



AlgoLabs

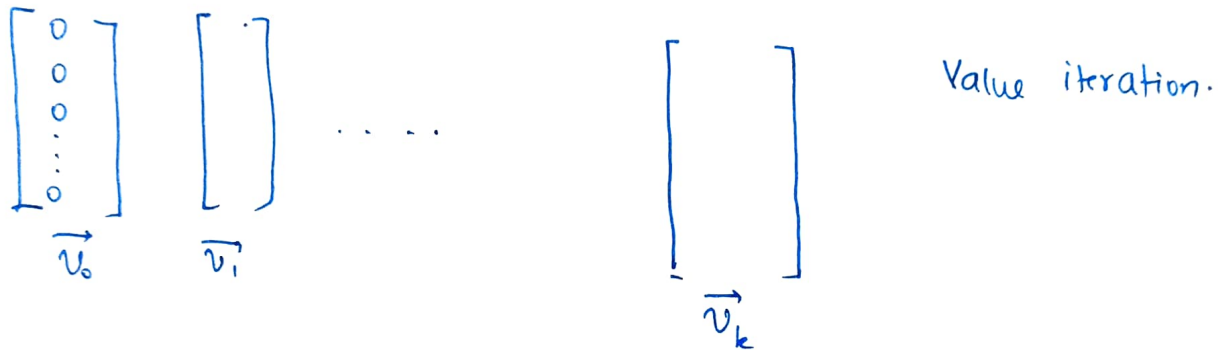
<http://www.algolabs.org.in>

(7)

The following equations can be shown by induction:

$$v_k(a) = \begin{cases} \max_{a \rightarrow b} (w_{ab} + v_{k-1}(b)) & a \in V_{\max} \\ \min_{a \rightarrow b} (w_{ab} + v_{k-1}(b)) & a \in V_{\min} \end{cases}$$

This gives an algorithm to compute \vec{v}_k for each k .



Claim: Compute \vec{v}_k for $k = \lceil n^3 W \rceil$. The value vector for the mean-payoff game can then be deciphered in constant-time.

Lemma 1: For every $a \in V$,

$$k \cdot v(a) - 2nW \leq v_k(a) \leq k \cdot v(a) + 2nW$$

Proof: Consider optimal strategy σ in F starting from 'a'. Show that playing that strategy ^{with stack-based extension $\tilde{\sigma}$} gives $v_k(a) \geq k \cdot v(a) - 2nW$

Similarly using τ and $\tilde{\tau}$ show other inequality.

(8)

In ~~from~~ above Lemma, substituting $k = 4n^3 W$ give:

$$\frac{v_k(a)}{k} \leq v(a) + \frac{2nW}{k}$$

ie,

$$\leq v(a) + \frac{1}{2n^2} \quad \text{--- } \textcircled{1}$$

$$n^2 \geq n(n-1)$$

$$\therefore \frac{1}{n^2} \leq \frac{\cancel{n(n-1)} 1}{n(n-1)}$$

$$\therefore -\frac{1}{n^2} \geq \frac{-1}{n(n-1)} \rightarrow \textcircled{2}$$

~~Let~~ $v'(a) = \frac{v_k(a)}{k}$

$$v(a) \geq v'(a) + \frac{1}{2n^2}$$

Similarly $\frac{v_k(a)}{k} \geq v(a) + \frac{1}{2n^2}$

Let $v'(a) = \frac{v_k(a)}{k}$

① gives: $v(a) \geq v'(a) - \frac{1}{2n^2}$

② give: $v(a) > v'(a) - \frac{1}{2n(n-1)}$ [strict inequality $\because n > 1$]

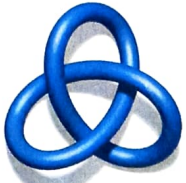
Similarly ~~$v'(a) \geq v(a) - \frac{2nW}{k}$~~ [previous Lemma]

$$v(a) \leq v'(a) + \frac{1}{2n^2} \quad [k = 4n^3 W]$$

From ②: $v(a) < v'(a) + \frac{1}{2n(n-1)}$ (cf)

Overall: $v'(a) - \frac{1}{2n(n-1)} < v(a) < v'(a) + \frac{1}{2n(n-1)}$

(9)



AlgoLabs

<http://www.algolabs.org.in>

From the previous discussion, it follows that if we do the value iteration for $k = 4n^3W$ steps, we get a vector $v'(a)$: and $v(a)$ is in the $\frac{1}{2n(n-1)}$ ball around it.

Can we uniquely spot it?

Observation about value: From the first cycle game, note that the value will be of the form p/q where $q \leq n$, [p, q are ~~the~~ integers].

Claim: Pick 2 no.s. $\frac{a}{k}, \frac{b}{l}$ ~~distinct~~ $\frac{a}{b}, \frac{p}{q}$ s.t. $0 \leq q, b \leq n$.
 $n \neq 1$.

Then $\left| \frac{a}{b} - \frac{p}{q} \right| \geq \frac{1}{n(n-1)}$

Proof: Suppose $b = q$.

$$\left| \frac{a}{b} - \frac{p}{q} \right| = \left| \frac{a-p}{q} \right| = \frac{|a-p|}{q} \geq \frac{1}{q} \geq \frac{1}{n} \geq \frac{1}{n(n-1)}$$

Suppose $b \neq q$.

$$\left| \frac{a}{b} - \frac{p}{q} \right| = \frac{|aq - bp|}{bq} \geq \frac{1}{bq} \geq \frac{1}{n(n-1)}$$

Therefore in the ball $\left(v'(a) - \frac{1}{2n(n-1)}, v'(a) + \frac{1}{2n(n-1)} \right)$ there is at most

one number of the form $\frac{p}{q}$ with $q \leq n$! Moreover we know

there exists one such number - which is the value $v(a)$.

How do we find it?

$$a' = v'(a) - \frac{1}{2n(n-1)}$$

$$b' = v'(a) + \frac{1}{2n(n-1)}$$

If there is an integer between a' and b' , then we are done.

Otherwise check if there is some $\frac{p}{q}$ with $q=2$. Multiply a', b' by 2.

Summary so far:

1. Proof of positional determinacy of finite as well as infinite version for the game starting at a_0 . ~~[positional strategy for infinite~~ [proof that was discussed can be applied to both versions. same strategy works for both cases].
2. Computing values uniformly for all vertices using a value iteration method. - Uri-Zwick's procedure.

Complexity: $4n^3 W \cdot |E|$

$$= O(|V|^3 \cdot |W| \cdot |E|)$$

→ pseudopolynomial.

Weights can be represented in binary. So all weights need $\log W$ bits.

An algo with $k|E|$ steps will then be exponential in representation.

→ Same as PRIMES. If number is p represented in unary (no. of bits same as the number) then simple algos work in PTIME. Main question was what happens when we have a binary representation.

Coming next: The above ~~comp~~ value iteration method computes

values. But it does not give a positional strategy immediately.

How to compute uniform positional strategies?

→ Theorem 3.1 in Zwick-Patterson's paper.