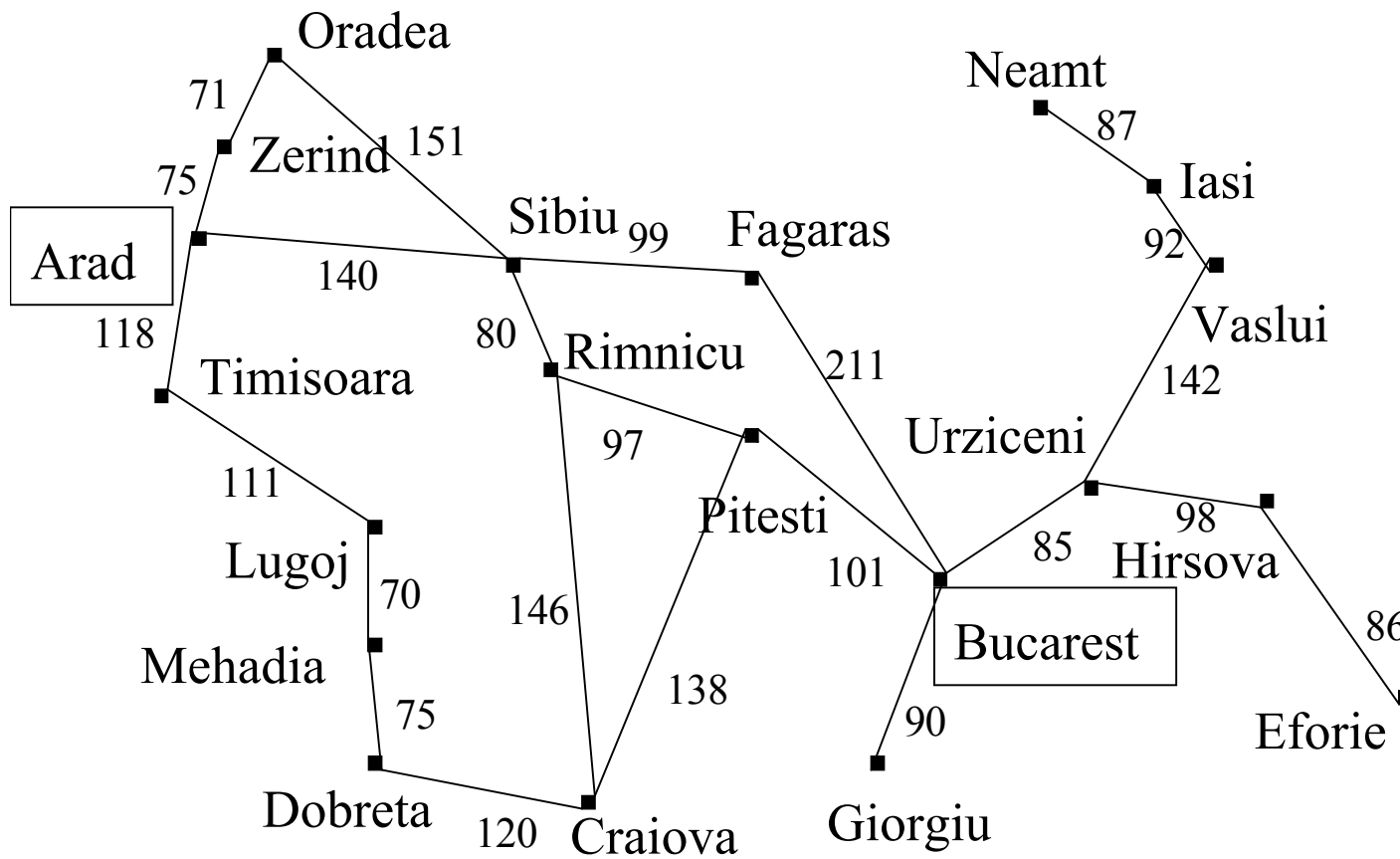


Ricerca golosa (“best first”)

- Prende in considerazione tutti i nodi che sono stati esaminati fino ad un certo momento ed espande il nodo che sembra più promettente.
- È chiamata **ricerca golosa**, perché espande il nodo che “localmente” sembra migliore, senza preoccuparsi se lo sarà anche nel lungo termine.
- Presenta gli stessi problemi della ricerca in profondità: non è né ottimale né completa:
 - Occorre risolvere il problema dei cicli,
- Problema delle false partenze:
 - Esempio: trovare un itinerario da Iasi a Fagaras

Ricerca Best-First su mappe stradali (1)



Distanze in Km.

Distanze in linea d'aria
da Bucarest

Arad	366
Bucarest	0
Craiova	160
Dobreta	242
Eforie	161
Giorgiu	77
Fagaras	178
Hirsova	151
Iasi	226
Lugoj	244
Mehadia	241
Neamt	234
Oradea	380
Pitesti	98
Rimnicu	193
Sibiu	253
Timisoara	329
Urziceni	80
Vaslui	199
Zerind	374

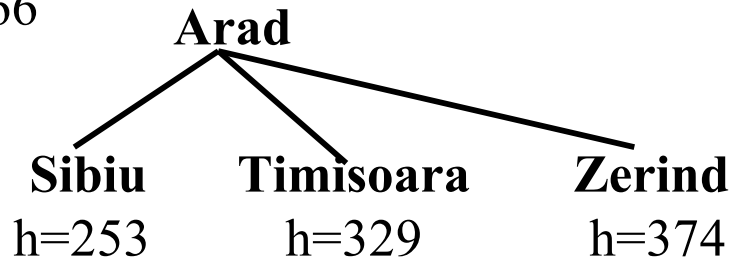
Ricerca Best-First su mappe stradali (2)

$h(\text{nodo}) = \text{Distanza in linea d'aria}$

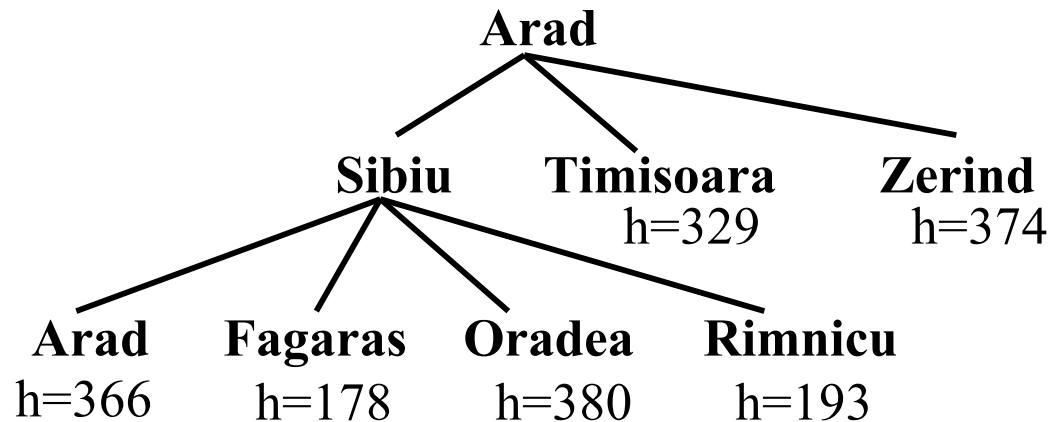
Lo stato iniziale

Arad
h=366

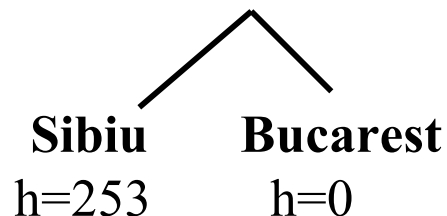
Dopo aver
espanso Arad



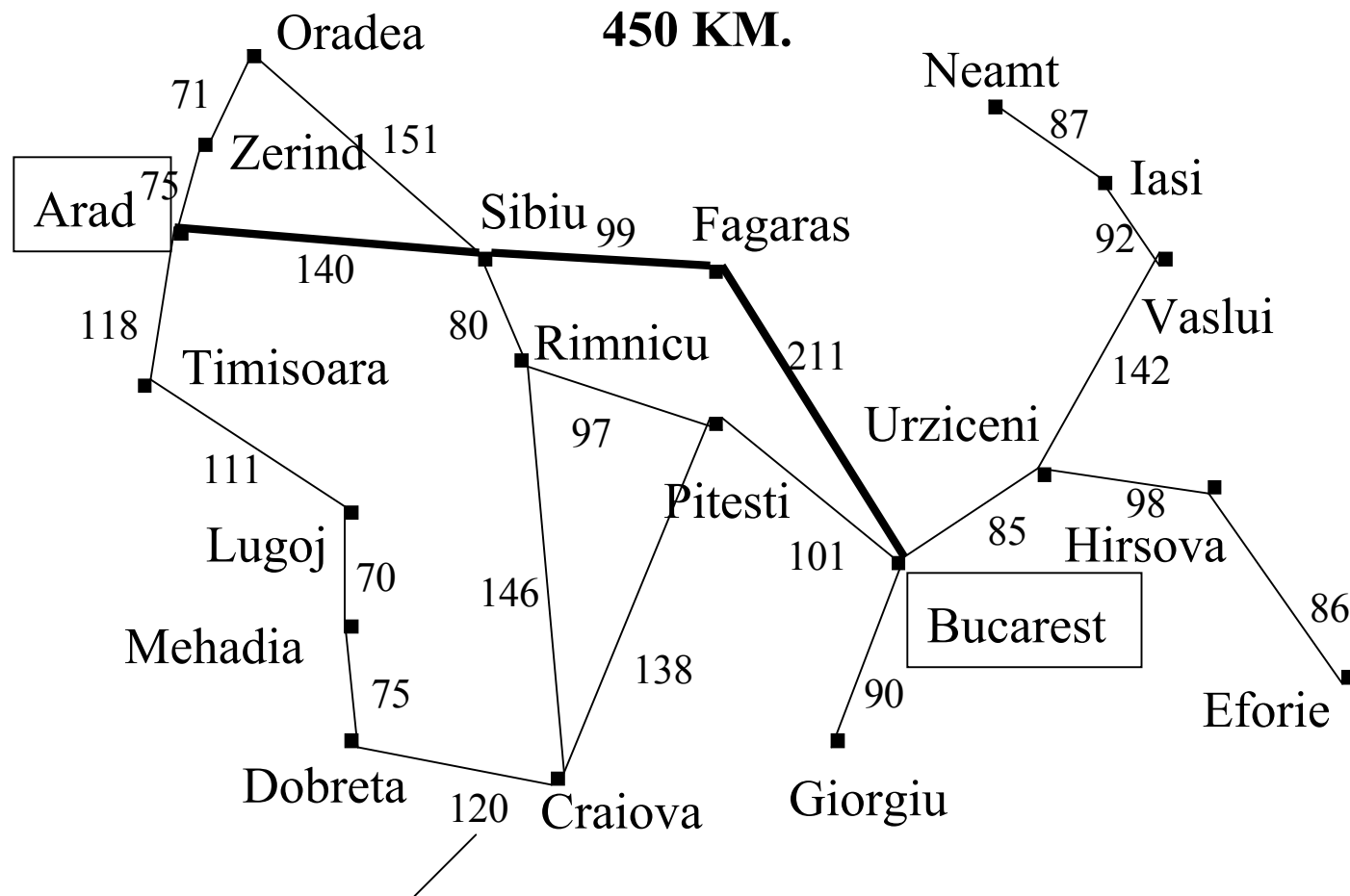
Dopo aver
espanso Sibiu



Dopo aver
espanso Faragas



Ricerca Best-First su mappe stradali (3)



Distanze in Km.

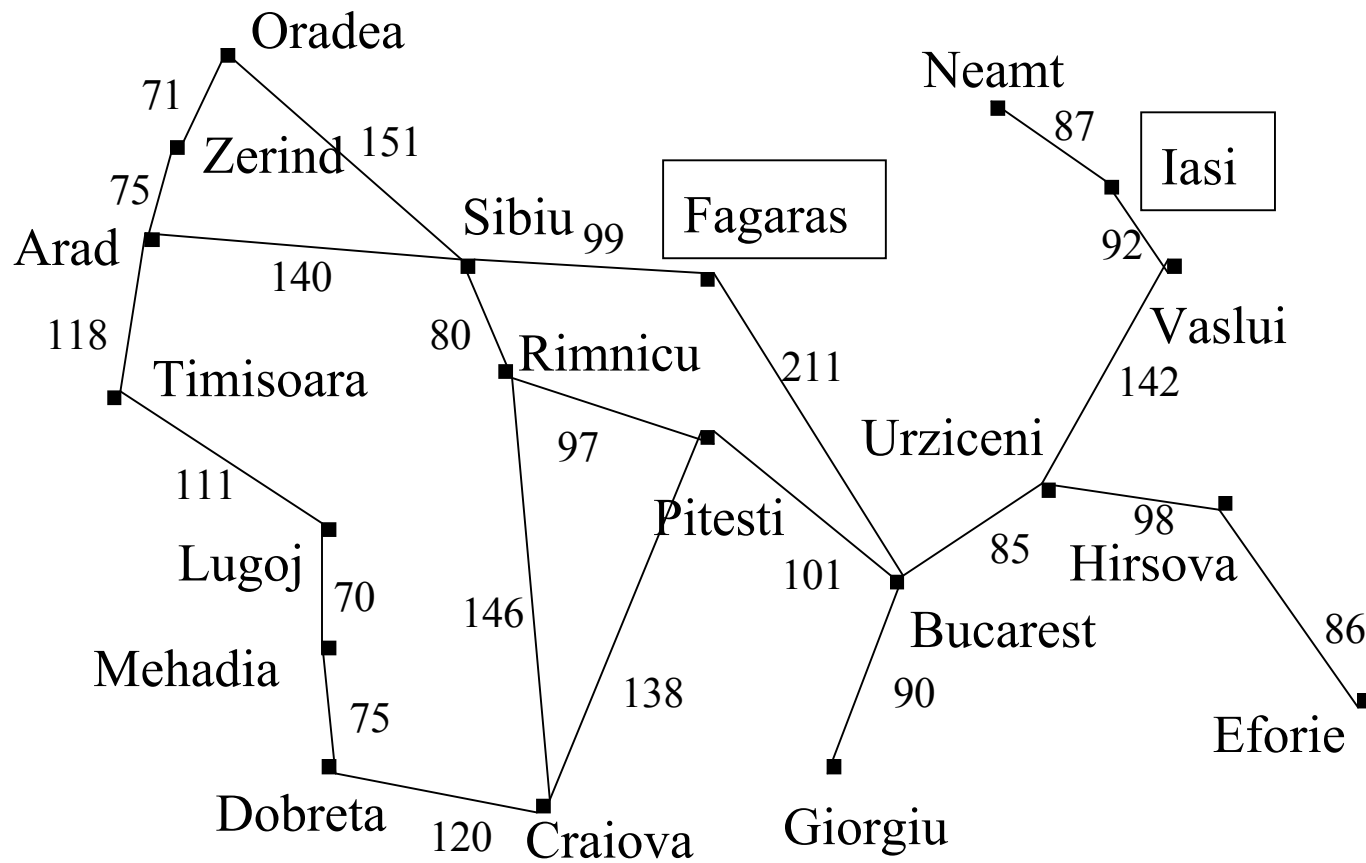
Distanze in linea d'aria
da Bucarest

Arad	366
Bucarest	0
Craiova	160
Dobreta	242
Eforie	161
Giorgiu	77
Fagaras	178
Hirsova	151
Iasi	226
Lugoj	244
Mehadia	241
Neamt	234
Oradea	380
Pitesti	98
Rimnicu	193
Sibiu	253
Timisoara	329
Urziceni	80
Vaslui	199
Zerind	374

Distanze in Km.

Lezione 6 12

False partenze



Distanze in Km.

Distanze in linea d'aria
da Bucarest

Arad	366
Bucarest	0
Craiova	160
Dobreta	242
Eforie	161
Giorgiu	77
Fagaras	178
Hirsova	151
Iasi	226
Lugoj	244
Mehadia	241
Neamt	234
Oradea	380
Pitesti	98
Rimnicu	193
Sibiu	253
Timisoara	329
Urziceni	80
Vaslui	199
Zerind	374

Ricerca “best first”

	Ampiezza	Profondità	Profondità limitata	“Best first”
Tempo	r^p	r^m	r^l	$< r^m$ nel caso medio
Spazio	r^p	$r * m$	$r * l$	$< r^m$ nel caso medio
Ottimale?	Sì (se gli operatori hanno costo uguale)	No	No	No
Completa?	Sì	No	Sì, se $l \geq p$	No

- Valutazione di strategie di ricerca. r è il fattore di ramificazione; p è la profondità stimata della soluzione; m è la profondità massima dell'albero di ricerca (nel caso peggiore p coincide con m); l è il limite di profondità massima deciso.

Ricerca A

- Migliora la ricerca best-first in modo che la soluzione trovata sia non solo accettabile, ma anche la soluzione **ottima**.

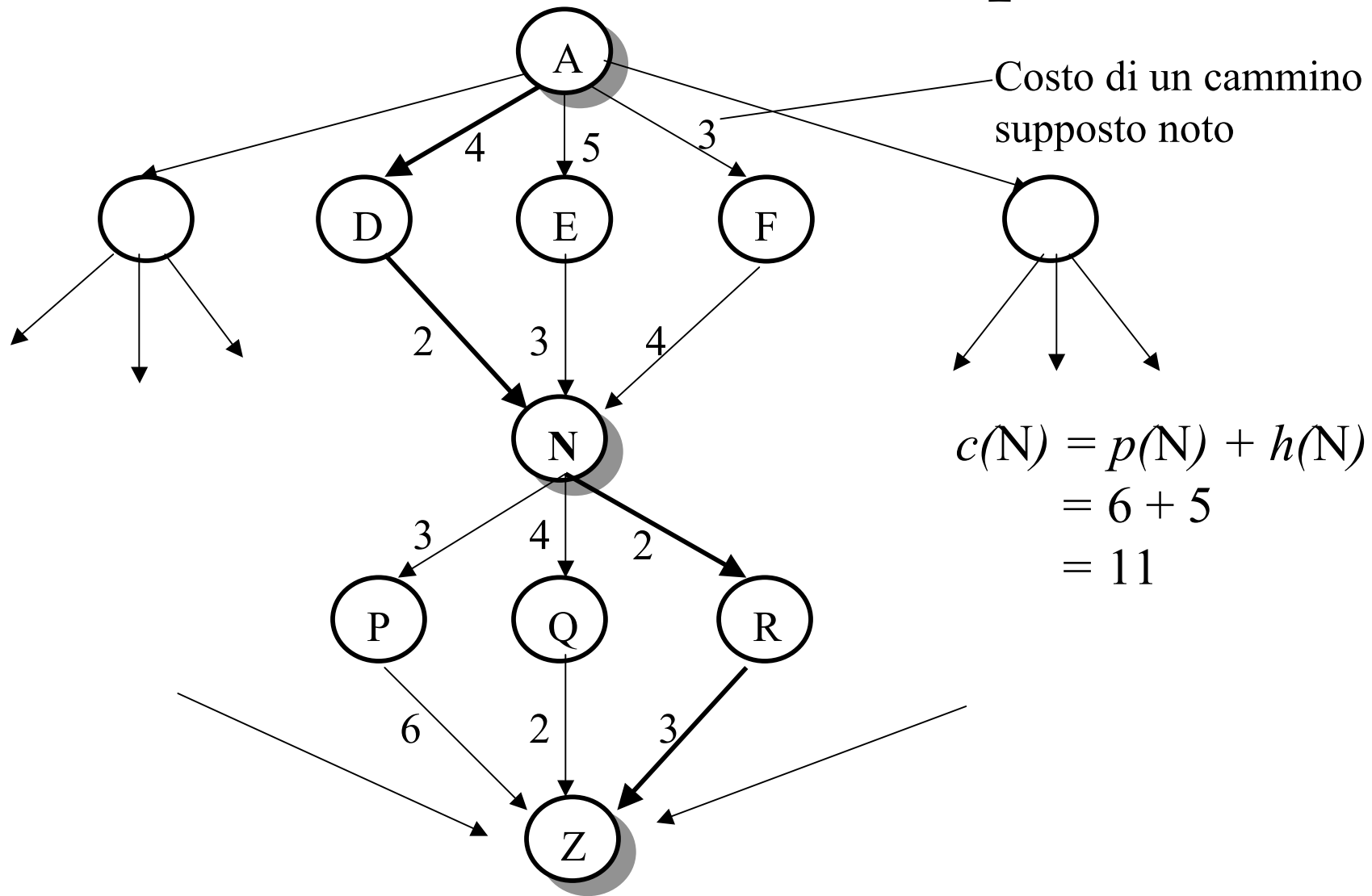
$$c(n) = p(n) + h(n)$$

$c(n)$ è il costo del cammino di soluzione ottimo passante per n , vale a dire il minimo cammino tra il nodo iniziale e il nodo obiettivo che passa per n .

Il valore di $c(n)$ deriva dalla somma di due componenti:

- $p(n)$: il costo del cammino ottimo dalla radice a n
(il miglior cammino prima di n)
- $h(n)$: il costo del cammino ottimo da n ad un nodo finale
(il miglior cammino dopo n).

Ricerca A: situazione (per assurdo) di conoscenza completa



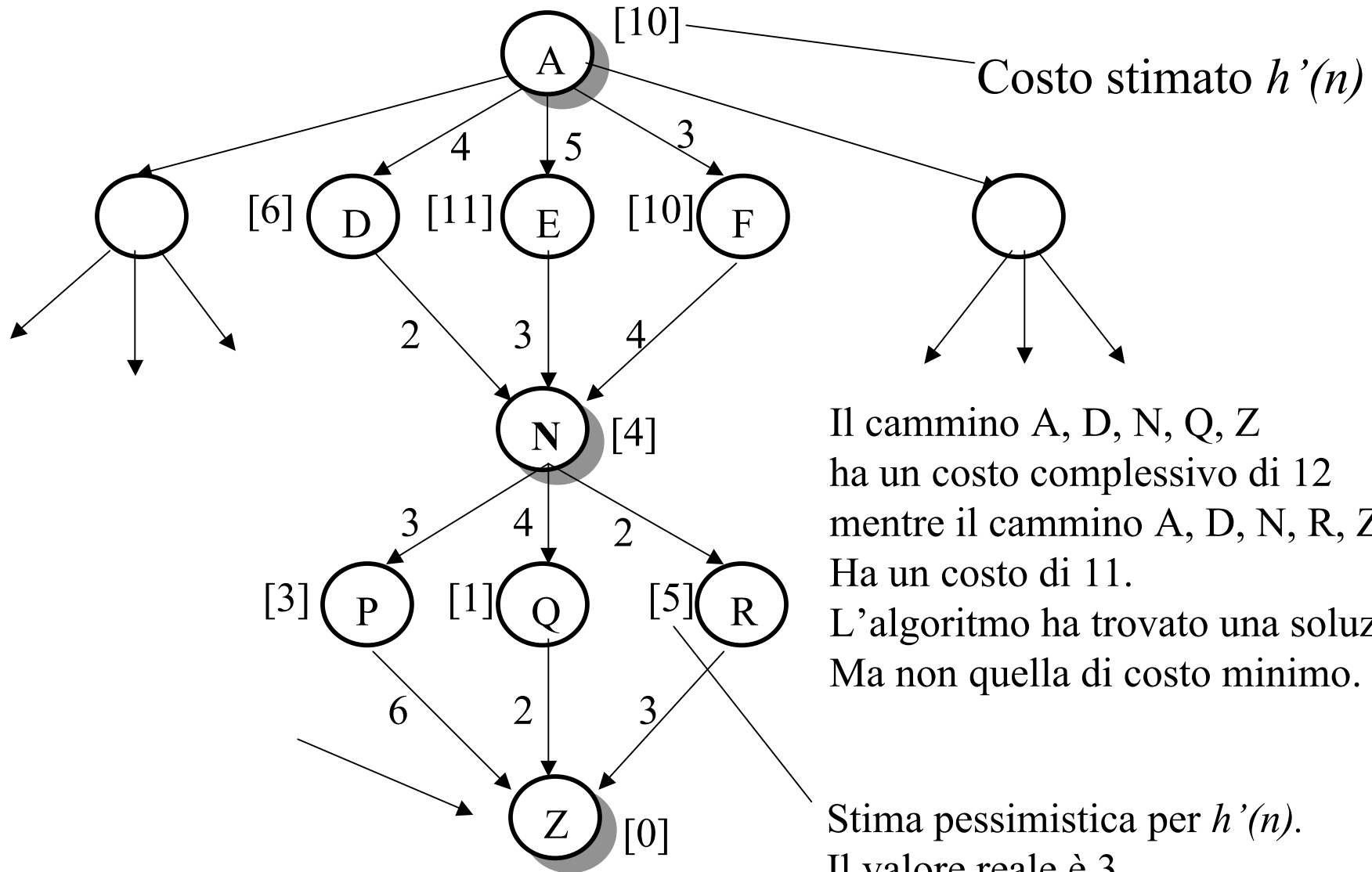
Ricerca A

- **Normalmente $c(n)$ non è noto per tutti i nodi di un albero di ricerca:** è quindi necessario stimarlo in modo approssimato.
- Non si usa quindi il *costo reale*, ma un *costo presunto* $c'(n)$:

$$c'(n) = p'(n) + h'(n)$$

- Come si possono calcolare $p'(n)$ e $h'(n)$?
 - $p'(n)$: possiamo usare il costo del cammino effettivamente percorso come stima del cammino minimo dalla radice a n .
 - $h'(n)$: adottiamo una stima euristica basata sulla conoscenza specifica del dominio, ad esempio la distanza in linea d'aria, per il caso di ricerca di itinerari.

Ricerca A



Il cammino A, D, N, Q, Z
ha un costo complessivo di 12
mentre il cammino A, D, N, R, Z
Ha un costo di 11.
L'algoritmo ha trovato una soluzione,
Ma non quella di costo minimo.

Stima pessimistica per $h'(n)$.
Il valore reale è 3.

Ricerca A^*

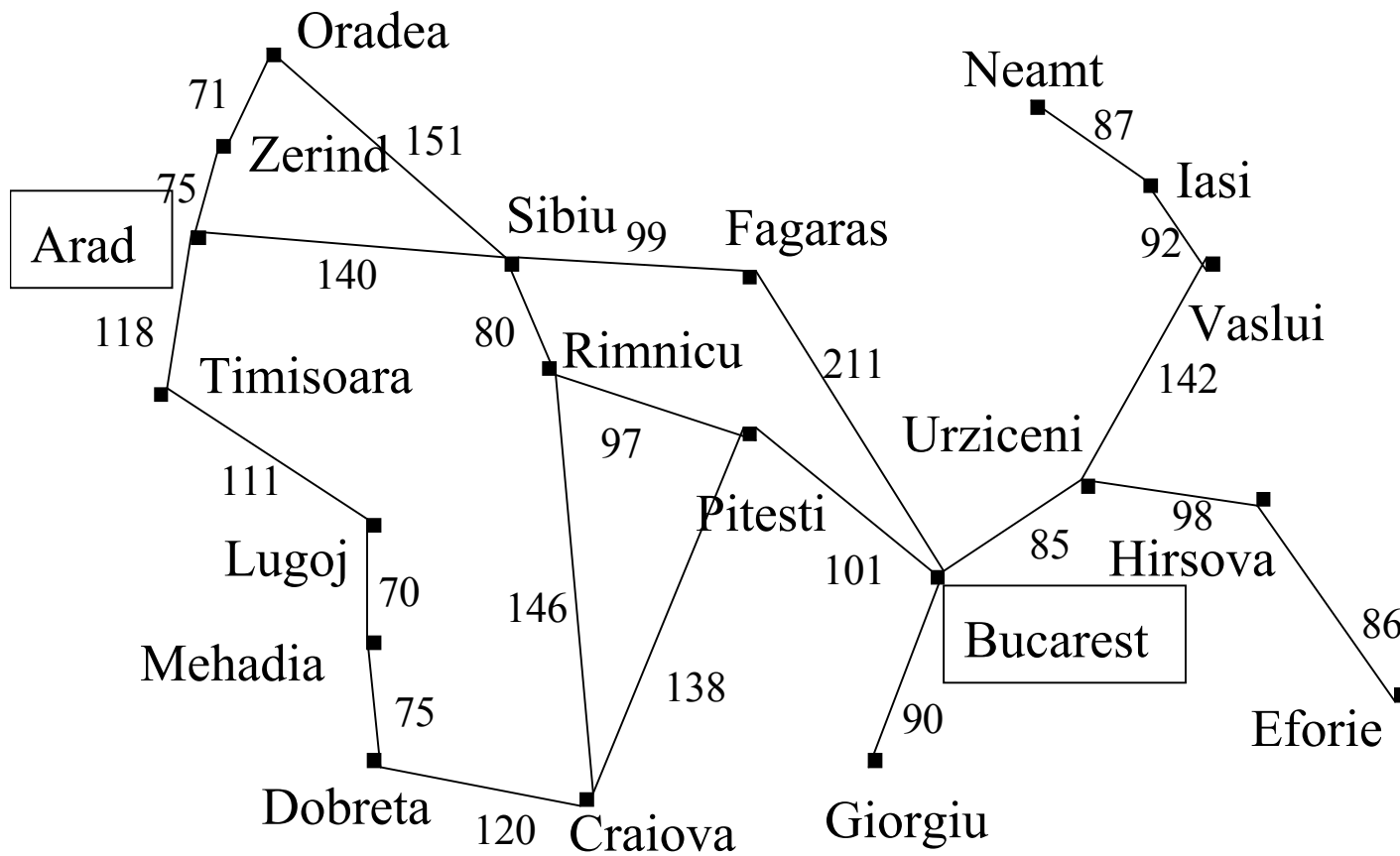
- In generale, non conviene essere troppo **pessimisti** nella stima della funzione $h'(n)$, perché si possono trascurare soluzioni ottime.
- Un algoritmo che usa una stima **ottimista** del costo necessario a raggiungere la soluzione partendo da un certo nodo n si chiama A^* .
- Si ha quindi una ricerca A^* quando il costo stimato dalla funzione euristica $h'(n)$ è inferiore al costo reale della funzione $h(n)$:

$$h'(n) \leq h(n)$$

Ricerca A*

- E' stato dimostrato [Pearl, 1984] che l'algoritmo A* consente di trovare sempre la **soluzione ottima** (si dice che l'algoritmo è **ammissibile**), se tale soluzione esiste.
- Data questa proprietà, per essere sicuri di trovare la soluzione ottima è sufficiente usare una stima ottimistica. Ad esempio, si potrebbe usare $d'(n) = 0$ per tutti i nodi.
- Tuttavia, affinché la ricerca sia anche efficiente è utile approssimarsi al valore reale di $d'(n)$.
- Una **funzione $d'(n)$ ideale** dovrebbe fornire valori uguali o inferiori a $d(n)$ per garantire la soluzione ottima e dovrebbe dare valori il più vicino possibile a $d(n)$ per garantire l'efficienza.

Ricerca A* su mappe stradali



Distanze in Km.

Distanze in linea d'aria
da Bucarest

Arad	366
Bucarest	0
Craiova	160
Dobreta	242
Eforie	161
Giorgiu	77
Fagaras	178
Hirsova	151
Iasi	226
Lugoj	244
Mehadia	241
Neamt	234
Oradea	380
Pitesti	98
Rimnicu	193
Sibiu	253
Timisoara	329
Urziceni	80
Vaslui	199
Zerind	374

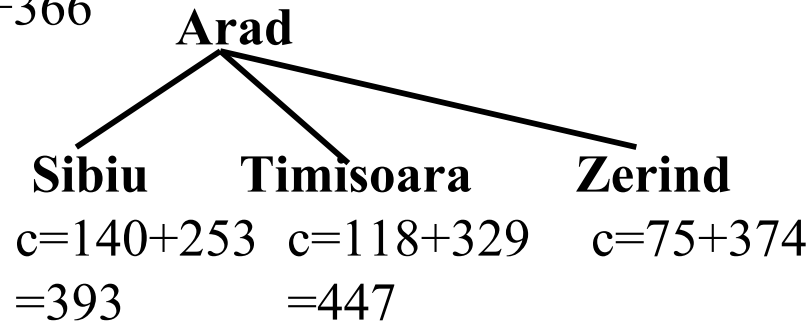
A* su mappe stradali

$c(nodo) = \text{Distanza in KM.} + \text{distanza in linea d'aria}$

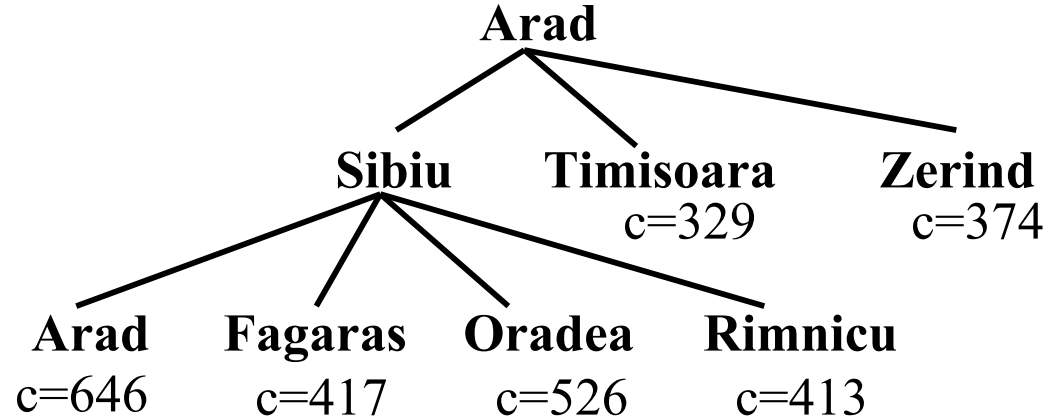
Lo stato iniziale

Arad
 $c=0+366$

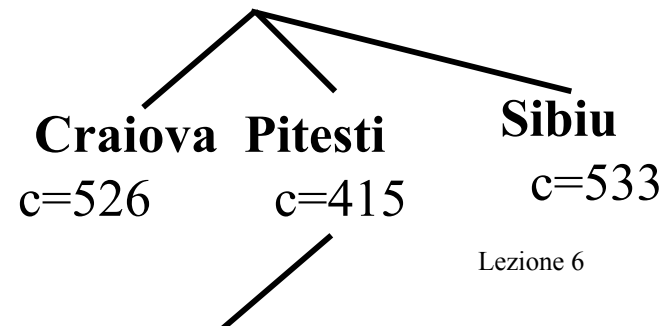
Dopo aver
espanso Arad



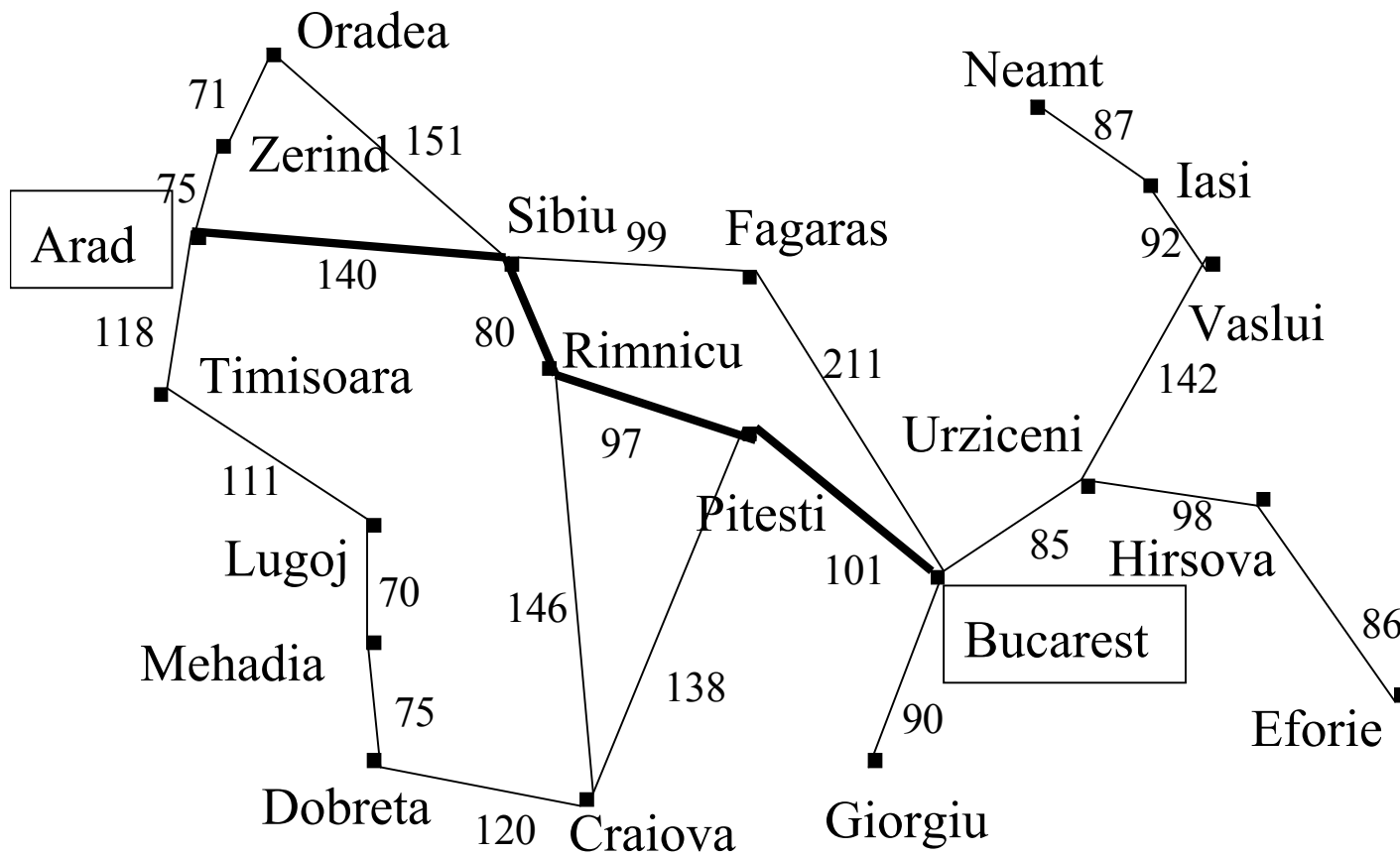
Dopo aver
espanso Sibiu



Dopo aver
Espanso Rimnicu



Ricerca A* su mappe stradali



Distanze in Km.

Distanze in linea d'aria
da Bucarest

Arad	366
Bucarest	0
Craiova	160
Dobreta	242
Eforie	161
Giorgiu	77
Fagaras	178
Hirsova	151
Iasi	226
Lugoj	244
Mehadia	241
Neamt	234
Oradea	380
Pitesti	98
Rimnicu	193
Sibiu	253
Timisoara	329
Urziceni	80
Vaslui	199
Zerind	374

Riassumendo

Criterio	Ampiezza	Profondità	Profondità limitata	Best first	A*
Tempo	r^p	r^m	r^l	$< r^m$ nel caso medio	$< r^m$ nel caso medio
Spazio	r^p	$r * m$	$r * l$	$< r^m$ nel caso medio	$< r^m$ nel caso medio
Ottimale?	Sì	No	No	No	Sì
Completo?	Sì	No	Sì, se $l \geq p$	No	Sì, con r finito

- Valutazione di strategie di ricerca. r è il fattore di ramificazione; p è la profondità stimata della soluzione; m è la profondità massima dell'albero di ricerca (nel caso peggiore p coincide con m); l è il limite di profondità massima deciso.