Analizando la heurística
“**piezas descolocadas**”

A

8	1	3
7	2	5
4		6

1	2	3
8		4
7	6	5

GOAL

La solución óptima para esta configuración son 9 pasos

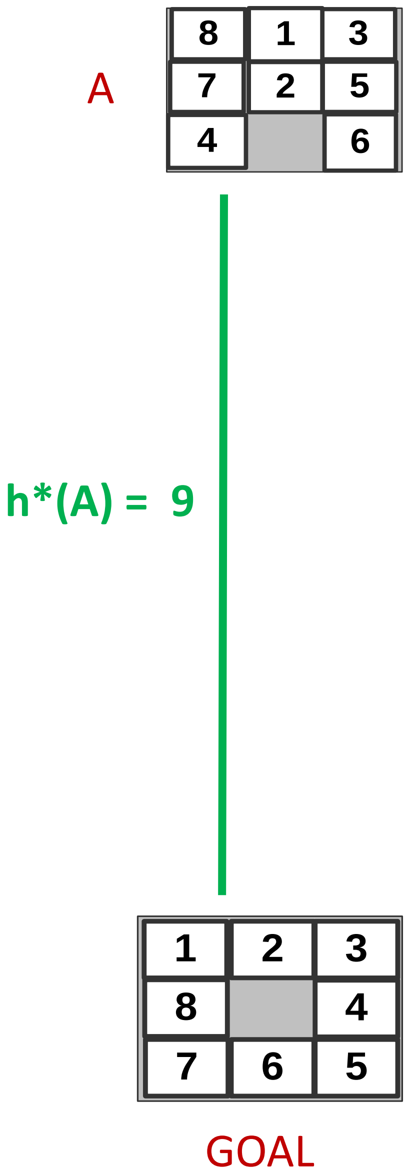
A

8	1	3
7	2	5
4		6

h*(A) = 9

1	2	3
8		4
7	6	5

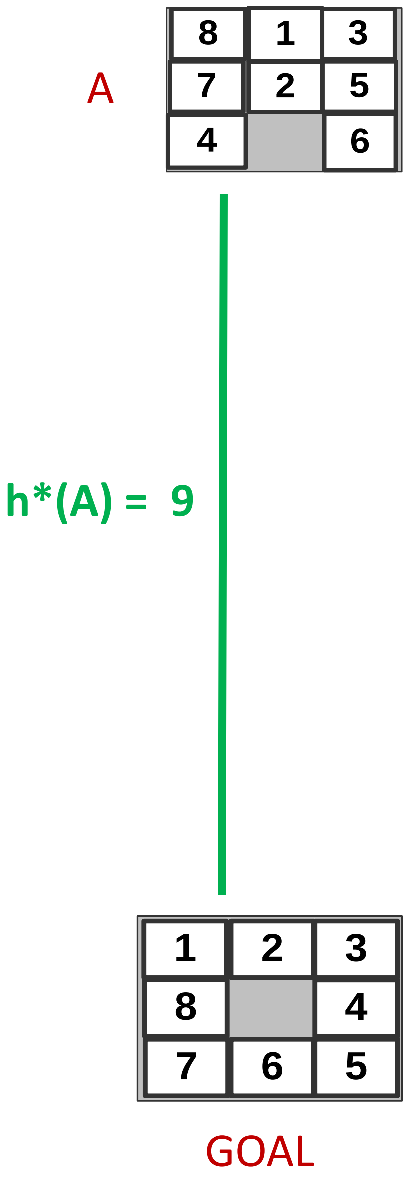
GOAL



La solución óptima para esta configuración son 9 pasos

es decir

El coste real del nodo **A** al **GOAL** es 9 ➔ **h*(A)=9**

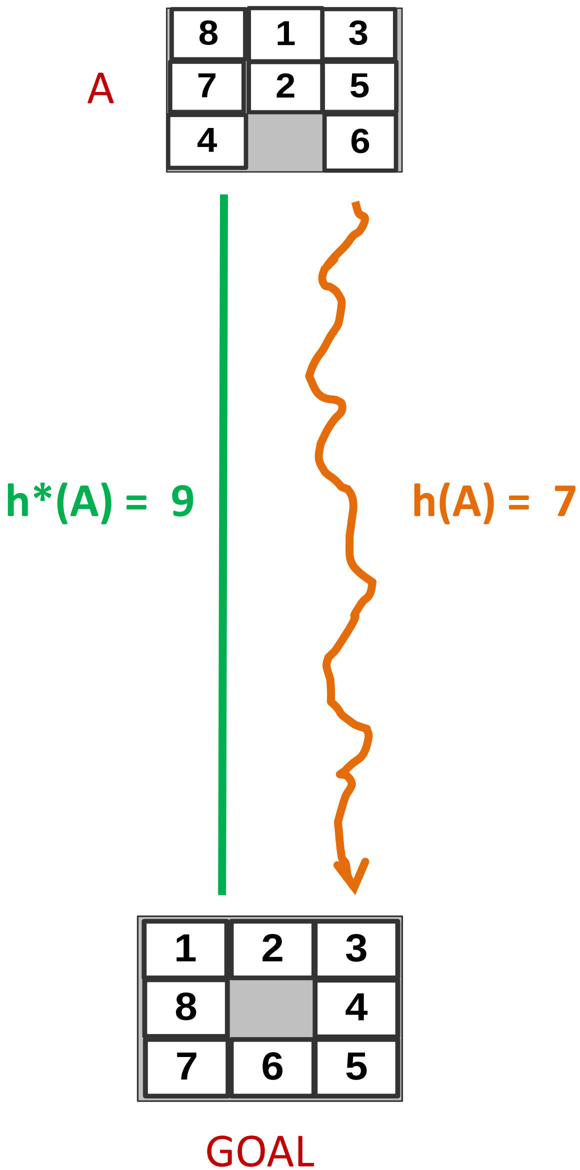


La solución óptima para esta configuración son 9 pasos

es decir

El coste real del nodo **A** al **GOAL** es 9 ➔ **h*(A)=9**

(Sé el coste de la solución porque he ejecutado el programa del puzzle)



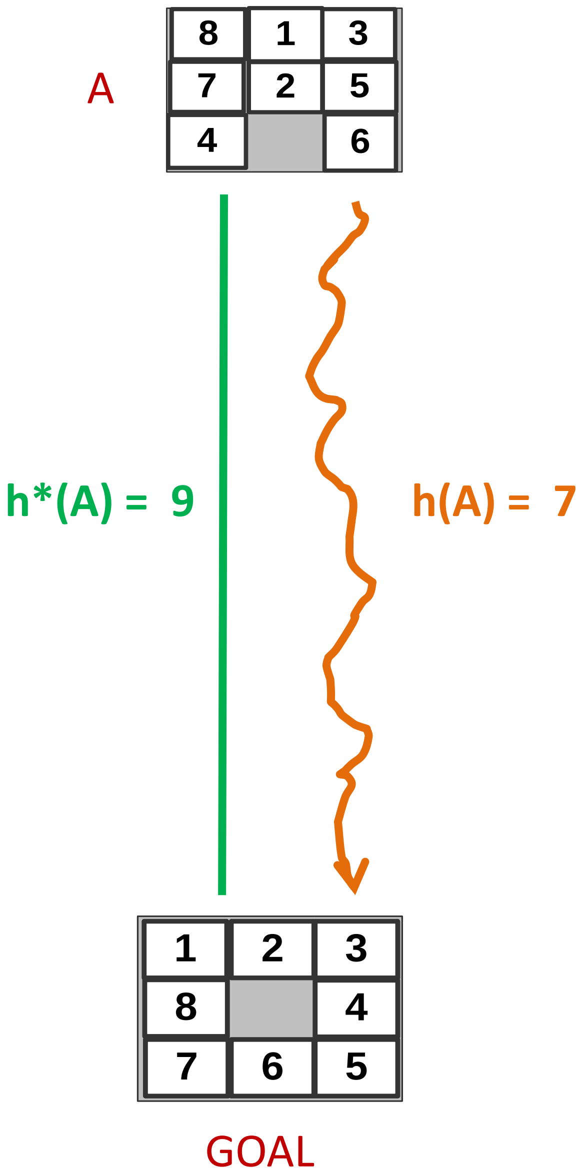
La solución óptima para esta configuración son 9 pasos

es decir

El coste real del nodo **A** al **GOAL** es 9 ➔ **h*(A)=9**

(Sé el coste de la solución porque he ejecutado el programa del puzzle)

Y la heurística "piezas descolocadas" estima un coste de 7 porque todas las fichas excepto la ficha 3 están descolocadas



La solución óptima para esta configuración son 9 pasos

es decir

El coste real del nodo **A** al **GOAL** es 9 ➔ **h*(A)=9**

(Sé el coste de la solución porque he ejecutado el programa del puzzle)

Y la heurística “piezas descolocadas” estima un coste de 7 porque todas las fichas excepto la ficha 3 están descolocadas

h*(A) = 9 ⇒ Coste real de **A** al **GOAL**

h(A) = 7 ⇒ Estimación de mi heurística de **A** al **GOAL**

Otro ejemplo ….

A

	1	4
7	2	6
3	8	5

1	2	3
8		4
7	6	5

GOAL

Otro ejemplo ....

A

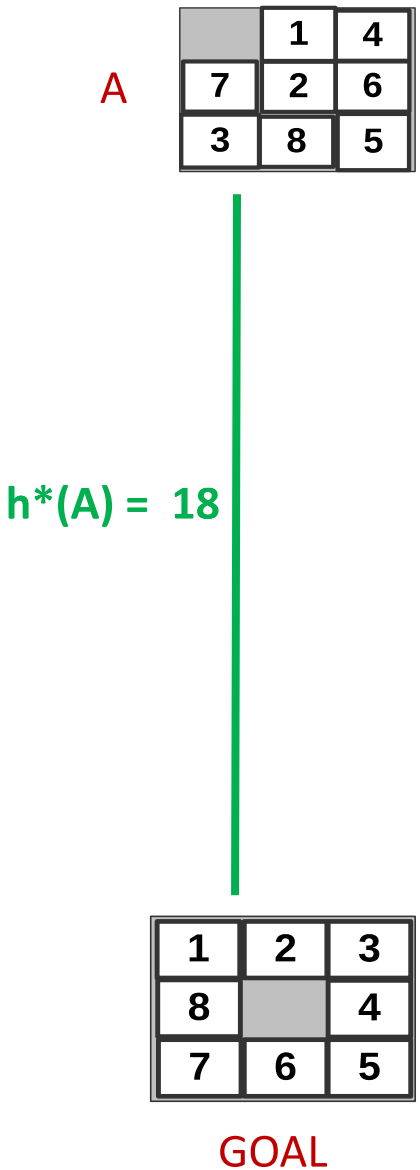
	1	4
7	2	6
3	8	5

h*(A) = 18

1	2	3
8		4
7	6	5

GOAL

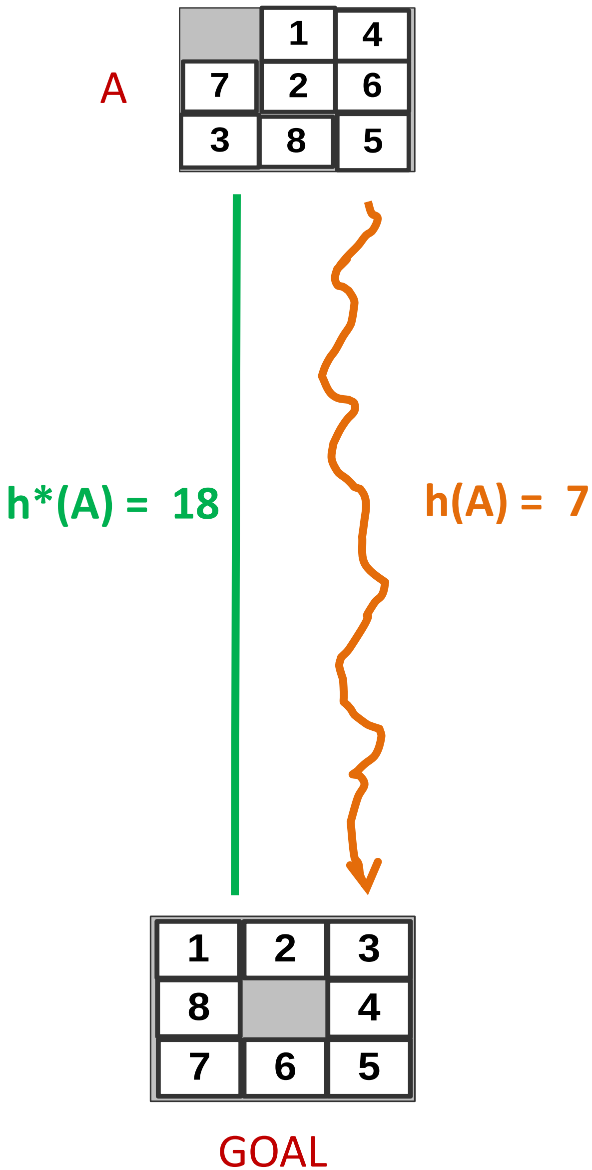
Otro ejemplo ….



h*(A) = 18 ⟹ Coste real de **A** al **GOAL**

Coste de la solución óptima

Otro ejemplo ….

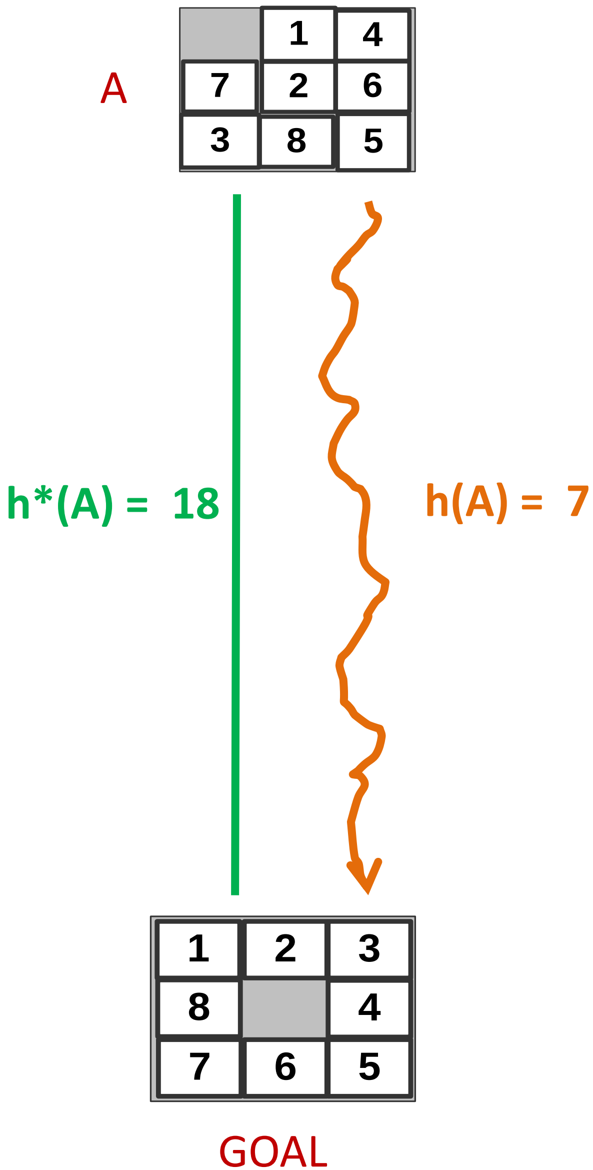


h*(A) = 18 ⟹ Coste real de **A** al **GOAL**

Coste de la solución óptima

(todas las fichas excepto la 5 están mal colocadas)

Otro ejemplo ....



(todas las fichas excepto la 5 están mal colocadas)

h*(A) = 18 ⇒ Coste real de **A** al **GOAL**

Coste de la solución óptima

h(A) = 7 ⇒ Coste estimado de **A** al **GOAL**
con la heurística DESCOLOCADAS

En resumen

En resumen

2	8	3
1		4
7	6	5

$h^*(A) = 4$ $h(A) = 3$

En resumen

2	8	3
1		4
7	6	5

$h^*(A) = 4$ $h(A) = 3$

8	1	3
7	2	5
4		6

$h^*(A) = 9$ $h(A) = 7$

En resumen

2	8	3
1		4
7	6	5

$h^*(A) = 4$ $h(A) = 3$

8	1	3
7	2	5
4		6

$h^*(A) = 9$ $h(A) = 7$

	1	4
7	2	6
3	8	5

$h^*(A) = 18$ $h(A) = 7$

En resumen

2	8	3
1		4
7	6	5

h*(A) = 4

h(A) = 3

La heurística 'Descolocadas' siempre SUBESTIMA el coste real

8	1	3
7	2	5
4		6

h*(A) = 9

h(A) = 7

	1	4
7	2	6
3	8	5

h*(A) = 18

h(A) = 7

En resumen

2	8	3
1		4
7	6	5

h*(A) = 4 h(A) = 3

La heurística 'Descolocadas' siempre SUBESTIMA el coste real

8	1	3
7	2	5
4		6

h*(A) = 9 h(A) = 7

Sí, porque solo estamos contando las fichas mal colocadas por tanto solo estamos estimando 1 movimiento por cada ficha mal colocada

	1	4
7	2	6
3	8	5

h*(A) = 18 h(A) = 7

En resumen

2	8	3
1		4
7	6	5

$h^*(A) = 4$ $h(A) = 3$

La heurística 'Descolocadas' siempre SUBESTIMA el coste real

8	1	3
7	2	5
4		6

$h^*(A) = 9$ $h(A) = 7$

Sí, porque solo estamos contando las fichas mal colocadas por tanto solo estamos estimando 1 movimiento por cada ficha mal colocada

	1	4
7	2	6
3	8	5

$h^*(A) = 18$ $h(A) = 7$

Y para colocar una ficha en su lugar correcto hacen falta más movimientos como poner la casilla vacía adyacente a la ficha

En resumen

h(n) = número de piezas descolocadas en el nodo ‘n’

Se cumple siempre $\forall n$ $h(n) <= h^*(n)$

h(n) es una heurística admisible

En resumen

h(n) = número de piezas descolocadas en el nodo ‘n’

Se cumple siempre $\forall n \;\; h(n) <= h^*(n)$

h(n) es una heurística admisible

$f(n) = g(n)+h(n)$ $\Longrightarrow$ **Algoritmo de tipo A**

En resumen

h(n) = número de piezas descolocadas en el nodo ‘n’

Se cumple siempre $\forall n$ $h(n) <= h^*(n)$

h(n) es una heurística admisible

$f(n) = g(n)+h(n)$ ⇒ Algoritmo de tipo A

si h(n) es admisible ⇒ Algoritmo A*

En resumen

h(n) = número de piezas descolocadas en el nodo ‘n’

Se cumple siempre $\forall n \;\; h(n) <= h^*(n)$

h(n) es una heurística admisible

---

f(n) = g(n)+h(n) → **Algoritmo de tipo A**

si h(n) es admisible → **Algoritmo A***

garantiza la SOLUCIÓN ÓPTIMA

¿Se puede encontrar una mejor estimación (heurística) para el problema del puzzle?

Una nueva heurística para el problema del 8-puzzle

Una nueva heurística para el problema del 8-puzzle

Distancias de Manhattan: para cada ficha mal colocada, calcular la distancia en horizontal y vertical a su posición objetivo y sumar todas las distancias.

Una nueva heurística para el problema del 8-puzzle

Distancias de Manhattan: para cada ficha mal colocada, calcular la distancia en horizontal y vertical a su posición objetivo y sumar todas las distancias.

	1	4
7	2	6
3	8	5

Estado inicial del puzzle

1	2	3
8		4
7	6	5

Estado final del puzzle

Una nueva heurística para el problema del 8-puzzle

Distancias de Manhattan: para cada ficha mal colocada, calcular la distancia en horizontal y vertical a su posición objetivo y sumar todas las distancias.

	1	4
7	2	6
3	8	5

Estado inicial del puzzle

1	2	3
8		4
7	6	5

Estado final del puzzle

Ficha 1: 1
Ficha 4: 1
Ficha 6: 2
Ficha 5: 0
Ficha 8: 2
Ficha 3: 4
Ficha 7: 1
Ficha 2: 1

Una nueva heurística para el problema del 8-puzzle

Distancias de Manhattan: para cada ficha mal colocada, calcular la distancia en horizontal y vertical a su posición objetivo y sumar todas las distancias.

	1	4
7	2	6
3	8	5

Estado inicial del puzzle

1	2	3
8		4
7	6	5

Estado final del puzzle

Ficha 1: 1
Ficha 4: 1
Ficha 6: 2
Ficha 5: 0
Ficha 8: 2
Ficha 3: 4
Ficha 7: 1
Ficha 2: 1

TOTAL: 12

Comparación de funciones heurísticas para el puzzle

	1	4
7	2	6
3	8	5

$h^*(A) = 18$

Comparación de funciones heurísticas para el puzzle

Descolocadas
h(n)=W(n)

	1	4
7	2	6
3	8	5

h*(A) = 18

Comparación de funciones heurísticas para el puzzle

Descolocadas

h(n)=W(n)

W(A) = 7

↓

Heurística admisible

	1	4
7	2	6
3	8	5

h*(A) = 18

Comparación de funciones heurísticas para el puzzle

Descolocadas
h(n)=W(n)

W(A) = 7

Heurística admisible

	1	4
7	2	6
3	8	5

h*(A) = 18

Distancia Manhattan
h(n)=D(n)

Comparación de funciones heurísticas para el puzzle

Descolocadas
h(n)=W(n)

W(A) = 7

Heurística admisible

	1	4
7	2	6
3	8	5

h*(A) = 18

Distancia Manhattan
h(n)=D(n)

D(A) = 12

Heurística admisible

Comparación de funciones heurísticas para el puzzle

Descolocadas
h(n)=W(n)

W(A) = 7

Heurística admisible

	1	4
7	2	6
3	8	5

h*(A) = 18

Distancia Manhattan
h(n)=D(n)

D(A) = 12

Heurística admisible

f(n) = g(n)+W(n) $\Longrightarrow$ **solución óptima**

Comparación de funciones heurísticas para el puzzle

Descolocadas
h(n)=W(n)

W(A) = 7

Heurística admisible

	1	4
7	2	6
3	8	5

h*(A) = 18

Distancia Manhattan
h(n)=D(n)

D(A) = 12

Heurística admisible

f(n) = g(n)+W(n) → **solución óptima**

f(n) = g(n)+D(n) → **solución óptima**

Comparación de funciones heurísticas para el puzzle

Descolocadas
h(n)=W(n)

W(A) = 7

Heurística admisible

	1	4
7	2	6
3	8	5

h*(A) = 18

Distancia Manhattan
h(n)=D(n)

D(A) = 12

Heurística admisible

f(n) = g(n)+W(n) ⇒ **solución óptima**

f(n) = g(n)+D(n) ⇒ **solución óptima**

Entonces, ¿cuál es la diferencia entre usar W(n) y D(n)?

D(n) está mucho más informada que W(n)

Los valores D(n) están mucho más cerca del coste real (h^*) que W(n)

D(n) está mucho más informada que W(n)

Los valores D(n) están mucho más cerca del coste real (h*) que W(n)

$D(n) >> W(n) \longrightarrow$ D(n) domina a W(n)

D(n) está mucho más informada que W(n)

Los valores D(n) están mucho más cerca del coste real (h*) que W(n)

D(n) >> W(n) $\longrightarrow$ D(n) domina a W(n)

h*

D(n) está mucho más informada que W(n)

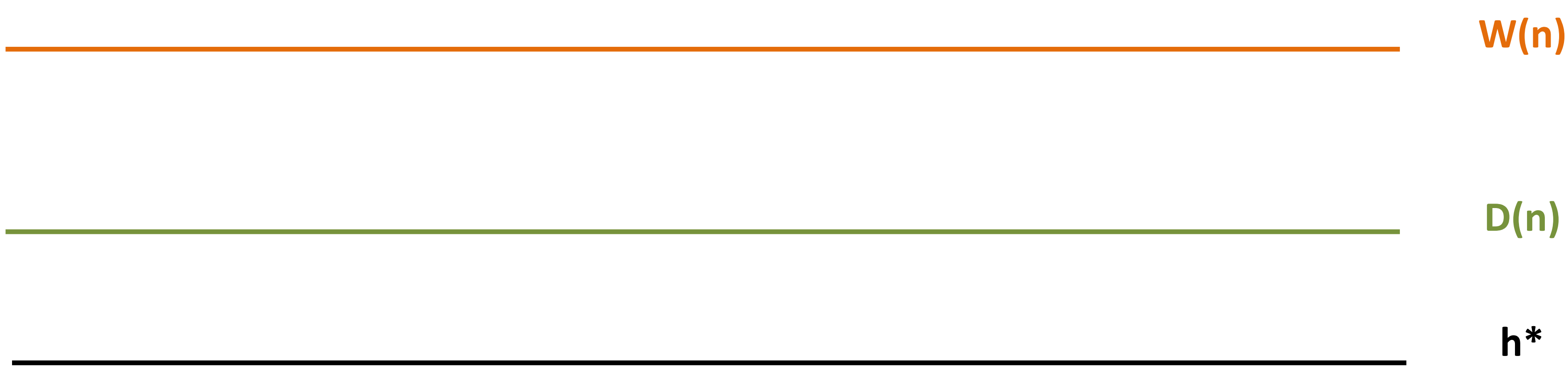
Los valores D(n) están mucho más cerca del coste real (h*) que W(n)

$D(n) >> W(n) \longrightarrow$ D(n) domina a W(n)

D(n) está mucho más informada que W(n)

Los valores D(n) están mucho más cerca del coste real (h*) que W(n)

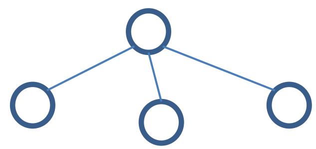
D(n) >> W(n) $\longrightarrow$ D(n) domina a W(n)



D(n) está mucho más informada que W(n)

Los valores D(n) están mucho más cerca del coste real (h*) que W(n)

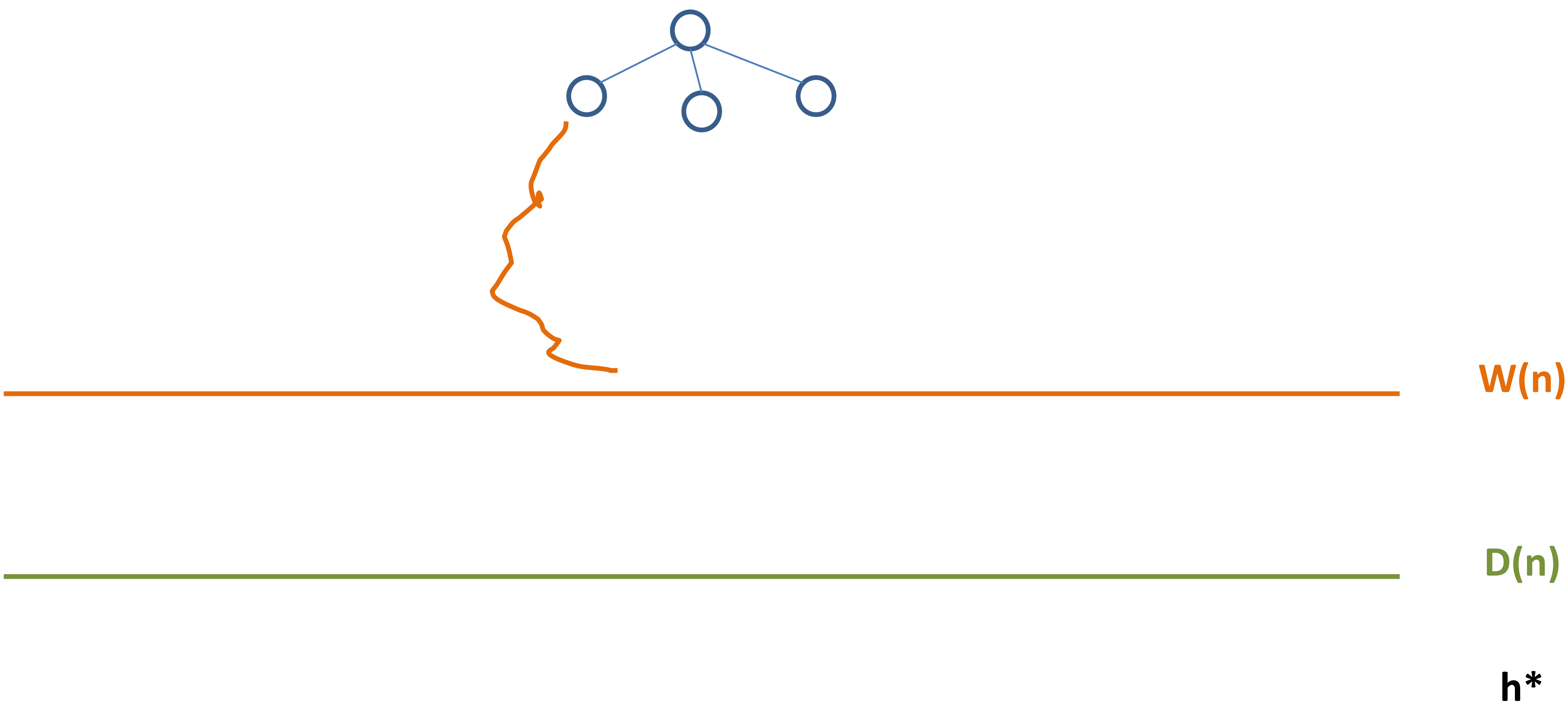
D(n) >> W(n) $\longrightarrow$ D(n) domina a W(n)



D(n) está mucho más informada que W(n)

Los valores D(n) están mucho más cerca del coste real (h*) que W(n)

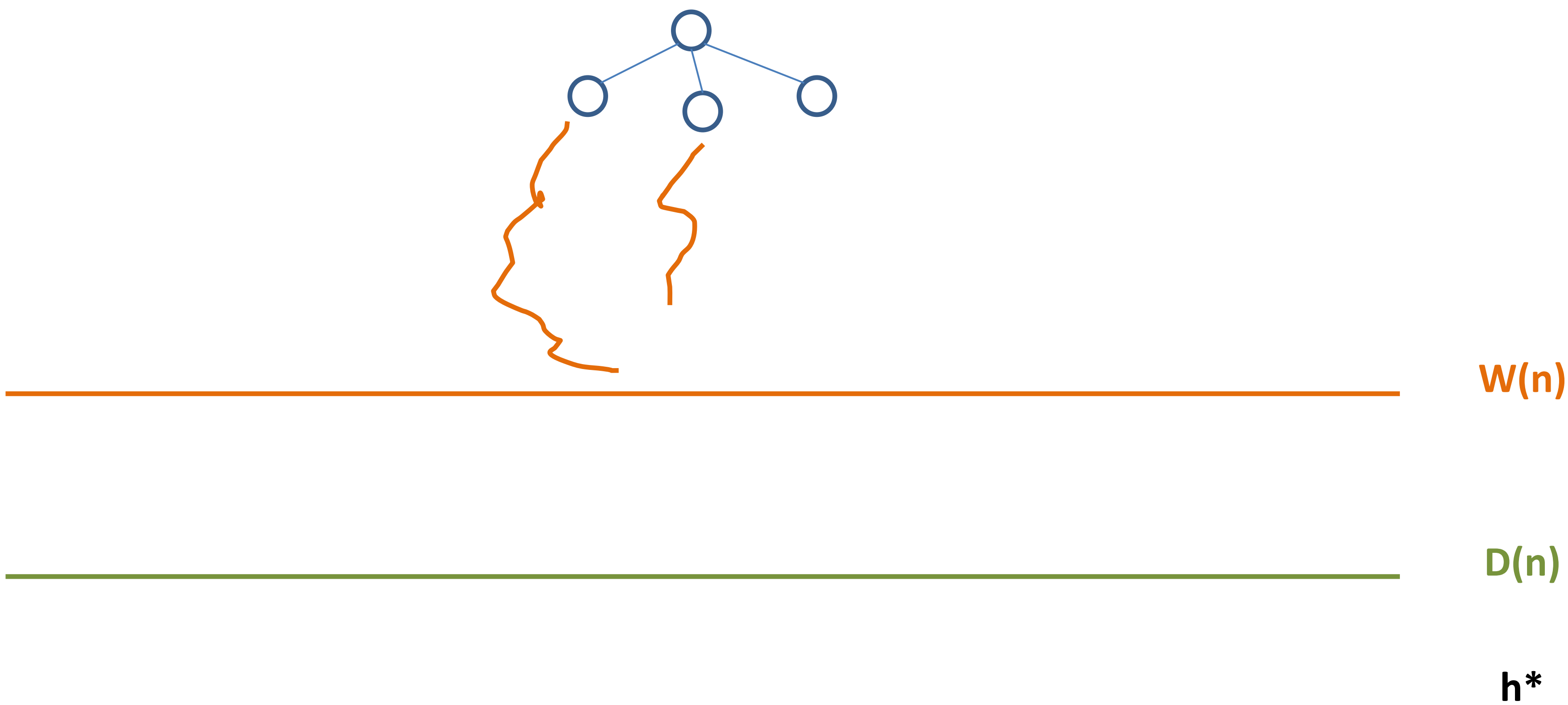
D(n) >> W(n) → D(n) domina a W(n)



D(n) está mucho más informada que W(n)

Los valores D(n) están mucho más cerca del coste real (h*) que W(n)

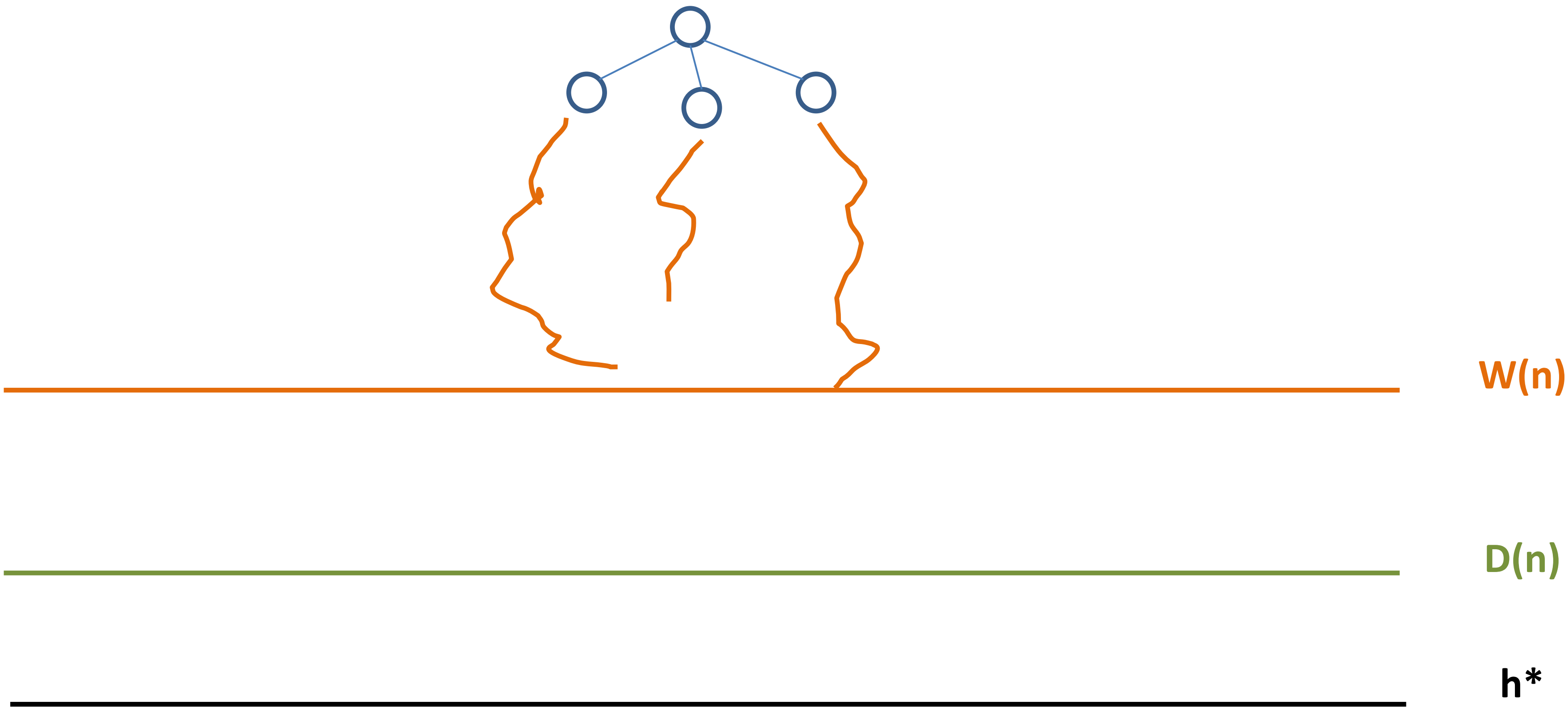
D(n) >> W(n) $\longrightarrow$ D(n) domina a W(n)



D(n) está mucho más informada que W(n)

Los valores D(n) están mucho más cerca del coste real (h*) que W(n)

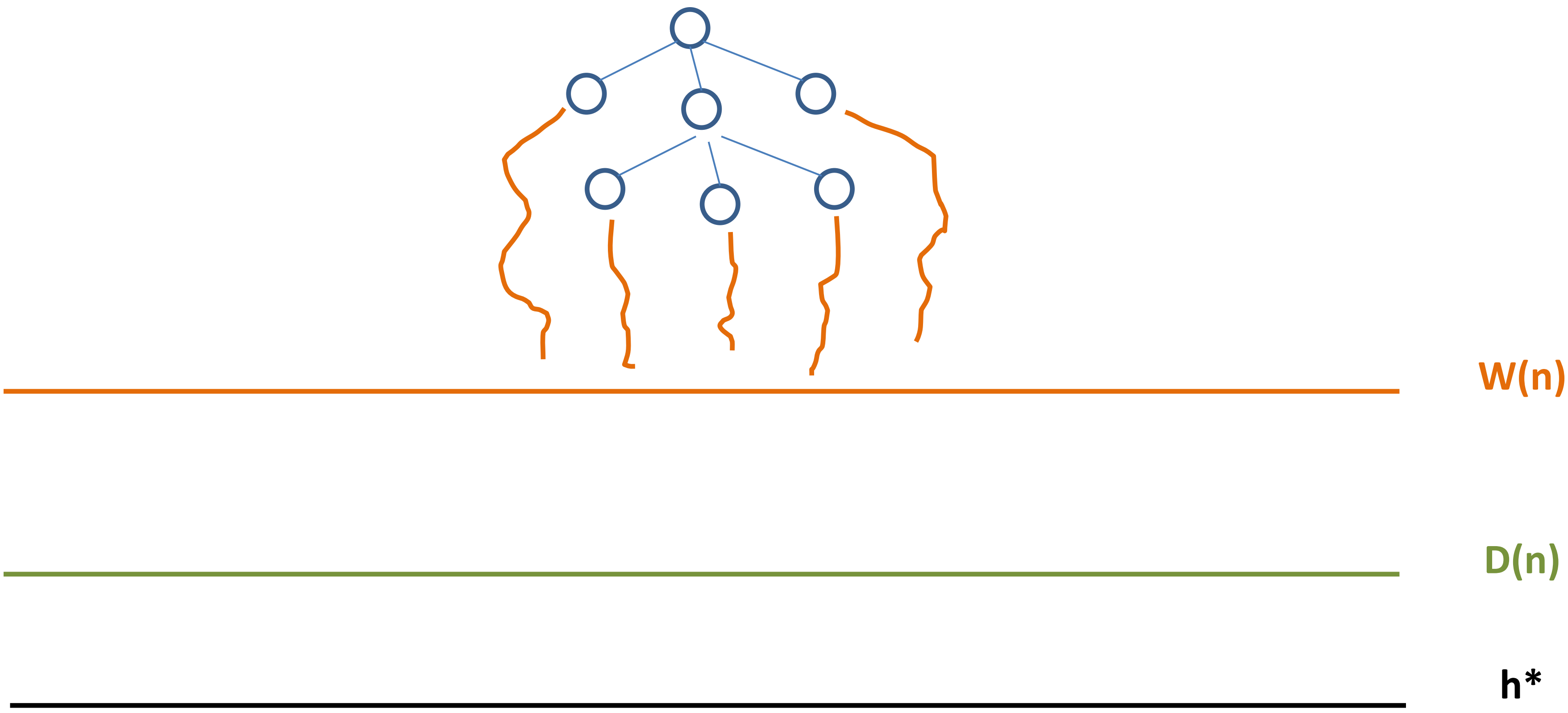
D(n) >> W(n) → D(n) domina a W(n)



D(n) está mucho más informada que W(n)

Los valores D(n) están mucho más cerca del coste real (h*) que W(n)

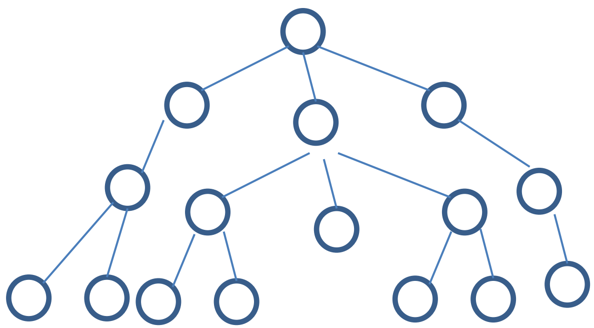
D(n) >> W(n) → D(n) domina a W(n)



D(n) está mucho más informada que W(n)

Los valores D(n) están mucho más cerca del coste real (h*) que W(n)

D(n) >> W(n) $\longrightarrow$ D(n) domina a W(n)



W(n)

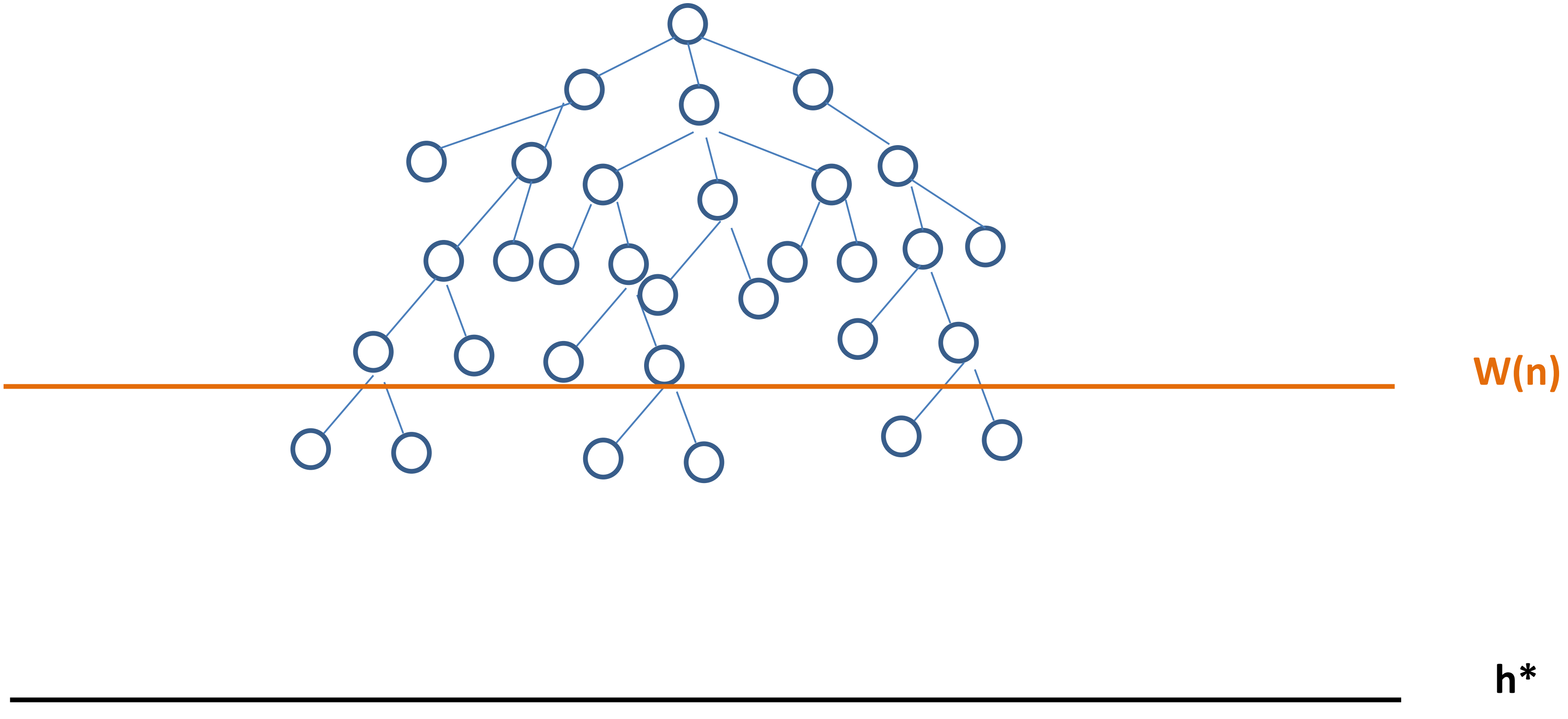
D(n)

h*

D(n) está mucho más informada que W(n)

Los valores D(n) están mucho más cerca del coste real (h*) que W(n)

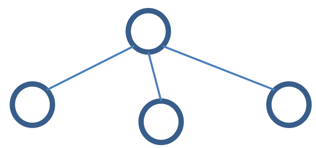
D(n) >> W(n) ⟶ D(n) domina a W(n)

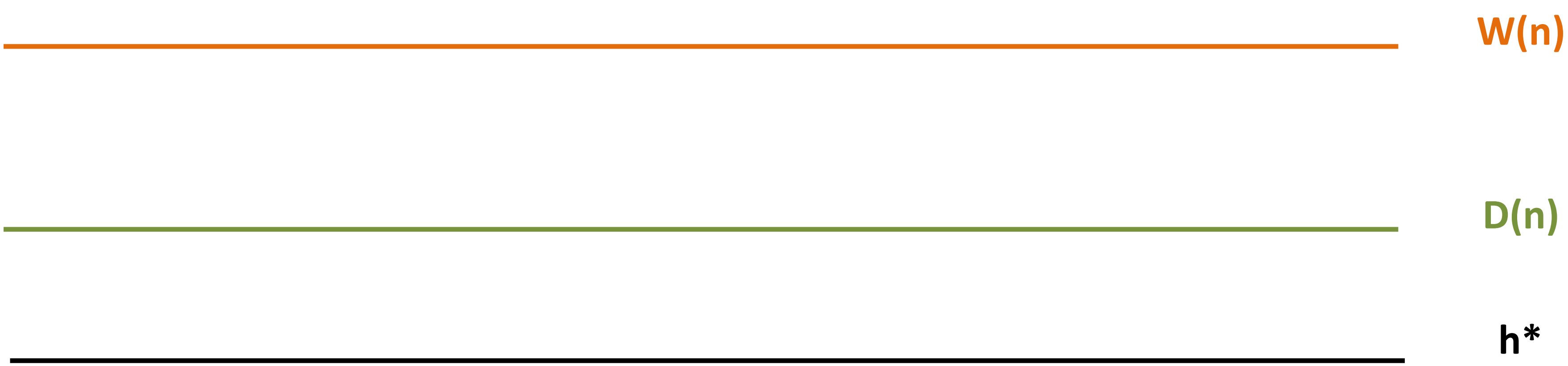


D(n) está mucho más informada que W(n)

Los valores D(n) están mucho más cerca del coste real (h*) que W(n)

$D(n) >> W(n) \longrightarrow$ D(n) domina a W(n)

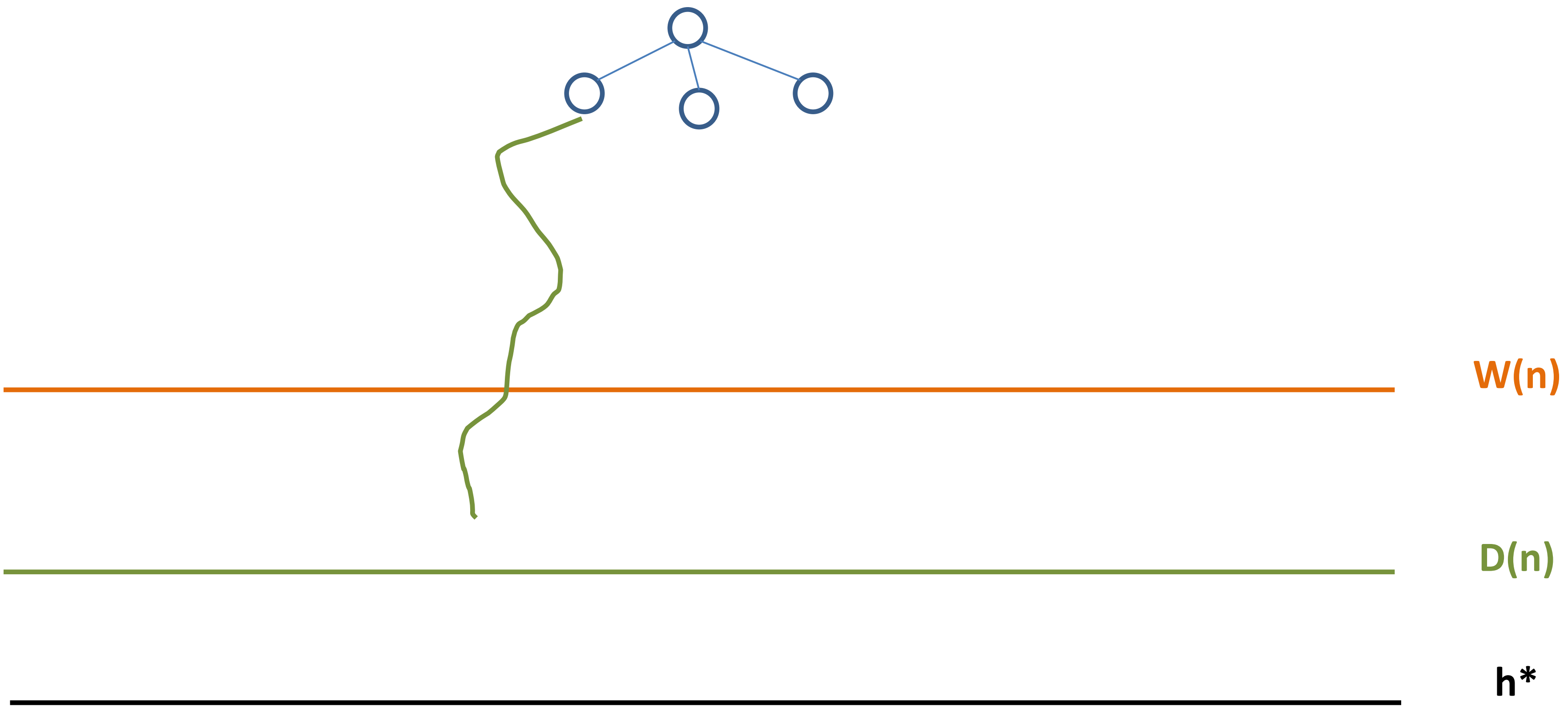




D(n) está mucho más informada que W(n)

Los valores D(n) están mucho más cerca del coste real (h*) que W(n)

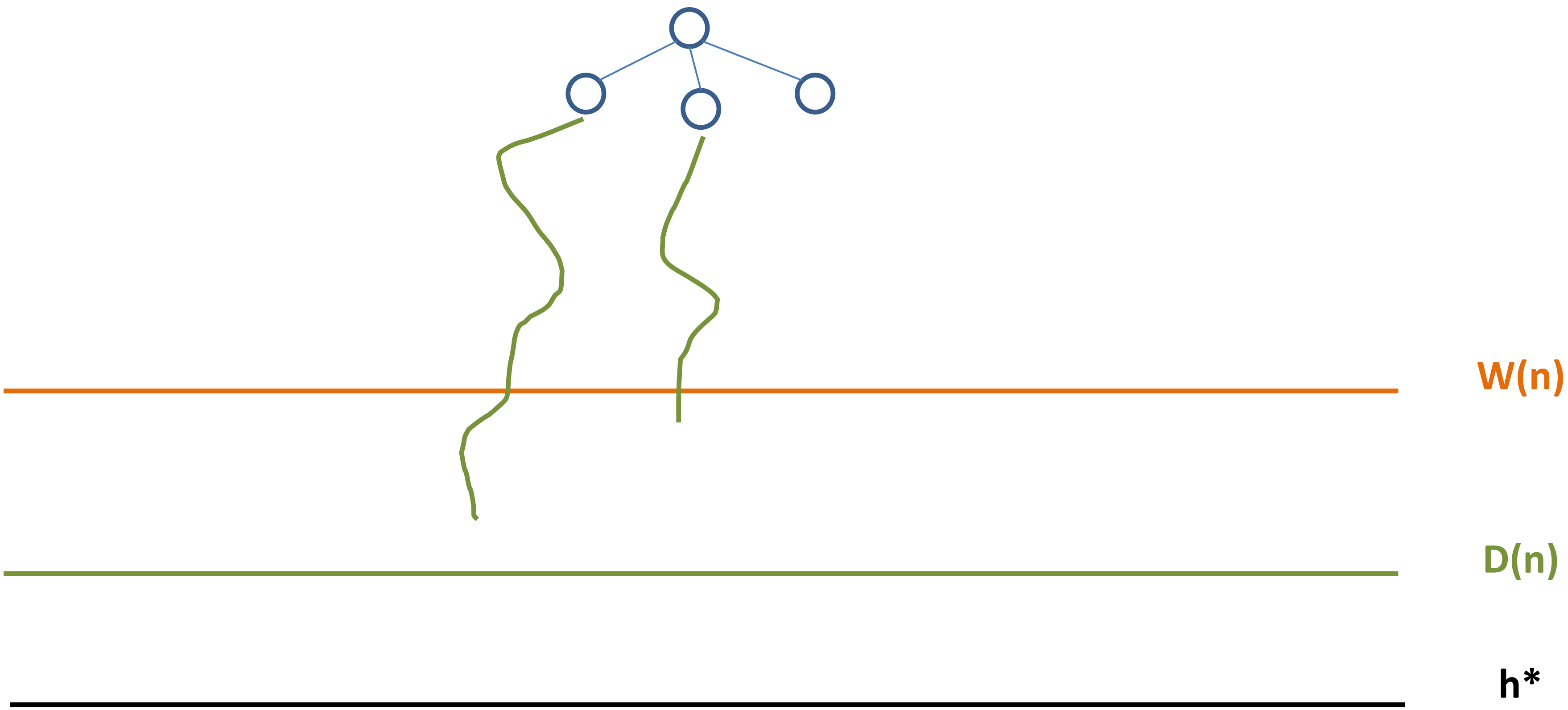
D(n) >> W(n) $\longrightarrow$ D(n) domina a W(n)



D(n) está mucho más informada que W(n)

Los valores D(n) están mucho más cerca del coste real (h*) que W(n)

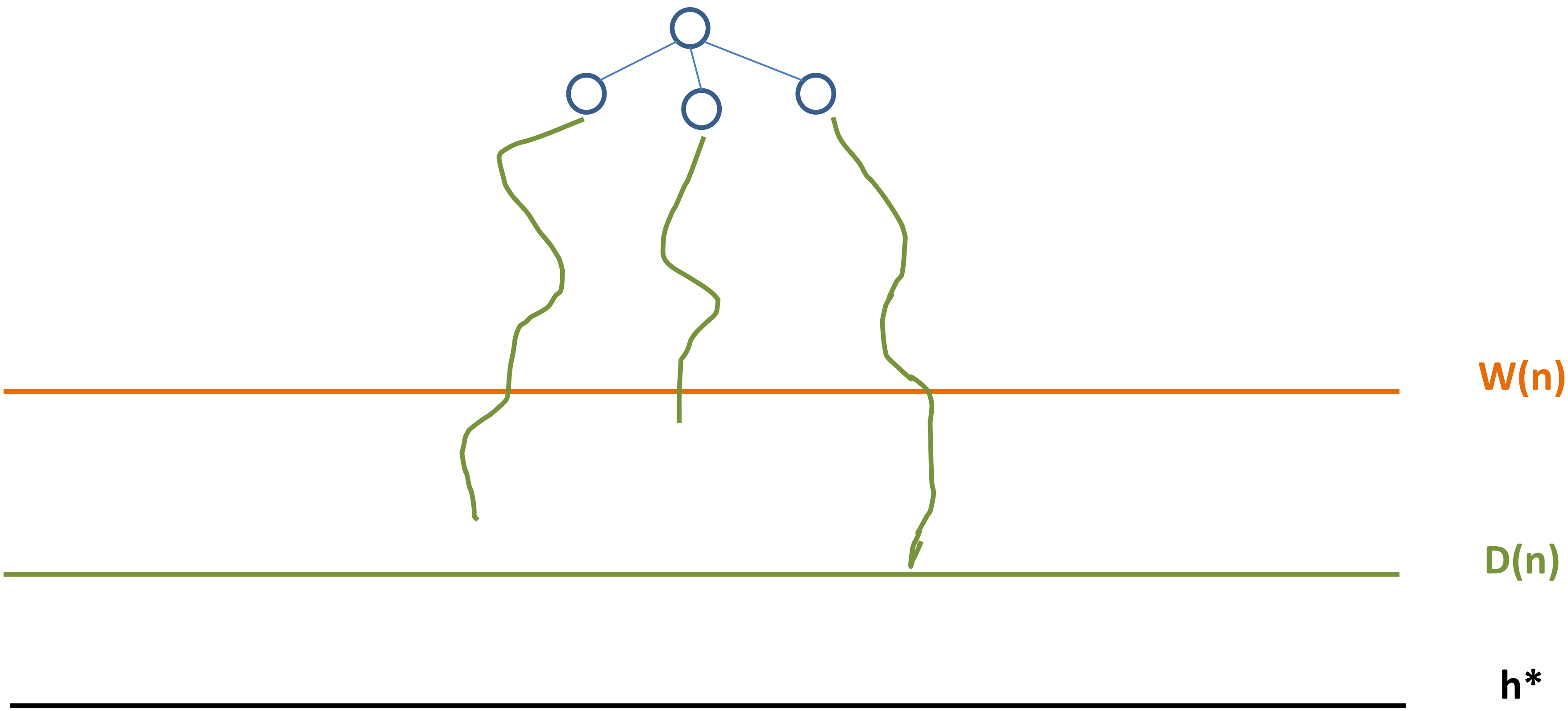
D(n) >> W(n) → D(n) domina a W(n)



D(n) está mucho más informada que W(n)

Los valores D(n) están mucho más cerca del coste real (h*) que W(n)

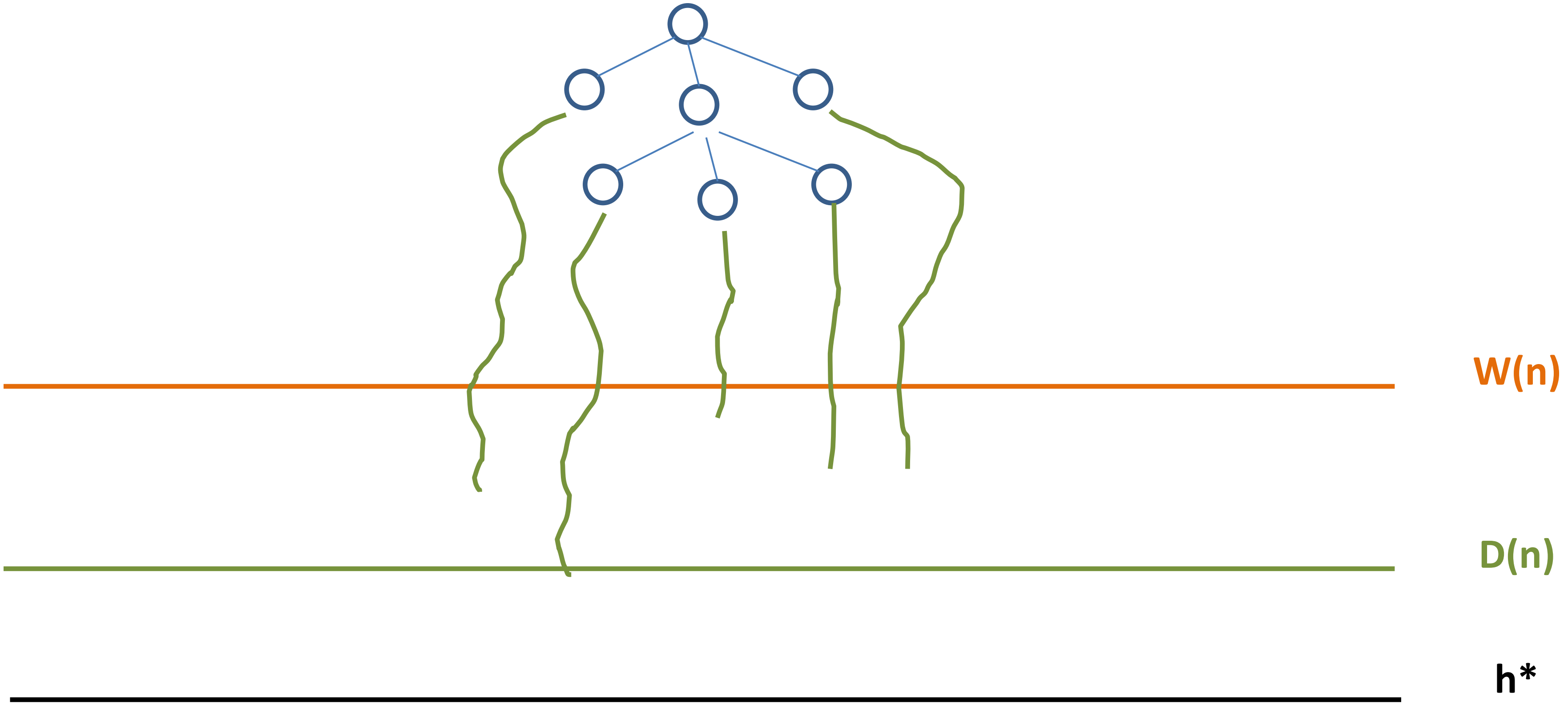
D(n) >> W(n) $\longrightarrow$ D(n) domina a W(n)



D(n) está mucho más informada que W(n)

Los valores D(n) están mucho más cerca del coste real (h*) que W(n)

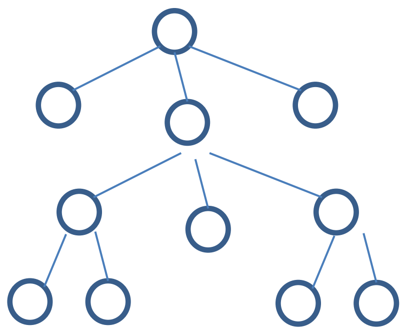
D(n) >> W(n) $\longrightarrow$ D(n) domina a W(n)



D(n) está mucho más informada que W(n)

Los valores D(n) están mucho más cerca del coste real (h*) que W(n)

D(n) >> W(n) $\longrightarrow$ D(n) domina a W(n)



W(n)

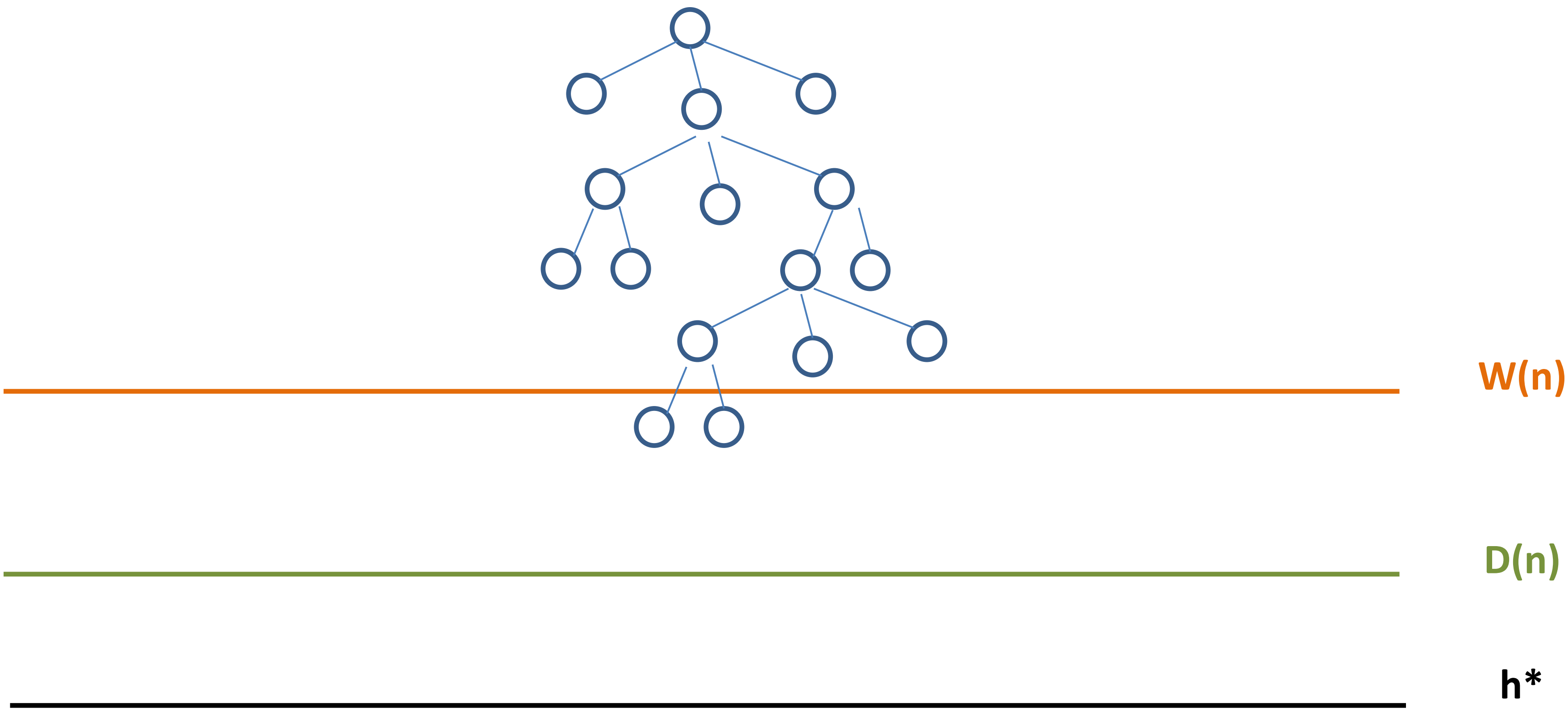
D(n)

h*

D(n) está mucho más informada que W(n)

Los valores D(n) están mucho más cerca del coste real (h*) que W(n)

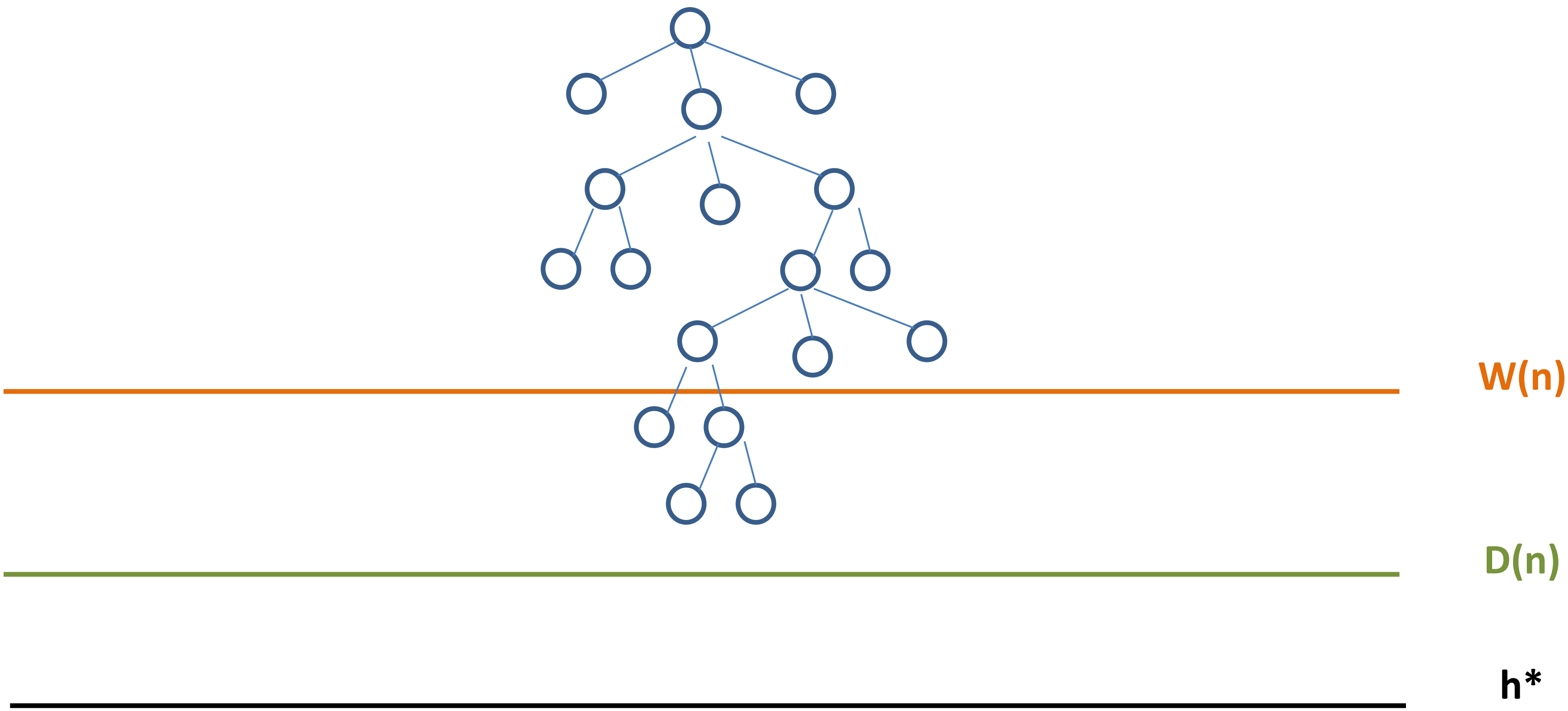
D(n) >> W(n) → D(n) domina a W(n)



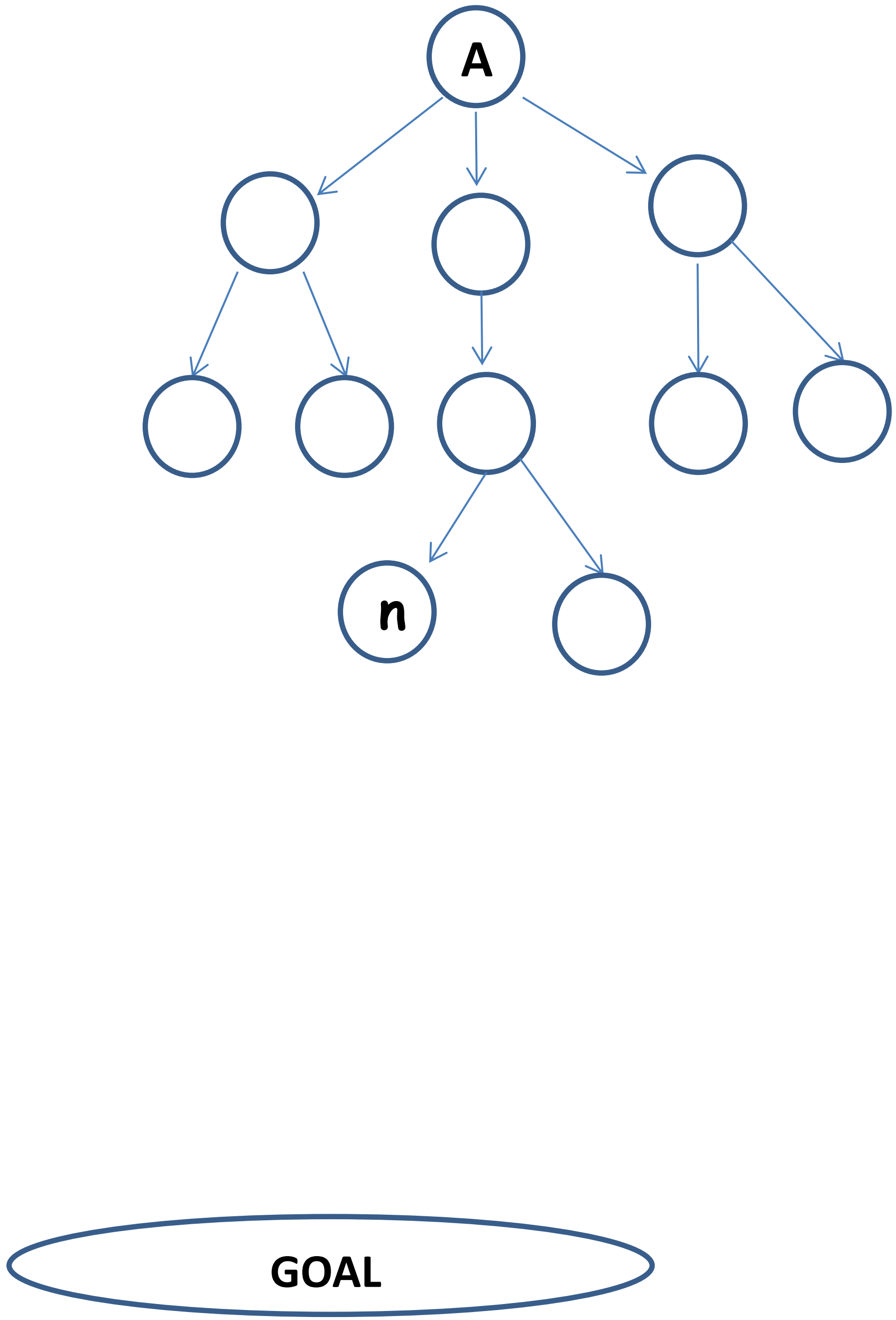
D(n) está mucho más informada que W(n)

Los valores D(n) están mucho más cerca del coste real (h*) que W(n)

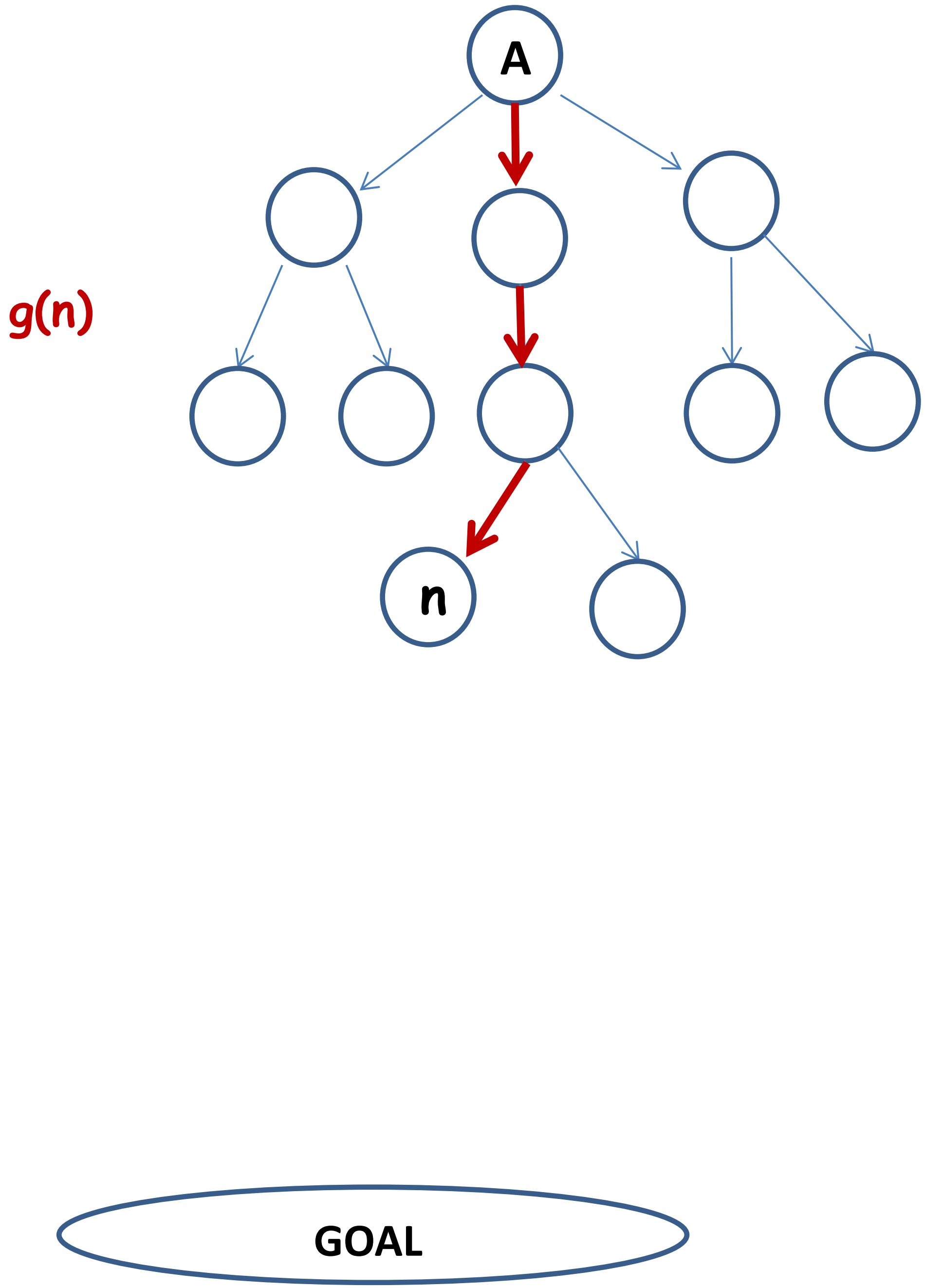
D(n) >> W(n) $\longrightarrow$ D(n) domina a W(n)



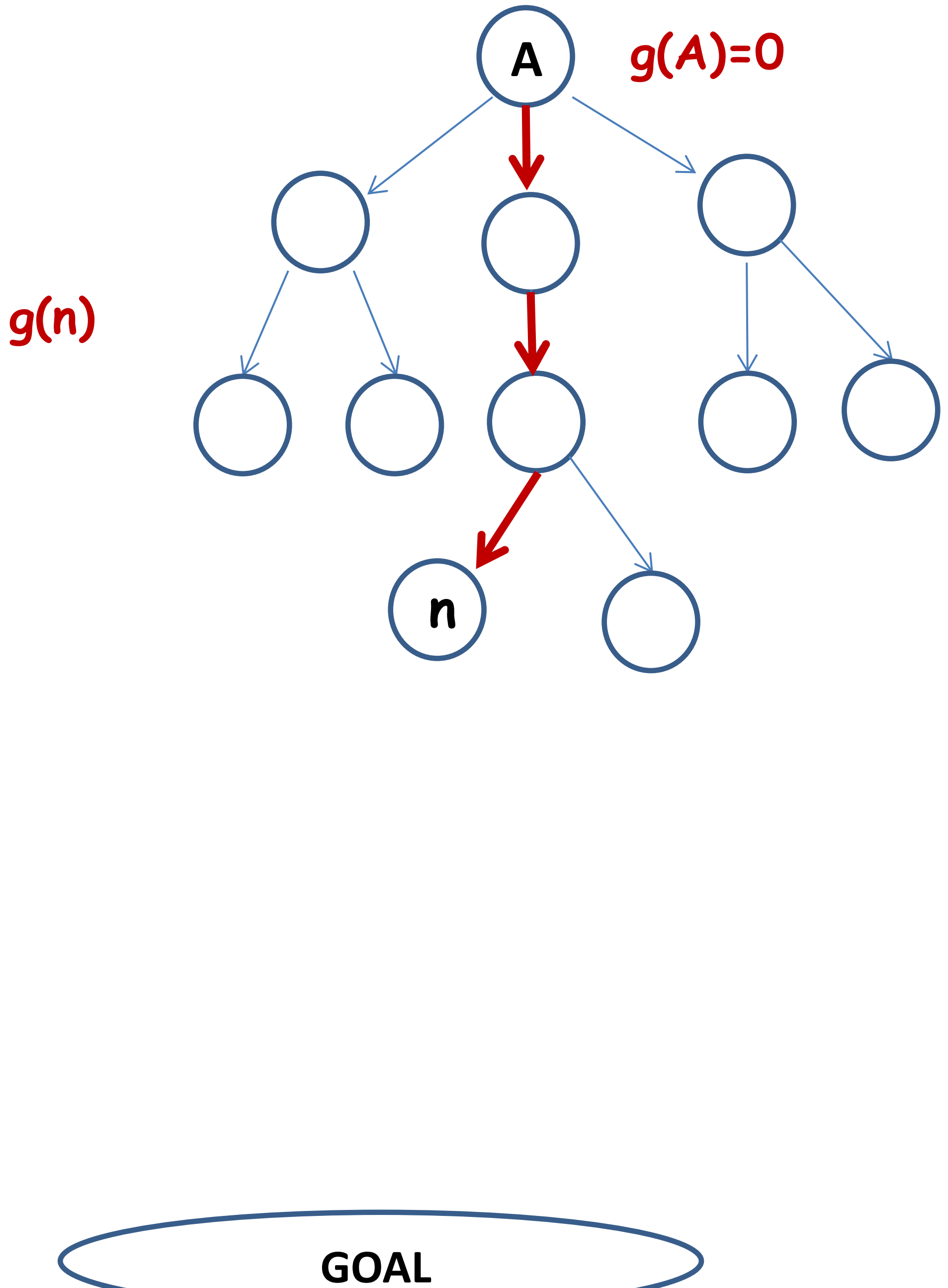
RESUMEN



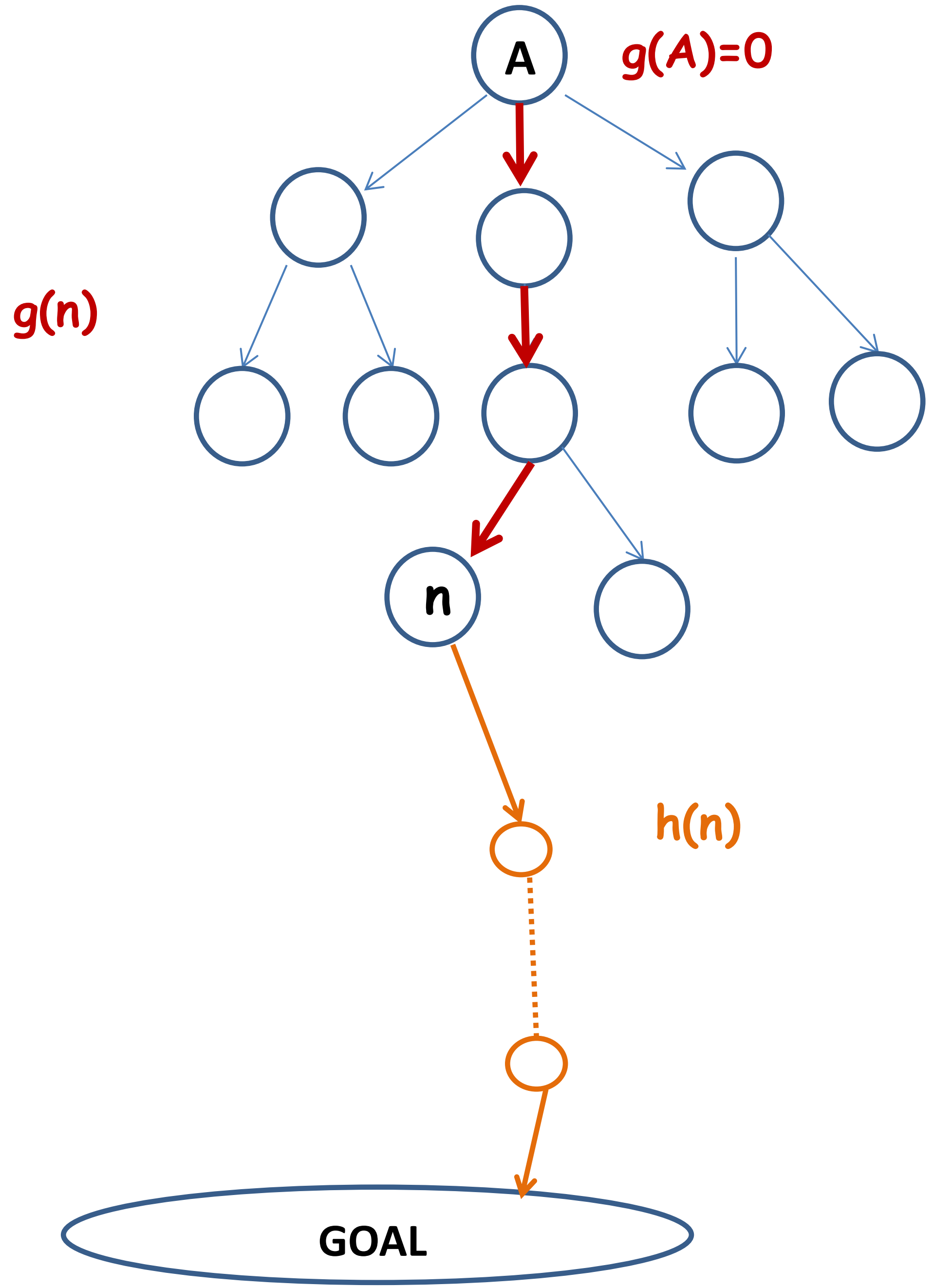
RESUMEN



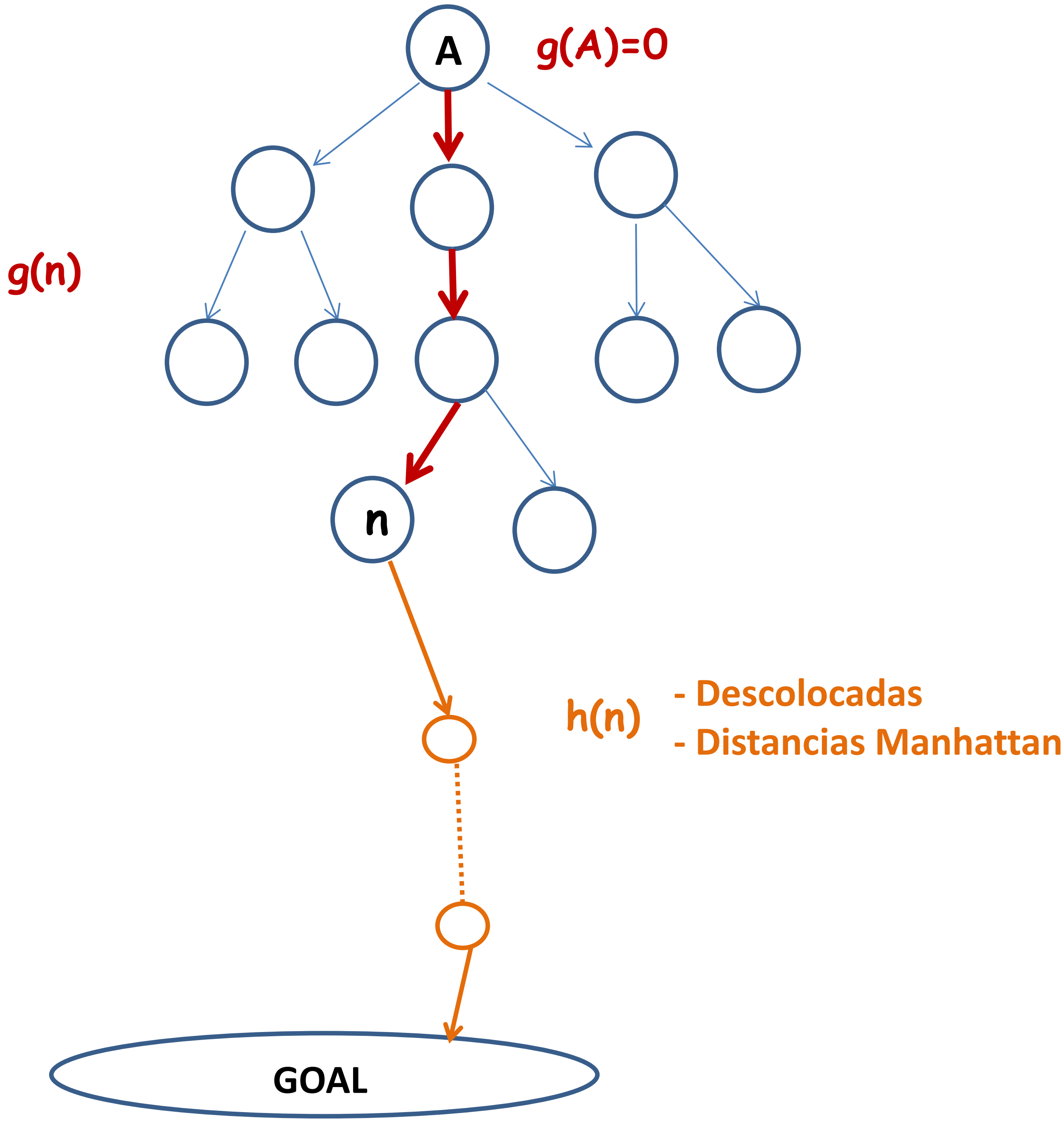
RESUMEN



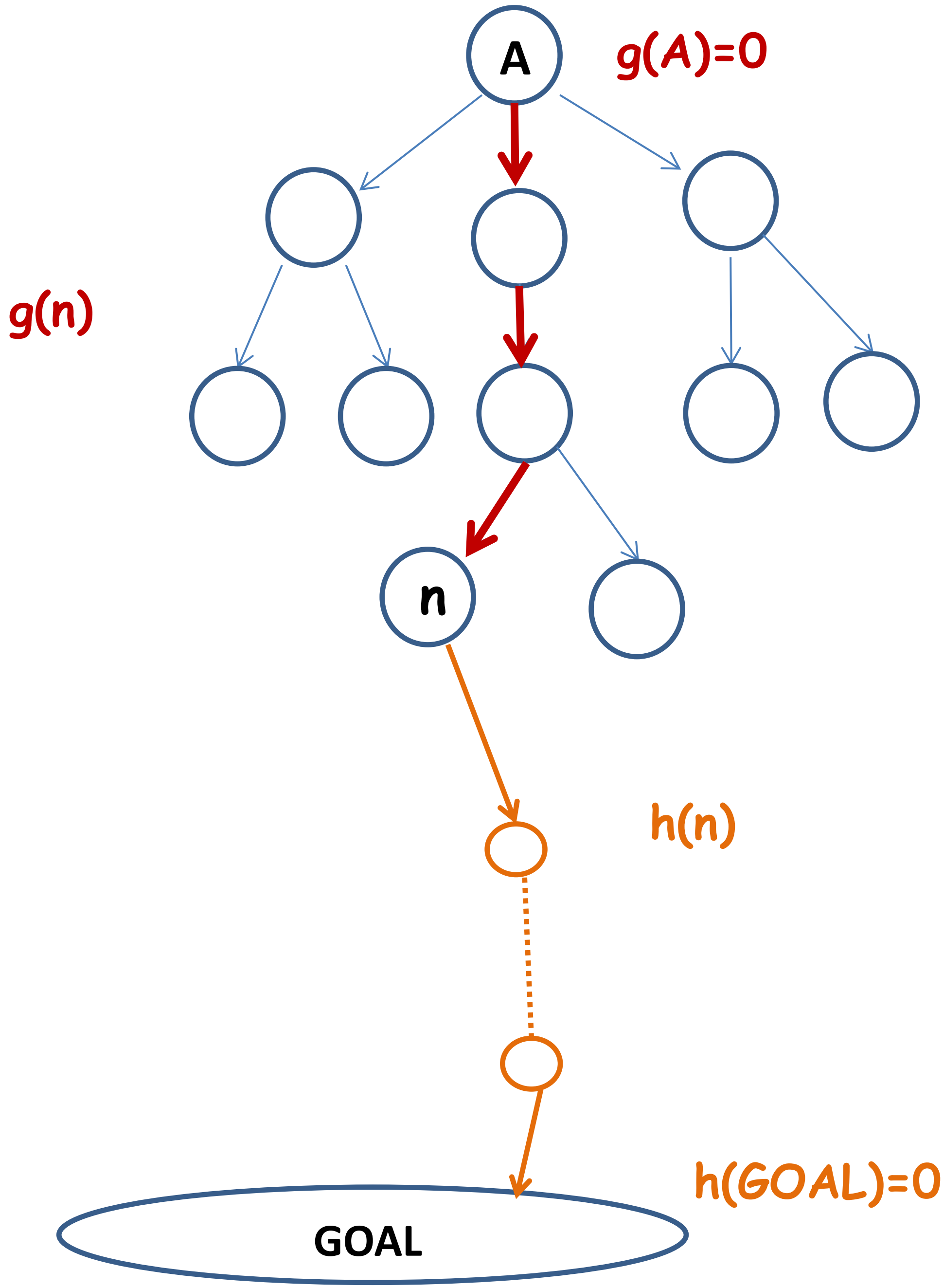
RESUMEN



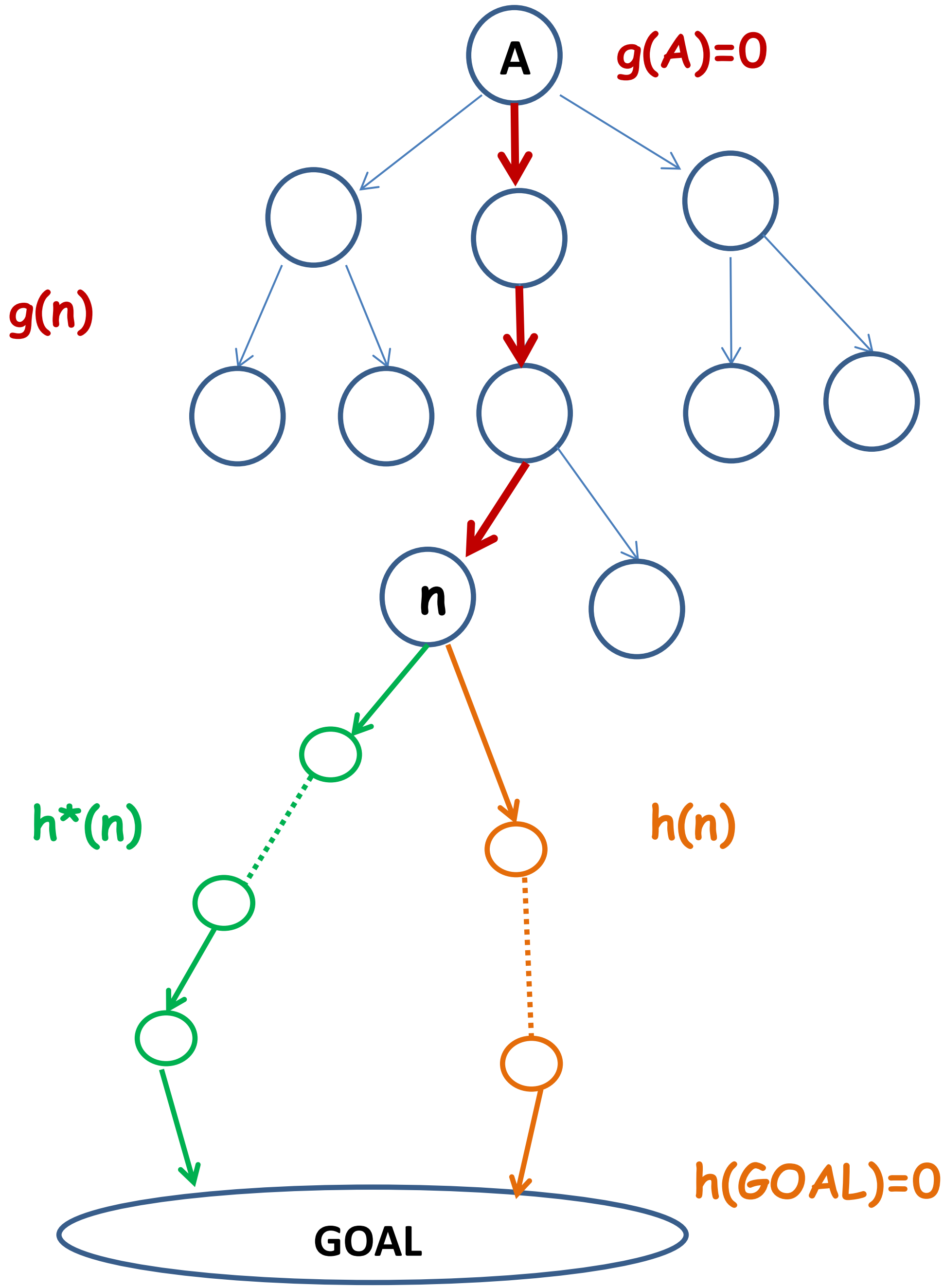
RESUMEN



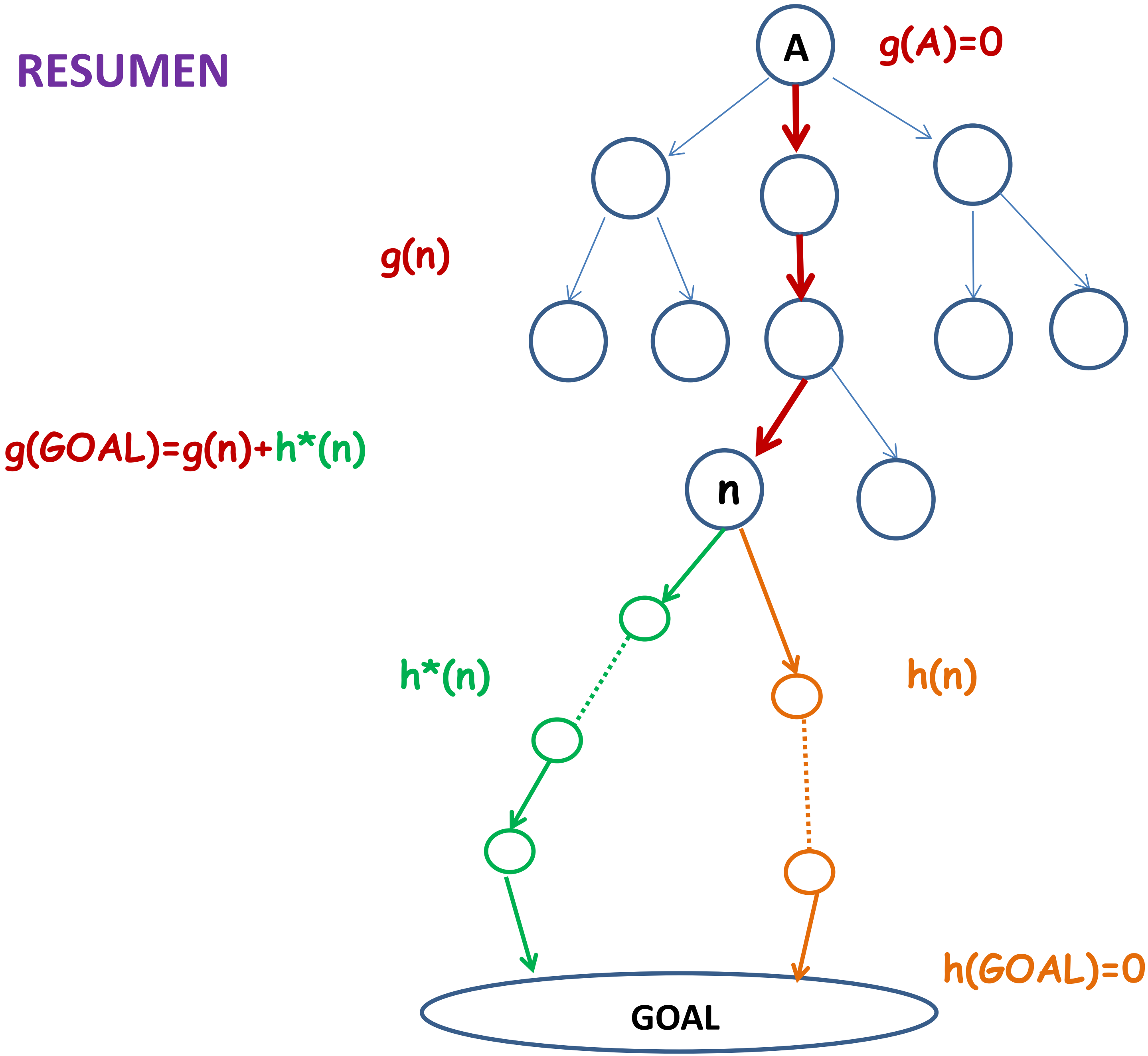
RESUMEN



RESUMEN



RESUMEN



RESUMEN

$g(GOAL) = g(n) + h^*(n)$

$g(GOAL) = h^*(A)$