

Propiedades de la poda alfa-beta

- No se sacrifica la optimalidad.
- Una correcta ordenación de los hijos puede mejorar la cantidad de ramas podadas.
- Con una ordenación perfecta se explora el doble de profundidad que sin poda.
- A pesar de todo... el ajedrez se resiste.

Pregunta

Si a cada uno de vosotros os pregunto “qué forma de evaluar el ejercicio prefieres?”:

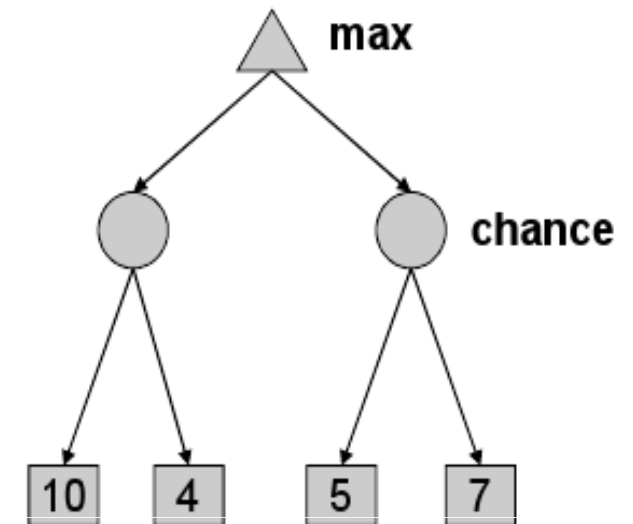
- a) Elegir por sorteo 12 ejercicios y a esos les pongo un 10 de nota y al resto 5.
- b) Poner a esos 12 un 8 y al resto un 7.
- c) Elegir los 12 mejores ejercicios, ponerles un 10 y al resto un 5
- d) Poner a esos 12 un 8 y al resto un 7.

Búsqueda expectimax

- ¿Qué pasa si no conocemos cuál será el resultado de una acción?
 - Pedir una carta en el solitario.
 - Probar una mina en el buscaminas.
 - Pacman contra fantasmas aleatorios.
- Podemos usar búsqueda expectimax:
 - Los nodos del "enemigo" en lugar de calcular el mínimo calcula la esperanza (el valor medio).
- Más adelante → Markov Decision Processes.

Utilidad máxima esperada

- ¿Por qué no usamos minimax?
- Principio de máxima utilidad esperada: Un agente debe seleccionar la acción que **maximice su utilidad esperada dado su conocimiento.**
- Principio general para la toma de decisiones.
- Definición de racionalidad.
- Aparecerá más a lo largo del curso.



Probabilidades

- Una **variable aleatoria** representa un hecho cuyo resultado desconocemos.
- Una **distribución de probabilidad** es una asignación de pesos a los diferentes eventos del hecho.
- Ejemplo: Tráfico en la autopista
 - Variable aleatoria: T = cuánto tráfico hay.
 - Eventos: ligero, normal, denso.
 - Distribución: $P(\{T=\text{ligero}\}) = 0.4$ $P(\{T=\text{normal}\})=0.5$
 $P(\{T=\text{denso}\})=0.1$

Probabilidades

- Las probabilidades son siempre no negativas.
- Las probabilidades sobre todos los eventos posibles suman 1.
- A medida que tenemos más información, las probabilidades cambian.
 - $P(\{T=\text{denso}\})=0.1$
 - $P(\{T=\text{denso}\}|\{hora=8am\})=0.4$
- Más adelante veremos métodos para razonar y actualizar las probabilidades.

¿Qué son las probabilidades?

- Visión frecuentista:
 - Medias sobre experimentos repetidos.
 - Se estiman a partir de observaciones históricas.
 - Nos permiten saber cómo funcionarán los experimentos futuros (a largo plazo).
 - Nos hace pensar en hechos *inherentemente aleatorios* como lanzar dados.
- Visión Bayesiana:
 - Grados de creencia sobre variables no observadas (en base al conocimiento).
 - La creencia de un agente en que está lloviendo, dada la temperatura.
 - La creencia de pacman de que un fantasma va a girar, dado el estado.
 - Se pueden *aprender* a partir de la experiencia, ya que las nuevas experiencias modifican nuestras creencias.

Incertidumbre por todos lados

- No sólo en juegos de azar:
 - Estoy resfriado. ¿Estornudaré en el próximo minuto?
 - Este correo contiene "Viagra". ¿Es spam?
 - Me duele un diente. ¿Tiene caries?
 - ¿Llegaré en 30 min al aeropuerto?
 - El robot ha hecho girar la rueda tres vueltas. ¿Cuán lejos se ha desplazado?
 - ¿Es seguro cruzar la calle ahora?
- Fuentes de incertidumbre:
 - Procesos aleatorios: Lanzamiento de dados...
 - Información insuficiente.
 - Ignorancia de los procesos subyacentes.
 - Variables que no se incluyen en el modelo.
 - El mundo es ruidoso.

Recordatorio: Esperanzas

- Podemos definir una función $f(X)$ de una variable aleatoria X .
- El valor esperado de una función es su valor medio, ponderando cada valor de su entrada por la distribución de probabilidad.
- Ejemplo: ¿Cuánto tarda en llegar al aeropuerto?
 - Es función del tráfico:
 - $f(\text{ligero})=10$, $f(\text{normal})=15$, $f(\text{denso})=40$
 - $P(\{T=\text{ligero}\}) = 0.4$ $P(\{T=\text{normal}\})=0.5$ $P(\{T=\text{denso}\})=0.1$
 - ¿Cuál es el tiempo esperado? $E[f(T)]$

Esperanzas

- Esperanza de una función de una variable aleatoria:

$$E_{P(X)}[f(X)] = \sum_x f(x)P(x)$$

- Ejemplo: ¿Cuál es el valor medio de un dado?

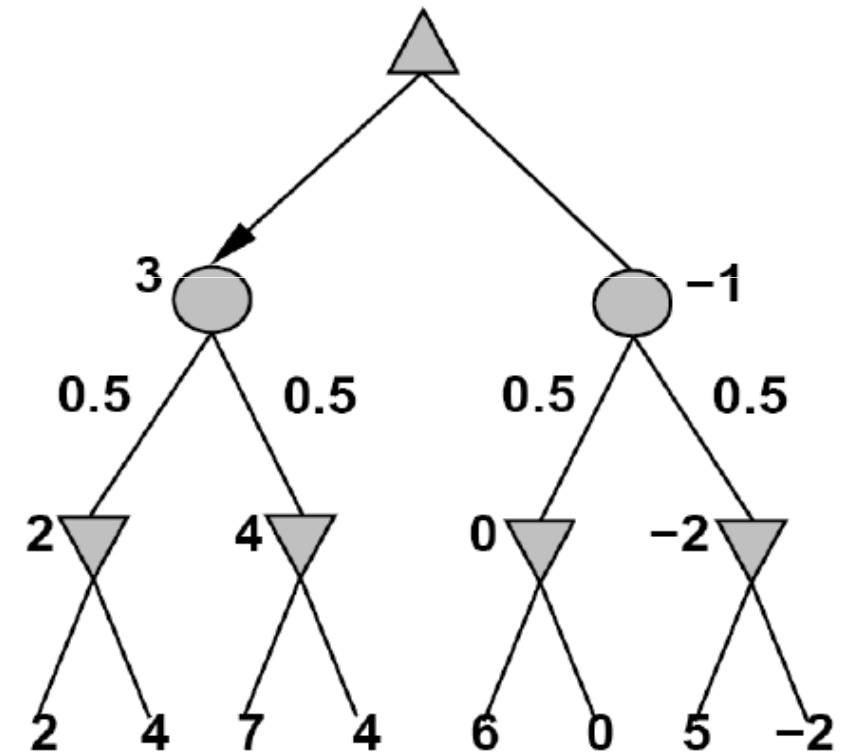
X	P	f
1		1
2		2
3		3
4		4
5		5
6		6

Utilidades

- Las utilidades son funciones que van de los resultados (estados del mundo) a números reales que representan las preferencias de un agente.
- ¿De dónde salen las utilidades?
 - En un juego, sencillo: ganar= +1, perder =-1
 - Las utilidades resumen los objetivos de un agente.
 - Teorema: Cualquier conjunto de preferencias coherente se puede representar mediante una función de utilidad.

Expectiminimax

- Expectiminimax:
 - El entorno es un jugador especial que juega después de cada jugador.
 - Los nodos aleatorios usan la esperanza y el resto como minimax.



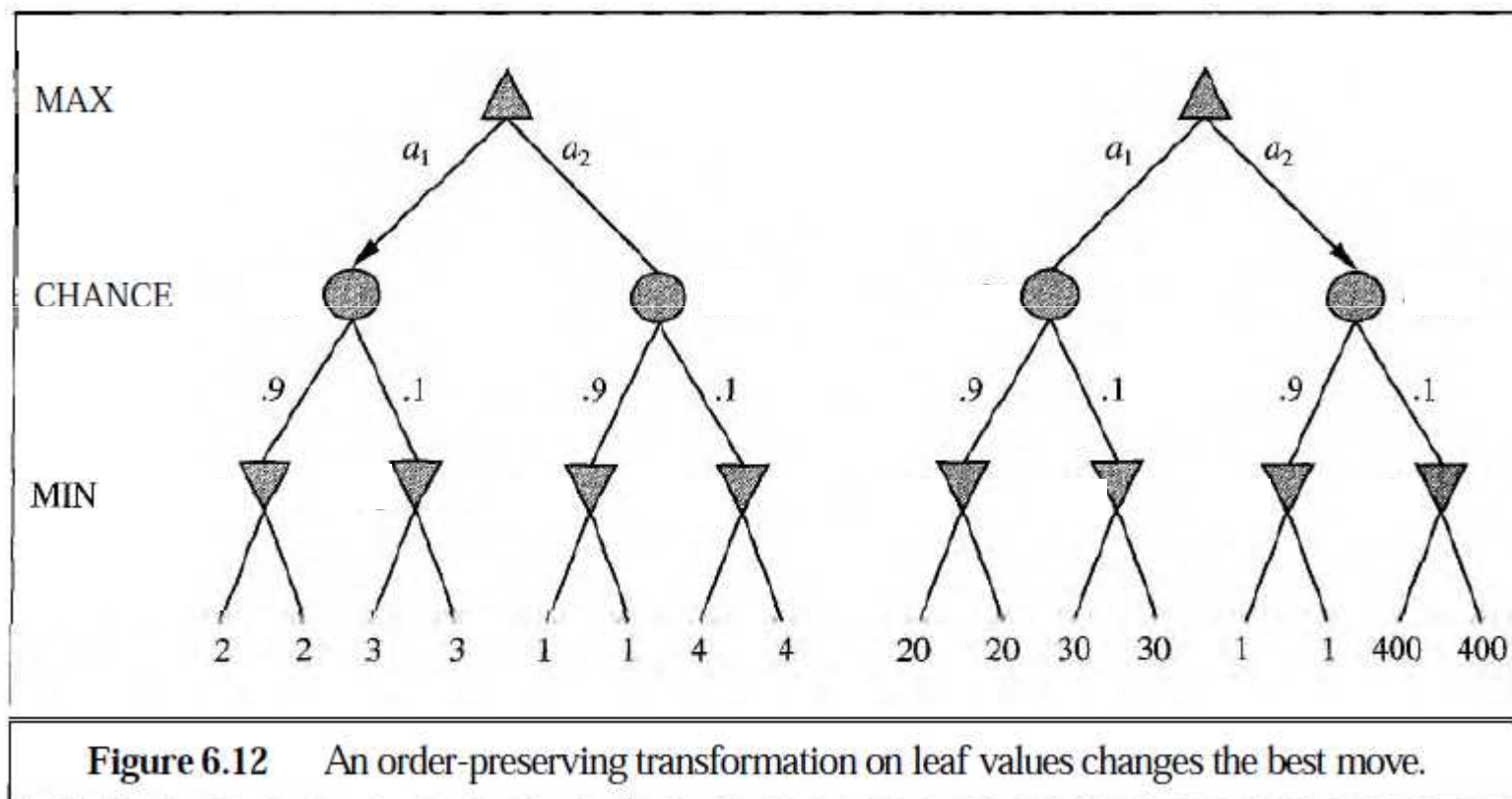
EXPECTIMINIMAX(n) =

$$\begin{cases} \text{UTILITY}(n) & \text{if } n \text{ is a terminal state} \\ \max_{s \in \text{Successors}(n)} \text{EXPECTIMINIMAX}(s) & \text{if } n \text{ is a MAX node} \\ \min_{s \in \text{Successors}(n)} \text{EXPECTIMINIMAX}(s) & \text{if } n \text{ is a MIN node} \\ \sum_{s \in \text{Successors}(n)} P(s) \cdot \text{EXPECTIMINIMAX}(s) & \text{if } n \text{ is a chance node} \end{cases}$$

- Ej: Backgammon

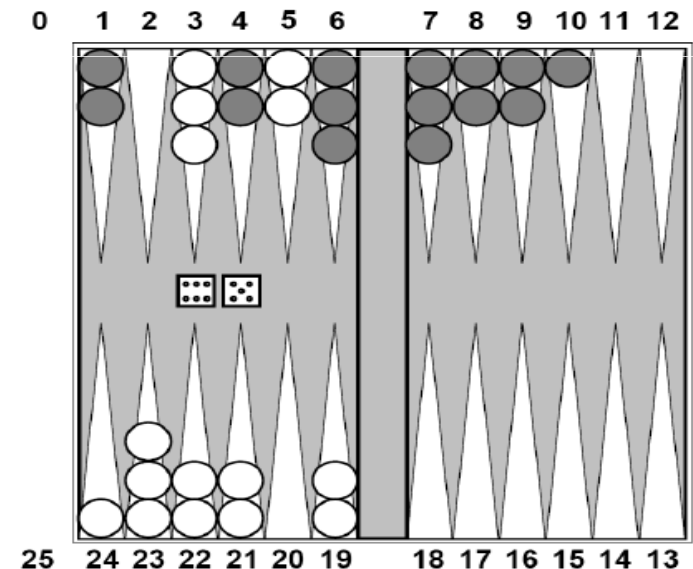
Funciones de evaluación en expectiminimax

- En minimax el valor de las funciones de evaluación no importa, tan sólo es importante el orden relativo (si un estado es mejor que otro o no)
- Para expectiminimax necesitamos además que las magnitudes de los valores sean correctas.



Backgammon

- TD-Gammon:
 - <http://en.wikipedia.org/wiki/TD-Gammon>
 - Búsqueda a 2 niveles + buena función de evaluación: Temporal Difference learning (red neuronal)
 - Al nivel del campeón del mundo.
- El lanzamiento de dados incrementa b:
 - posibilidades diferentes con 2 dados ?



Backgammon

- A medida que bajamos niveles disminuye la probabilidad de que lleguemos a esa configuración.

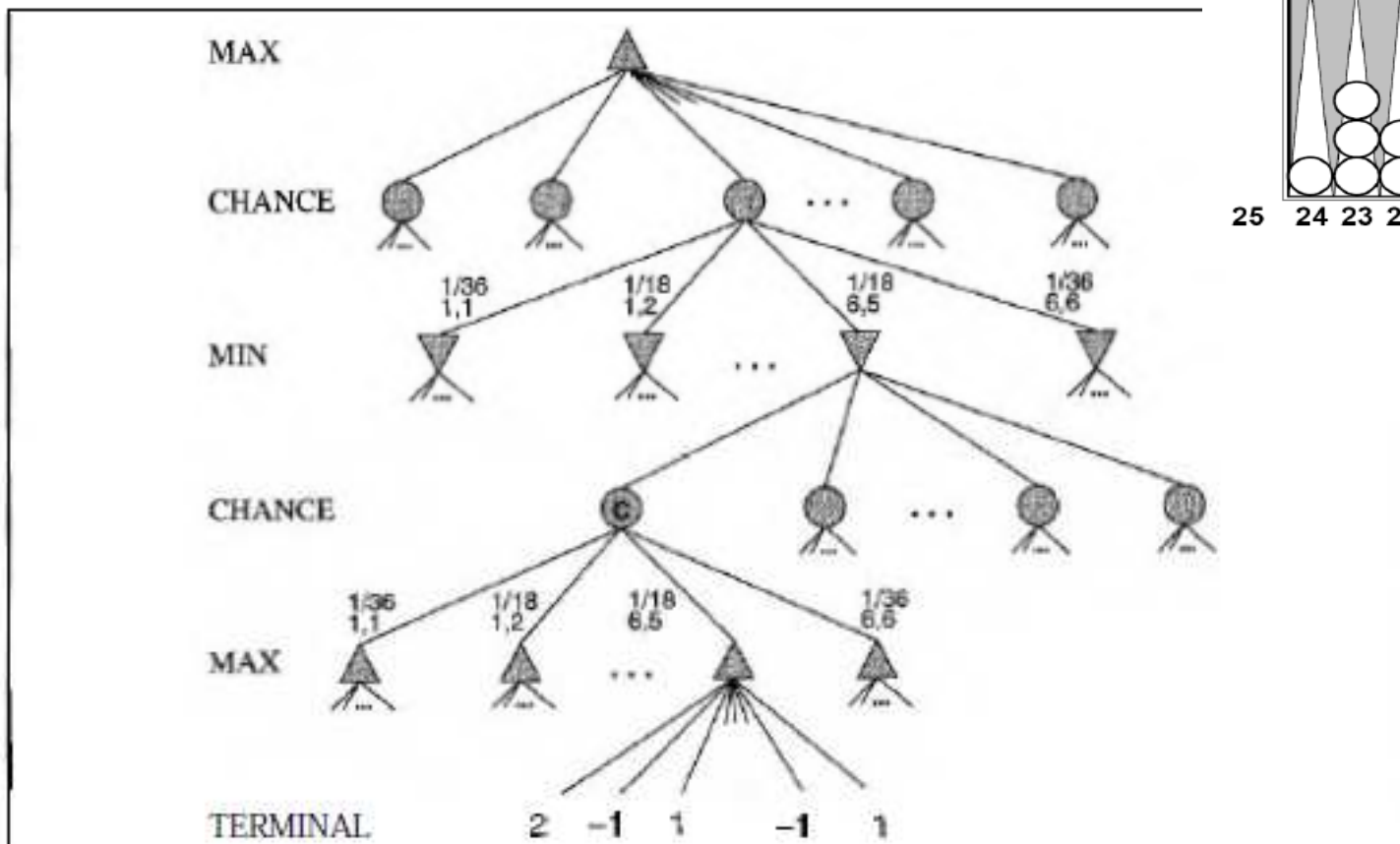
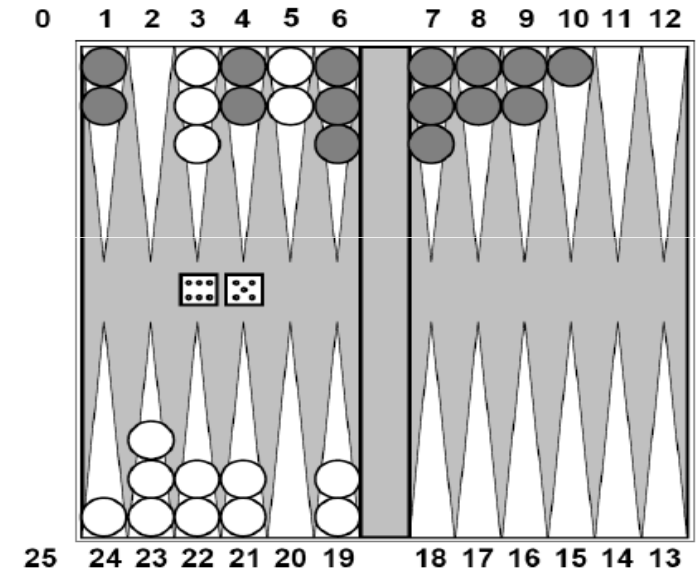
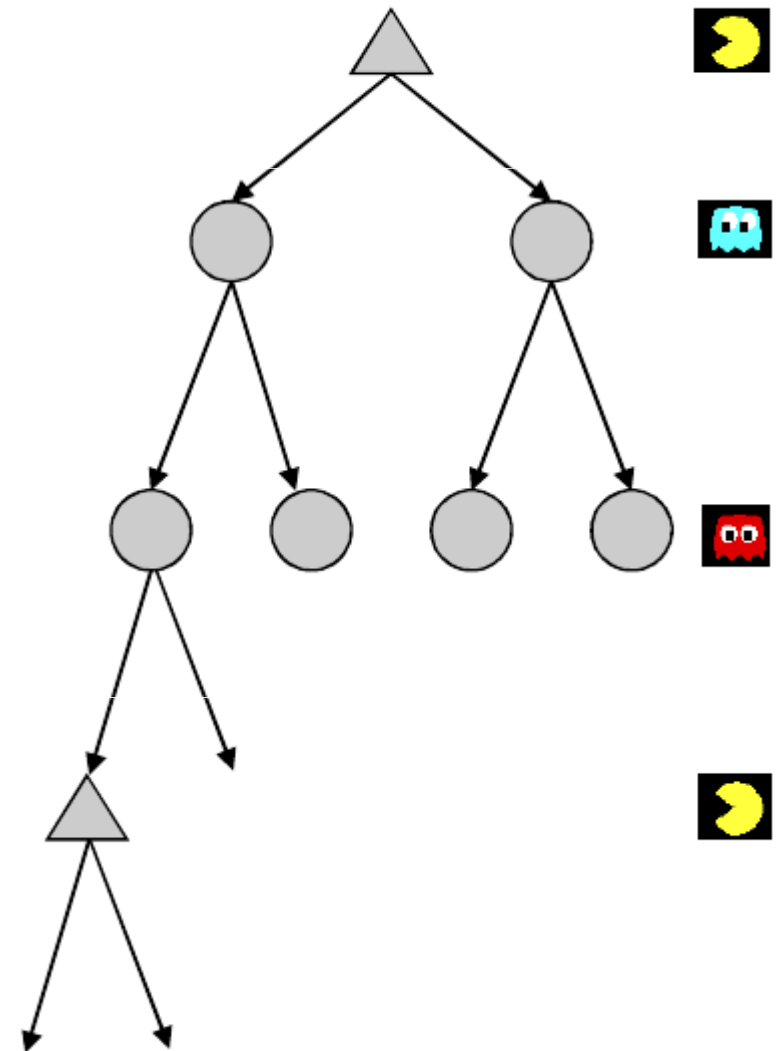


Figure 6.11 Schematic game tree for a backgammon position.

Búsqueda expectimax

- Tenemos un modelo probabilístico de como se comportan los rivales:
 - Sencillo: Lanzar un dado.
 - Complejo.
 - Un nodo representa cada hecho fuera de nuestro control (enemigo o naturaleza).
 - El modelo podría decir que el agente se comporta como un adversario ideal.

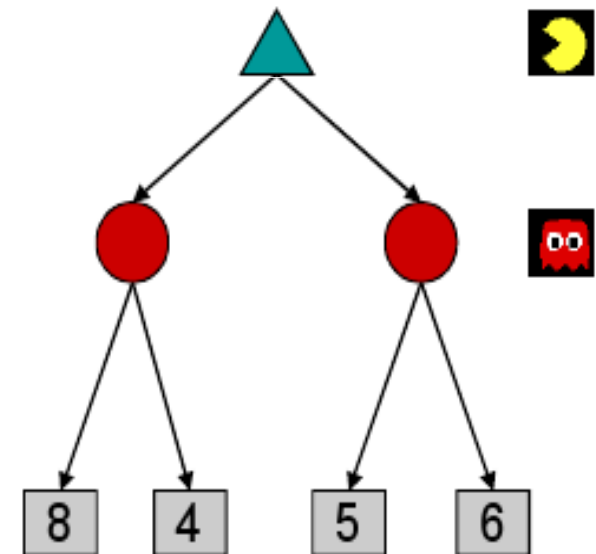


Pseudocódigo para expectimax

```
def value(s)
  if s is a max node return max(s)
  if s is an exp node return expValue(s)
  if s is a terminal node return evaluation(s)
```

```
def max(s)
  values = [value(s') for s' in successors(s)]
  return max(values)
```

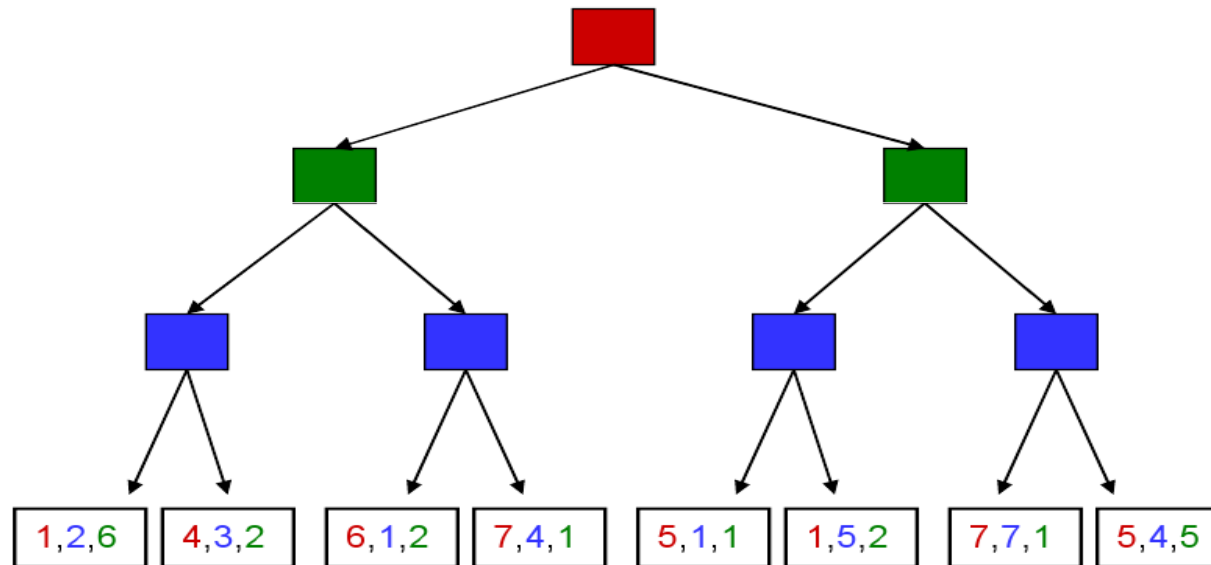
```
def expValue(s)
  values = [value(s') for s' in successors(s)]
  weights = [probability(s, s') for s' in successors(s)]
  return expectation(values, weights)
```



Expectimax para Pacman

- Nos hemos alejado de la idea de que los fantasmas tratan de ganarnos el juego. Ahora los consideramos "parte del entorno".
- Pacman tiene una distribución de creencias sobre como se comportarán.
- ¿Podemos ver minimax como un caso especial de expectimax?
- ¿Cómo se comportaría Pacman si asumiera que los fantasmas realizan minimax a un nivel el 80% del tiempo y juegan aleatoriamente el resto del tiempo?
- Podemos modelar al rival pensando que él tiene un modelo de nosotros...

Juegos de suma no cero o multijugador



- Similar al minimax:
 - Las utilidades son ahora tuplas.
 - Cada jugador **maximiza** su propia entrada y propaga el resultado al siguiente nivel.
 - Diplomacy game