Absorbing States and z-Transforms

ECE 341: Random Processes

Lecture #23

note: All lecture notes, homework sets, and solutions are posted on www.BisonAcademy.com

Absorbing States

Markov chains solve problems of the form

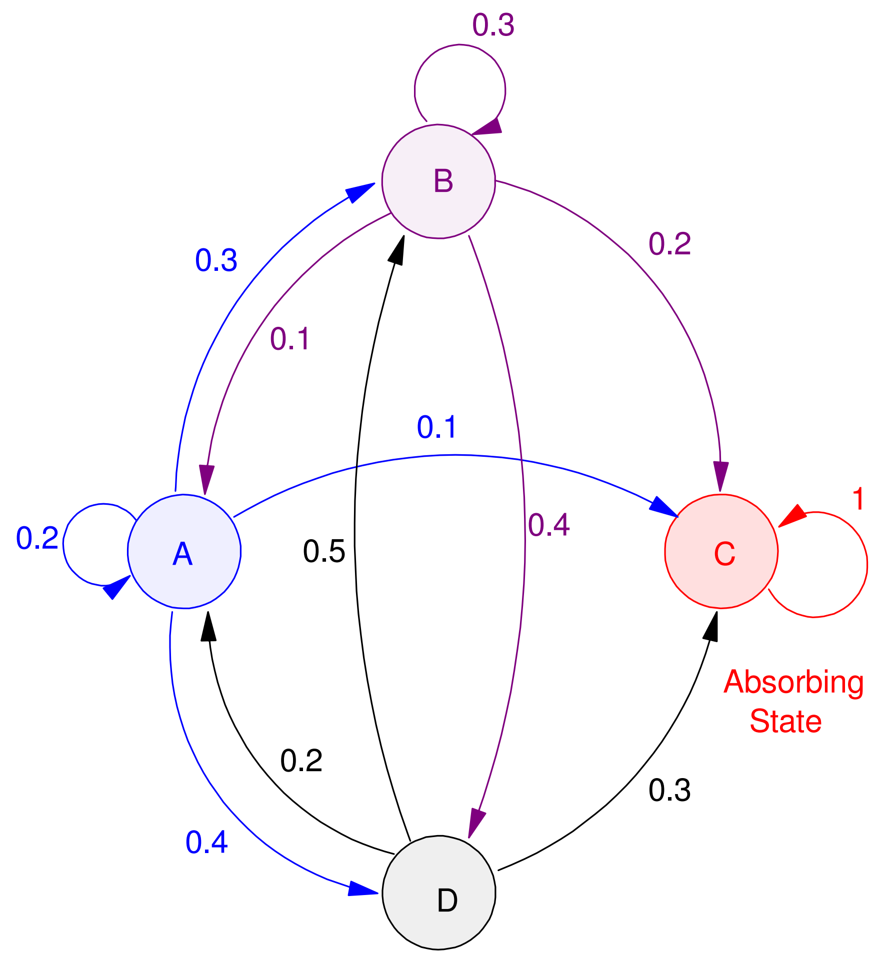
$$x(k+1) = A\ x(k) \qquad\qquad x(k=0) = X_0$$

The previous lecture looked at tossing a ball around

- Four players

An absorbing state is where a player keeps the ball

- State C in this case

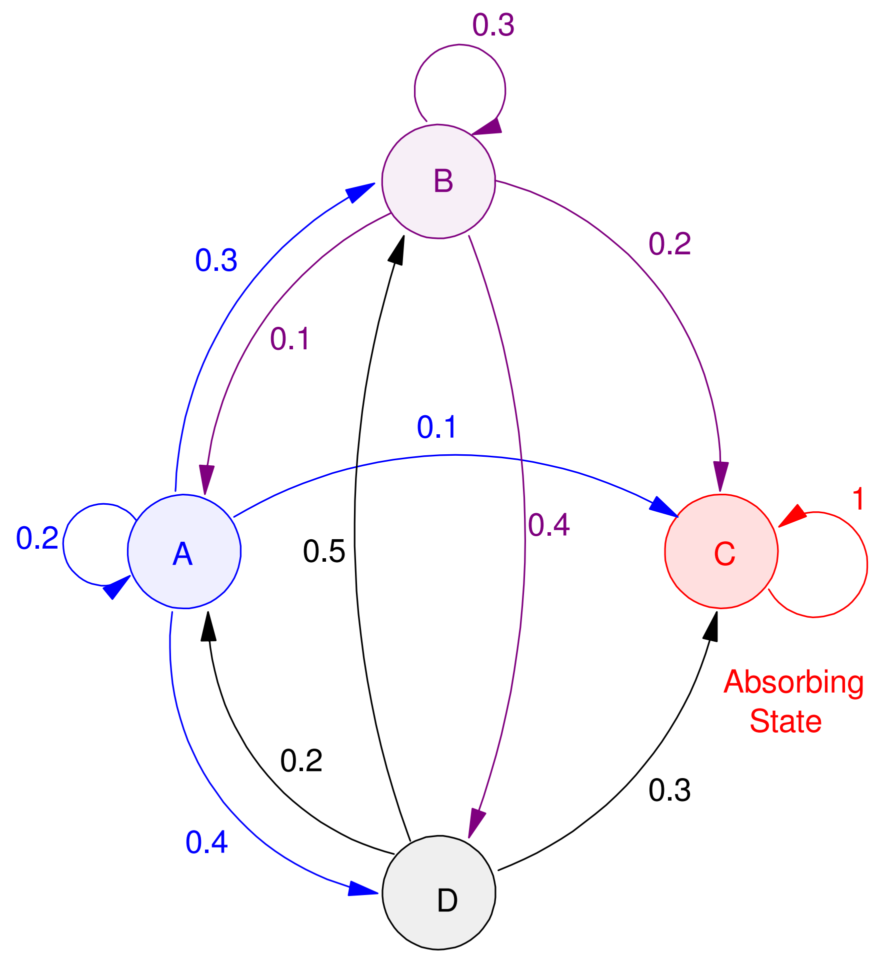


State Transistion Matrix with Absorbing States

A diagonal element of '1' indicated an absorbing state

```
>> A = [0.2, 0.1, 0, 0.2
        0.3, 0.3, 0, 0.1
        0.1, 0.2, 1, 0.4
        0.4, 0.4, 0, 0.3]
```

0.2000	0.1000	**0**	0.2000
0.3000	0.3000	**0**	0.1000
0.1000	0.2000	**1.0000**	0.4000
0.4000	0.4000	**0**	0.3000



Steate-State Solution

If

- There is only one absorbing state, and
- It is reachable

The ball always ends up at the absorbing state

```
>> A = [0.2,0.1,0,0.2 ; 0.3,0.3,0,0.1 ; 0.1,0.2,1,0.4 ; 0.4,0.4,0,0.3]

    0.2000    0.1000         0    0.2000
    0.3000    0.3000         0    0.1000
    0.1000    0.2000    1.0000    0.4000
    0.4000    0.4000         0    0.3000

>> A^100

    0.0000    0.0000         0    0.0000
    0.0000    0.0000         0    0.0000
    1.0000    1.0000    1.0000    1.0000
    0.0000    0.0000         0    0.0000
```

Eigenvalues & Eigenvectors

This also shows up with the eigenvalues and eigenvectrors

- The absorbing state has an eigenvalue of 1.0000
  - It never goes away
- It's eigenvector is the absorbing staet
  - State C

Eigenvectors

A	**0**	0.2023	0.6250	0.6250
B	**0**	0.2390	-0.5359 - 0.2959i	-0.5359 + 0.2959i
C	**1.0000**	-0.8549	0.1538 - 0.0847i	0.1538 + 0.0847i
D	**0**	0.4135	-0.2429 + 0.3806i	-0.2429 - 0.3806i

Eigenvalues

1.0000	0.7269	0.0365 - 0.0745i	0.0365 + 0.0745i

The z-transform for the absorbing state has a pole at z=1

- A starts with the ball

The z-transform for each state is:

$$A(z) = \left(\frac{(z-0.1)(z-0.5)z}{(z-0.7269)(z^2-0.07308z+0.006878)}\right)$$

$$B(z) = \left(\frac{0.3z(z-0.1667)}{(z-0.7269)(z^2-0.07308z+0.006878)}\right)$$

$$C(z) = \left(\frac{0.1z(z+1.631)(z-0.03066)}{(z-1)(z-0.7269)(z^2-0.07308z+0.006878)}\right)$$

<<< pole at z = 1

$$D(z) = \left(\frac{0.4z^2}{(z-0.7269)(z^2-0.07308z+0.006878)}\right)$$

note:

- Only state C has a pole at z=1
- C is an absorbing state

As you run a simulation, you wind up at the absorbing state

k	p(A)	p(B)	**p(C)**	p(D)
0	0.2500	0.2500	**0.2500**	0.2500
1	0.1250	0.1750	**0.4250**	0.2750
2	0.0975	0.1175	**0.5825**	0.2025
3	0.0718	0.0847	**0.6967**	0.1468
4	0.0522	0.0616	**0.7796**	0.1066
5	0.0379	0.0448	**0.8398**	0.0775
6	0.0276	0.0326	**0.8835**	0.0563
7	0.0200	0.0237	**0.9153**	0.0410
8	0.0146	0.0172	**0.9385**	0.0298
9	0.0106	0.0125	**0.9553**	0.0216
10	0.0077	0.0091	**0.9675**	0.0157
11	0.0056	0.0066	**0.9764**	0.0114
12	0.0041	0.0048	**0.9828**	0.0083
13	0.0030	0.0035	**0.9875**	0.0060
14	0.0021	0.0025	**0.9909**	0.0044
15	0.0016	0.0018	**0.9934**	0.0032
16	0.0011	0.0013	**0.9952**	0.0023
17	0.0008	0.0010	**0.9965**	0.0017
18	0.0006	0.0007	**0.9975**	0.0012
19	0.0004	0.0005	**0.9982**	0.0009
20	0.0003	0.0004	**0.9987**	0.0006

There can be several absorbing states

Example: Win By Two game

- Example: Tennis

Absorbing States

- A is up 2 games
  - A wins
- B is up 2 games
  - B wins

Win Two vs Win By Two

There's a big difference between

- First to win 2 games and
- First to win *by* 2 games

The former is a binomial distribution

- 3 game series

The latter is a Markov chain

- Series can go on forever
- Example: 2022 US Open
  - Alcaraz vs. Sinner
  - 5 hour 15 minute tennis match

Example: First to Win 2 Games

What is the chance A wins a 3-game series?

- A has a 70% chance of winning any given game
- Example: NBA semi-finals

Solution: A can win three ways (enumeration):

- AA
- ABA
- BAA

The total odds of A winning is

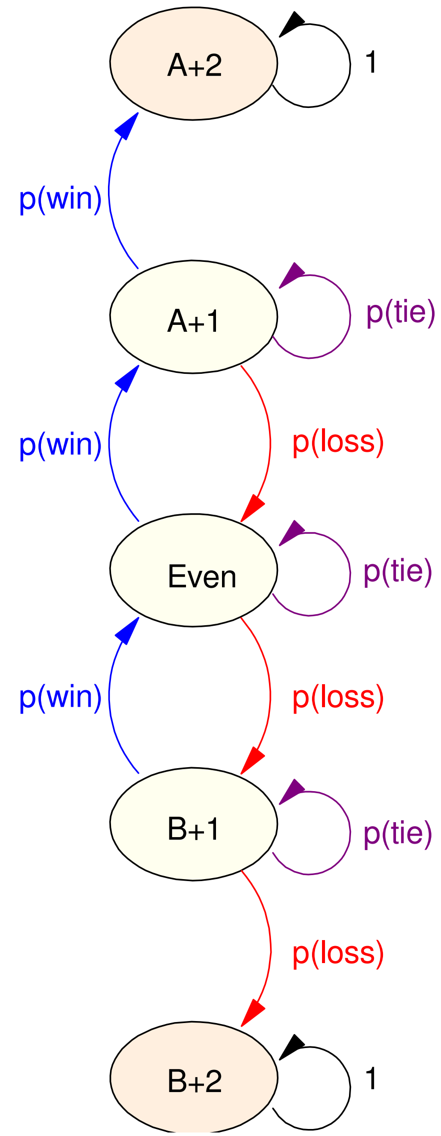
$$p(A) = p^2 + 2p^2q$$

$$p(A) = 0.7840$$

Example: First to Win By 2 Games

There are five states:

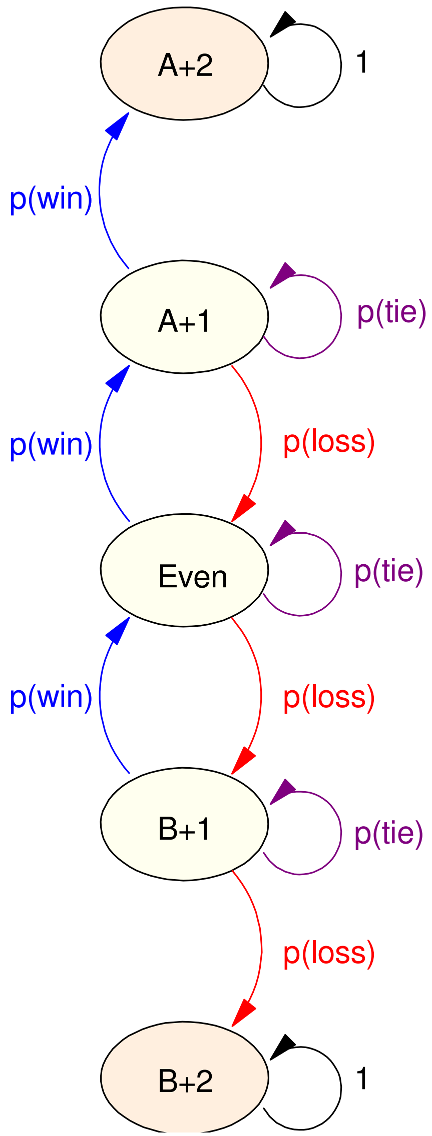
- A is up 2 games (A wins)
- A is up 1 game
- Tied
- B is up 1 game
- B is up 2 games (B wins)



Win By 2 Games: State Transition Matrix

- 70% chance of a win
- 0% chance of a tie
- 30% chance of a loss

$$X(k+1) = \begin{bmatrix} 1.0 & 0.7 & 0 & 0 & 0 \\ 0 & 0 & 0.7 & 0 & 0 \\ 0 & 0.3 & 0 & 0.7 & 0 \\ 0 & 0 & 0.3 & 0 & 0 \\ 0 & 0 & 0 & 0.3 & 1.0 \end{bmatrix} X(k)$$



Win-By-2 Series: Who will Win?

Find the steady-state solution

$$x(k+1) = Ax(k) = x(k)$$

$$(A-I)x(k) = 0$$

This doesn't help in this case due to the absorbing states. If you try to solve, you get

$$\begin{bmatrix} 0 & 0.7 & 0 & 0 & 0 \\ 0 & -1 & 0.7 & 0 & 0 \\ 0 & 0.3 & -1 & 0.7 & 0 \\ 0 & 0 & 0.3 & -1 & 0 \\ 0 & 0 & 0 & 0.3 & 0 \end{bmatrix} \begin{bmatrix} a \\ b \\ c \\ d \\ e \end{bmatrix} = 0 \qquad X(\infty) = \begin{bmatrix} a \\ 0 \\ 0 \\ 0 \\ e \end{bmatrix}$$

result: someone wins

Option 2: Eigenvectors.

The eigenvalues and eigenvectors tell you

- How the system behaves (eigenvalues) and
- What behaves that way.

```
[M,V] = eig(A)

     1.0000         0   -0.8033    0.2846   -0.5384
          0         0    0.4039   -0.6701    0.7692
          0         0    0.3739    0.6203   -0.0000
          0         0    0.1731   -0.2872   -0.3297
          0    1.0000   -0.1476    0.0523    0.0989

V:   1.0000    1.0000    0.6481   -0.6481         0
```

The eigenvectors tell you that eventually,

- A wins (first eigenvector), or
- B wins (second eivenvector).

Option 3: Play the game a large number of times (100 times).

In Matlab:

```
A = [1,0,0,0,0;0.7,0,0.3,0,0;0,0.7,0,0.3,0;0,0,0.7,0,0.3;0,0,0,0,1]'

    1.0000    0.7000         0         0         0
         0         0    0.7000         0         0
         0    0.3000         0    0.7000         0
         0         0    0.3000         0         0
         0         0         0    0.3000    1.0000

A^100

    1.0000    0.9534    0.8448    0.5914         0
         0         0    0.0000         0         0
         0    0.0000         0    0.0000         0
         0         0    0.0000         0         0
         0    0.0466    0.1552    0.4086    1.0000
```

A^100 tells you the probability of

- A winning (first row) and
- B winning (last row)

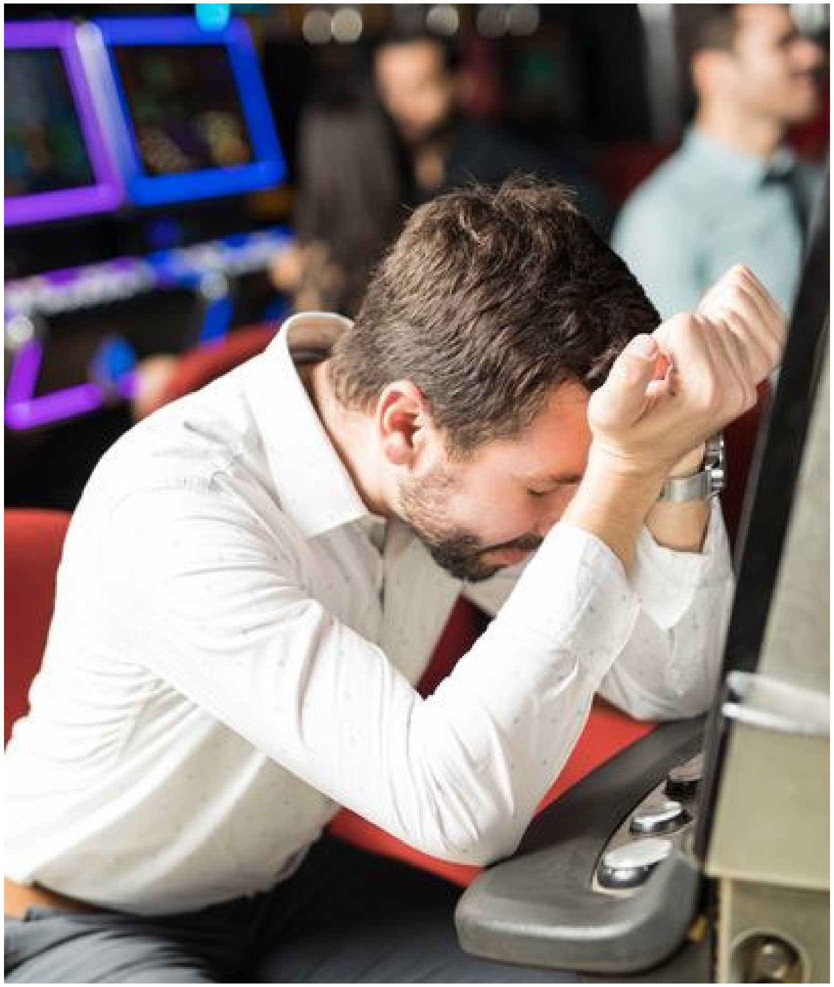
The columns tell you the probability if you offer odds:

- Column 1: Player A starts with a +2 game advantage (player A always wins)
- Column2: Player A starts with a +1 game advantage (A wins 95.34% of the time)
- Column 3: Player A starts with a +0 game advantage (A wins 84.48% of the time)
- Column 4: Player B starts with a +1 game advantage (A wins 59.14% of the time)
- Column 5: Player B starts with a +2 game advantage (B always wins)

Odds	+2	+1	+0	-1	-2	
	1.0000	**0.9534**	**0.8448**	**0.5914**	**0**	**A wins**
	0	0	0.0000	0	0	
	0	0.0000	0	0.0000	0	
	0	0	0.0000	0	0	
	0	**0.0466**	**0.1552**	**0.4086**	**1.0000**	**B wins**

Good Money After Bad

- If you start losing, keep gambling to recoup your losses
- This tends to result in you getting further behind
- You're risking money you have (good money) to recoup money you lost (bad money)



Good Money After Bad

For example, suppose you play a game of chance.

- 55% of the time you win and earn $1.
- 45% of the time you lose and lose $1.

Keep playing until you are up $10

- Absorbing state

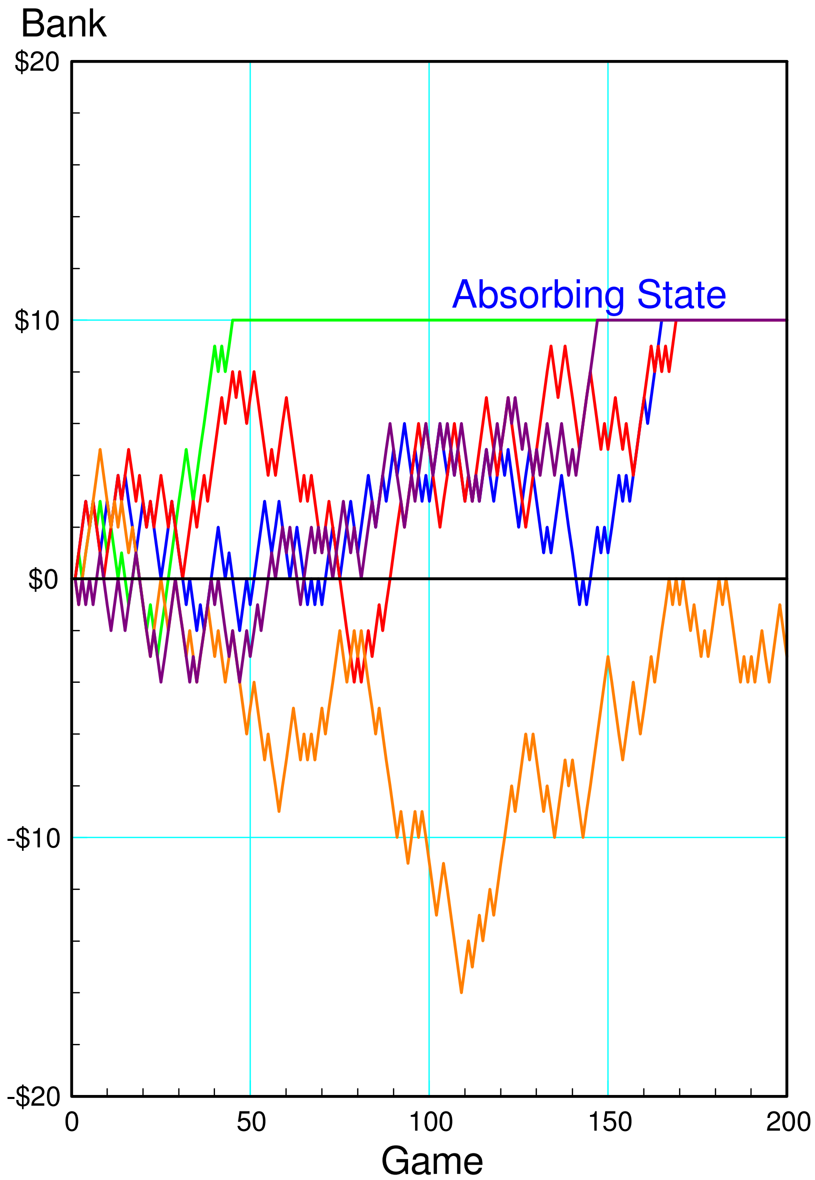
In theory, you always up $10

- Only one absorbing state
- Sounds like a safe way to make $10

Monte-Carlo Simulation

- Keep playing until you are up $10.
- After 200 games, you're almost always up $10

```
% game of chance

X = zeros(200,5);
for k=1:5
    Bank = 0;
    for n=2:200;
        if(Bank < 10)
            if (rand < 0.55)
                Bank = Bank + 1;
            else
                Bank = Bank - 1;
            end
        end
        X(n,k) = Bank;
    end
end

k = [1:200]';
plot(k,X)
```



Change the problem

- p(winning) = 45%

There still is only one absorbing state

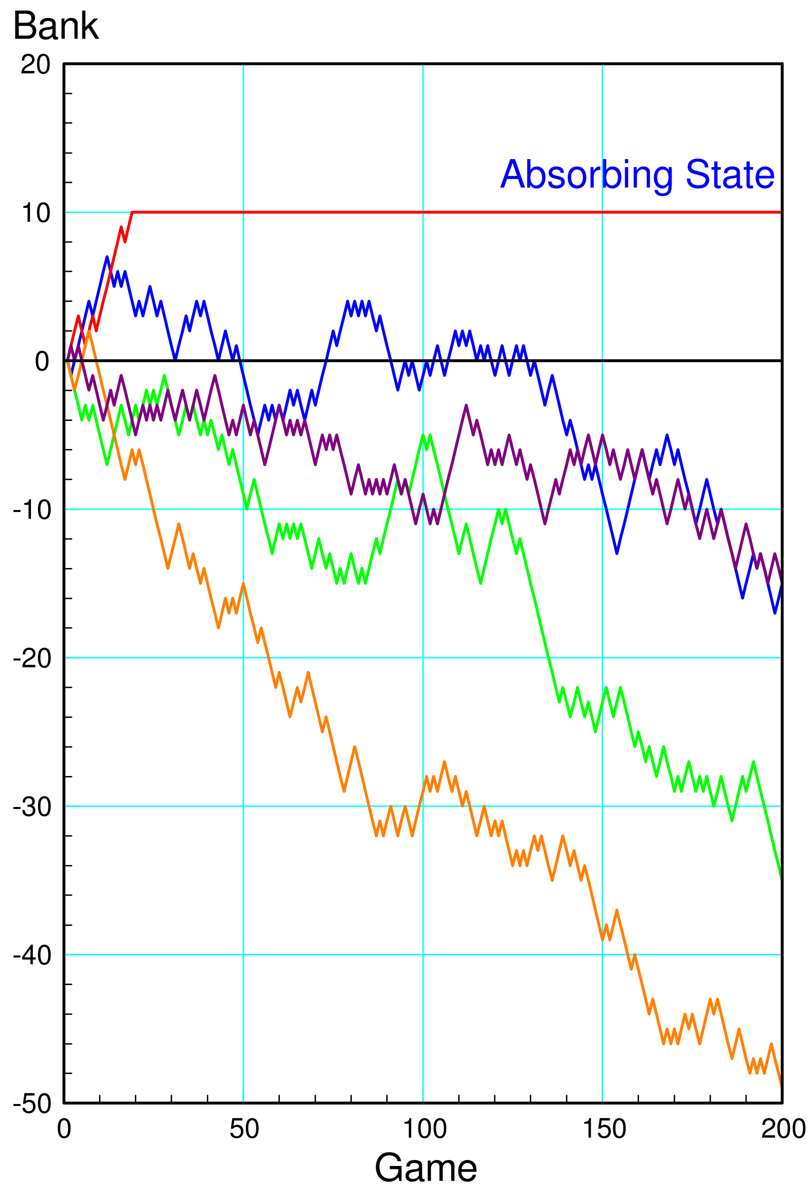
- Quit when you're up $10
- In theory, you *always* end at the absorbing state

Sometimes you win

- Reach the absorbing state

Other times the game keeps going

- What's happening?
- Why don't you always end at the absorbing state?



Better Model

Add a second absorbing state:

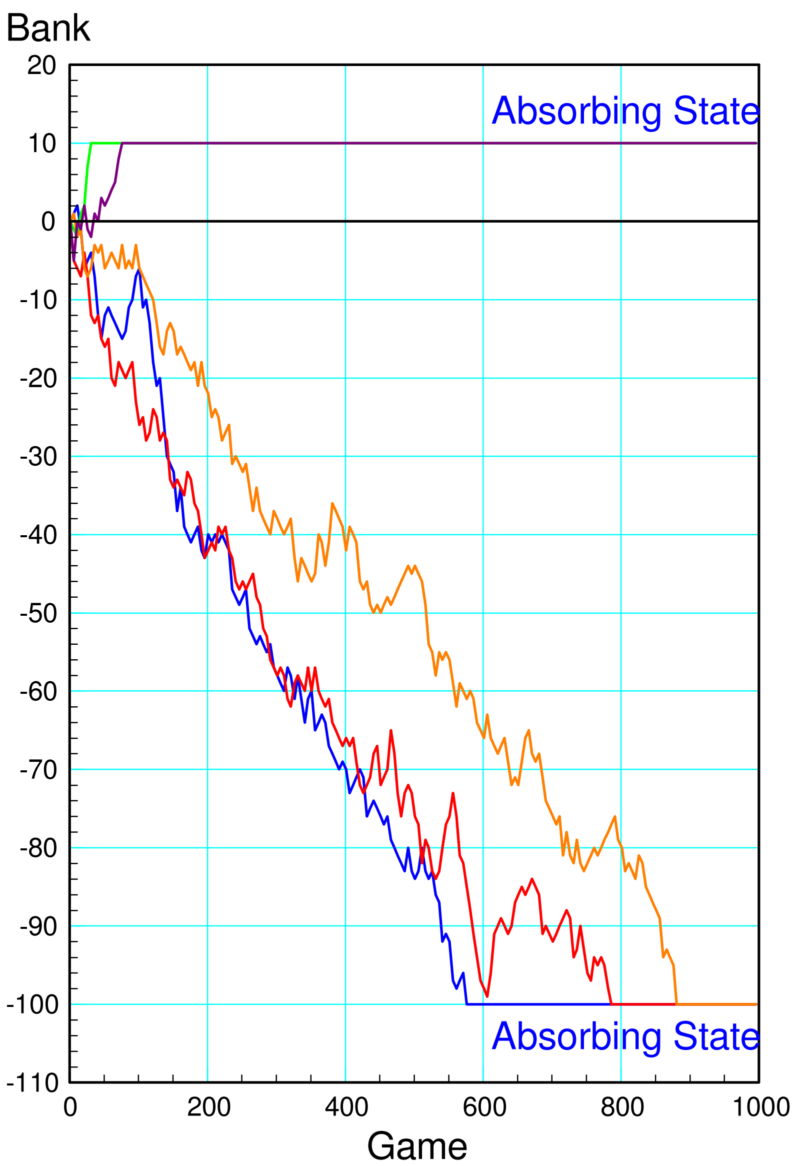
- Quit when up $10
- Quit when down $100
  - The house no longer accepts your money.

Monte-Carlo Simulation: In 10,000 games

- 6,647 times you're up $10
- 3,353 times you're down $100

If you're losing, walk away.

- It could be you're losing because you're just not that good
- Your odds of winning are actually 45%, not 55%



z-Transforms & Markov Chains

Problem: Determine the probability of player A winning after k games.

- New topic
- This is where z-transforms shine.

Solution: Express the system in state-space form:

$$X(k+1) = AX(k) + BU(k)$$

$$Y = CX(k)$$

If U(k) is an impulse function, you get the impulse response:

$$zX = AX + B$$

$$Y = CX$$

This also works for Markov chains where

- B is the initial condition: X(0)
- C tells you which state you want to look at.

For example, for the problem of winning by 2 games,

```
A =

    1.0000    0.7000         0         0         0
         0         0    0.7000         0         0
         0    0.3000         0    0.7000         0
         0         0    0.3000         0         0
         0         0         0    0.3000    1.0000

X0 = [0;0;1;0;0]

     0
     0
     1
     0
     0

C = [1,0,0,0,0]     % A wins

C =     1     0     0     0     0
```

You can now find the Y(z) (or the impulse response)

- Multiply by z (as per last lecture)

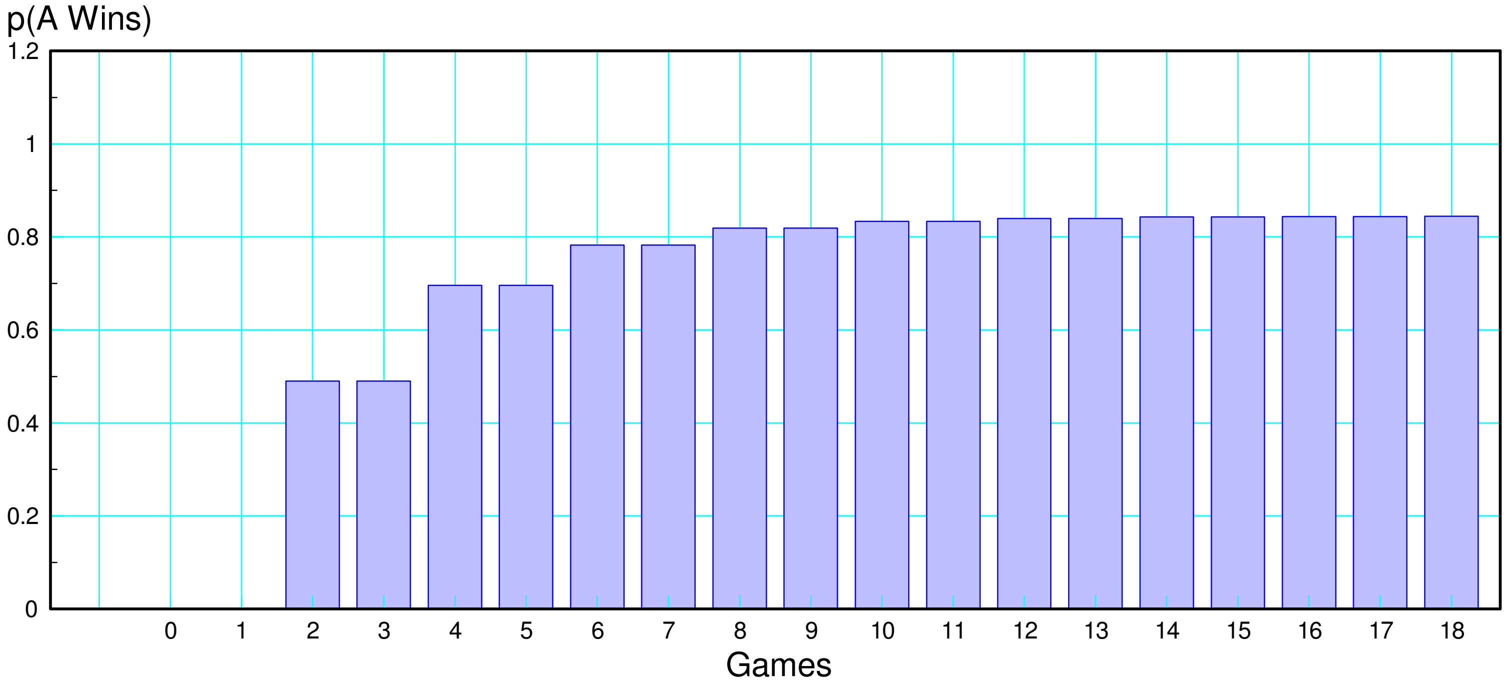
```
G  = ss(A,X0,C,0,1);

tf(G)

                  0.49 z
------------------------------------------
z^4 - z^3 - 0.42 z^2 + 0.42 z + 1.821e-018

sampling time (seconds): 1

zpk(G)

           0.49 z
-----------------------------
z (z-1) (z-0.6481) (z+0.6481)

Sampling time (seconds): 1
```

This tells you that

$$Y(z) = \left(\frac{0.49z}{(z-1)(z-0.6481)(z+0.6481)} \right)$$

The time response is from the *impulse* function

```
y = impulse(G)
```



You can also find the explicit function for y(k) using z-transfoms.

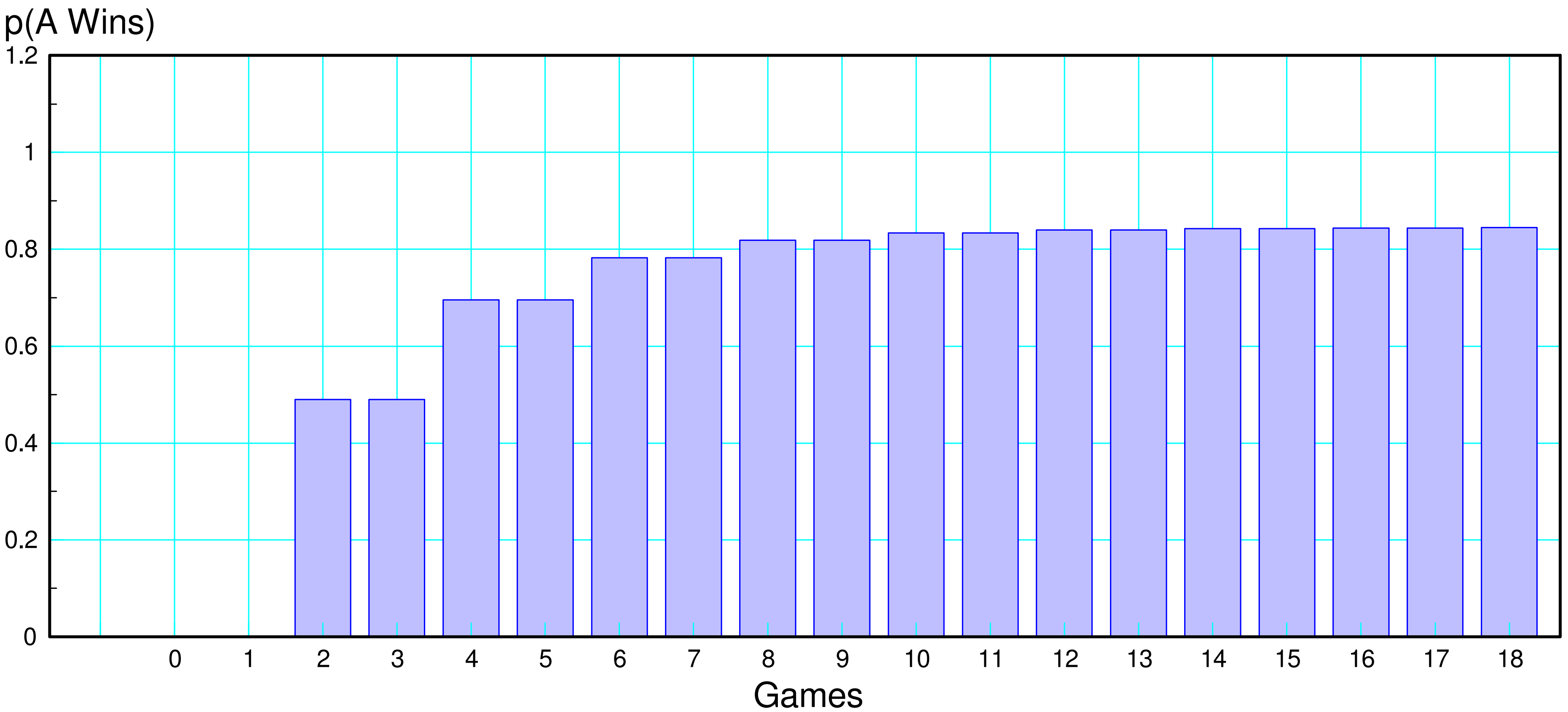
$$Y(z) = \left(\frac{0.49z}{(z-1)(z-0.6481)(z+0.6481)}\right)$$

Pull ou a z and do partial fractions

$$Y = \left(\left(\frac{0.8449}{z-1}\right) + \left(\frac{-1.0742}{z-0.6481}\right) + \left(\frac{0.2294}{z+0.6481}\right)\right)z$$

Take the inverse z-transform

$$y(k) = \left(0.8449 - 1.0742\ (0.6481)^k + 0.2294\ (-0.6481)^k\right)\ u(k)$$


p(A Wins)
1.2
1
0.8
0.6
0.4
0.2
0
0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18
Games

Summary

An absorbing state is state you never leave once you enter that state

- Example: Match is over once you're up 2 games

If there is only one absorbing state and it's reachable, you always end up at the absorbing state

- z-Transforms are helpful in computing the path to that state

If there are two absorbing states, you can find the probability of ending at each absorbing state by

- Matrix multiplication,
- Monte-Carlo simulations, or
- z-Transforms