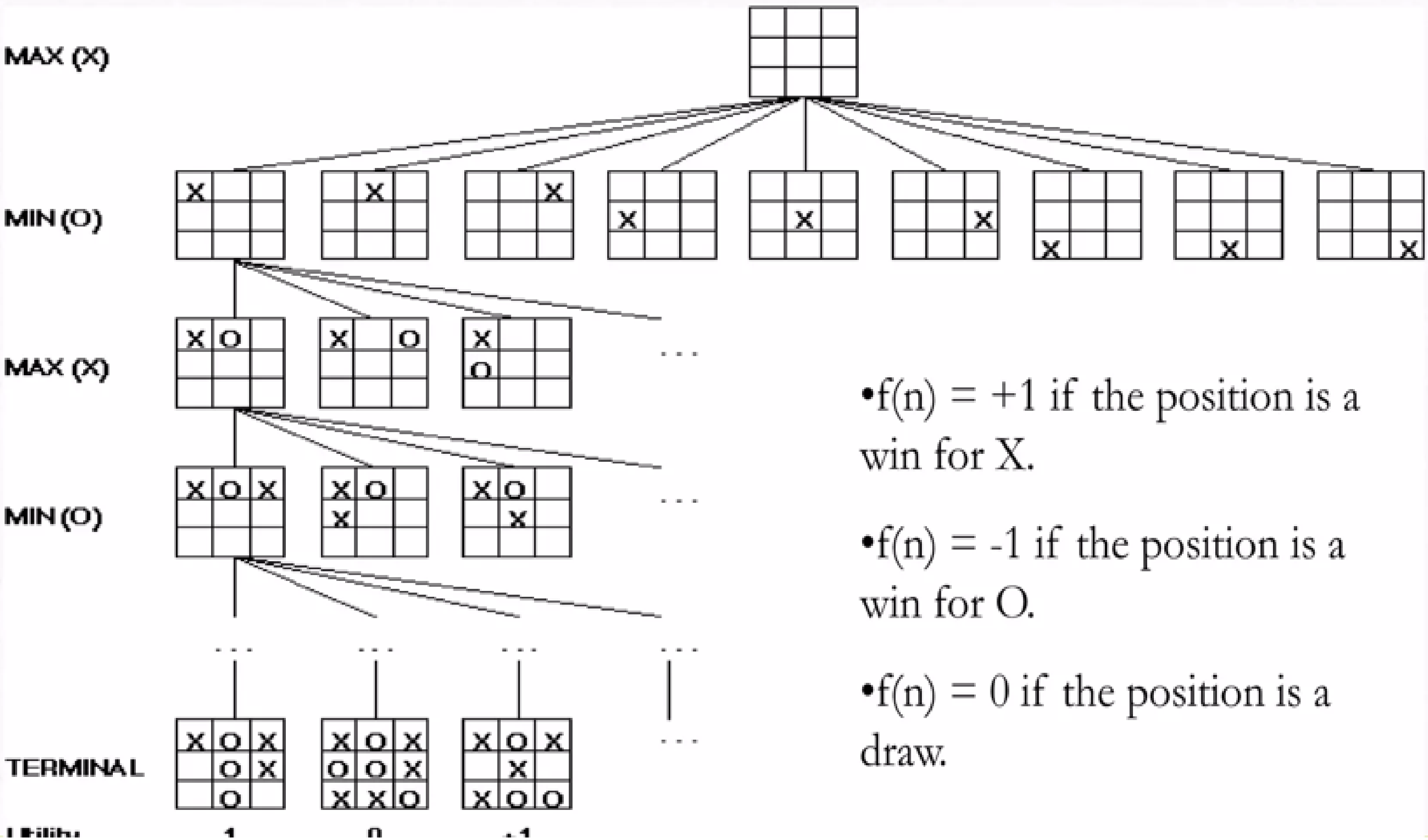
FUNDAMENTALS OF DIGITAL ECONOMY AND FINTECH

PROF. ANDY L. LIU

FBM & CENTER OF BLOCKCHAIN

EXTENSIVE FORM GAMES

An example (partial) game tree for Tic-Tac-Toe

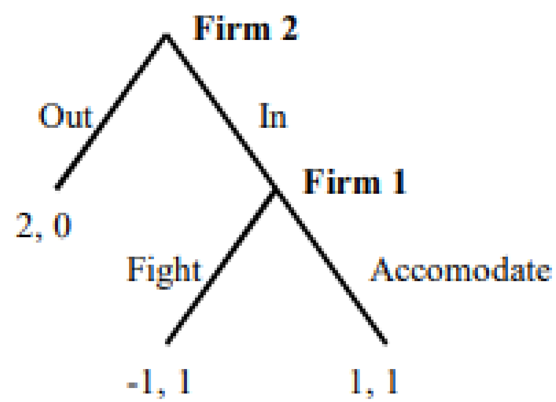


- $f(n) = +1$ if the position is a win for X.
- $f(n) = -1$ if the position is a win for O.
- $f(n) = 0$ if the position is a draw.

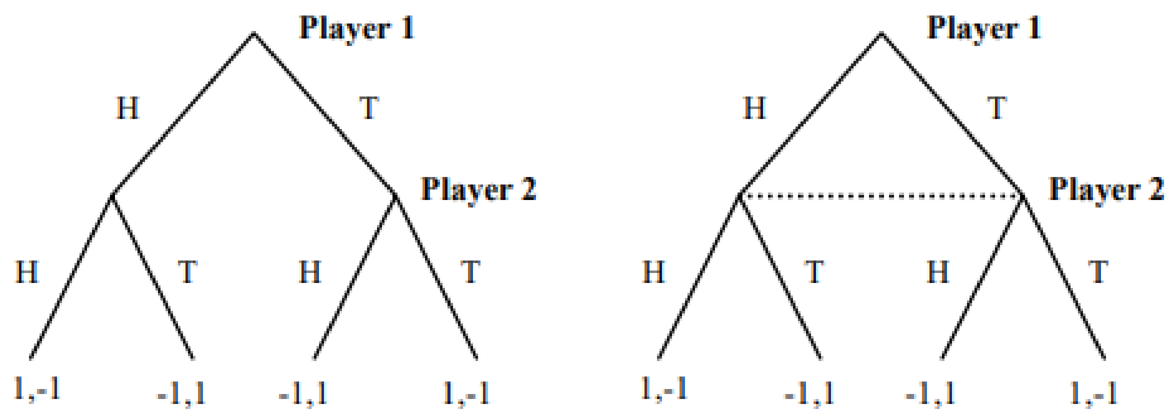
The extensive form of a game is a complete description of:

1. The set of players
2. Who moves when and what their choices are
3. What players know when they move
4. The players' payoffs as a function of the choices that are made.

An Entry Model Firm 1 is an incumbent monopolist. A second firm, Firm 2, has the opportunity to enter. After Firm 2 enters, Firm 1 will have to choose how to compete: either aggressively (Fight), or by ceding some market share (Accomodate). The strategic situation can be represented as follows.



Matching Pennies Or consider two variants of the matching pennies game. In the first variant, player one moves first, and then player two moves second *after having observed player one's action*. In the second, player two does not observe player one's action.



2.2 Formal Definitions

Formally, a finite extensive form game consists of:

- A finite set of players $i = 1, ..., I$.
- A finite set X of nodes that form a tree, with $Z \subset X$ being the terminal nodes.
- A set of functions that describe for each $x \notin Z$,
  - The player $i(x)$ who moves at x.
  - The set $A(x)$ of possible actions at x.
  - The successor node $n(x, a)$ resulting from action a.
- Payoff functions $u_i : Z \to \mathbb{R}$ assigning payoffs to players as a function of the terminal node reached.
- An information partition: for each x, let $h(x)$ denote the set of nodes that are possible given what player $i(x)$ knows. Thus, if $x' \in h(x)$, then $i(x') = i(x)$, $A(x') = A(x)$ and $h(x') = h(x)$.

We will sometimes use the notation $i(h)$ or $A(h)$ to denote the player who moves at information set h and his set of possible actions.

In an extensive form game, write H_i for the set of information sets at which player i moves.

$$H_i = \{S \subset X : S = h(x) \text{ for some } x \in X \text{ with } i(x) = i\}$$

Write A_i for the set of actions available to i at any of his information sets.

Definition 1 *A pure strategy for player i in an extensive form game is a function $s_i : H_i \to A_i$ such that $s_i(h) \in A(h)$ for each $h \in H_i$.*

A strategy is a complete contingent plan explaining what a player will do in every situation. Let S_i denote the set of pure strategies available to player i, and $S = S_1 \times ... \times S_I$ denote the set of pure strategy profiles. As before, we will let $s = (s_1, ..., s_I)$ denote a strategy profile, and s_{-i} the strategies of i's opponents.

Matching Pennies, cont. In the first version of matching pennies, $S_1 = \{H, T\}$ and $S_2 = \{HH, HT, TH, TT\}$. In the second version, $S_1 = S_2 = \{H, T\}$.

There are two ways to represent mixed strategies in extensive form games.

Definition 2 *A mixed strategy for player i in an extensive form game is a probability distribution over pure strategies, i.e. some $\sigma_i \in \Delta(S_i)$.*

Definition 3 *A behavioral strategy for player i in an extensive form game is a function $\sigma_i : H_i \to \Delta(A_i)$ such that $support(\sigma_i(h)) \subset A(h)$ for all $h \in H_i$.*

THE NORMAL FORM AND NASH EQUILIBRIUM

Any extensive form game can also be represented in the normal form. If we adopt a normal form representation, we can solve for the Nash equilibrium.

Matching Pennies, cont. For our two versions of Matching Pennies, the normal forms are:

	HH	HT	TH	TT
H	$1, -1$	$1, -1$	$-1, 1$	$-1, 1$
T	$-1, 1$	$1, -1$	$-1, 1$	$1, -1$

	H	T
H	$1, -1$	$-1, 1$
T	$-1, 1$	$1, -1$

In the first version, Player two has a winning strategy in the sense that she can always create a mismatch if she adopts the strategy TH. Any strategy for player one, coupled with this strategy for player two is a Nash equilibrium. In the second version, the Nash equilibrium is for both players to mix $\frac{1}{2}H + \frac{1}{2}T$.

Entry Game, cont. For the entry game above, the normal form is:

	Out	In
F	$2, 0$	$-1, 1$
A	$2, 0$	$1, 1$

There are several Nash equilibria: (A, In), (F, Out) and $(\alpha F + (1 - \alpha)A, Out)$ for any $\alpha \geq 1/2$.

SUBGAME PERFECT EQUILIBRIUM

Definition 4 *Let G be an extensive form game, a subgame G' of G consists of (i) a subset Y of the nodes X consisting of a single non-terminal node x and all of its successors, which has the property that if $y \in Y$, $y' \in h(y)$ then $y' \in Y$, and (ii) information sets, feasible moves, and payofs at terminal nodes as in G.*

Definition 5 *A strategy profile s is a* ***subgame perfect equilibrium*** *of G if it induces a Nash equilibrium in every subgame of G.*

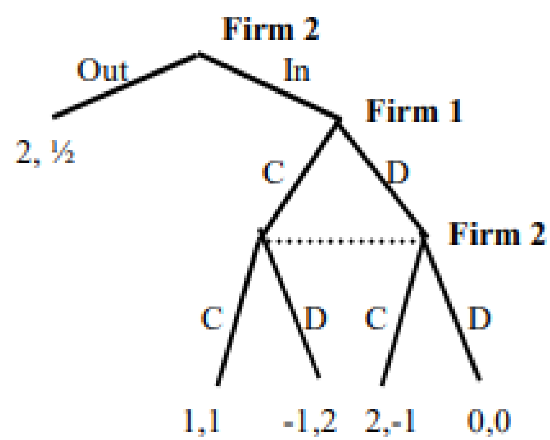
Entry Game, cont. In the entry game, only (A, In) is subgame perfect.

BACKWARD INDUCTION

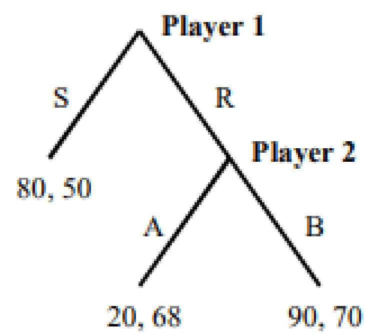
subgame perfect equilibria is to start at the end of the game and work back to the front. This process is called *backward induction*.

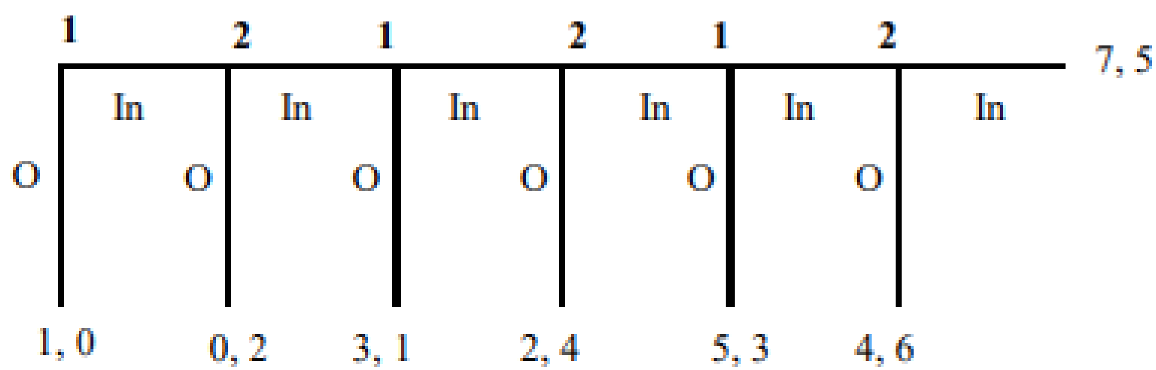
Definition 6 *An extensive form game is said to have* **perfect information** *if each information set contains a single node.*

Proposition 7 *(Zermelo's Theorem) Any finite game of perfect information has a pure strategy subgame perfect equilibrium. For generic payoffs, there is a unique SPE.*

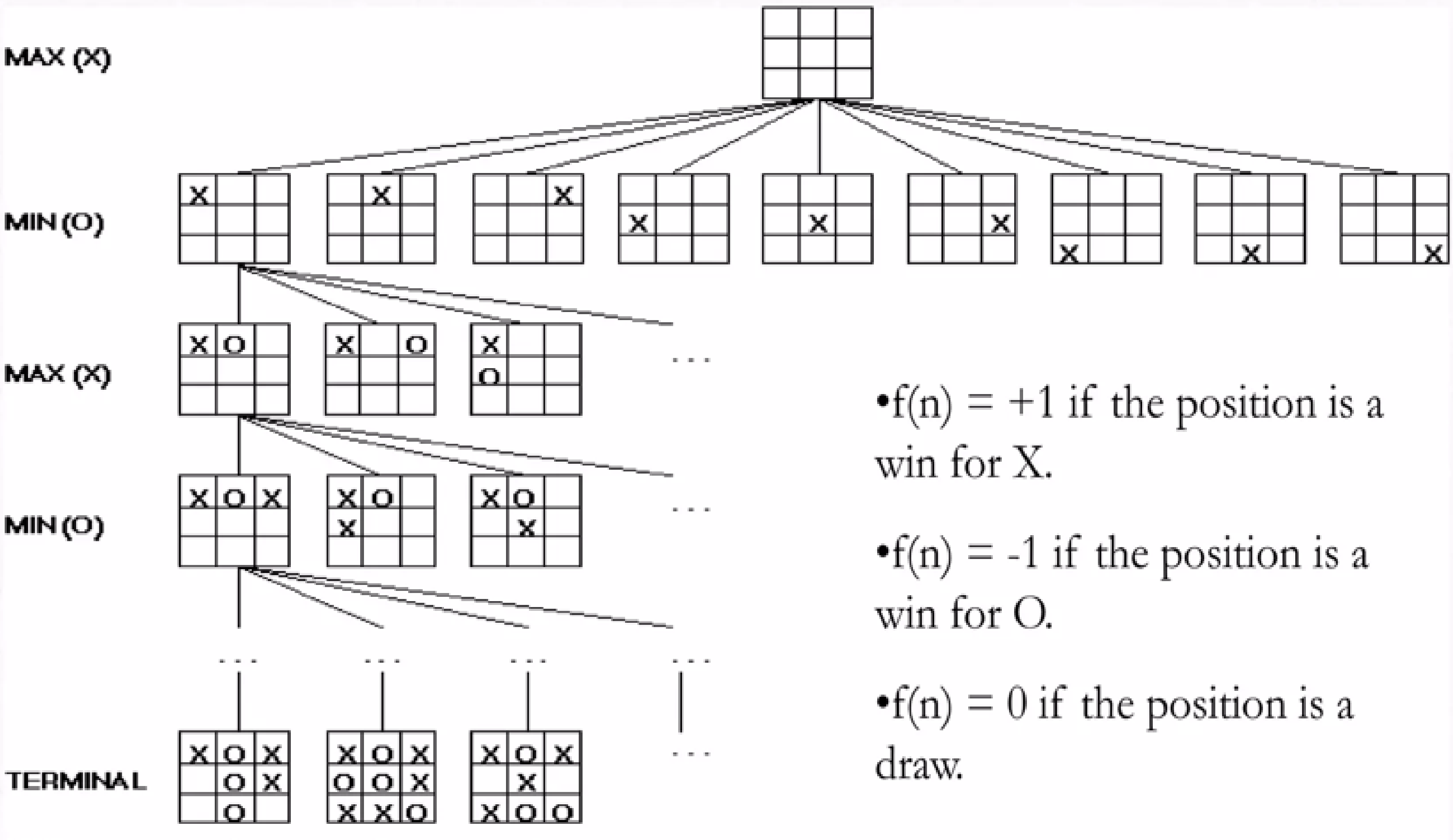


CRITICISMS OF SUBGAME PERFECTION





An example (partial) game tree for Tic-Tac-Toe



- f(n) = +1 if the position is a win for X.
- f(n) = -1 if the position is a win for O.
- f(n) = 0 if the position is a draw.

REPEATED GAME

Definition 3.5 (Repeated Game). *In a repeated game G^T, the same simultaneous-move game $G = (N, A, \tilde{u})$ (the stage game) is played by the same players for T periods, with every agent having perfect information about the history of actions in all previous periods. In an infinitely repeated game G^∞ the stage game G is repeated forever.*

		Player 2 C	Player 2 D
Player 1	C	3, 3	0, 5
	D	5, 0	1, 1

Figure 3.5: The Prisoner's Dilemma.

to current payoff. Based on this, the utility of player i for an infinite sequence of play $h = (a^0, a^1, \ldots)$ in an infinitely repeated game is,

$$u_i(h) = \sum_{k=0}^{\infty} \delta^k \tilde{u}_i(a^k), \tag{3.3}$$

where history h is an infinite sequence of action profiles. To calculate the total discounted payoff, we use the formula for the geometric series,

$$\sum_{k=0}^{\infty} \delta^k v = \frac{v}{1-\delta}, \tag{3.4}$$

which holds for any value $v \in \Re$, and any discount factor $0 < \delta < 1$. From this, we see that the total payoffs are finite.

To get some intuition for this expression, let's consider two different discount factors $\delta \in \{0.5, 0.9\}$ and two different per-period payoffs, either 1 or 4; see Table 3.1. Notice,

		per-period payoff	
		1	4
discount factor δ	0.5	2	8
	0.9	10	40

The discounted sum payoff for different per-period payoffs and discount factors

Theorem 3.4 (Uniqueness). *If the stage game has a unique Nash equilibrium, then a finitely repeated game has a unique subgame-perfect equilibrium, which is to play the stage game Nash equilibrium in every period.*

Example 3.4. *Since* (D, D) *is the unique Nash equilibrium in the* Prisoner's Dilemma *stage game, then by Theorem 3.4, the only subgame-perfect equilibrium in a finitely repeated* Prisoner's Dilemma *is defect-defect in every period.*

AUTOMATA
AND THEORY OF COMPUTATION

Theory of Computation

- The ***Theory of Computation*** is the branch of computer science that deals with how efficiently problems can be solved on a model of computation, using an algorithm.
- The field is divided into three major branches:
  - Automata theory and language
  - Computability theory
  - Complexity theory

2.3 Formal Definition of Finite State Automata

Definition 3 : A finite state automaton(FA) $M = (\Sigma, Q, \delta, q_0, F)$, where

Σ is a finite set alphabet,

Q is a finite set of states,

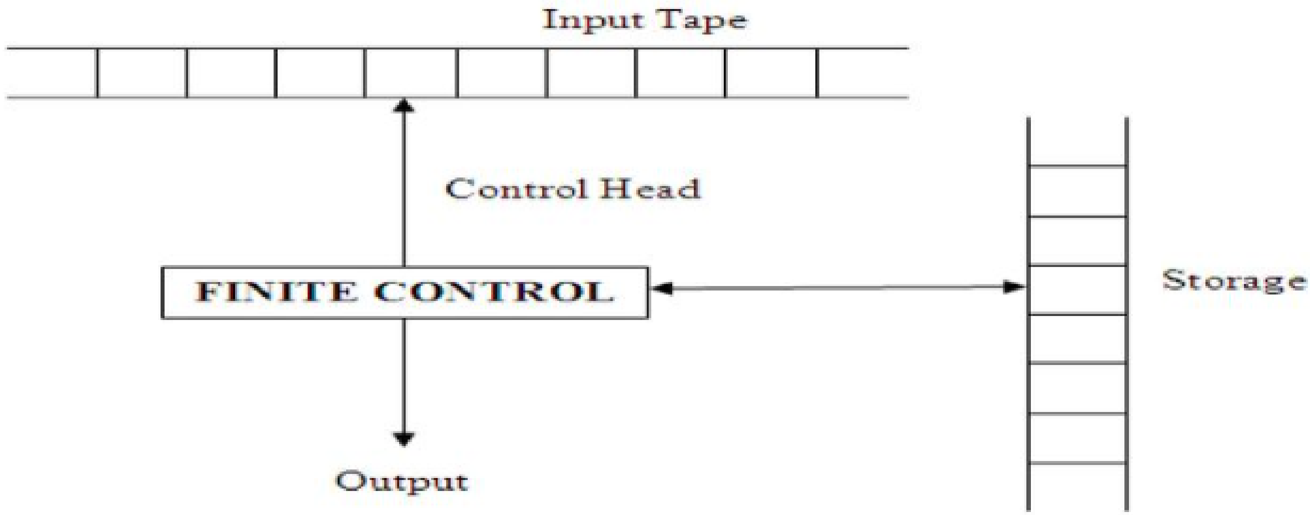
$\delta : Q \times \Sigma \to Q$ is a transition function,

$q_0 \in Q$ is the initial state,

$F \subseteq Q$ is the set of final states.

Note : A finite state automaton (FA) M defined above is a deterministic finite state automaton (DFA), since each move of the machine is uniquely determined.

Finite Automaton Model



Input tape

It is a linear tape which holds the input string. The tape is divided into a finite number of cells. Each cell holds a symbol from Σ.

Finite Control

It Indicates current state and decides next state on receiving a particular input from input tape. It consists of 3 things:

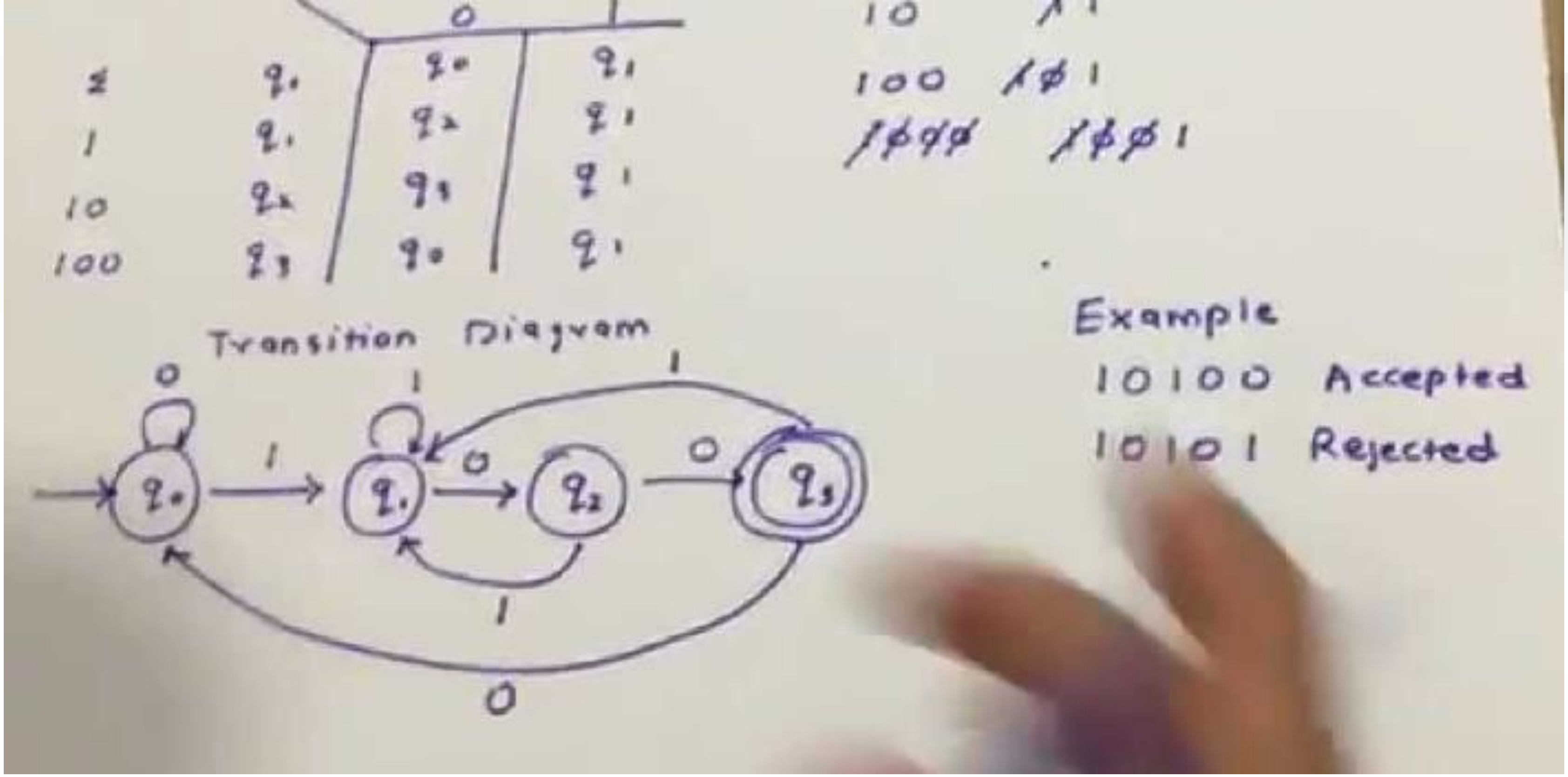
- finite number of states that the machine is allowed to be in (zero or more states are designated as accept or final states),
- a current state, initially set to a start state,
- a state transition functions for changing the current state.

Storage

Automaton have temporary storage which has unlimited number of cells

Control Head (or) Tape Head

It reads symbol from input tape and moves right side with or without changing the state.


Transition Diagram
Example
10100 Accepted
10101 Rejected

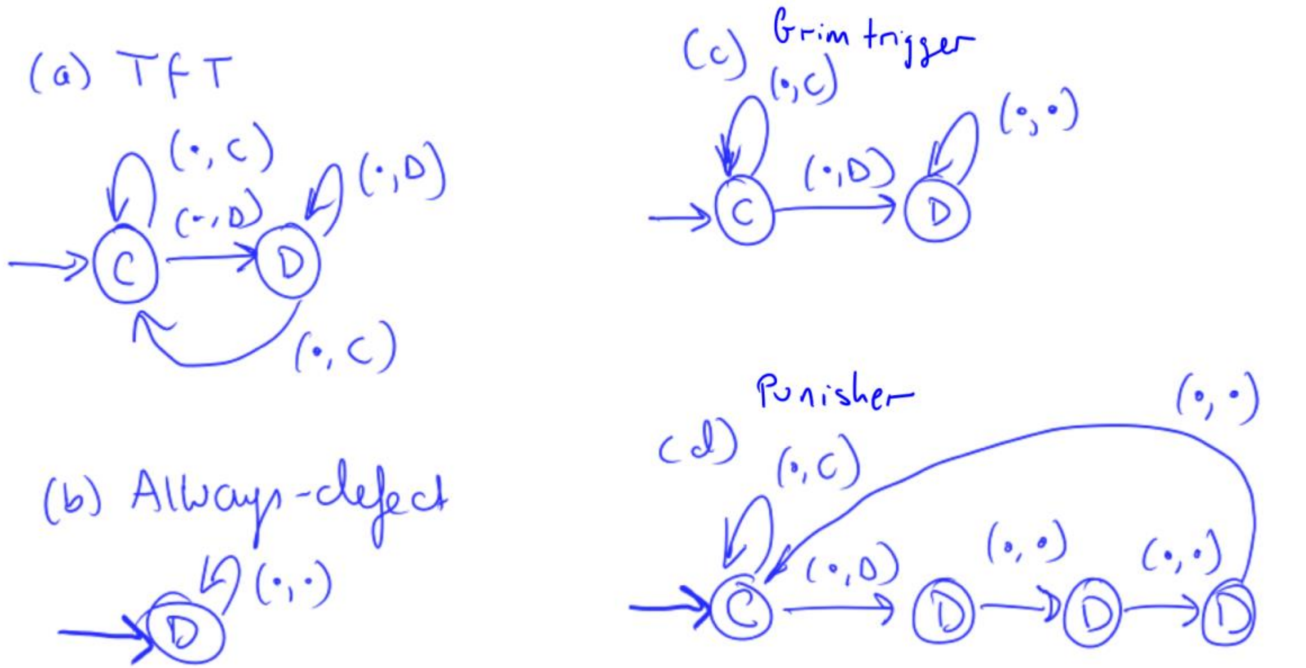
AUTOMATON STRATEGIES

Given the possibility of an infinite number of periods, we cannot adopt an explicit representation of an agent's strategy for all possible histories. Instead, we adopt *finite state automata* to provide a succinct representation of an agent's strategy. Automata can be used to represent strategies in both finite and infinitely repeated games, and a strategy represented in this way is an *automaton strategy.*

Definition 3.7 (Automaton Strategy). *An automaton strategy m_i for player i in a repeated game is defined by $(Q_i, q_i^0, succ_i, h_i)$ where:*

- *Q_i is the set of machine states*
- *$q_i^0 \in Q_i$ is the start state*
- *next state $q_i' = succ_i(q_i, a)$, defined for all states q_i and action profiles $a = (a_1, \ldots, a_n) \in A$*
- *action $h_i(q_i) \in A_i$, defined for all machine states q_i*

four examples of automaton strategies for the infinitely repeated *Prisoner's Dilemma.* The example strategies are (a) *Tit-for-Tat* (TfT), (b) *always-defect*, (c) *grim trigger* and (d) *punisher.* All four automaton strategies are illustrated from the perspective of agent 1.



Four example automaton strategies for the repeated *Prisoner's Dilemma*, expressed from the perspective of agent 1: (a) *Tit-for-Tat*, (b) *always-defect*, (c) *grim trigger* and (d) *punisher*. The arrow coming from the left points to the start state. Each state is annotated with the action taken in that state. Transitions between states depend on the actions selected by players in the current period. The notation "$(\cdot, a_2)$" on an edge indicates this transition occurs whenever agent 2 plays action a_2, and "$(\cdot, \cdot)$" indicates that this transition always occurs.

FOLK THEOREM

a folk theorem, say that an action profile a of the stage game is *enforceable* if there exists a (possibly mixed) Nash equilibrium strategy a^* of the stage game, for which,

$$\tilde{u}_i(a) > \tilde{u}_i(a^*) = e_i, \tag{3.8}$$

for all agents i, where e_i is the expected utility to agent i given strategy a^*. We do not require that action profile a is a Nash equilibrium of the stage game. The strict inequality is important in obtaining the folk theorem.

Example 3.7. *In the* Prisoner's Dilemma, *the unique Nash equilibrium of the stage game is* $a^* = (D, D)$, *with payoff* $(1, 1)$ *to each player. Given that the payoffs for the other action profiles are* $\{(3, 3), (0, 5), (5, 0)\}$, *the only enforceable action profile is* $a = (C, C)$.

Theorem 3.6 (Folk Theorem). *Given a stage game* G *with an enforceable action profile* a, *there exists a subgame-perfect equilibrium of the infinitely repeated game* G^∞, *for some discount factor* $\delta < 1$, *where action profile* a *is played in equilibrium in every period.*

KNOWLEDGE AND EPISTEMIC GAME THEORY

Imagine three girls sitting in a circle, each wearing either a red hat or a white hat. Suppose that all the hats are red. When the teacher asks if any student can identify the color of her own hat, the answer is always negative, since nobody can see her own hat. But if the teacher happens to remark that there is at least one red hat in the room, a fact which is well-known to every child (who can see two red hats in the room) then the answers change. The first student who is asked cannot tell, nor can the second. But the third will be able to answer with confidence that she is indeed wearing a red hat.

Information. We describe an agent's knowledge in each state using an *information function.*

Definition 1 *An* ***information function*** *for Ω is a function h that associates with each state $\omega \in \Omega$ a nonempty subset $h(\omega)$ of Ω.*

Knowledge. We refer to a set of states $E \subset \Omega$ as an *event*. If $h(\omega) \subset E$, then in state ω, the agent views $\neg E$ as impossible. Hence we say that the agent *knows* E. We define the agent's *knowledge function* K by:

$$K(E) = \{\omega \in \Omega : h(\omega) \subset E\}.$$

Thus, $K(E)$ is the set of states at which the agent knows E.

		STATES OF THE WORLD							
		a	*b*	*c*	*d*	*e*	*f*	*g*	*h*
	1	*R*	*R*	*R*	*R*	*W*	*W*	*W*	*W*
PLAYER	2	*R*	*R*	*W*	*W*	*R*	*R*	*W*	*W*
	3	*R*	*W*	*R*	*W*	*R*	*W*	*R*	*W*

In the notation we have introduced, the set of all possible states of nature Ω can be summarized as $\{a, b, c, d, e, f, g, h\}$, with a letter designating each state. Then, the partitions of the three agents are given by: $P_1 = \{\{a, e\}, \{b, f\}, \{c, g\}, \{d, h\}\}$, $P_2 = \{\{a, c\}, \{b, d\}, \{e, g\}, \{f, h\}\}$, $P_3 = \{\{a, b\}, \{c, d\}, \{e, f\}, \{g, h\}\}$.

Definition. *State* ω *is* reachable *from state* ω_0 *if there exists a sequence of agents such that* $\omega \in \prod_{i \in N} P^i(\omega_0)$.

Example. White and Red Hats without teaching hitting. Starting from any $\omega \in \Omega$ could reach the whole Ω.

From RRR to WWW:

$$(WRR) \in K_1^1(\Omega \mid RRR, K_0)$$

$$(WWR) \in K_1^2(K_2^1(\Omega \mid K_1^1(\Omega \mid RRR, K_0))$$

$$(WWW) \in K_1^3(K_2^2(K_3^1(\Omega \mid (K_2^1(\Omega \mid K_1^1(\Omega \mid RRR, K_0)))))$$

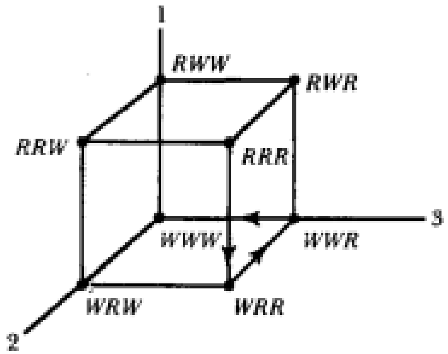


Figure 1a

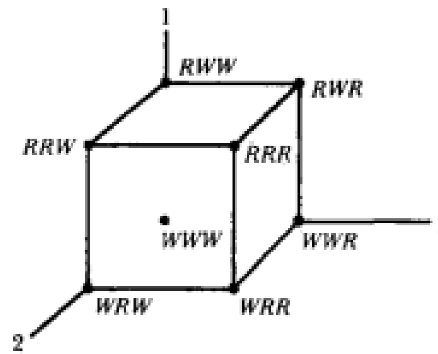


Figure 1b

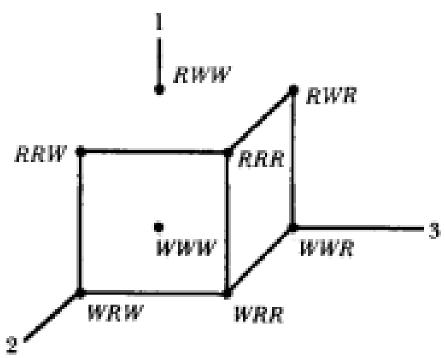


Figure 1c

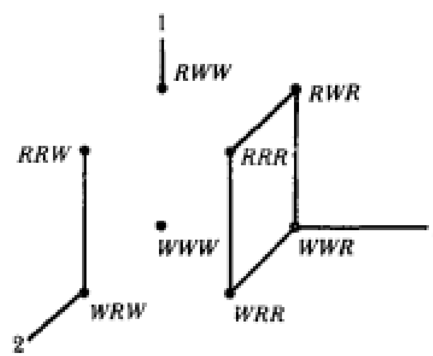
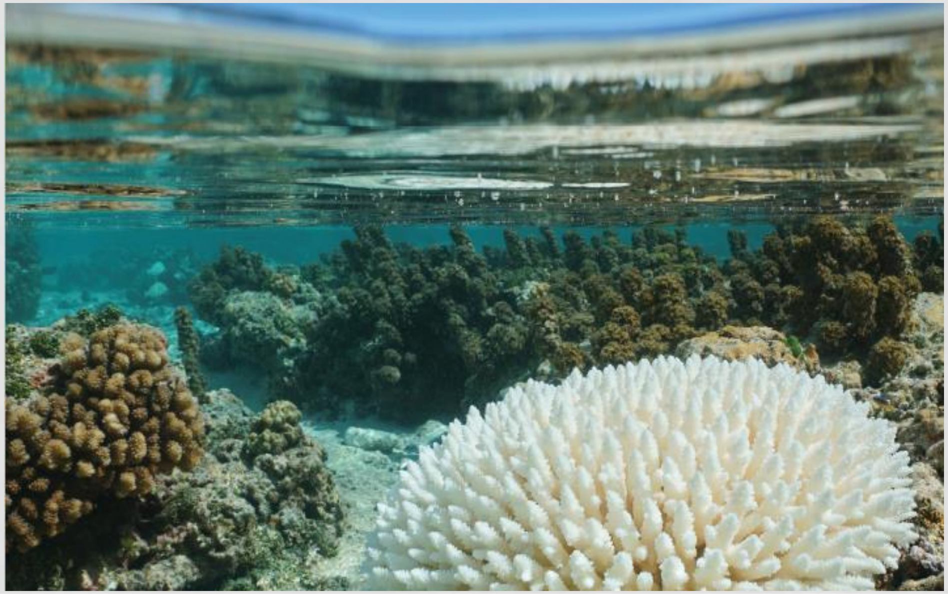


Figure 1d



HOMEWORK 4

OCTOBER 15

4.0 (OPTIONAL)

3.2 Tit-for-tat in the infinitely repeated *Prisoner's Dilemma*

(a) Prove that the *TfT* strategy does not form a subgame-perfect equilibrium of the infinitely repeated *Prisoner's Dilemma*, for any discount factor $\delta < 1$.

(b) Prove that *TfT* does form a Nash equilibrium for some discount factor $\delta < 1$.

4.1

- According to the definition of automaton strategy, work out the pseudocode and python code (optional) of two of the four examples for the infinitely repeated prisoner's dilemma.

Definition 3.7 (Automaton Strategy). *An automaton strategy m_i for player i in a repeated game is defined by $(Q_i, q_i^0, succ_i, h_i)$ where:*

- *Q_i is the set of machine states*
- *$q_i^0 \in Q_i$ is the start state*
- *next state $q_i' = succ_i(q_i, a)$, defined for all states q_i and action profiles $a = (a_1, \ldots, a_n) \in A$*
- *action $h_i(q_i) \in A_i$, defined for all machine states q_i*

four examples of automaton strategies for the infinitely repeated *Prisoner's Dilemma*. The example strategies are (a) *Tit-for-Tat* (TfT), (b) *always-defect*, (c) *grim trigger* and (d) *punisher*. All four automaton strategies are illustrated from the perspective of agent 1.

4.2

2. To study the puzzle of red or white hats in Geanakoplos' *Common Knowledge*.
3. To study Hart and Tauman Paper, and program their model.
4. To practice the epistemic analysis of Stratagem of empty-city with paradigm of common knowledge.

END OF LECTURE

LECTURE 4