



Τεχνητή Νοημοσύνη

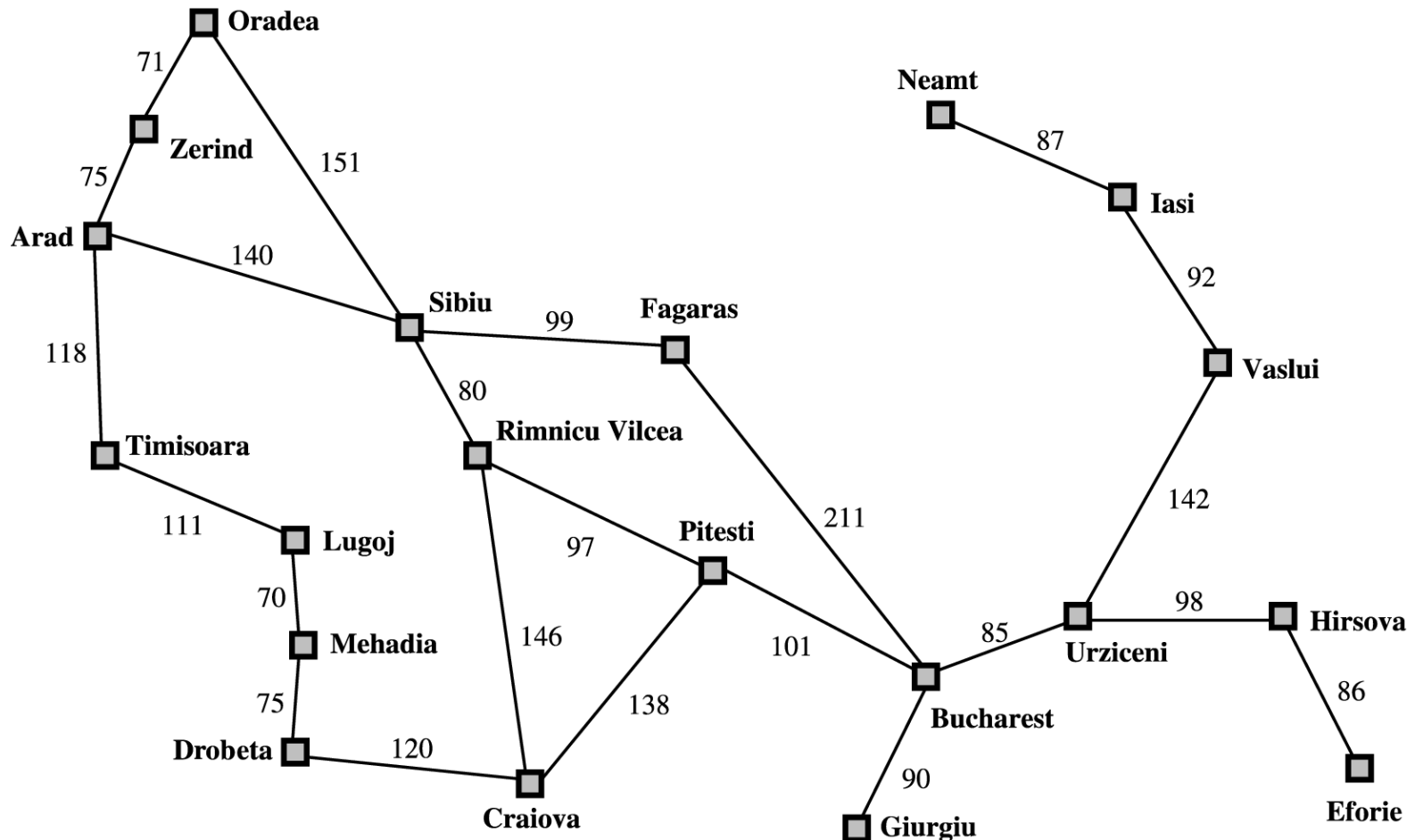
*Πληροφορημένη αναζήτηση
και εξερεύνηση*

*Διδάσκων: Τσίπουρας Μάρκος
Εκπαιδευτικό Υλικό: Τσίπουρας Μάρκος
<http://ai.uom.gr/aima/>*

Στρατηγικές πληροφορημένης αναζήτησης

- Πληροφορημένη αναζήτηση (informed search)
- Συνάρτηση αξιολόγησης (evaluation function), $f(n)$
 - Προτιμώνται οι μικρότερες τιμές
- Ευρετική συνάρτηση (heuristic function), $h(n)$
 - Συνήθως εκτίμηση απόστασης από στόχο

Αναζήτηση πρώτα στο καλύτερο (1/5)



Αναζήτηση πρώτα στο καλύτερο (2/5)

- Τιμές h_{SLD} για απόσταση από Βουκουρέστι

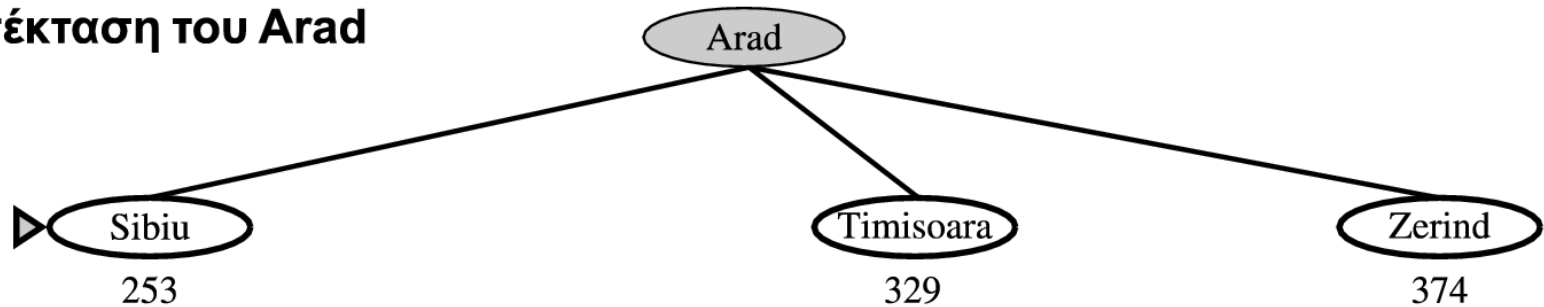
Arad	366	Mehadia	241
Bucharest	0	Neamt	234
Craiova	160	Oradea	380
Dobreta	242	Pitesti	100
Eforie	161	Rimniscu Vilcea	193
Fagaras	176	Sibiu	253
Giurgiu	77	Timisoara	329
Hirsova	151	Urziceni	80
Iasi	226	Vaslui	199
Lugoj	244	Zerind	374

Αναζήτηση πρώτα στο καλύτερο (3/5)

(α) Η αρχική κατάσταση

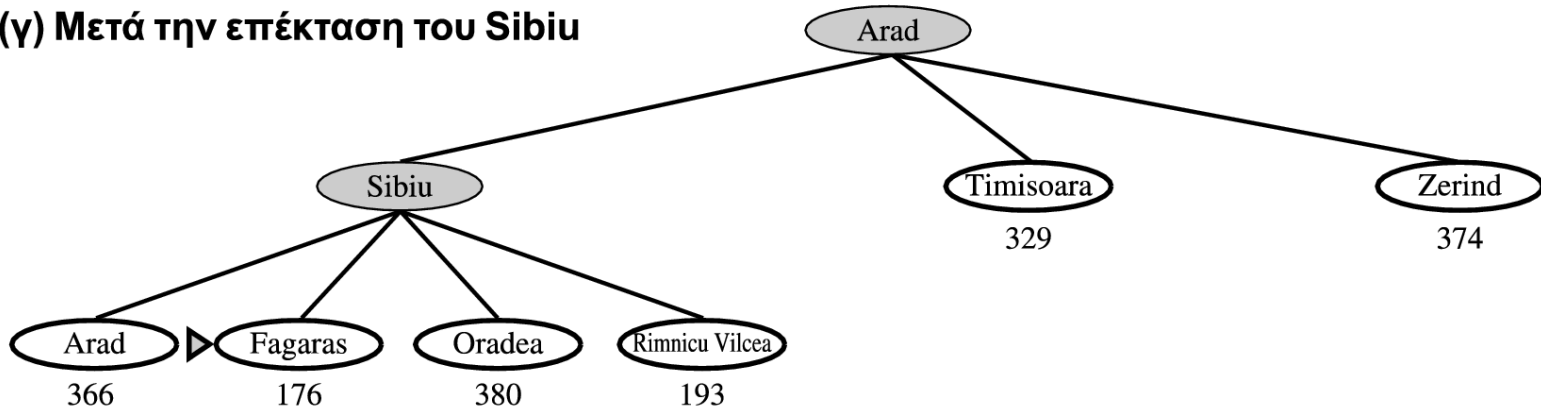


(β) Μετά την επέκταση του Arad

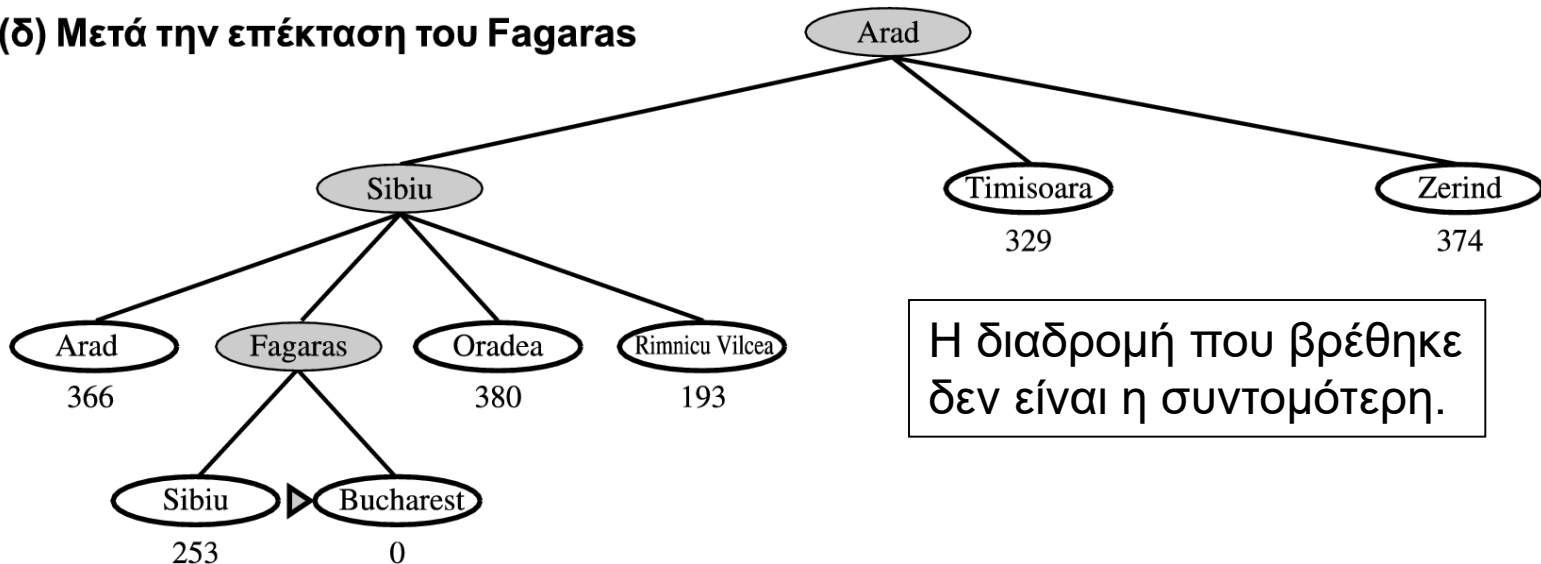


Αναζήτηση πρώτα στο καλύτερο (4/5)

(γ) Μετά την επέκταση του Sibiu



(δ) Μετά την επέκταση του Fagaras



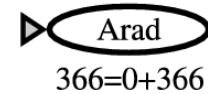
Αναζήτηση πρώτα στο καλύτερο (5/5)

- Επιρρεπής σε λανθασμένες εκτιμήσεις:
 - Μετάβαση από Iasi σε Fagaras
- Έχει ομοιότητα με την αναζήτηση πρώτα κατά βάθος.
 - Μπορεί να μπλέξει σε ατέρμονα κλαδιά.
- Χρονική, χωρική πολυπλοκότητα χειρότερης περίπτωσης: $O(b^m)$

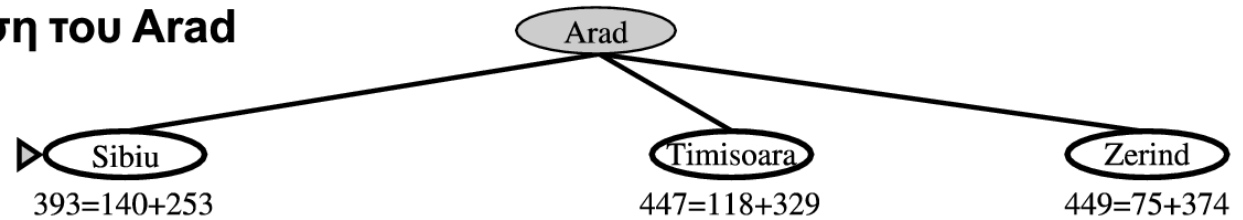
Αναζήτηση A^* (1/3)

- $f(n)=g(n)+h(n)$
- Παραδεκτός ευρετικός μηχανισμός $h(n)$: Δεν κάνει υπερκτιμήσεις.
 - π.χ. h_{SLD}
- Η αναζήτηση A^* χωρίς κλάδεμα επαναλαμβανόμενων καταστάσεων είναι βέλτιστη αν η $h(n)$ είναι παραδεκτή.

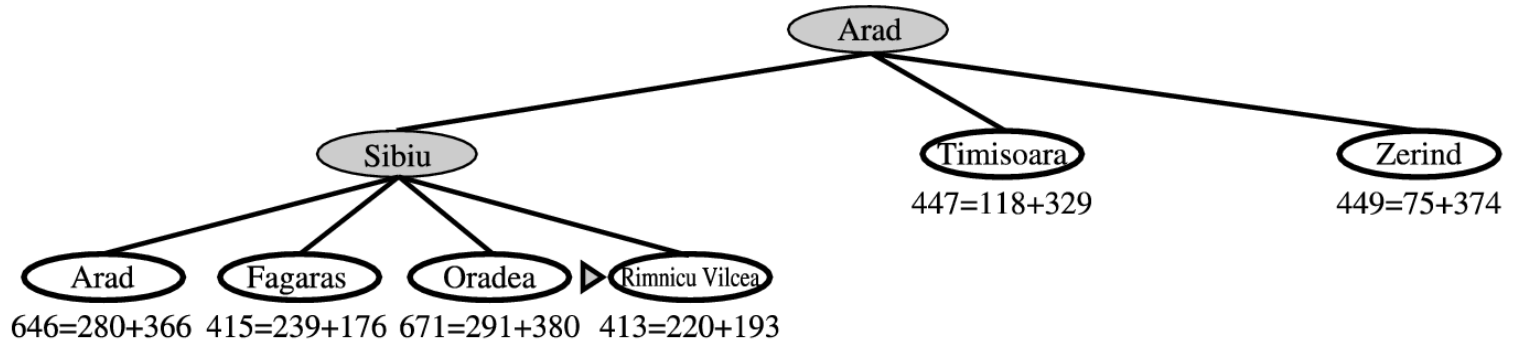
(α) Η αρχική κατάσταση



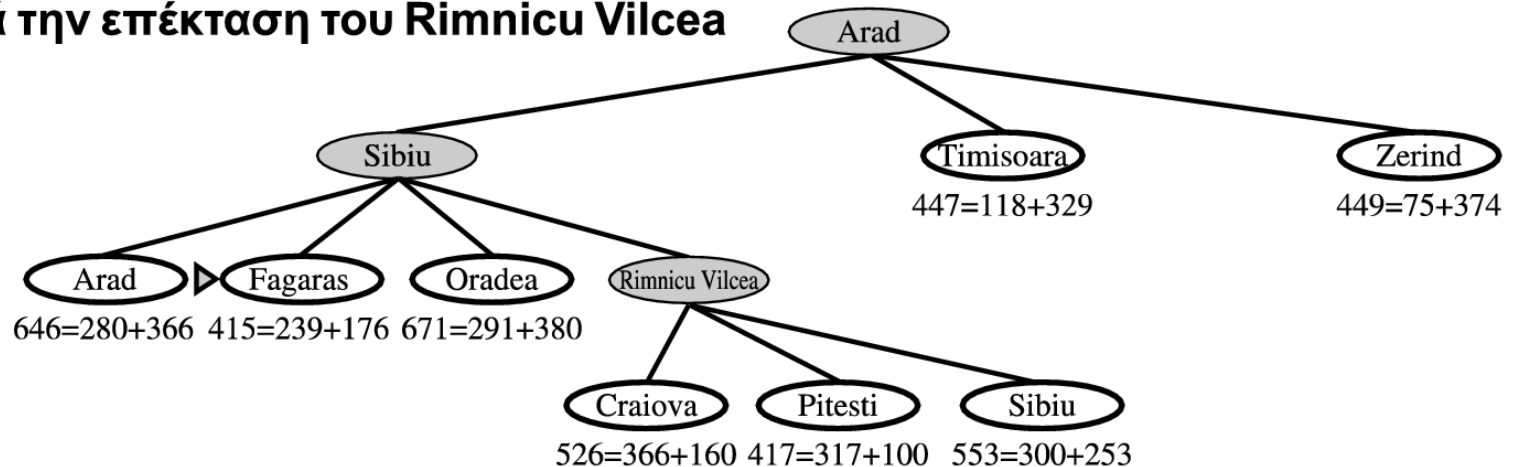
(β) Μετά την επέκταση του Arad



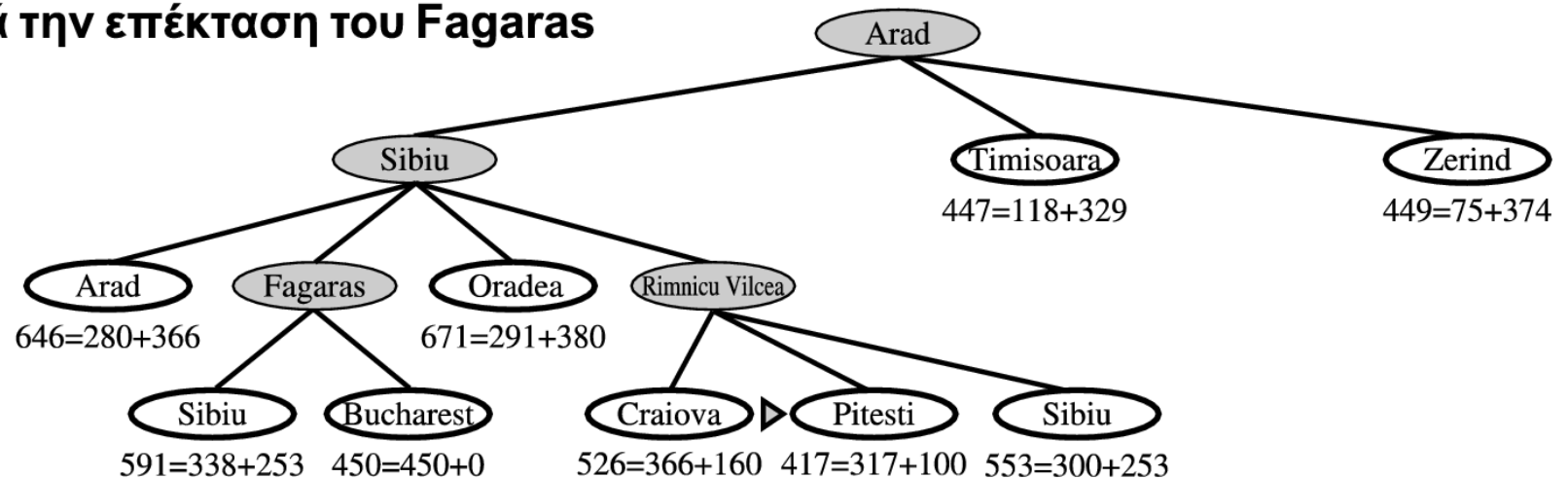
(γ) Μετά την επέκταση του Sibiu



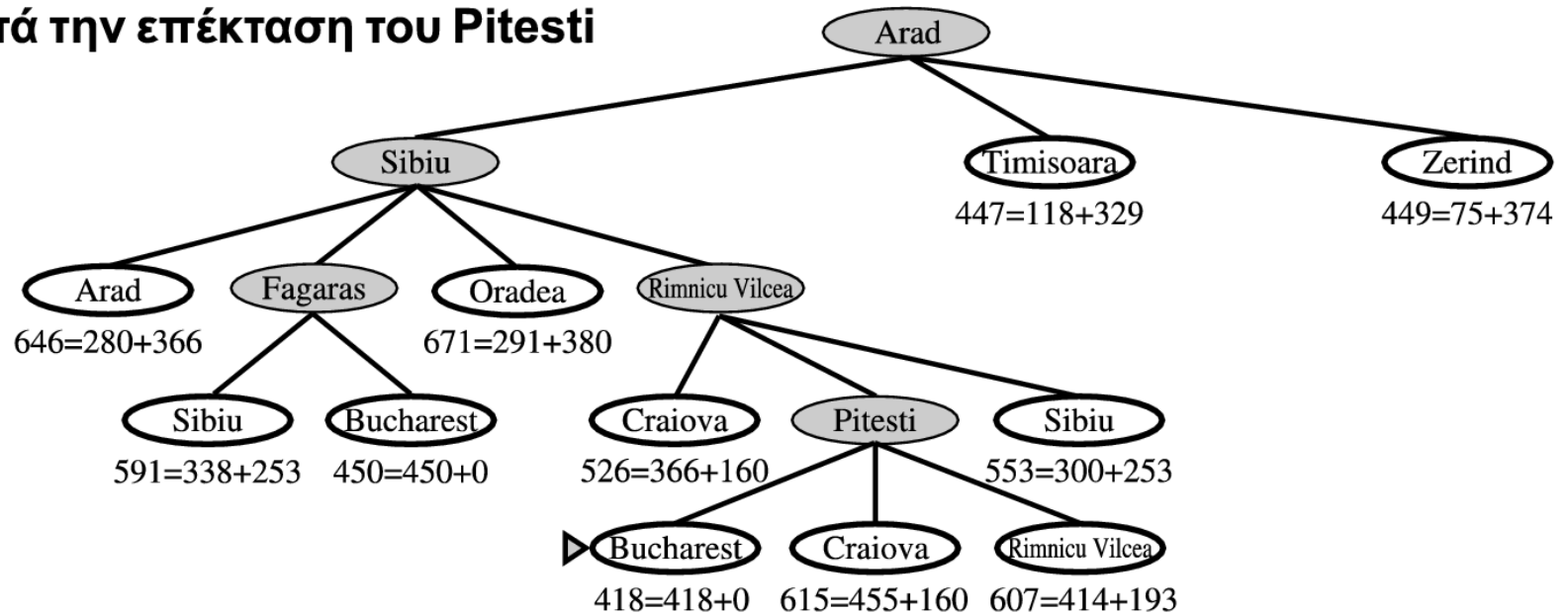
(δ) Μετά την επέκταση του Rimnicu Vilcea



(ε) Μετά την επέκταση του Fagaras



(στ) Μετά την επέκταση του Pitesti

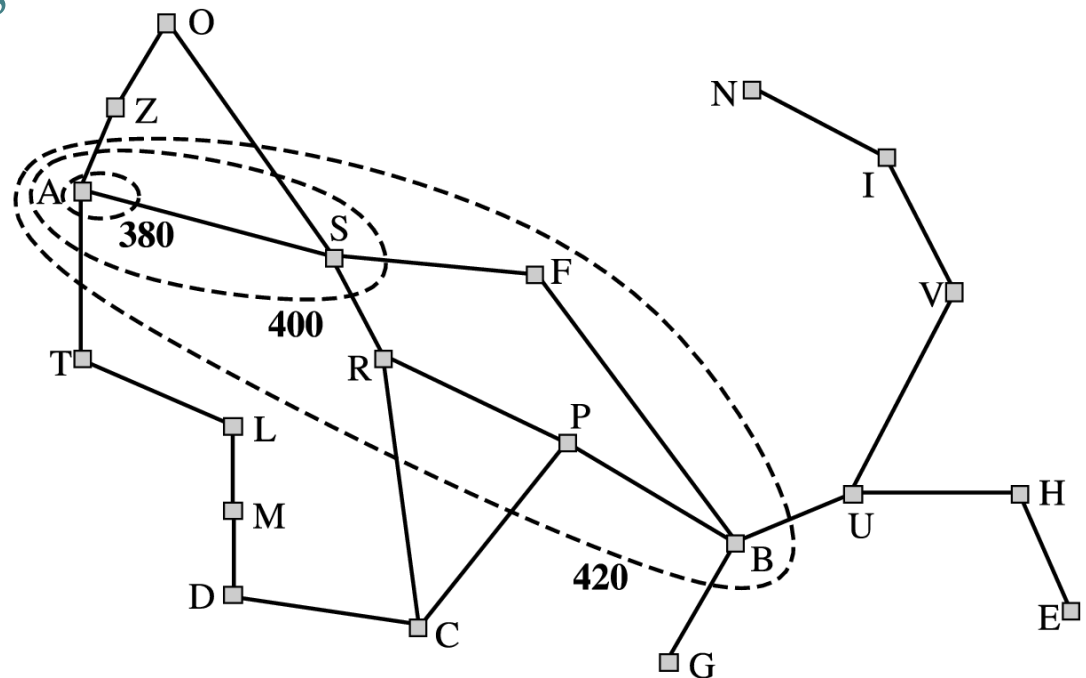


Συνεπείς ευρετικές συναρτήσεις

- Μια ευρετική συνάρτηση λέγεται συνεπής (consistent) ή μονοτονική, εάν ισχύει:
 - $h(n) \leq c(n, a, n') + h(n')$
- Η αναζήτηση A^* με κλάδεμα επαναλαμβανόμενων καταστάσεων, χωρίς να λαμβάνεται υπόψη εάν η νέα κατάσταση έχει μικρότερη τιμή $f(n)$ από την παλιά, είναι βέλτιστη αν η $h(n)$ είναι συνεπής.
- Μια συνεπής ευρετική συνάρτηση είναι και παραδεκτή.
- Οι περισσότερες παραδεκτές ευρετικές συναρτήσεις είναι και συνεπείς (π.χ. h_{SLD})

Ισοϋψείς καμπύλες

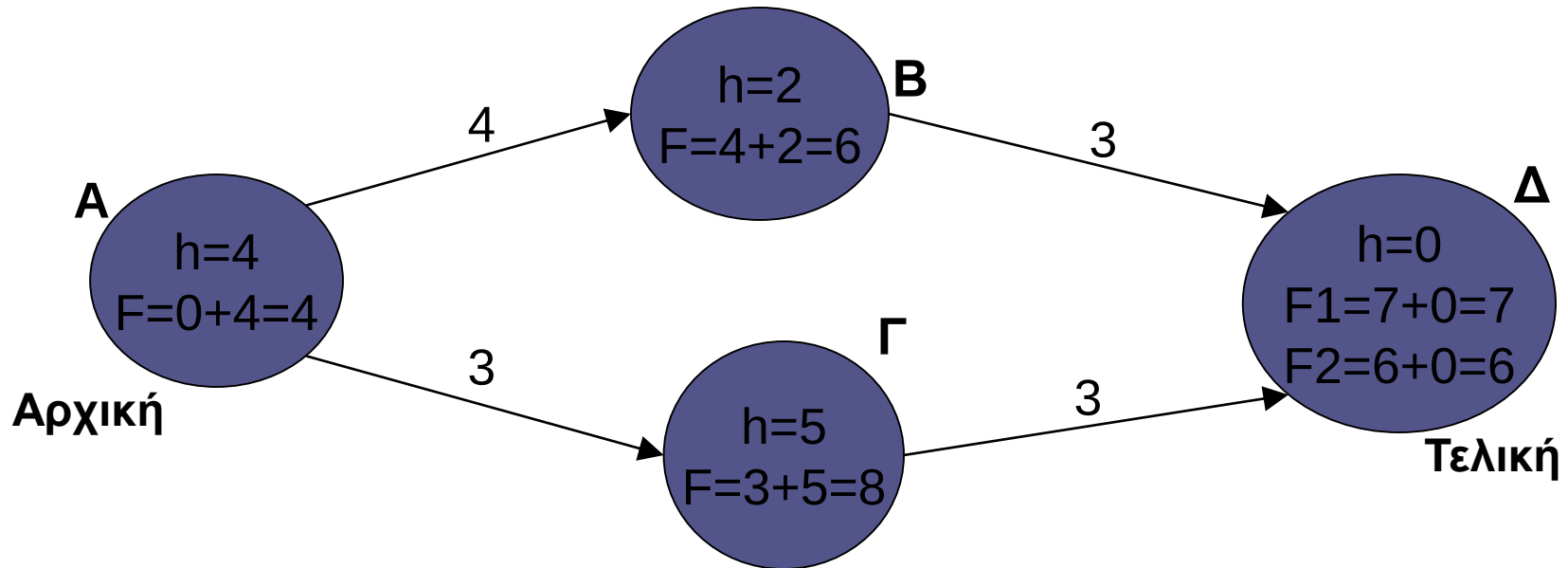
- Αν η $h(n)$ είναι συνεπής, τότε οι τιμές της $f(n)$ πάνω σε οποιαδήποτε διαδρομή είναι μη φθίνουσες.
 - Ισοϋψείς καμπύλες



Αποδοτικότητα A^*

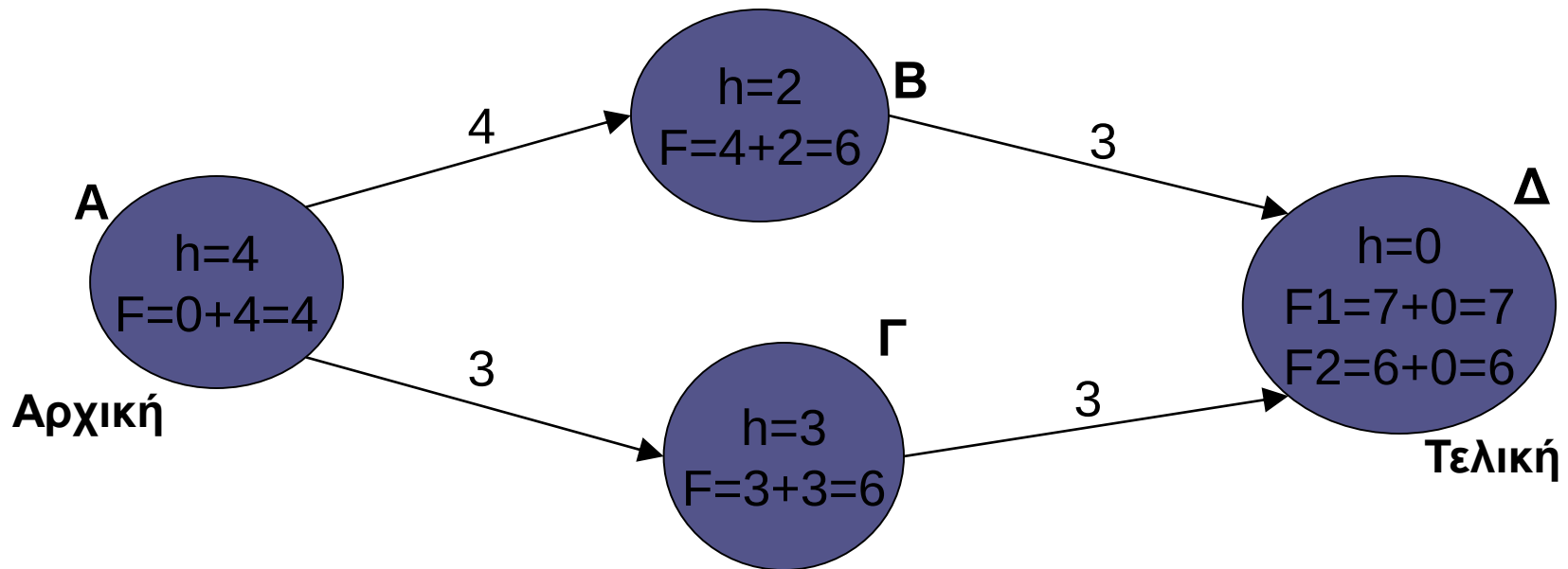
- Βέλτιστος και πλήρης
- Βέλτιστα αποδοτικός: Εξερευνά τις λιγότερες καταστάσεις από όλους τους βέλτιστους και πλήρεις αλγορίθμους.
- Μεγάλες (εκθετικές) απαιτήσεις σε χρόνο και (κυρίως) σε μνήμη.

Παράδειγμα: Μη-Παραδεκτή Συνάρτηση



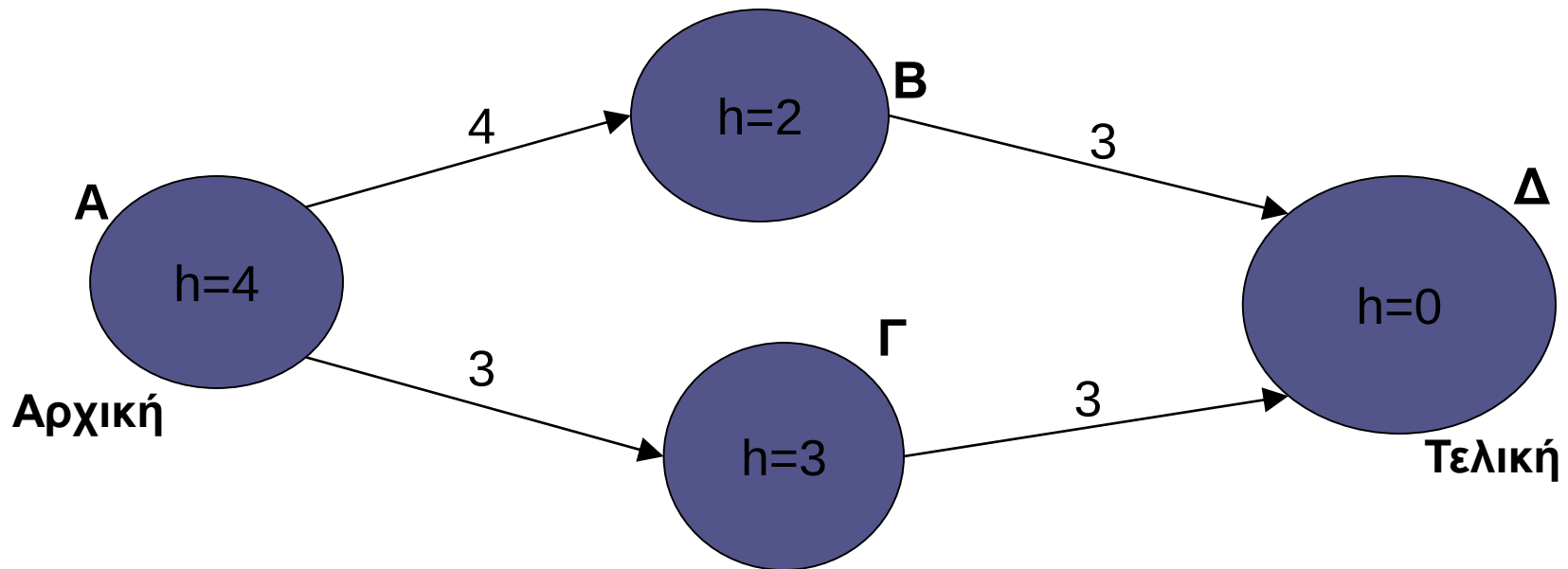
Μέτωπο αναζήτησης	Κλειστό σύνολο	Τρέχουσα κατάσταση	Παιδιά
A^4		A	$AB^6, A\Gamma^8$
$AB^6, A\Gamma^8$	A	AB^6	$AB\Delta^7$
$AB\Delta^7, A\Gamma^8$	A, AB	$AB\Delta^7$	Λύση

Παράδειγμα: Παραδεκτή Συνάρτηση



Μέτωπο αναζήτησης	Κλειστό σύνολο	Τρέχουσα κατάσταση	Παιδιά
A^4		A	$AB^6, A\Gamma^6$
$AB^6, A\Gamma^6$	A	AB^6	$AB\Delta^7$
$A\Gamma^6, AB\Delta^7$	A, AB	$A\Gamma^6$	$A\Gamma\Delta^6$
$A\Gamma\Delta^6, AB\Delta^7$	A, AΓ, AB	$A\Gamma\Delta^6$	Λύση

Παράδειγμα: Σύγκριση BestFS και A*



Λύση στο προηγούμενο παράδειγμα με χρήση BestFS:

Μέτωπο αναζήτησης	Κλειστό σύνολο	Τρέχουσα κατάσταση	Παιδιά
A		A	AB, AΓ
AB, AΓ	A	AB	ABΔ
ABΔ, AΓ	A, AB	ABΔ	Λύση

Ευρετική αναζήτηση περιορισμένης μνήμης

A series of horizontal lines of varying lengths and colors (teal, light blue, and white) extending from the left edge of the slide towards the right, positioned below the title.

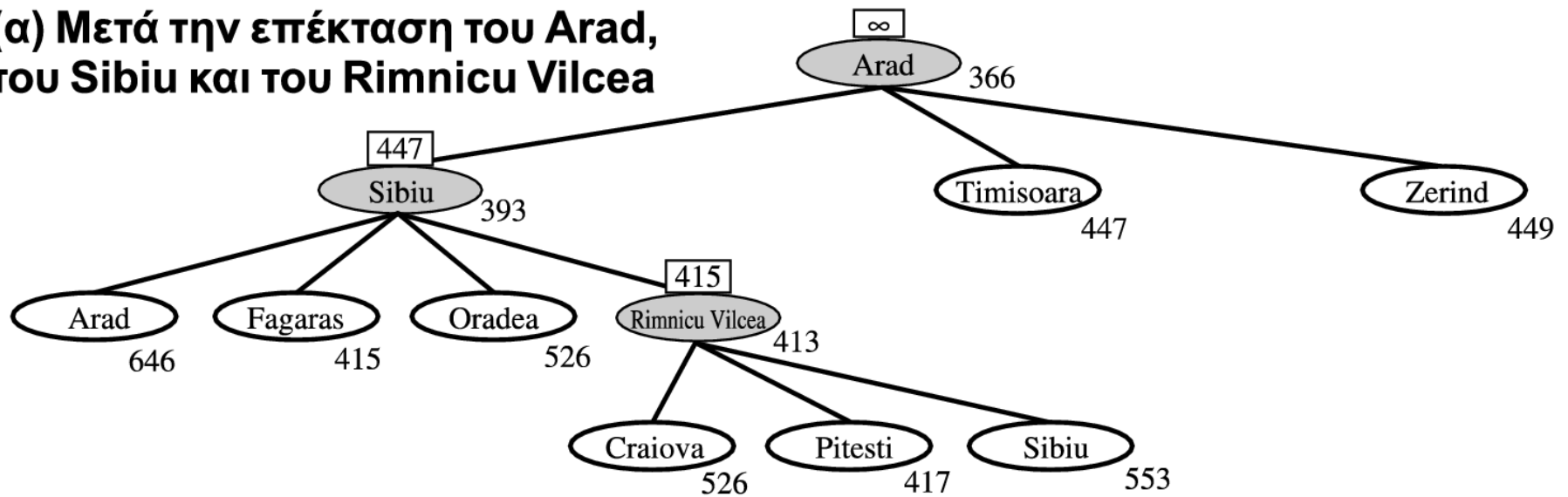
A* με επαναληπτική εκβάθυνση

- Iterative Deepening A* (IDA*)
- Επέκταση των κλαδιών πρώτα κατά βάθος.
- Όριο αποκοπής στη συνάρτηση $f(n) = g(n) + h(n)$
- Αν δεν βρεθεί λύση σε κάποιο κόστος, αυξάνεται το όριο αποκοπής στο κόστος του «φθηνότερου» κόμβου που κλαδεύτηκε.
- Δεν ανιχνεύει επαναλαμβανόμενες καταστάσεις.

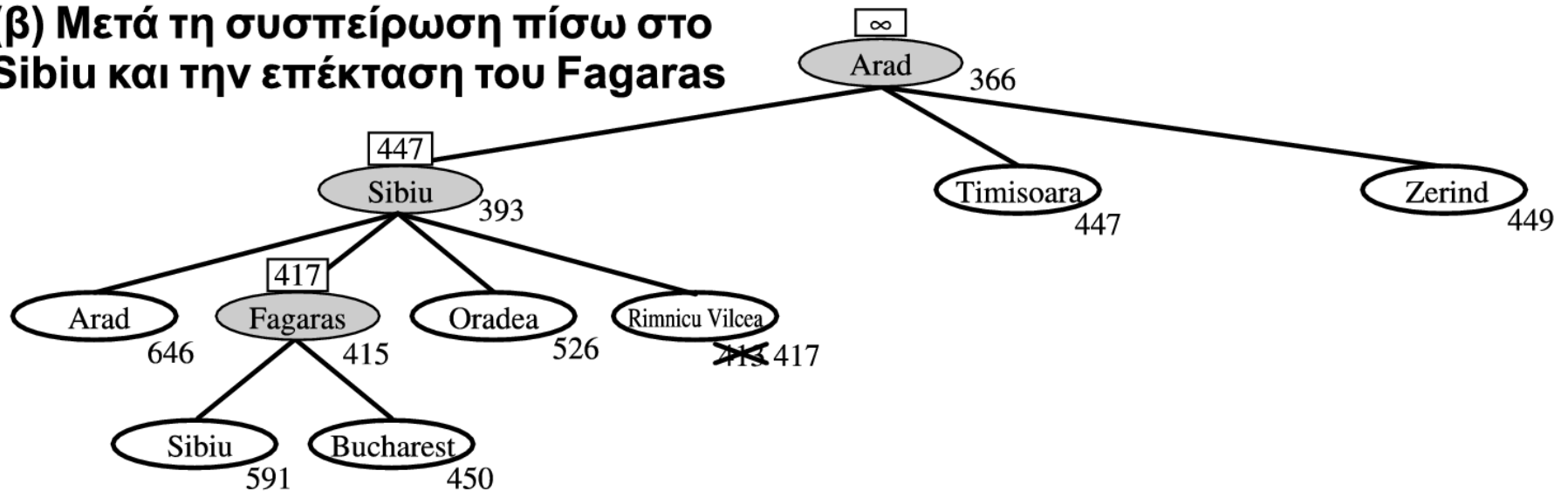
Αναδρομική αναζήτηση πρώτα στο καλύτερο (1/3)

- Recursive best-first search (RBFS)
- Σε κάθε κόμβο που επιλέγει να επεκτείνει, θυμάται την καλύτερη τιμή f των γειτονικών του κόμβων.
- Εάν δεν υπάρχει παιδί με τιμή f καλύτερη του βέλτιστου γειτονικού κόμβου, η αναζήτηση συνεχίζει από τον βέλτιστο γειτονικό, ενημερώνοντας τον τρέχοντα κόμβο με την τιμή του καλύτερου παιδιού του.
- Γραμμικές απαιτήσεις σε μνήμη.
- Δεν ανιχνεύει επαναλαμβανόμενες καταστάσεις.

(α) Μετά την επέκταση του Arad, του Sibiu και του Rimnicu Vilcea

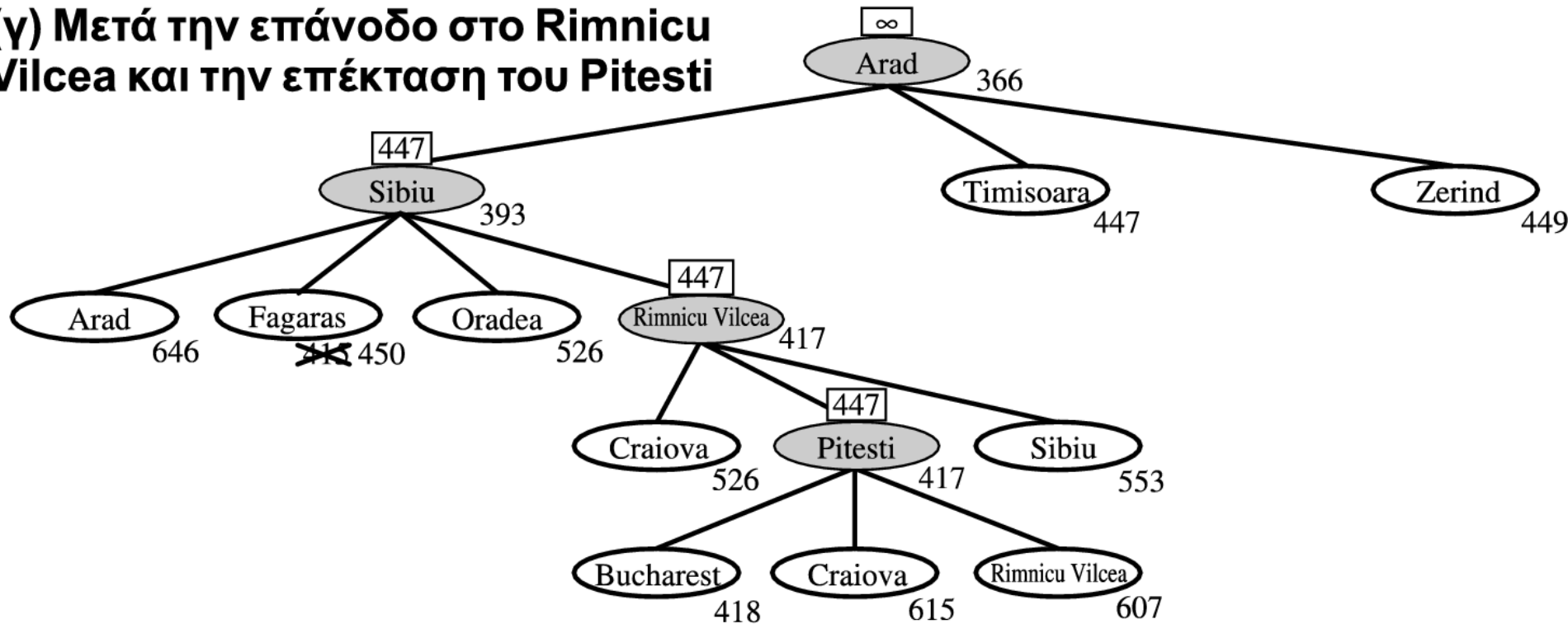


(β) Μετά τη συσπείρωση πίσω στο Sibiu και την επέκταση του Fagaras



Αναδρομική αναζήτηση πρώτα στο καλύτερο (3/3)

(γ) Μετά την επάνοδο στο Rimnicu
Vilcea και την επέκταση του Pitesti



Απλουστευμένος A^* περιορισμένης μνήμης

- Simplified memory-bounded A^* (SMA*)
- Ίδιος καταρχήν με τον A^* .
- Όταν γεμίσει η μνήμη, για κάθε νέο κόμβο διαγράφει έναν παλιό.
- Διαγράφει πάντα τον κόμβο με το χειρότερο $f(n)$.
 - Ο κόμβος αυτός θα είναι πάντα ένας κόμβος-φύλλο, μιας και για συνεπείς ευρετικούς μηχανισμούς τα παιδιά έχουν πάντα μεγαλύτερη τιμή f από τους γονείς.
 - Σε περίπτωση ισοβαθμίας, διαγράφεται ο παλαιότερος κόμβος.
- Η τιμή $f(n)$ του κόμβου που διαγράφηκε διατηρείται στον γονικό κόμβο.
- Πλήρης αλγόριθμος, εκτός και αν η βέλτιστη λύση δεν χωρά στη μνήμη!

Ευρετικές συναρτήσεις

A series of horizontal lines of varying lengths and colors (teal, light blue, and white) extending from the left side of the slide towards the right, positioned below the title.

Παζλ των 8 πλακιδίων

$h_1=8$

$h_2=18$

$h^*=26$

7	2	4
5		6
8	3	1

Αρχική κατάσταση

	1	2
3	4	5
6	7	8

Κατάσταση στόχου

- h_1 = ο αριθμός των πλακιδίων που δεν είναι στη θέση τους.
- h_2 = το άθροισμα των αποστάσεων των πλακιδίων από τις θέσεις προορισμού τους.
 - Απόσταση Manhattan
- Παραδεκτές και οι δύο.
- Η h_2 κυριαρχεί της h_1 (χωρίς να υπερεκτιμά)
 - Η h_2 επεκτείνει λιγότερους κόμβους

Δραστικός παράγοντας διακλάδωσης

- $N + 1 = 1 + b^* + (b^*)^2 + \dots + (b^*)^d$

d	Κόστος αναζήτησης			Δραστικός παράγοντας διακλάδωσης		
	IDS	$A^*(h_1)$	$A^*(h_2)$	IDS	$A^*(h_1)$	$A^*(h_2)$
2	10	6	6	2,45	1,79	1,79
4	112	13	12	2,87	1,48	1,45
6	680	20	18	2,73	1,34	1,30
8	6384	39	25	2,80	1,33	1,24
10	47127	93	39	2,79	1,38	1,22
12	3644035	227	73	2,78	1,42	1,24
14	—	539	113	—	1,44	1,23
16	—	1301	211	—	1,45	1,25
18	—	3056	363	—	1,46	1,26
20	—	7276	676	—	1,47	1,27
22	—	18094	1219	—	1,48	1,28
24	—	39135	1641	—	1,48	1,26

Επινόηση παραδεκτών μηχανισμών: Χαλαρά προβλήματα

- **Χαλαρά προβλήματα** (relexed problems): Απλοποίηση των κανόνων του προβλήματος.
 - Το κόστος μιας βέλτιστης λύσης για ένα χαλαρό πρόβλημα είναι ένας παραδεκτός ευρετικός μηχανισμός για το αρχικό πρόβλημα.
- **Κανόνας:**
 - Ένα πλακίδιο μπορεί να μετακινηθεί από το τετράγωνο A στο τετράγωνο B αν το A συνορεύει οριζόντια ή κάθετα με το B **και** το B είναι κενό
- **Χαλαροί κανόνες:**
 - Ένα πλακίδιο μπορεί να μετακινηθεί από το τετράγωνο A στο τετράγωνο B αν το A συνορεύει με το B. ($\rightarrow h_2$)
 - Ένα πλακίδιο μπορεί να μετακινηθεί από το τετράγωνο A στο τετράγωνο B αν το B είναι κενό.
 - Ένα πλακίδιο μπορεί να μετακινηθεί από το τετράγωνο A στο τετράγωνο B. ($\rightarrow h_1$)

Επινόηση παραδεκτών μηχανισμών: Υποπροβλήματα

- Βάσεις δεδομένων προτύπων (pattern databases)
 - Αποθήκευση λύσεων υποπροβλημάτων.

*	2	4
*		*
*	3	1

Αρχική κατάσταση

	1	2
3	4	*
*	*	*

Κατάσταση στόχου

- Βάσεις δεδομένων ξένων προτύπων (disjoint pattern databases)
 - Ξεχωριστές βάσεις δεδομένων, πρόσθεση των εκτιμήσεών τους, αφαίρεση (πλήρης) των κοινών κινήσεων.

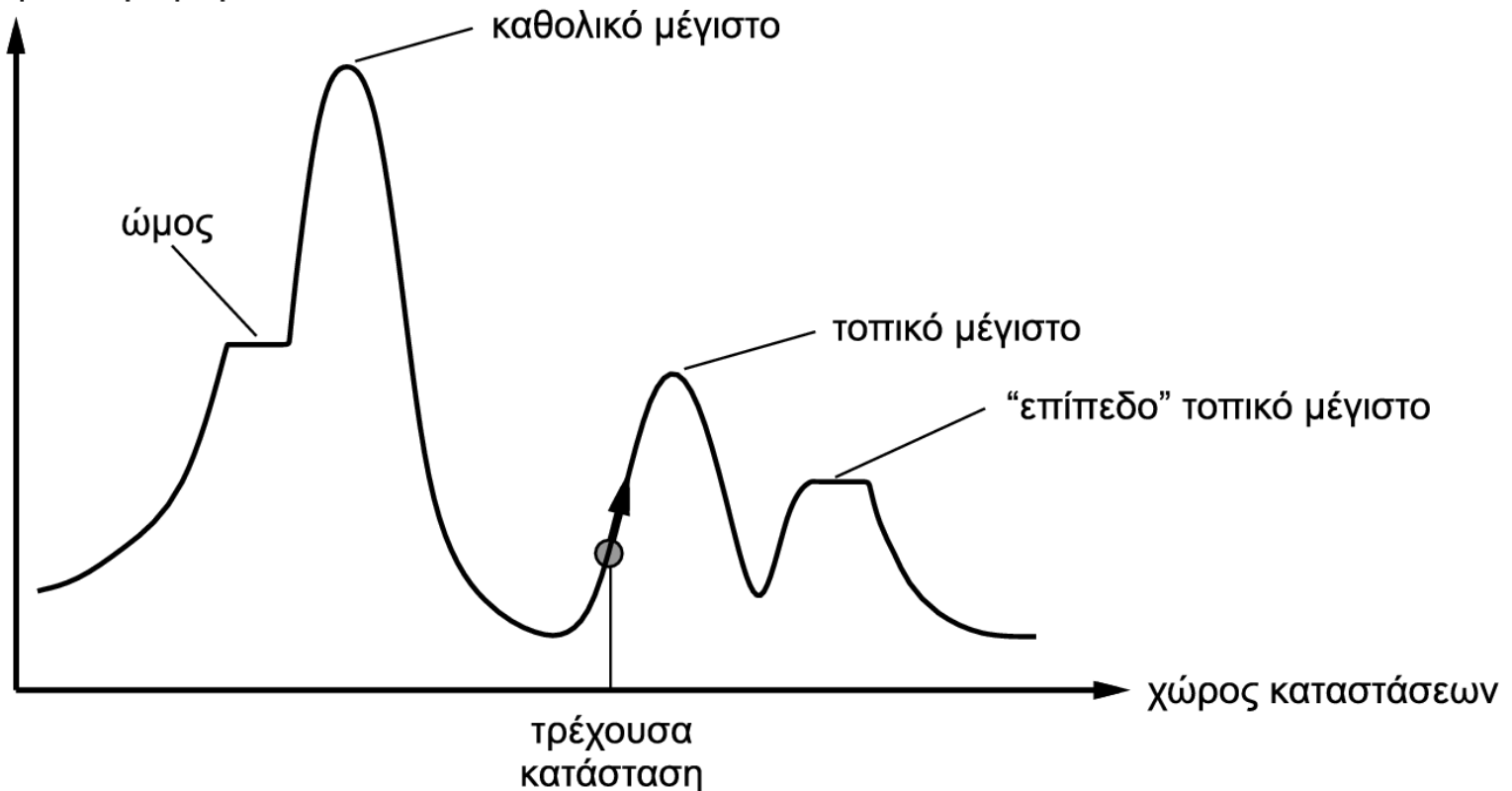
Αλγόριθμοι τοπικής αναζήτησης και προβλήματα βελτιστοποίησης

A series of horizontal lines in teal and light blue colors, with varying lengths and thicknesses, extending from the left edge of the slide towards the right, positioned below the title.

Τοπική αναζήτηση

- Προβλήματα βελτιστοποίησης, αντικειμενική συνάρτηση
 - Τοπίο χώρου καταστάσεων

αντικειμενική συνάρτηση

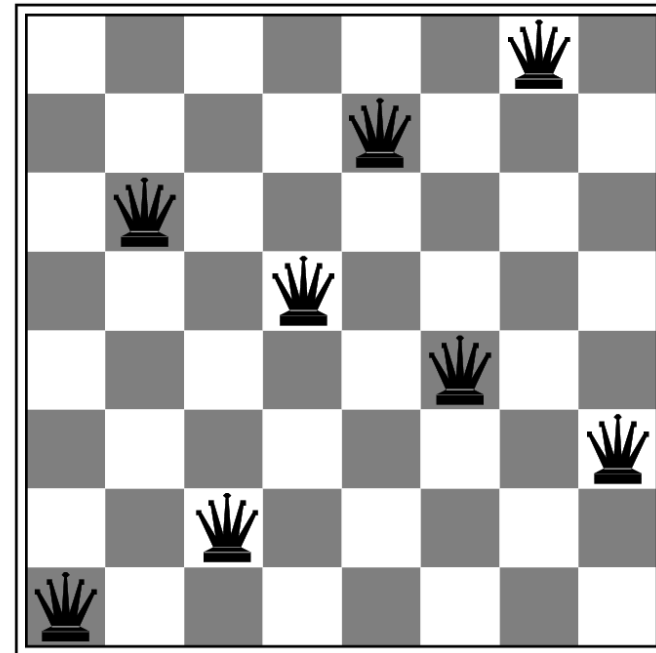


Αναρρίχηση λόφων

- Μετακινείται συνεχώς στην καλύτερη γειτονική κατάσταση.
 - Συνήθως διατύπωση με πλήρεις καταστάσεις

18	12	14	13	13	12	14	14
14	16	13	15	12	14	12	16
14	12	18	13	15	12	14	14
15	14	14	♔	13	16	13	16
♔	14	17	15	♔	14	16	16
17	♔	16	18	15	♔	15	♔
18	14	♔	15	15	14	♔	16
14	14	13	17	12	14	12	18

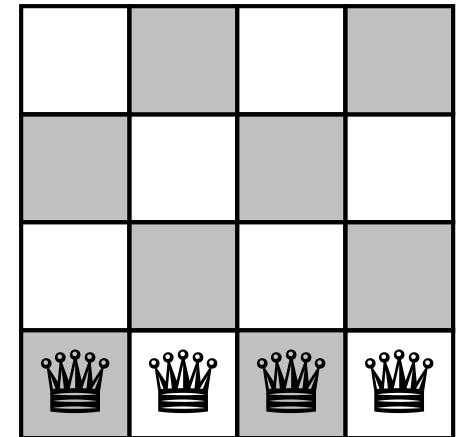
Τρέχουσα τιμή $h=17$
(α)



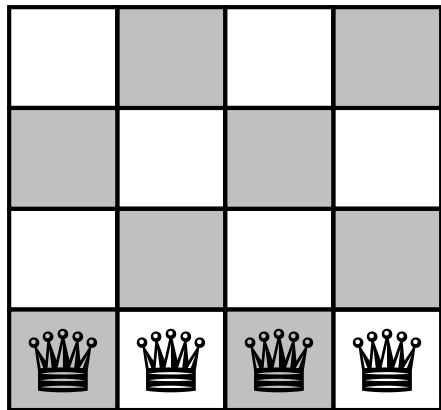
Τοπικό ελάχιστο με $h=1$
(β)

Παράδειγμα με 4 βασίλισσες (1/2)

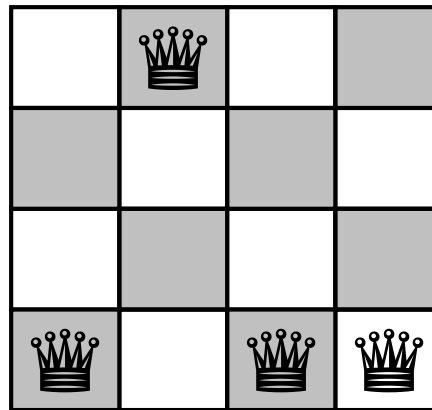
- Σε κάθε στήλη θα υπάρχει μία μόνο βασίλισσα.
- Ξεκινάμε με όλες τις βασίλισσες στην κάτω γραμμή.
- Σε κάθε βήμα μπορούμε να μετακινήσουμε μια βασίλισσα σε μια άλλη θέση στη στήλη της, άρα οι δυνατές κινήσεις είναι $4 \times 3 = 12$.
- Χρησιμοποιούμε ως ευρετική συνάρτηση το πόσες απειλές υπάρχουν κάθε φορά (όλες οι απειλές είναι διπλές, εμείς τις μετράμε ως μία απειλή κάθε φορά).



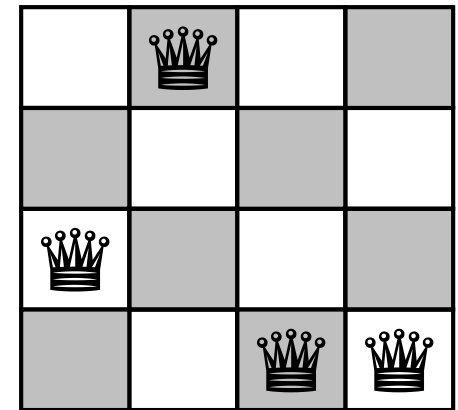
Παράδειγμα με 4 βασίλισσες (2/2)



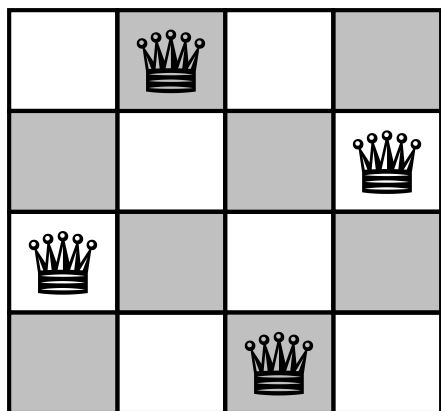
6



3



1



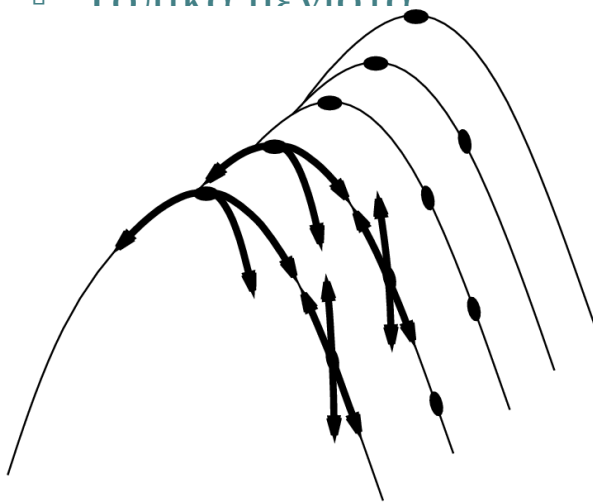
0

- Στο συγκεκριμένο παράδειγμα είμασταν τυχεροί, μιας και όλες οι επιλογές μας βγήκαν σωστές.

Προβλήματα αναρρίχησης λόφων

- Προβλήματα:

- Τοπικά μέγιστα



- Αντιμετώπιση:

- Πλάγιες κινήσεις
 - Στοχαστική αναζήτηση
 - Επιλέγει τυχαία από τις «καλές» κινήσεις
 - Αναρρίχηση λόφων με την πρώτη επιλογή
 - Τυχαίες επανεκκινήσεις (πλήρης αλγόριθμος)

- Οροπέδια

Προσομοιωμένη ανόπτηση

(Simulated annealing)

- Προ-επιλέγει τυχαία μια κίνηση.
- Αν αυτή βελτιώνει τα πράγματα, γίνεται αποδεκτή.
- Αν τα χειροτερεύει, γίνεται αποδεκτή με πιθανότητα $P = e^{\Delta E/T}$,
 - $\Delta E < 0$, η χειροτέρευση
 - $T > 0$, «θερμοκρασία»
- Η θερμοκρασία μειώνεται με την πάροδο του χρόνου.

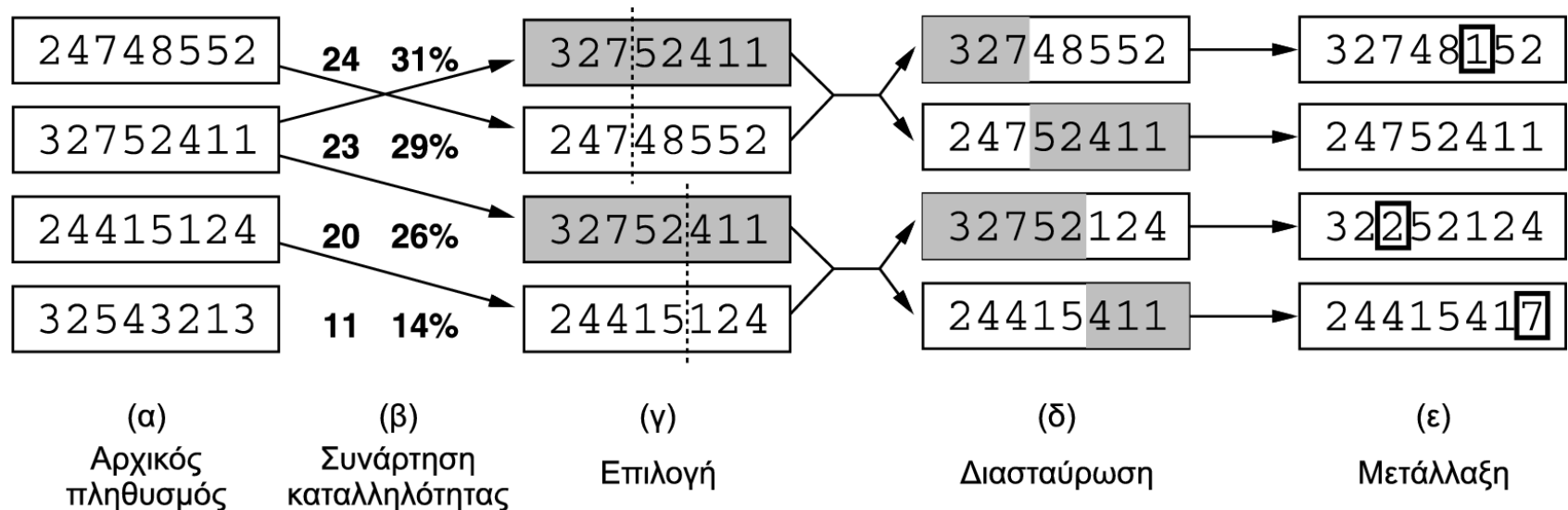
Τοπική ακτινική αναζήτηση

(local beam search)

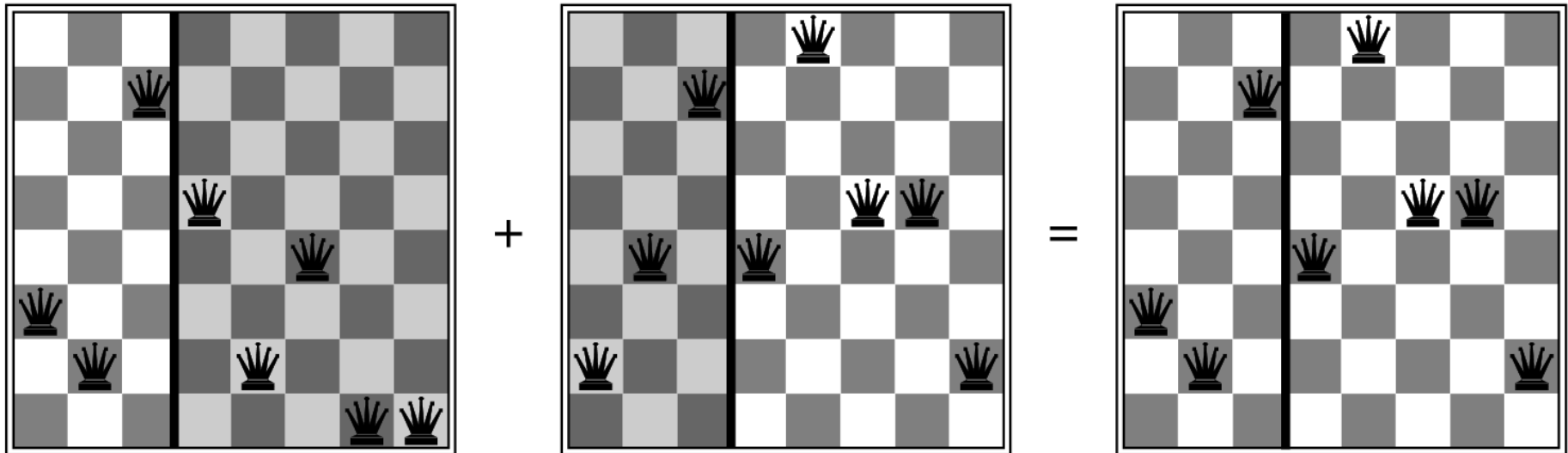
- Επιλέγονται k τυχαίες αρχικές καταστάσεις.
- Εκτελούνται k αναζητήσεις παράλληλα.
- Σε κάθε βήμα επιλέγονται τα k καλύτερα παιδιά από όλα τα παιδιά του προηγούμενου βήματος.
- Κίνδυνος συγκέντρωσης όλων των αναζητήσεων γύρω από το ίδιο κλαδί.
- Αντιμετώπιση: Στοχαστική επιλογή των k παιδιών.

Γενετικοί αλγόριθμοι (1/2)

- Οι διάδοχες καταστάσεις παράγονται με το συνδυασμό δύο γονικών καταστάσεων, και όχι με την τροποποίηση μιας μεμονωμένης κατάστασης.



Γενετικοί αλγόριθμοι (2/2)



Τοπική αναζήτηση σε συνεχείς χώρους

A series of horizontal lines of varying lengths and colors (teal, light blue, and white) extending from the left edge of the slide towards the right, positioned below the title.

Συνεχείς χώροι

- Άπειρος αριθμών διάδοχων καταστάσεων
- Παράδειγμα: Τοποθέτηση 3 αεροδρομίων στη Ρουμανία με ελαχιστοποίηση του αθροίσματος των αποστάσεων των πόλεων.
 - Χώρος 6 διαστάσεων
 - Αντικειμενική συνάρτηση: $f(x_1, y_1, x_2, y_2, x_3, y_3)$
- Διακριτοποίηση, βήμα $\pm\delta$

Πλέον απότομη ανάβαση

(steepest-ascent hill climbing)

- Κλίση:

$$\nabla f = \left(\frac{\partial f}{\partial x_1}, \frac{\partial f}{\partial y_1}, \frac{\partial f}{\partial x_2}, \frac{\partial f}{\partial y_2}, \frac{\partial f}{\partial x_3}, \frac{\partial f}{\partial y_3} \right)$$

- Πλέον απότομη ανάβαση:

- $\mathbf{x} \leftarrow \mathbf{x} + \alpha \nabla f(\mathbf{x})$

- Εμπειρική κλίση
- Ευθύγραμμη αναζήτηση

Πράκτορες online αναζήτησης και άγνωστα περιβάλλοντα

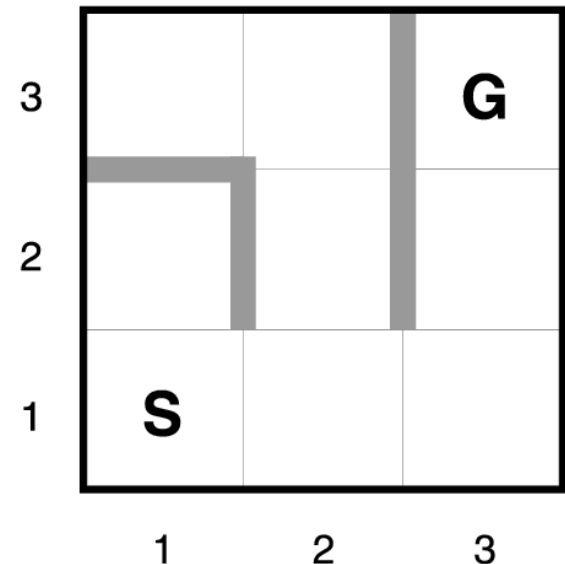
A series of horizontal lines in teal and light blue colors, with varying lengths and thicknesses, extending from the left edge of the slide towards the right, positioned below the main title.

Online αναζήτηση

- Επεξεργασία δεδομένων πριν γίνει διαθέσιμο το σύνολό τους
- Διαπλοκή (interleaving) υπολογισμών και ενεργειών
- Κατάλληλη για πεδία:
 - Δυναμικά
 - Ημιδυναμικά
 - Στοχαστικά
- Μια offline αναζήτηση θα έπρεπε να καταστρώνει ένα εκθετικά μεγάλο πλάνο ενδεχομένων που να παίρνει υπόψη όλα όσα είναι δυνατό να συμβούν.
- Προβλήματα εξερεύνησης: Οι καταστάσεις του κόσμου και τα αποτελέσματα των ενεργειών είναι άγνωστα.

Προβλήματα online αναζήτησης (1/2)

- Ο πράκτορας πρέπει να εκτελέσει τις ενέργειες για να μάθει τα αποτελέσματά τους!
- Γνώση πράκτορα:
 - Ξέρει ποιες είναι οι ενέργειές του
 - Γνωρίζει το κόστος κάθε βήματος
 - Μπορεί να καταλάβει εάν πέτυχε το στόχο

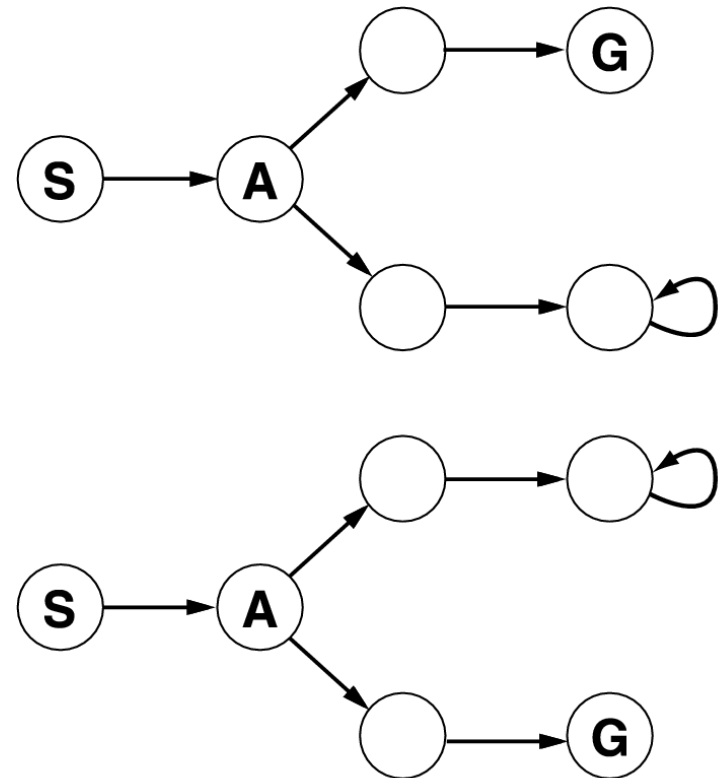


Προβλήματα online αναζήτησης (2/2)

- Παραδοχές (απλουστευτικές):
 - Θυμάται τις καταστάσεις που επισκέφθηκε
 - Αιτιοκρατικές ενέργειες
 - Γνωρίζει μια παραδεκτή ευρετική συνάρτηση (π.χ. απόσταση Manhattan)
- Στόχος: Να φθάσει στον προορισμό
- Κόστος: Τα βήματα που εκτέλεσε
- Ανταγωνιστικός λόγος (competitive ratio):
 - Πραγματικό κόστος / βέλτιστο κόστος

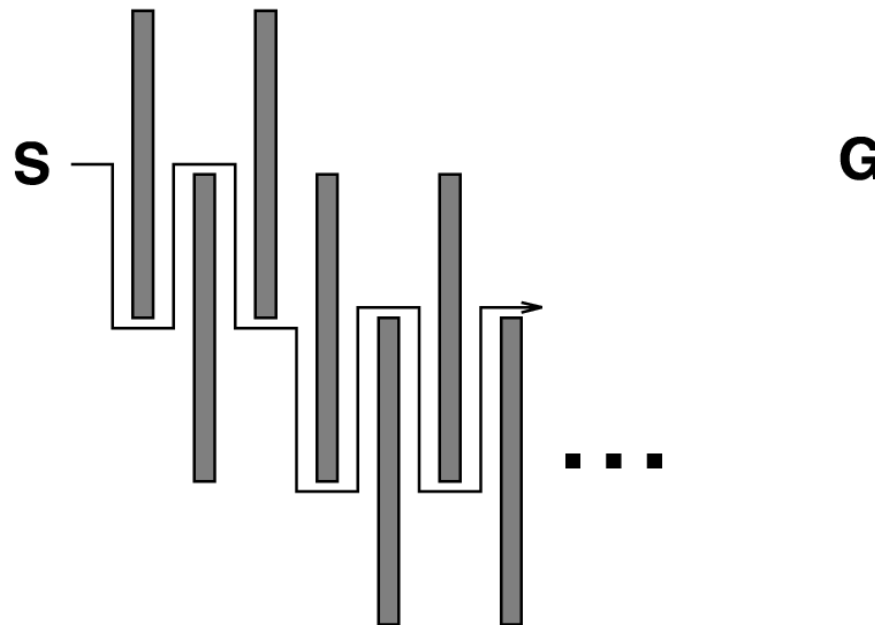
Αδιέξοδα

- Εάν οι ενέργειες δεν αντιστρέφονται, ο πράκτορας μπορεί να παγιδευτεί.
 - Κανένας αλγόριθμος δεν μπορεί να αποφεύγει τα αδιέξοδα σε όλους τους χώρους καταστάσεων.



Ασφαλώς εξερευνηήσιμοι χώροι

- Από κάθε προσπελάσιμη κατάσταση είναι προσπελάσιμη κάποια κατάσταση στόχου.
 - Ωστόσο το κόστος μπορεί να είναι μη-φραγμένο.

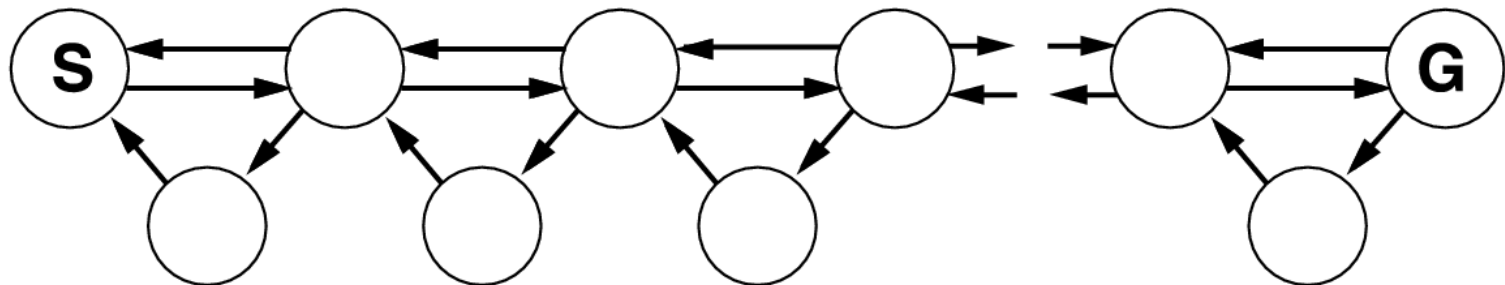


Πράκτορες online αναζήτησης

- Έπειτα από κάθε ενέργεια, ένας πράκτορας online προσλαμβάνει μια αντίληψη που του λέει σε ποια κατάσταση έχει φτάσει.
 - Μπορεί να βελτιώσει το χάρτη του περιβάλλοντος που έχει.
- Επέκταση κόμβων με τοπική προτεραιότητα.
 - Δεν μπορούν να χρησιμοποιηθούν αλγόριθμοι που εκτελούν άλματα (όπως π.χ. ο A^*).
- Εξερεύνηση πρώτα κατά βάθος
 - Πρόβλημα οι επιστροφές, αναστρέψιμες ενέργειες
- Επαναληπτική εκβάθυνση

Τοπική online αναζήτηση

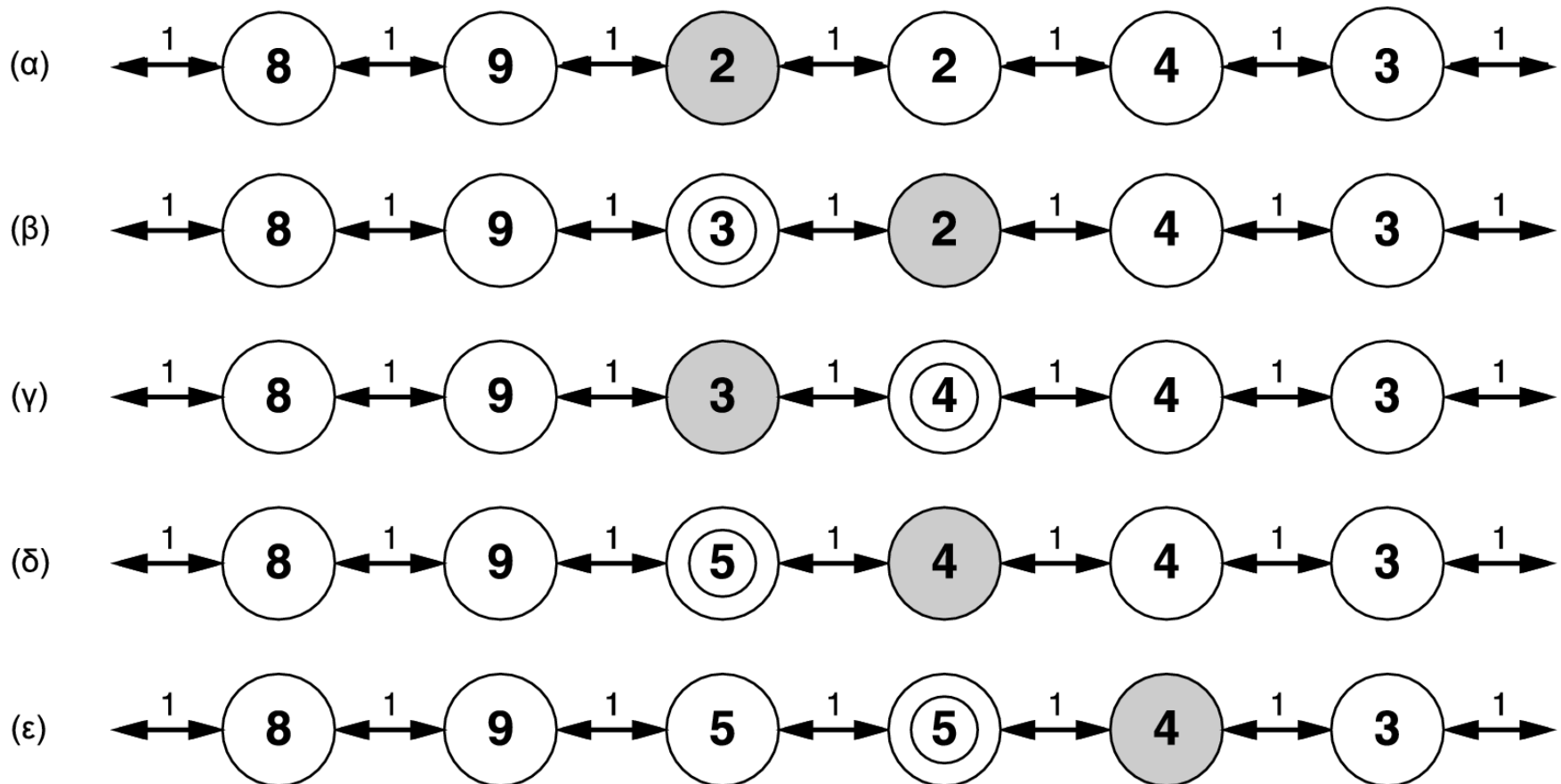
- Τοπική αναζήτηση αναρρίχησης λόφων
 - Πρόβλημα με τα οροπέδια, δεν μπορούμε να εκτελέσουμε τυχαίες επανεκκινήσεις.
- Τυχαίος περίπατος (random walk)
 - Τυχαία επιλογή βημάτων, αποφυγή επανάληψης
 - Στο παρακάτω σχήμα, η επιστροφή προς το S είναι πάντα 2 φορές πιο πιθανή από την πορεία προς το G. Εκθετικό κόστος.



A* πραγματικού χρόνου που μαθαίνει (1/3) (Learning real-time A*, LRTA*)

- Ο πράκτορας μεταβαίνει πάντα στην καλύτερη γειτονική κατάσταση.
- Εάν πρόκειται για κατάσταση που επισκέπτεται ο πράκτορας για πρώτη φορά, αυτή βαθμολογείται με την ευρετική συνάρτηση, αλλιώς διατηρεί την όποια βαθμολογία της.
 - Αισιοδοξία υπό αβεβαιότητα
- Η κατάσταση από την οποία έγινε η μετάβαση βαθμολογείται με το άθροισμα του κόστους μετάβασης και της βαθμολογίας της νέας κατάστασης.
- Ο πράκτορας πρέπει να θυμάται τις νέες βαθμολογίες, γιατί αυτές δεν προκύπτουν πλέον από την ευρετική συνάρτηση.

A^* πραγματικού χρόνου που μαθαίνει (2/3) (Learning real-time A^* , LRTA*)



A* πραγματικού χρόνου που μαθαίνει (3/3) (Learning real-time A*, LRTA*)

- Είναι πλήρης για πεπερασμένα, ασφαλώς εξερευνήσιμα περιβάλλοντα.
 - Δεν είναι πλήρης για άπειρους χώρους καταστάσεων.
- Πολυπλοκότητα: $O(n^2)$, όπου n το πλήθος των καταστάσεων.