Mean-Payoff games:

7/9/2017

Lecture Notes: I

AlgoLabs
http://www.algolabs.org.in

Organization:

Part 1: Definition of the game, objectives of the Players

Part 2: Illustrative example

Part 3: Overview of results

Part 4: First-cycle version of the game

Part 5: Equivalence of values between finite and infinite versions.

Part 1: Mean-Payoff game definition

Arena: $G = (V_0, V_1, E)$

$w : E \longrightarrow \{-W, \ldots +W\}$ (weight function)

integers

Two Players Maximizer and Minimizer

Initial vertex $a_0 \in V$

Play: Sequence $a_0\ a_1\ a_2 \ldots$

of vertices, edges

$\pi : a_0 \xrightarrow{w_0} a_1 \xrightarrow{w_1} a_2 \xrightarrow{w_2} \ldots$

Play continues ~~for~~ indefinitely

- Suppose Play stops at 'n' steps:

$$a_0 \xrightarrow{w_0} a_1 \xrightarrow{w_1} a_2 \longrightarrow \cdots\cdots a_{n-1} \xrightarrow{w_{n-1}} a_n$$

(Average) Mean-~~Pay~~ weight $= \dfrac{\sum_{i=0}^{n-1} w_i}{n}$

Minimizer pays $\dfrac{\sum_{i=0}^{n-1} w_i}{n}$ to Maximizer

- In the infinite ~~ver~~ game, we need a definition of the average of the play (play).

Mean-Payoff $(\pi) \stackrel{?}{:=} \lim_{n\to\infty} \dfrac{\sum_{i=0}^{n-1} w_i}{n}$ [Can we define it this way?]

↓

play

This seems like a natural definition ~~at~~ for the infinity case, but unfortunately the above limit might not exist always.

Example: (credits to Soumyajit).

w_i sequence:	2	−2	2	2	−2	−2	2	2	2	2	2	····
		↓		↓		↓				↓		
$\frac{\sum w_i}{n}$		0		1		0				1		···

$\lim_{n\to\infty} \frac{\sum w_i}{n}$ oscillates between 0 and 1.

- So we consider $\lim_{n\to\infty} \inf \frac{\sum w_i}{n}$

- How is this defined?

$$\lim_{n\to\infty} \inf x_n = \lim_{n\to\infty} \left(\inf_{m \geq n} \{x_m\} \right)$$

In the above example $\lim_{n\to\infty} \inf \left(\frac{\sum w_i}{n} \right) = 0$

Why should lim inf always exist in our case?

- Note that each $\frac{\sum w_i}{n} \geq -W$ and $\leq W$
- Sequence ~~$\{\sum_{i=0}^{n}$~~ $\inf_{m}$ $\{x_m, x_{m+1} \cdots\}\}$ is increasing.
- Sequence is ~~mono~~ increasing and bounded. So its limit should exist.

Objective in Mean Payoff:

Maximizer wants to maximize $\liminf_{n\to\infty} \frac{\sum w_i}{n}$ of a play.

Minimizer want to minimize it.

We can think of it as: Min paying the "Mean Payoff" to Max.

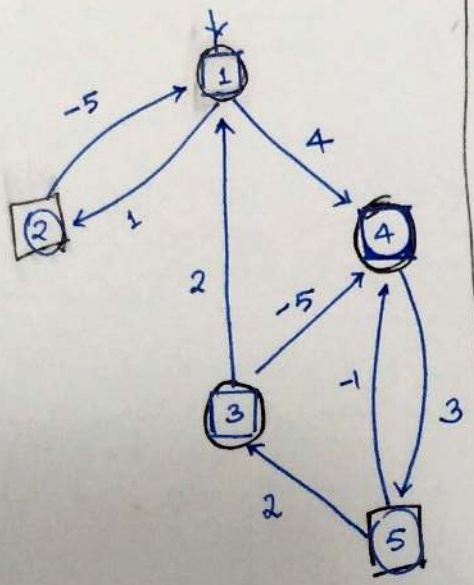
Strategies: σ for Max, τ for Min } convention.

$\sigma : V^* V_1 \mapsto V$
$\tau : V^* V_0 \mapsto V$
} ~~not necc~~ positional if $\sigma : V_1 \to V$, $\tau : V_0 \mapsto V$

Objective for Min is to play a strategy that minimizes payoff

Objective for Max is to play a strategy that maximizes payoff.

Part 2: Example (from Raskin's tutorial, with notations changed).



Play: $1 \xrightarrow{1} 2 \xrightarrow{-5} 1 \xrightarrow{1} 2 \xrightarrow{-5} \cdots$

$\frac{\sum w_i}{n}$: ~~$\frac{1}{1}$ $\frac{-2}{2}$ $\frac{-3}{3}$ $\frac{-4}{4}$ $\frac{-3}{5}$~~

$\frac{1}{1}$ $\frac{-4}{2}$ $\frac{-3}{3}$ $\frac{-8}{4}$ $\frac{-7}{5}$ $\frac{-12}{6}$ .....

1 -2 -1 -2 $-7/5$ -2 ....

$\frac{-4n+1}{2n+1} = \frac{1-4n}{2n+1}$ $n : 0$ to ∞

$= \frac{(-4n-2)+3}{2n+1}$

$= -2 + \frac{3}{2n+1}$

$\liminf_{n\to\infty} = -2.$

1: Maximizer

0: Minimizer.

* Consistent with Raskin's notes + Jurdzinski's paper on UP ∩ co-UP

H1, SIPCOT IT Park, Siruseri, Kelambakkam 603 103, INDIA
Phone: +91 44 6748 0900 **Email:** office@algolabs.org.in

Another play:

$\pi_2 : 1\,(45)^\omega$

$1 \xrightarrow{4} 4 \xrightarrow{3} 5 \xrightarrow{-1} 4 \xrightarrow{3} 5 \xrightarrow{-1} 4 \longrightarrow \cdots$

$\frac{4}{1} \quad \frac{4+3}{2} \quad \frac{4+2}{3} \quad \frac{4+5}{4} \quad \frac{4+4}{5} \quad \frac{4+7}{6} \quad \cdots$

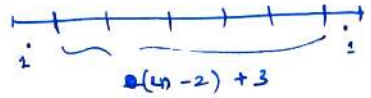
$\frac{4+2n}{2n+1}$

$n=2$

$2n-2=2$

$2 \cdot 2 = 4$

$\frac{4+2n-1}{2n}$ $\qquad$ $\frac{4+2(2n-2)+3}{2n} = \frac{4+4n+1}{2n}$



$\frac{4}{2n+1} + \frac{2n}{2n+1}$ $\qquad$ $\frac{4}{2n} + \frac{(2n+1)}{2n}$

First look at: $3 \quad -1 \quad 3 \quad -1 \quad 3 \quad -1 \quad \cdots$

$\frac{3}{1} \quad \frac{2}{2} \quad \frac{5}{3} \quad \frac{4}{4} \quad \frac{7}{5} \quad \frac{6}{6} \quad \frac{9}{7} \quad \cdots$

$\lim_{n\to\infty} \frac{(4+2n)}{2n+1} = 1$

$\lim_{n\to\infty} \left(\frac{4+2n+1}{2n}\right) = 1$

$\lim_{n\to\infty} \text{int} \; \frac{\quad}{\quad} = 1$

What happens by adding 4 in front?

$4 \quad 3 \quad -1 \quad 3 \quad -1 \quad 3 \quad -1 \quad \cdots$

$\frac{4+2}{3} \quad \frac{4+5}{4} \quad \frac{4+4}{5} \quad \frac{4+7}{6} \quad \cdots$

$\frac{4+2n}{2n+1} \to > 1$

at these points the numbers are > 1

So the payoff of $1(45)^\omega = 1$

→ so would be the payoff of $(45)^\omega$.

→ ~~exten~~ intuitively, the finite prefix does not play a role in the limit average payoff.

Exercise: Suppose π is a ~~path~~ play. Show that: Removing a finite prefix does not change the payoff. Similary show that adding a finite prefix does not change the payoff. [in the sequences, the differences will be due to a constant factor]. → Use fact that: 1) if $\{a_n\}_{n\geq 0}$, $\{b_n\}_{n\geq 0}$ are 2 bdd. real sequences and $\{b_n\}_{n\geq 0}$ converges to b

Then: $\liminf (a_n b_n) = b \liminf (a_n)$.

2) $\liminf (a_n + b_n) = \liminf (a_n) + \liminf (b_n)$ if (a_n), (b_n) are bounded real sequences.

Another play:

$\pi_3 : (1\,4\,5\,3)^\omega$

$= \frac{11}{4}$ → mean payoff.

How should Maximizer play? From 1 if she always plays 2, the payoff that she can get is -2. So ~~i~~ If she plays 4 always, the payoffs can be 1 or $11/4$ provided Min plays positionally.

→ ~~It turns~~

How should Minimizer play? At 5, she should choose to go to 4.

Currently, we have only described positional strategies and we have seen what payoffs you can get from such positional strategies. Will history-dependent strategies help? Can any of them play better ? with such strategies

Part 3: Overview of results.

Theorem: Given (G, V, a_0) a mean-payoff game. There exists a number $\tilde{v}(a_0)$ and positional strategies $\tilde{\sigma}, \tilde{\tau}$ s.t.

Payoff $(\tilde{\sigma}, \tau) \geq \tilde{v}(a_0)$ for every strategy τ of Min.

Payoff $(\sigma, \tilde{\tau}) \leq \tilde{v}(a_0)$ for every strategy σ of Max.

The value $\tilde{v}(a_0)$ is called the value of the game.

→ also called Minimax equilibrium.

Illustration on example:

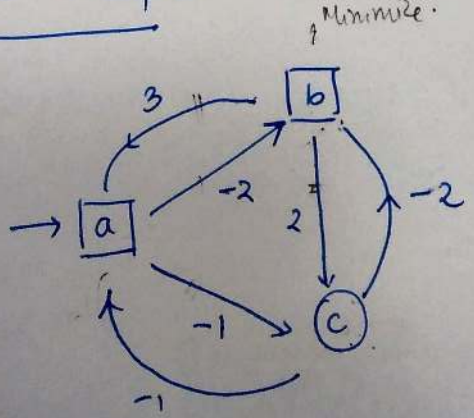
Max: $1 \longrightarrow 4$ $\quad 3 \longrightarrow 1$ she can ensure ≥ 1 no matter however Min plays.

Min: $5 \longrightarrow -1$ she can ensure ≤ 1 no matter how Max plays.

Value 1 is the value of the game.

More examples:

What is the value of this game?



$(ac)^\omega = 0$

$a(cb)^\omega = 0$

$(acc)^\omega = -1/2$

$(acb)^\omega = 0$

$(abc)^\omega = (-1/3)$

$(ab)^\omega = 1/2$

$a(bc)^\omega = 0$

0

- definition of game
- computing values manually
  (using result abt positional strategies)
- lim inf of sequences

Part 4: First cycle version of the game

In the previous two examples, the value that we computed turns out to be the mean weight of some cycle. Is this a coincidence or does it always happen? (if the theorem about positional strategies is true, this should always happen).

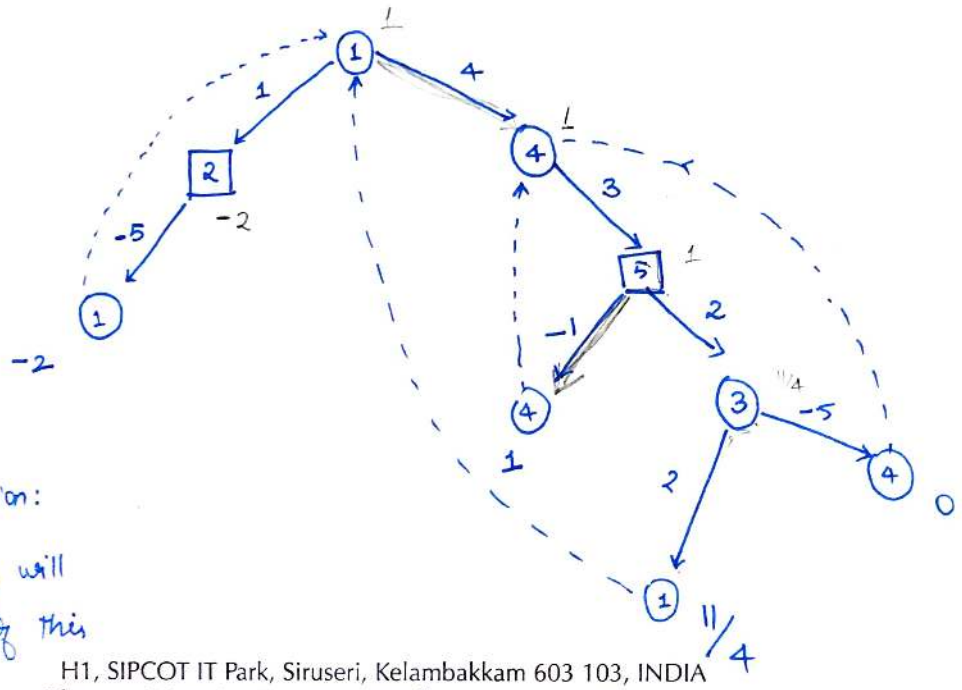
Motivated by this observation, we investigate a finite version of mean payoff games called the first-cycle version.

- Given (G, w, a_0). Play starts at a_0 and continues as before. However the play stops as soon as a vertex which was seen previously is encountered again.

$$a_0 \longrightarrow a_1 \longrightarrow a_2 \longrightarrow \cdots a_p \longrightarrow \cdots \longrightarrow a_q$$

first cycle formed.

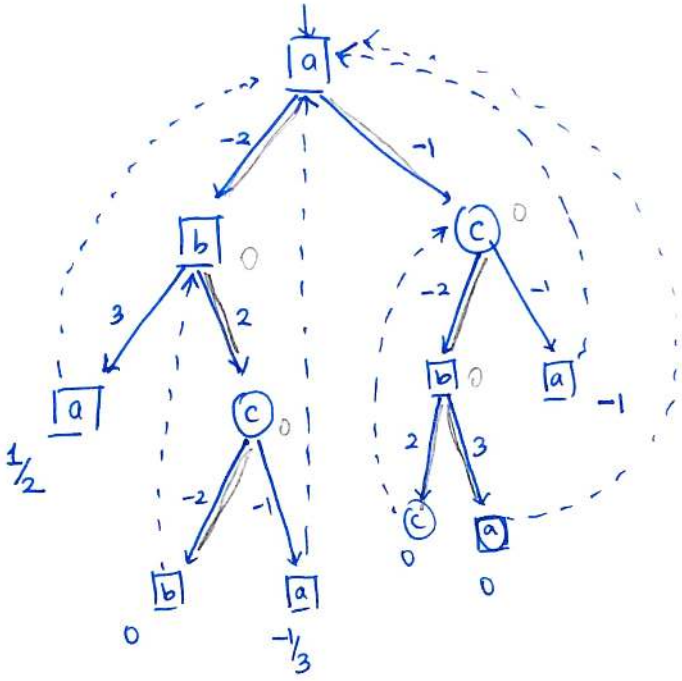
Payoff = Mean-weight of the cycle formed.



Idea of backward min max computation:

- Marked strategies will give the value of this finite game.

H1, SIPCOT IT Park, Siruseri, Kelambakkam 603 103, INDIA
Phone: +91 44 6748 0900 **Email:** office@algolabs.org.in



Observations: Different occurrences of a node could mean different possible values from each occurrence. The mean weight of the cycle formed depends on the history. For instance, for ⓒ, on the left there is a possibility to go to $-1/3$; and on the right, there is a possibility for -1.

→ The min - max computation starting from leaves all the way up to root will give a unique value. # The arg min / arg max gives strategies τ and σ for Minimizer and Maximizer that achieve this value.

→ It is not immediately clear that there exist positional strategies σ, τ for this game.

→ It is also not still clear why the value computed this way should be same as that for the infinite version game.

AlgoLabs

http://www.algolabs.org.in

Summary so far:

- 1. Mean-payoff games (infinite duration)
  Payoff, objectives, Result about existence of a value and positional strategies achieving this value.
- 2. First-cycle ~~Mean pa~~ mean payoff games
  Algorithm to compute value and (not necessarily positional) strategies.

Coming next:

- 1. Value computed in FC game = Value in Mean Payoff game
- 2. There exist positional strategies σ_{a_0}, τ_{a_0} which can achieve this value (in both the games).

Part 5: Equivalence between first cycle game and mean payoff game.

F → First cycle game starting at a_0

G → Infinite duration mean payoff game starting at a_0.

$\tilde{v}(a_0)$ → value in infinite game (existence not proved yet)

$v(a_0)$ → value in finite game (algorithm to compute $v(a_0)$ strategies σ, τ securing $v(a_0)$)

Pick the strategy σ for Max in F (not necessarily position).

- We will define strategy $\tilde{\sigma}$ as follows. It uses some finite memory (in the form of a bounded stack).

σ is a strategy defined on "simple paths".

- Game starts at a_0. start playing using σ

Initial memory $m_0 = a_0$.

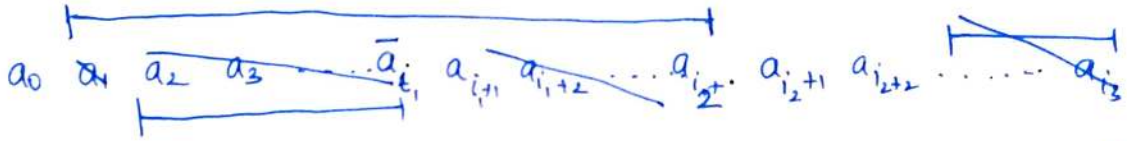
At the $(k+1)^{th}$ step, play $a_{k+1} = \sigma(m_k)$ if it is Max's turn.

Update m_{k+1} as follows.

- Suppose $m_k = u_0 u_1 u_2 \dots u_r$. $r \geq 0$, $r \leq n$

$$m_{k+1} = \begin{cases} u_0 u_1 \dots u_s & \text{if } v_{k+1} = u_s \text{ for some } s \leq r \\ u_0 u_1 \dots u_r v_{k+1} & \text{otherwise.} \end{cases}$$

Idea:



Think of this as a stack, each time there is a repetion pop the ~~contents~~ cycle, except the root.

Example: ~~σ~~ (first running example). $\sigma[1 \to 4,\ 14 \to 5,\ 1453 \to 1]$

$\tilde{\sigma}[1 \to 4]$ 1 ~~4 5 4 5 3 1~~ 4

$\tilde{\sigma}[14 \to 5]$ $\tilde{\sigma}$ uses a bounded stack as a memory

$\tilde{\sigma}[1454 \to 5]$ (finite memory strategy).

$\tilde{\sigma}[14543 \to 1]$

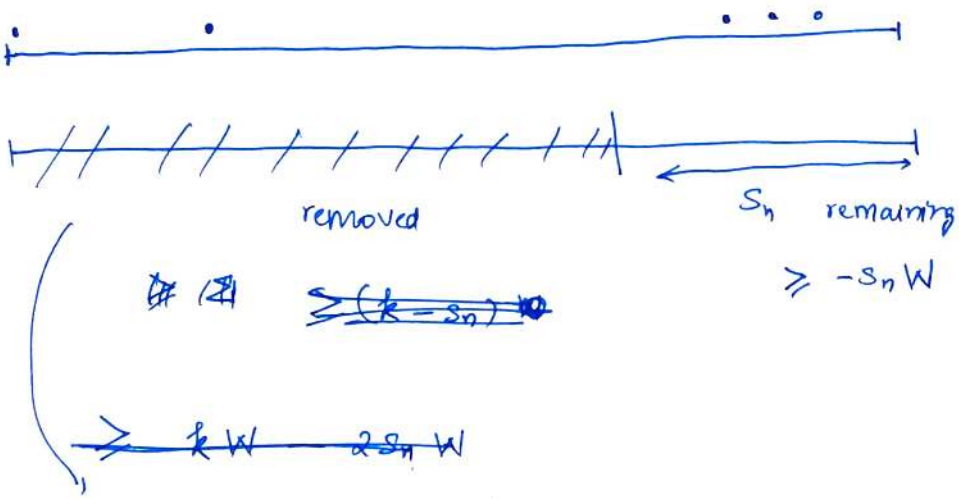
⋮

Claim: Playing $\tilde{\sigma}$ in G ensures that the value obtained for any play according to this strategy $\geq v(a_0)$.

Proof: Consider the mean payoff game played upto k steps, according to $\tilde{\sigma}$.

What is $\frac{\sum_{i=0}^{k-1} w_i}{k}$?

~~After k steps~~ In this process of playing k steps, say $C_1, C_2, \ldots C_m$ were the cycles removed. Let there be s_k elements left in the memory.



~~$\geq k W - 2 S_n W$~~

$\geq -S_n W$

$\geq \left(\sum |C_i|\right) \cdot \frac{\text{val}(a_0)}{V(a_0)}$

Sum of wts. $\geq (k - S_n)\, V(a_0) - S_n W$

$S_n \leq n$

$-S_n \geq -n$

$\geq (k - n)\, V(a_0) - n W$

Mean $\frac{\sum w_i}{k} \geq V(a_0) - \frac{n}{k} \cdot V(a_0) - \frac{nW}{k}$

$V(a_0) \geq -W$

$\geq V(a_0) - \frac{2nW}{k}$

$$\liminf_{k\to\infty} \frac{\Sigma_i w_i}{k} \geq \liminf_{k\to\infty} \frac{v(a_0) - 2nW}{k}$$

$$\geq v_0(a_0)$$

This proves that using $\tilde{\sigma}$ gives value at least $v(a_0)$.

- Using similar arguments, we can construct $\tilde{\tau}$ which ensures that the value of any play has payoff $\leq v(a_0)$.
- This gives us two strategies $\tilde{\sigma}$, $\tilde{\tau}$ and a value $\tilde{v}(a_0) = v(a_0)$ s.t. $\tilde{v}(a_0)$ is a minimax equilibrium.
- We also know how to get $\tilde{\sigma}$, $\tilde{\tau}$ using algorithm to compute σ and τ.
- However we still have not obtained positional strategies for any of these games. To do this, we need to go back and forth between the finite and infinite versions. This is a novel technique. ~~to prove such etc~~