

# Math 525: Lecture 22

April 05, 2018

In the last lecture, we showed that the running cost  $v$  of a Markov decision process satisfies a so-called *Bellman equation*:

$$\sup_{\pi_0 \in \Pi_0} \{(I - B(\pi_0))v - c(\pi_0)\} = 0 \quad (1)$$

where

$$B(\pi_0) = \begin{pmatrix} b_1(\pi_0(1))^\top \\ b_2(\pi_0(2))^\top \\ \vdots \\ b_m(\pi_0(m))^\top \end{pmatrix} \text{ with } b_i(\cdot) \geq 0 \text{ and } b_i(\cdot)^\top e \leq 1 \quad \text{and} \quad c(\pi_0) = \begin{pmatrix} c(\pi_0(1), 1) \\ c(\pi_0(2), 1) \\ \vdots \\ c(\pi_0(m), m) \end{pmatrix}.$$

In the case of a fixed discount factor  $0 \leq d < 1$ , we saw that  $B$  simplified to

$$B(\pi_0) = d \underbrace{\begin{pmatrix} p_1(\pi_0(1))^\top \\ p_2(\pi_0(2))^\top \\ \vdots \\ p_m(\pi_0(m))^\top \end{pmatrix}}_{P(\pi_0)} \text{ with } p_i(\cdot) \geq 0 \text{ and } p_i(\cdot)^\top e = 1.$$

In this lecture, we will establish uniqueness of a solution  $v$  to (1) and discuss how to compute it. Throughout, we assume

$\Pi_0$  is a finite set

(i.e., there are only finitely many actions the controller can take at each state). While this assumption can be relaxed, we do not do so here. In this case, 1 becomes

$$\max_{\pi_0 \in \Pi_0} \{A(\pi_0)v - c(\pi_0)\} = 0 \quad \text{where} \quad A(\pi_0) = I - B(\pi_0). \quad (2)$$

# 1 Matrix classes

First, we will need to recall some more linear algebra.

## 1.1 Monotone matrices

**Definition 1.1.** A *monotone matrix* is a real square matrix  $A$  such that  $Ax \geq 0$  implies  $x \geq 0$  for all real vectors  $x$ .

**Proposition 1.2.** *Monotone matrices are nonsingular.*

*Proof.* Let  $A$  be a monotone matrix and assume there exists  $x$  with  $Ax = 0$ . Then, by monotonicity,  $x \geq 0$  and  $-x \geq 0$ , and hence  $x = 0$ .  $\square$

**Proposition 1.3.** *A real square matrix  $A$  is monotone if and only if  $A^{-1}$  exists and is nonnegative.*

*Proof.* Suppose  $A$  is monotone. Denote by  $x$  the  $i$ -th column of  $A^{-1}$ . Then,  $Ax$  is the  $i$ -th standard basis vector, and hence  $x \geq 0$  by monotonicity. For the reverse direction, suppose  $A$  admits a nonnegative inverse. Then, if  $Ax \geq 0$ ,  $x = A^{-1}Ax \geq A^{-1}0 = 0$ , and hence  $A$  is monotone.  $\square$

## 1.2 M-matrices

**Definition 1.4.** An *M-matrix* is any square matrix  $A$  which can be written in the form

$$A = sI - B \tag{3}$$

where  $s \geq \rho(B)$  and  $B$  is nonnegative.

**Proposition 1.5.** *The M-matrix (3) is nonsingular if and only if  $s > \rho(B)$ .*

*Proof.* Note that  $A$  is nonsingular if and only if  $s$  is an eigenvalue of  $B$  since

$$Ax = sx - Bx = 0 \iff Bx = sx.$$

Therefore, if  $s > \rho(B)$ , then  $A$  is nonsingular. Conversely, if  $s = \rho(B)$ , by the Perron-Frobenius theorem,  $s$  is an eigenvalue of  $B$ .  $\square$

**Proposition 1.6.** *Nonsingular M-matrices are monotone.*

*Proof.* Divide (3) by  $s$  so that

$$s^{-1}A = I - s^{-1}B.$$

Noting that  $\rho(s^{-1}B) < 1$ , the inverse of the right hand side of the above is the Neumann series

$$(I - s^{-1}B)^{-1} = \sum_{k \geq 0} (s^{-1}B)^k.$$

In particular, this Neumann series consists only of powers of nonnegative matrices, and therefore converges to a nonnegative matrix. In other words,  $sA^{-1}$  is nonnegative, and hence so too is  $A^{-1}$ .  $\square$

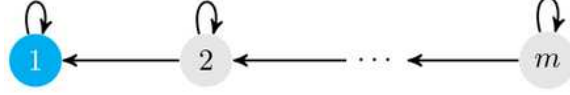


Figure 1: Graph of (5)

### 1.3 Weakly chained diagonally dominant matrices

**Definition 1.7.** Let  $A = (A_{ij}) \in \mathbb{C}^{m \times n}$  be a matrix. We say its  $i$ -th row is *strictly diagonally dominant* (s.d.d.) if

$$|A_{ii}| > \sum_{i \neq j} |A_{ij}|. \quad (4)$$

We say the matrix is s.d.d. if all of its rows are s.d.d. Weakly diagonally dominant (w.d.d.) is defined with weak inequality ( $\geq$ ) instead.

**Example 1.8.** The matrix

$$\begin{pmatrix} 1 & & & \\ -1 & 1 & & \\ & -1 & 1 & \\ & & -1 & 1 \end{pmatrix} \quad (5)$$

is not strictly diagonally dominant, but it is weakly diagonally dominant.

**Definition 1.9.** Let  $A = (A_{ij}) \in \mathbb{C}^{m \times m}$  be a matrix. Let

$$J = \{i: i \text{ satisfies (4)}\}$$

denote the set of all s.d.d. rows of  $A$ . We say  $A$  is *weakly chained diagonally dominant* (w.c.d.d.) if

1.  $A$  is w.d.d.
2. For each row  $i \notin J$ , there is a walk  $i \rightarrow j$  with  $j \in J$ .

**Example 1.10.** The matrix (5) is w.c.d.d. (see Figure 1).

The following first appeared in Shivakumar, P. N., and Kim Ho Chew. “A sufficient condition for nonvanishing of determinants.” *Proceedings of the American Mathematical Society* (1974).

**Proposition 1.11.** *w.c.d.d. matrices are nonsingular.*

*Proof.* Let  $A$  be w.c.d.d. If  $A$  is singular, we can find a nonzero vector  $x$  such that  $Ax = 0$ . Without loss of generality, let  $i_1$  be such that  $|x_{i_1}| = 1 \geq |x_j|$  for all  $j$ . Since  $A$  is w.c.d.d., we may pick a walk  $i_1 \rightarrow i_2 \rightarrow \dots \rightarrow i_k$  ending at an s.d.d. row  $i_k \in J$ .

Taking moduli on both sides of

$$-a_{i_1 i_1} x_{i_1} = \sum_{j \neq i_1} a_{i_1 j} x_j$$

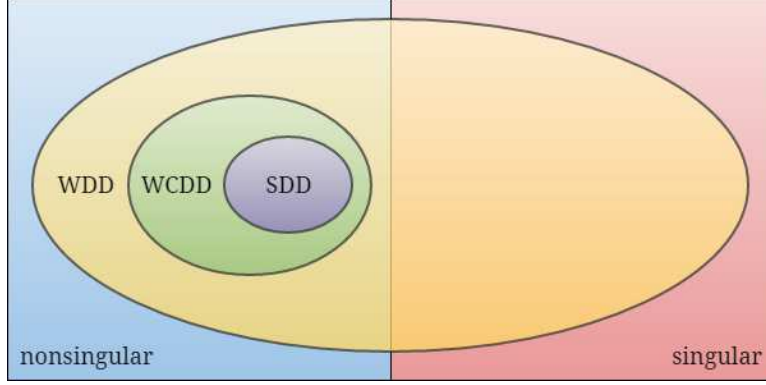


Figure 2: Containment of matrix classes

yields

$$|a_{i_1 i_1}| = |a_{i_1 i_1} x_{i_1}| = \left| \sum_{j \neq i_1} a_{i_1 j} x_j \right| \leq \sum_{j \neq i_1} |a_{i_1 j}| |x_j| \leq \sum_{j \neq i_1} |a_{i_1 j}|.$$

Since  $A$  is w.d.d., the above must hold with equality. Therefore,  $|x_j| = 1$  whenever  $a_{i_1 j}$  is nonzero. In particular,  $|x_{i_2}| = 1$ , and we can repeat the same argument as above to get that row  $i_2$  is not s.d.d., row  $i_3$  is not s.d.d., etc. until we conclude that row  $i_k$  is not s.d.d., a contradiction.  $\square$

**Theorem 1.12.** *Let  $A$  be a square w.d.d. matrix with nonnegative diagonals ( $A_{ii} \geq 0$ ) and nonpositive off-diagonals (i.e.,  $A_{ij} \leq 0$  for  $i \neq j$ ). Then, the following are equivalent:*

1.  $A$  is a nonsingular M-matrix.
2.  $A$  is a nonsingular matrix with positive diagonals.
3.  $A$  is w.c.d.d.

Throughout the proof we use the fact that  $A$  can be written in the form  $A = sI - B$  by taking  $s = \max_i A_{ii}$ ,  $B_{ii} = s - A_{ii}$ , and  $B_{ij} = -A_{ij}$  for  $j \neq i$ . Since  $A$  is w.d.d., this implies

$$s - B_{ii} = A_{ii} \geq \sum_{j \neq i} |A_{ij}| = \sum_{j \neq i} B_{ij}$$

and hence

$$\rho(B) \leq \|B\|_\infty = \max_i \left\{ B_{ii} + \sum_{j \neq i} B_{ij} \right\} \leq \max_i \{ B_{ii} + s - B_{ii} \} = s.$$

In other words,  $A$  is a (possibly singular) M-matrix.

*Proof of (1)  $\implies$  (2).* It is sufficient to prove that if  $A$  is nonsingular, then  $A$  has positive diagonals. We prove the contrapositive. Suppose  $A_{ii} = 0$  for some  $i$ . Then,  $|A_{ij}| = B_{ij} = 0$  for all  $j$ . That is, the  $i$ -th row of  $A$  is zero, and hence  $A$  is singular.  $\square$

*Proof of (2)  $\implies$  (3).* This proof is nontrivial, and hence I refer you to (self-plug) Azimzadeh, P. “A fast and stable test to check if a weakly diagonally dominant matrix is a nonsingular M-matrix.” *Mathematics of Computation* (2018).  $\square$

*Proof of (3)  $\implies$  (1).* If  $A$  is w.c.d.d., it is nonsingular (Proposition 1.11).  $\square$

**Corollary 1.13.** *A square s.d.d. matrix with nonnegative diagonals and nonpositive off-diagonals is a nonsingular M-matrix.*

*Proof.* An s.d.d. matrix is a w.c.d.d. matrix.  $\square$

## 2 Bellman equation

We start with a trivial observation.

**Lemma 2.1.**  *$A(\pi_0)$  is a square w.d.d. matrix with nonnegative diagonals and nonpositive off-diagonals.*

*Proof.* This is a result of the following observations:

- $(B(\pi_0))_{ii} \leq 1$  so that  $(A(\pi_0))_{ii} = 1 - (B(\pi_0))_{ii} \geq 0$ ,
- $(B(\pi_0))_{ij} \geq 0$  so that  $(A(\pi_0))_{ij} = -(B(\pi_0))_{ij} \leq 0$  whenever  $i \neq j$ , and
- $\sum_j (B(\pi_0))_{ij} \leq 1$  so that  $\sum_j (A(\pi_0))_{ij} = 1 - \sum_j (B(\pi_0))_{ij} \geq 0$ .  $\square$

The above implies that we can quickly check to see if  $A(\pi_0)$  is a nonsingular M-matrix (and hence monotone) by verifying if it is w.c.d.d. In the constant discount factor case,  $A(\pi_0)$  is trivially w.c.d.d.:

**Example 2.2.** Suppose  $A(\pi_0) = I - dP(\pi_0)$  for some  $0 \leq d < 1$ . Then,

$$\sum_j (A(\pi_0))_{ij} = 1 - d \sum_j (P(\pi_0))_{ij} = 1 - d > 0.$$

In other words,  $A(\pi_0)$  is s.d.d. since  $(A(\pi_0))_{ii} > \sum_{j \neq i} |(A(\pi_0))_{ij}|$ .

### 2.1 Uniqueness

**Theorem 2.3.** *Suppose  $A(\pi_0)$  is w.c.d.d. for each  $\pi_0$ . Then, solutions of (2) are unique.*

*Proof.* Let  $v$  and  $v'$  satisfy equation (2). Pick  $\pi_0^v$  such that

$$0 = \max_{\pi_0 \in \Pi_0} \{A(\pi_0)v - c(\pi_0)\} = A(\pi_0^v)v - c(\pi_0^v).$$

Then,

$$0 = \max_{\pi_0 \in \Pi_0} \{A(\pi_0)w - c(\pi_0)\} \geq A(\pi_0^v)w - c(\pi_0^v).$$

Putting these two together,

$$A(\pi_0^v)(v - w) \geq 0.$$

Since  $A(\pi_0^v)$  is monotone, its inverse exists and is nonnegative. Therefore, we can multiply both sides of the above inequality by  $(A(\pi_0^v))^{-1}$  to get  $v - w \geq 0$ . Switching the roles of  $v$  and  $w$  in this argument, we get  $w - v \geq 0$ .  $\square$

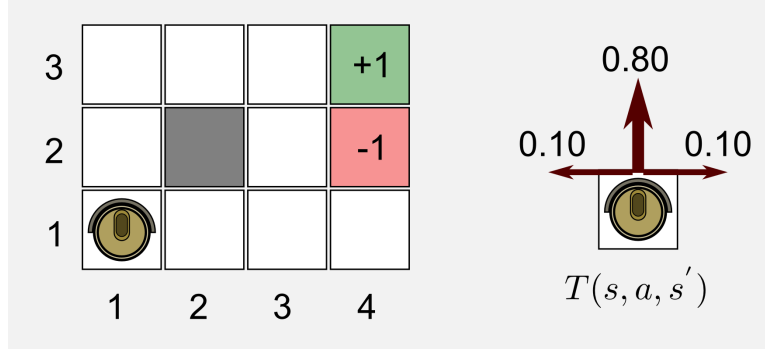


Figure 3: Robot navigation

## 2.2 Policy iteration

One way to compute the solution  $v$  is by the so-called *policy iteration algorithm*:

1. Pick an “initial guess”  $\pi_0^{(0)} \in \Pi_0$ . Set  $v^{(0)} = (-\infty, \dots, -\infty)$  and  $k \leftarrow 1$ .
2. Solve the linear system  $A(\pi_0^{(k-1)})v^{(k)} = b(\pi_0^{(k-1)})$  for  $v^{(k)} \in \mathbb{R}^m$ .
3. Pick  $\pi_0^{(k)} \in \Pi_0$  such that  $A(\pi_0^{(k)})v^{(k)} - b(\pi_0^{(k)}) = \max_{\pi_0 \in \Pi_0} \{A(\pi_0)v^{(k)} - b(\pi_0)\}$ .
4. If  $v^{(k)} = v^{(k-1)}$ , **stop**. Otherwise, set  $k \leftarrow k + 1$  and **go to** step 2.

**Theorem 2.4.** *Suppose  $A(\pi_0)$  is w.c.d.d. for each  $\pi_0$ . Then, the above policy iteration algorithm converges to the solution  $v$  of (2) in at most  $|\Pi_0|$  iterations (i.e.,  $v = v^{(|\Pi_0|)} = v^{(|\Pi_0|+1)} = \dots$ ).*

*Proof.* A nice reference for this fact is Bokanowski, Olivier, Stefania Maroso, and Hasnaa Zidani. “Some convergence results for Howard’s algorithm.” *SIAM Journal on Numerical Analysis* 47.4 (2009): 3001-3026.  $\square$

## 2.3 Example

- This example is by Massimiliano Patacchiola (<https://mpatacchiola.github.io/>).
- A robot, started in position  $(1, 1)$ , has to find the best way to reach the charging station (Reward +1) and to avoid falling down the flight of stairs (Reward -1).
- This is a Markov chain with state space  $S = \{(1, 1), (1, 2), \dots, (4, 3), \emptyset\}$  ( $\emptyset$  is an extra state which both  $(4, 3)$  and  $(4, 2)$  transition to deterministically to signify the end of the game).
- At each state  $i$ , the robot gets to choose which direction to face:  $\mathcal{A}_i = \{N, S, W, E\}$ . Then, the robot tries to move forward. However, being faulty, 10% of the time it moves to its left and 10% of the time it moves to its right. This determines the transition matrix  $P(\pi_0)$ .

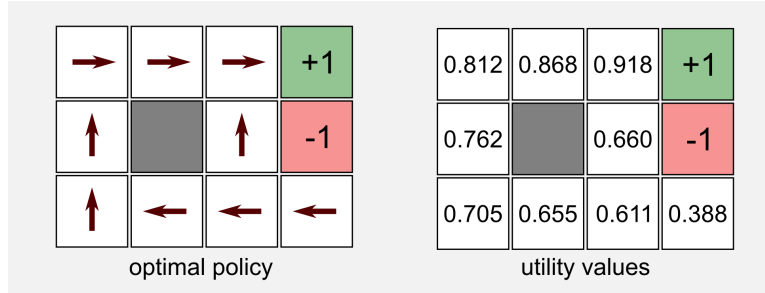


Figure 4: Result of running policy iteration

- The reward function is  $r(i) = I_{\{(4,3)\}}(i) - I_{\{(4,2)\}}(i)$ .
- We take a discount factor of  $d = 0.999 < 1$  so that  $A(\pi_0) = I - dP(\pi_0)$ .
- The Markov decision process is

$$\underbrace{\max_{\pi \in \Pi} \mathbb{E} \left[ \sum_{n \geq 0} 0.999^n r(X_n^\pi) \right]}_{\text{maximize rewards } r} = - \underbrace{\min_{\pi \in \Pi} \mathbb{E} \left[ \sum_{n \geq 0} 0.999^n (-r(X_n^\pi)) \right]}_{\text{minimize costs } c = -r}$$