

Επίλυση Προβλημάτων με Αναζήτηση

Πράκτορες (μέθοδοι) επίλυσης προβλημάτων

- Αποφασίζουν τι θα κάνουν βρίσκοντας ακολουθίες ενεργειών που οδηγούν σε επιθυμητές καταστάσεις.
- Διατύπωση στόχου: Σύνολο καταστάσεων του κόσμου
 - Αποτελεί, με βάση τις τρέχουσες περιστάσεις και το μέτρο απόδοσης του πράκτορα, το πρώτο βήμα για την επίλυση των προβλημάτων
- Διατύπωση προβλήματος: Η διαδικασία με την οποία αποφασίζεται ποιες ενέργειες και καταστάσεις θα πρέπει να εξετάζονται με δεδομένο ένα στόχο.
 - Επιλογή επιπέδου λεπτομέρειας
 - **Αφαίρεση**: παράλειψη λεπτομερειών από μια αναπαράσταση

Πράντορες (μέθοδοι) επίλυσης προβλημάτων

- Κάθε μία από τις «καταστάσεις» που εξετάζει ένας πράντορας (ανάλογα με το επίπεδο αφαίρεσης) στην πραγματικότητα αντιστοιχεί σε ένα μεγάλο σύνολο καταστάσεων του πραγματικού κόσμου.
 - Μια κατάσταση του πραγματικού κόσμου καθορίζει όλες τις απόψεις της πραγματικότητας
 - Υπάρχει διάκριση μεταξύ των καταστάσεων επίλυσης προβλημάτων και των καταστάσεων του κόσμου

Πράκτορες (μέθοδοι) επίλυσης προβλημάτων

- Ένας πράκτορας που έχει στη διάθεσή του πολλές άμεσες επιλογές άγνωστης αξίας μπορεί να αποφασίζει τι να κάνει εξετάζοντας πρώτα διάφορες δυνατές ακολουθίες ενεργειών που οδηγούν σε καταστάσεις γνωστής αξίας και μετά επιλέγοντας την καλύτερη ακολουθία

Πράκτορες (μέθοδοι) επίλυσης προβλημάτων

- Φάσεις επίλυσης προβλημάτων
 - Διατύπωση
 - Αναζήτηση
 - Η διαδικασία εύρεσης μιας ακολουθίας ενεργειών που οδηγεί σε κατάσταση γνωστής αξίας
 - Λύση
 - Ένας αλγόριθμος αναζήτησης παίρνει ως είσοδο ένα πρόβλημα και επιστρέφει μια «λύση» με τη μορφή μιας ακολουθίας ενεργειών
 - Εκτέλεση
 - Πραγματοποίηση των ενεργειών που υποδεικνύονται από τη λύση

Χαρακτηριστικά του περιβάλλοντος

■ Στατικό

- Δε συμβαίνουν αλλαγές στο περιβάλλον (κατά τη διατύπωση και επίλυση του προβλήματος)

■ Παρατηρήσιμο

- Η αρχική κατάσταση είναι γνωστή

■ Διακριτό

- Οι εναλλακτικοί τρόποι ενέργειας μπορούν να απαριθμηθούν

■ Αιτιοκρατικό

- Οι λύσεις των προβλημάτων είναι μοναδικές ακολουθίες ενεργειών και επομένως δε μπορούν να χειριστούν οποιαδήποτε απροσδόκητα συμβάντα

Προβλήματα και Λύσεις

- Συνιστώσες προβλήματος:

- **Αρχική κατάσταση**

- Η κατάσταση από την οποία ξεκινά ο πράκτορας

- **Συνάρτηση διαδόχων**

- Περιγράφει τις δυνατές ενέργειες που μπορεί να εκτελέσει από μια συγκεκριμένη κατάσταση ο πράκτορας

- **Χώρος καταστάσεων**

- Το σύνολο των καταστάσεων που είναι προσπελάσιμες από την αρχική κατάσταση
 - Ορίζεται έμμεσα από την αρχική κατάσταση και τη συνάρτηση διαδόχων
 - Αναπαράσταση με γράφημα

Προβλήματα και Λύσεις

■ Συνιστώσες προβλήματος:

■ Διαδρομή (στο χώρο καταστάσεων)

- Μια ακολουθία καταστάσεων που συνδέονται με μια ακολουθία ενεργειών

■ Έλεγχος στόχου

- Προσδιορίζει αν μια δεδομένη κατάσταση είναι κατάσταση στόχου
- Περιπτώσεις στόχων:
 - Μία κατάσταση (π.χ. να πάμε στην Αθήνα)
 - Ρητή απαρίθμηση συνόλου καταστάσεων (π.χ. να πάμε στην Ευρώπη)
 - Περιγραφή ιδιοτήτων (π.χ. να γίνει ματ στο σιάι)

Προβλήματα και Λύσεις

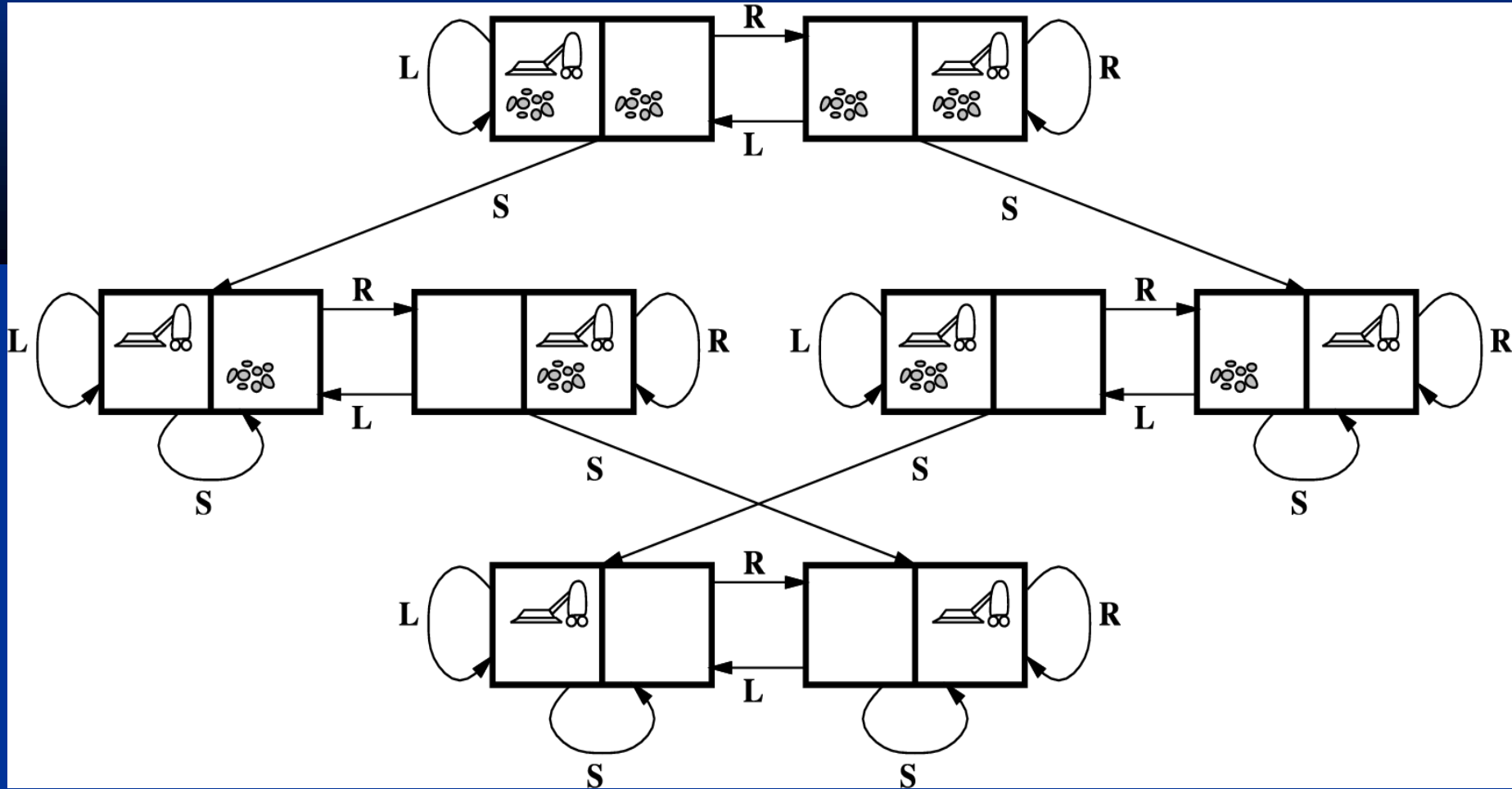
- Συνιστώσες προβλήματος:
 - **Κόστος διαδρομής**
 - Ένα αριθμητικό κόστος που αποδίδεται σε κάθε διαδρομή από μια αντίστοιχη συνάρτηση κόστους
 - Αρκετές φορές ισούται με το άθροισμα από τα κόστη των μεμονωμένων ενεργειών κατά μήκος της διαδρομής
 - Κόστος βήματος: $c(x,a,y)$
- Λύση (ενός προβλήματος)
 - Διαδρομή από την αρχική κατάσταση σε μια κατάσταση στόχου
 - **Βέλτιστη λύση**
 - Έχει το μικρότερο κόστος διαδρομής μεταξύ όλων των λύσεων

Προβλήματα – παιχνίδια

Κόσμος της ηλεκτρονικής σιούπας

- Καταστάσεις
 - Δες σχήμα επόμενης διαφάνειας (2×2^2)
- Αρχική κατάσταση
 - Οποιαδήποτε κατάσταση
- Συνάρτηση διαδοχών
 - Οι ενέργειες είναι:
 - Κίνηση της σιούπας Αριστερά
 - Κίνηση της σιούπας Δεξιά
 - Αναρρόφηση σιόνης
- Έλεγχος στόχου
 - Είναι και τα δύο τετράγωνα καθαρά;
- Κόστος διαδρομής
 - Ο αριθμός των βημάτων της διαδρομής
 - Κάθε βήμα έχει κόστος 1

Κόσμος της ηλεκτρικής σκούπας



L=Αριστερά, R=Δεξιά, S=Αναρρόφηση

Το παζλ των 8 πλακιδίων

- Καταστάσεις
 - Κάθε διαφορετική διάταξη των 8 πλακιδίων
- Αρχική κατάσταση
 - Οποιαδήποτε κατάσταση
- Συνάρτηση διαδοχών
 - Οι ενέργειες είναι:
 - Κενό πλακίδιο Αριστερά
 - Κενό πλακίδιο Δεξιά
 - Κενό πλακίδιο Πάνω
 - Κενό πλακίδιο Κάτω
- Έλεγχος στόχου
 - Ταυτίζεται η τρέχουσα κατάσταση με την κατάσταση στόχου;
- Κόστος διαδρομής
 - Ο αριθμός των βημάτων της διαδρομής
 - Κάθε βήμα έχει κόστος 1

Το παζλ των 8 πλακιδίων

7	2	4
5		6
8	3	1

Αρχική κατάσταση

	1	2
3	4	5
6	7	8

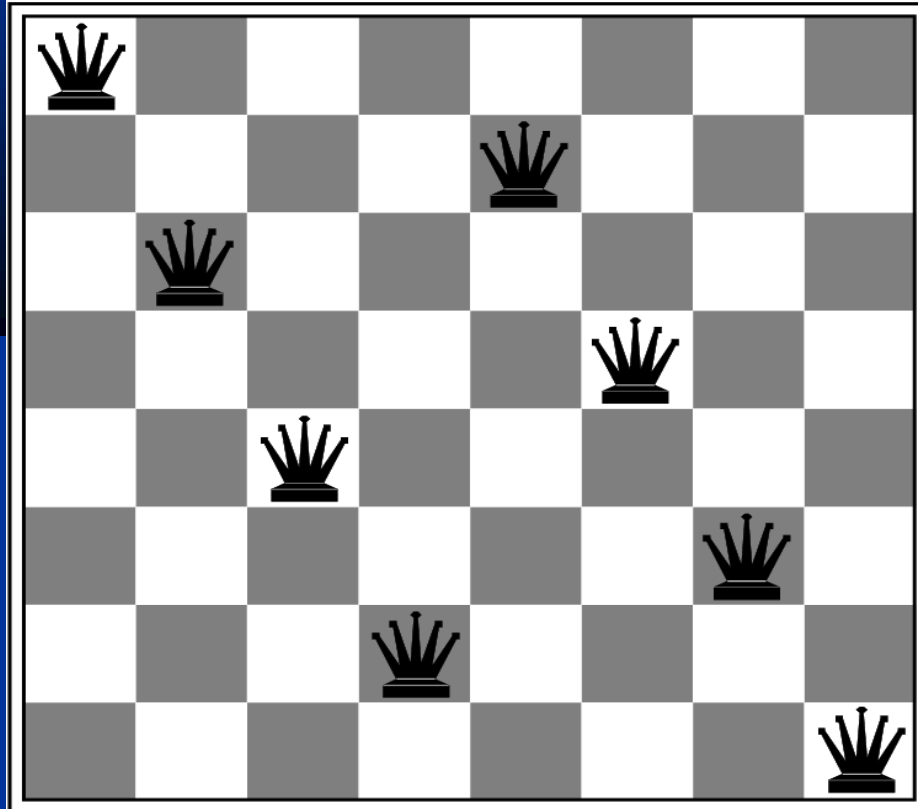
Κατάσταση στόχου

- Παζλ 15 πλακιδίων: Λύνεται εύκολα βέλτιστα
- Παζλ 24 πλακιδίων: Δε λύνεται βέλτιστα (ακόμη)

Το πρόβλημα των 8 βασιλισσών (αυξητική διατύπωση)

- Καταστάσεις
 - Οποιαδήποτε διάταξη 0 έως 8 βασιλισσών στη σιακιέρα
- Αρχική κατάσταση
 - Καμία βασίλισσα στη σιακιέρα
- Συνάρτηση διαδοχών
 - Μετακινεί μια βασίλισσα σε οποιοδήποτε κενό τετράγωνο
 - Εναλλακτικά, μετακινεί μια βασίλισσα σε οποιοδήποτε τετράγωνο ώστε να μην απειλείται από κάποια άλλη βασίλισσα
- Έλεγχος στόχου
 - 8 βασίλισσες στη σιακιέρα, καμία δεν απειλείται

Το πρόβλημα των 8 βασιλισσών



- Δύο εναλλακτικές διατυπώσεις:
 - Αυξητική
 - Πλήρεις καταστάσεις

Προβλήματα του πραγματικού κόσμου

Εύρεση δρομολογίου

- Με βάση καθορισμένες θέσεις και μεταβάσεις μέσω συνδέσμων μεταξύ τους να βρεθεί το βέλτιστο δρομολόγιο που πρέπει να ακολουθηθεί

Εύρεση δρομολογίου (παράδειγμα)

■ Αεροπορικό δρομολόγιο

■ Καταστάσεις

- Κάθε κατάσταση αναπαρίσταται με μια τοποθεσία (π.χ. ένα αεροδρόμιο) και με την τρέχουσα ώρα

■ Αρχική κατάσταση

- Καθορίζεται από το πρόβλημα

■ Συνάρτηση διαδόχων

- Επιστρέφει τις καταστάσεις που προκύπτουν από την επιλογή οποιασδήποτε προγραμματισμένης πτήσης

■ Έλεγχος στόχου

- Βρισκόμαστε στον προορισμό κάποια προγραμματισμένη ώρα;

■ Κόστος διαδρομής

- Εξαρτάται από το κόστος του εισιτηρίου, το χρόνο αναμονής, το χρόνο πτήσης, την ποιότητα της θέσης, κλπ.

Προβλήματα περιήγησης

- Έχουν στενή σχέση με τα προβλήματα εύρεσης δρομολογίων
- Οι ενέργειες είναι ίδιες
 - Αντιστοιχούν σε μετακινήσεις μεταξύ γειτονικών θέσεων
- Ο χώρος καταστάσεων είναι εντελώς διαφορετικός
 - Κάθε κατάσταση πρέπει να περιλαμβάνει όχι μόνο την τρέχουσα θέση αλλά και το σύνολο των θέσεων που έχει επισκεφτεί ο πράκτορας

Προβλήματα περιήγησης (παραδείγματα)

- Το πρόβλημα του πλανόδιου πωλητή (Traveling Salesman Problem – **TSP**)
 - Ο πράκτορας πρέπει να επισκεφτεί κάθε πόλη ακριβώς μία φορά κάνοντας το συντομότερο (φθηνότερο) δρομολόγιο
- Πλοήγηση ρομπότ
 - Μετακίνηση ενός ρομπότ σε ένα συνεχή χώρο με ένα (θεωρητικά) άπειρο σύνολο ενεργειών και καταστάσεων
- Σχεδίαση πρωτεϊνών
 - Εύρεση μιας ακολουθίας αμινοξέων που να καταλήγει σε μια τρισδιάστατη πρωτεΐνη με τις επιθυμητές ιδιότητες για τη θεραπεία κάποιας ασθένειας
- Αναζήτηση στο διαδίκτυο

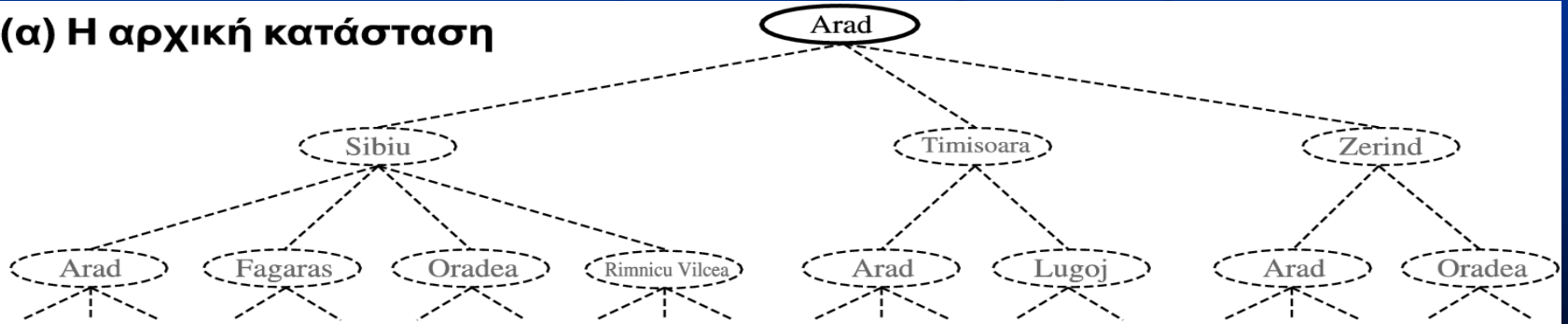
Αναζήτηση λύσεων

Δένδρο αναζήτησης

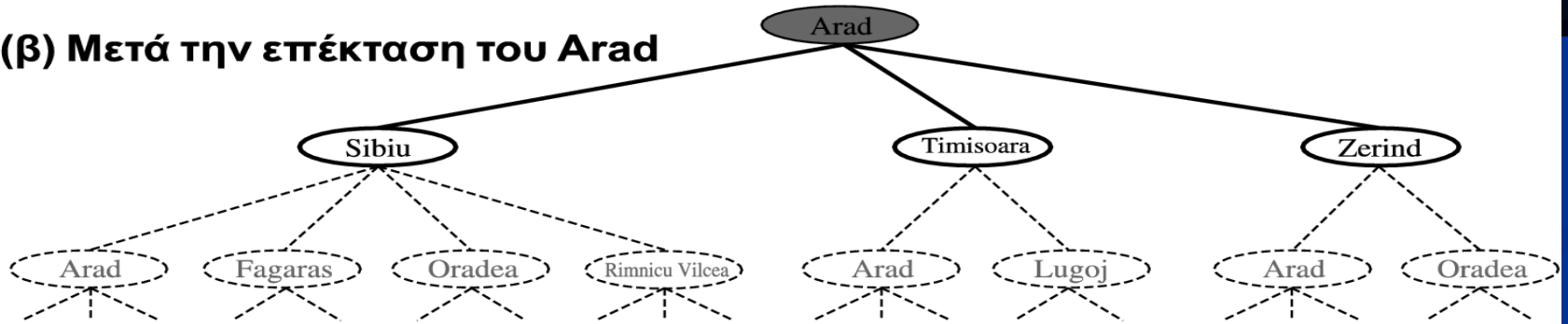
- **Κόμβος αναζήτησης**
 - Ένας για κάθε κατάσταση
- **Επέκταση** (της τρέχουσας κατάστασης)
 - Εφαρμογή της συνάρτησης διαδοχών στην τρέχουσα κατάσταση
- **Παραγωγή**
 - Από την επέκταση παράγεται ένα νέο σύνολο καταστάσεων
- **Στρατηγική αναζήτησης**
 - Προσδιορίζει την επιλογή της κατάστασης που θα επεκταθεί

Δένδρο αναζήτησης

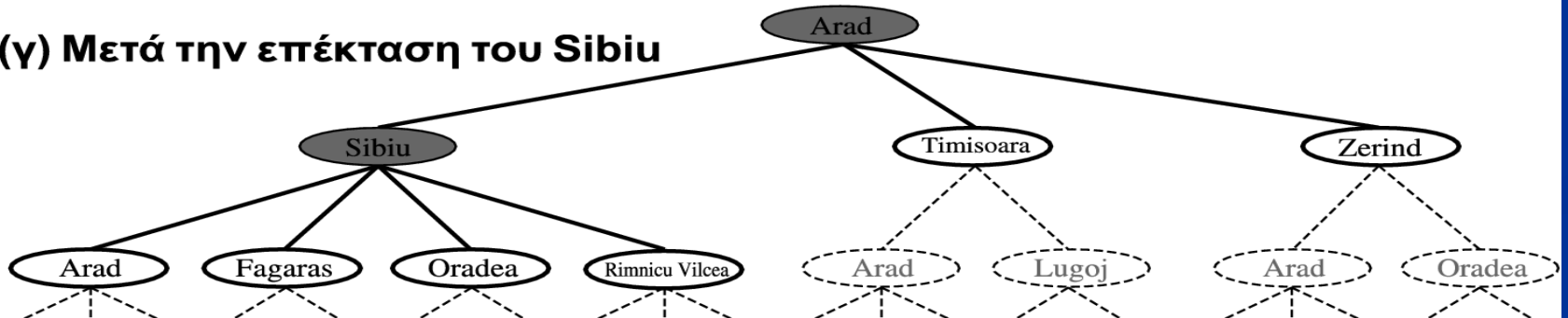
(α) Η αρχική κατάσταση



(β) Μετά την επέκταση του Arad



(γ) Μετά την επέκταση του Sibiu

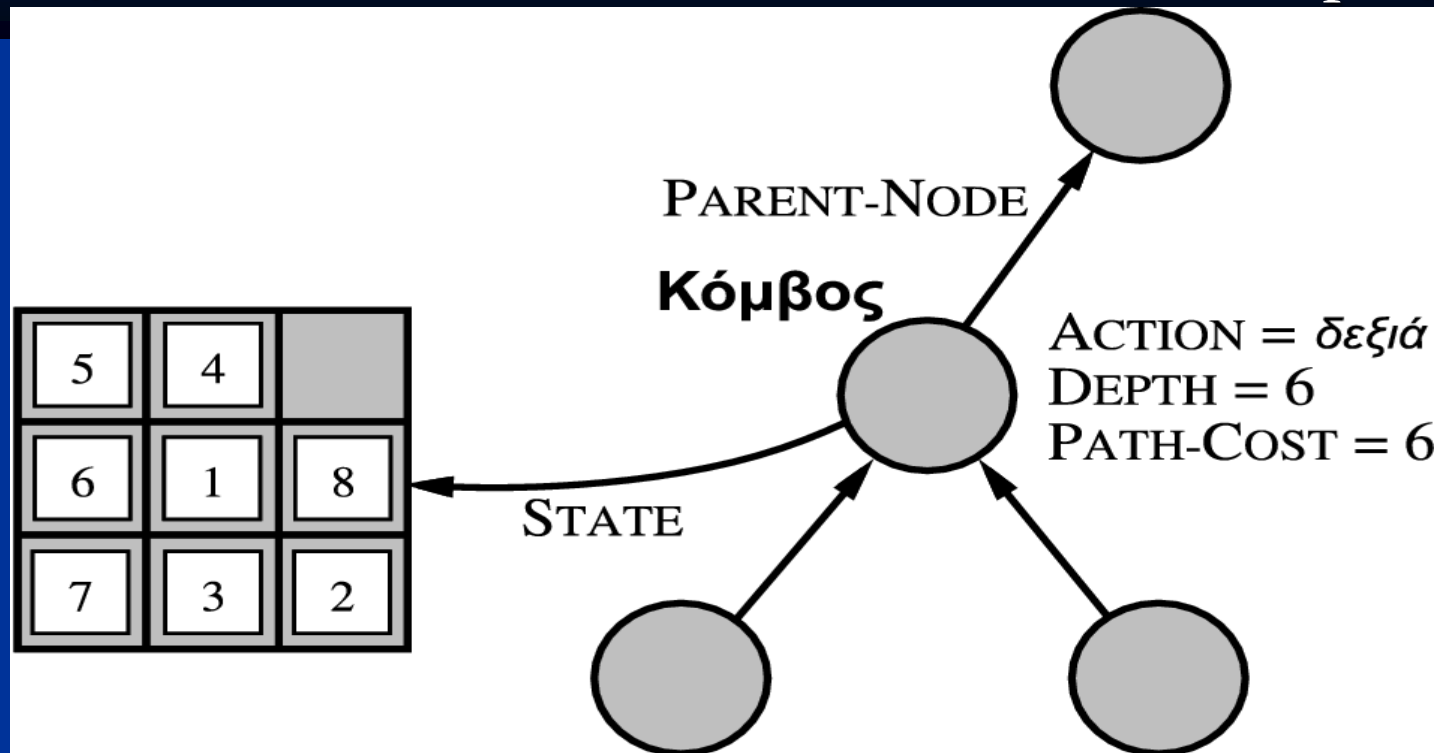


Δένδρο αναζήτησης

- Αναπαράσταση κόμβου αναζήτησης
 - **Κατάσταση** (State)
 - Η κατάσταση του χώρου καταστάσεων στην οποία αντιστοιχεί ο κόμβος
 - **Γονικός κόμβος** (Parent-Node)
 - Ο κόμβος του δέντρου αναζήτησης ο οποίος παρήγαγε αυτόν τον κόμβο
 - **Ενέργεια** (Action)
 - Η ενέργεια που εφαρμόστηκε στο γονικό κόμβο για να παραχθεί αυτός ο κόμβος
 - **Κόστος διαδρομής** (Path-Cost)
 - Το κόστος της διαδρομής από την αρχική κατάσταση μέχρι αυτόν τον κόμβο
 - **Βάθος** (Depth)
 - Ο αριθμός των βημάτων από την αρχική κατάσταση

Δένδρο αναζήτησης

- Αναπαράσταση κόμβου αναζήτησης
 - $\langle \text{State, Parent-Node, Action, Path-Cost, Depth} \rangle$



Δένδρο αναζήτησης

- Κόμβος φύλλο (leaf node)
 - Ένας κόμβος χωρίς διαδόχους στο δέντρο
- Σύνορο (fringe)
 - Η συλλογή των κόμβων που έχουν παραχθεί αλλά δεν έχουν επεκταθεί ακόμα
 - Υλοποίηση με ουρά
 - Η κλασσική ουρά είναι δομή δεδομένων **FIFO**

Μέτρηση απόδοσης αλγορίθμων αναζήτησης

- **Πληρότητα (Completeness)**
 - Είναι εγγυημένο ότι ο αλγόριθμος θα βρει λύση όταν υπάρχει;
- **Βέλτιστη συμπεριφορά (Optimality)**
 - Βρίσκει ο αλγόριθμος τη βέλτιστη λύση;
- **Χρονική πολυπλοκότητα (Time complexity)**
 - Πόσο χρόνο χρειάζεται για να βρει τη λύση;
- **Χωρική πολυπλοκότητα (Space complexity)**
 - Πόση μνήμη χρειάζεται για να πραγματοποιήσει την αναζήτηση;

Μέτρηση απόδοσης αλγορίθμων αναζήτησης

- Μέγεθος προβλήματος (πολυπλοκότητα):
 - Παράγοντας διακλάδωσης b (branching factor)
 - Ο μέγιστος αριθμός διαδόχων οποιουδήποτε κόμβου
 - Βάθος d (depth) του πιο ρηχού κόμβου στόχου
 - Μέγιστο μήκος m , οποιασδήποτε διαδρομής του χώρου καταστάσεων

Μέτρηση απόδοσης αλγορίθμων αναζήτησης

■ Κόστος αναζήτησης (search cost)

- Εξαρτάται κατά κανόνα από τη χρονική πολυπλοκότητα αλλά μπορεί να περιλαμβάνει και έναν όρο για τη χρησιμοποίηση μνήμης (χωρική πολυπλοκότητα)

■ Ολικό κόστος (total cost)

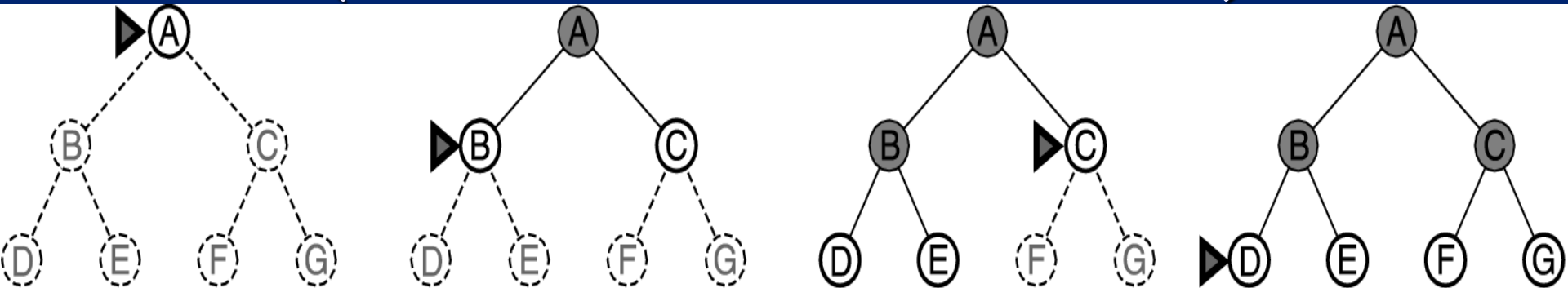
= Κόστος αναζήτησης + Κόστος εκτέλεσης πλάνου
(κόστος διαδρομής της λύσης που βρέθηκε)

Στρατηγικές απληροφόρητης αναζήτησης

Στρατηγικές απληροφόρητης αναζήτησης

- Δεν έχουν καμία επιπλέον πληροφορία για τις καταστάσεις, πέρα από τις πληροφορίες που παρέχονται από τον ορισμό του προβλήματος
- Το μόνο που μπορούν να κάνουν είναι να παράγουν διαδόχους και να διακρίνουν μια κατάσταση στόχου από μια κατάσταση που δεν ικανοποιεί το στόχο

Αναζήτηση πρώτα κατά πλάτος (Breadth-first search – BFS)



- Πρώτα επεκτείνεται ο κόμβος ρίζα, μετά όλοι οι διάδοχοί του, μετά οι δικοί τους διάδοχοι, κ.ο.κ.
- Επεκτείνονται όλοι οι κόμβοι που βρίσκονται σε ένα δεδομένο βάθος στο δέντρο αναζήτησης, πριν επεκταθούν οποιοδήποτε κόμβοι του επόμενου επιπέδου

Αναζήτηση πρώτα κατά πλάτος (Breadth-first search – BFS)

- Πλήρης
 - Όταν ο παράγοντας διακλάδωσης είναι πεπερασμένος
- Βέλτιστος
 - Όταν όλες οι ενέργειες έχουν το ίδιο κόστος
- Χωρική, χρονική πολυπλοκότητα $O(b^{d+1})$

Αναζήτηση πρώτα κατά πλάτος (Breadth-first search – BFS)

- π.χ. για $b=10$, 10.000 κόμβοι/sec, 1000 bytes/κόμβο

Βάθος	Κόμβοι	Χρόνος	Μνήμη
2	1100	0.11 δευτερόλεπτα	1 megabyte
4	111.100	11 δευτερόλεπτα	10 ⁶ megabytes
6	10 ⁷	19 λεπτά	10 gigabytes
8	10 ⁹	31 ώρες	1 terabyte
10	10 ¹¹	129 ημέρες	101 terabytes
12	10 ¹³	35 χρόνια	10 petabytes
14	10 ¹⁵	3.523 χρόνια	1 exabyte

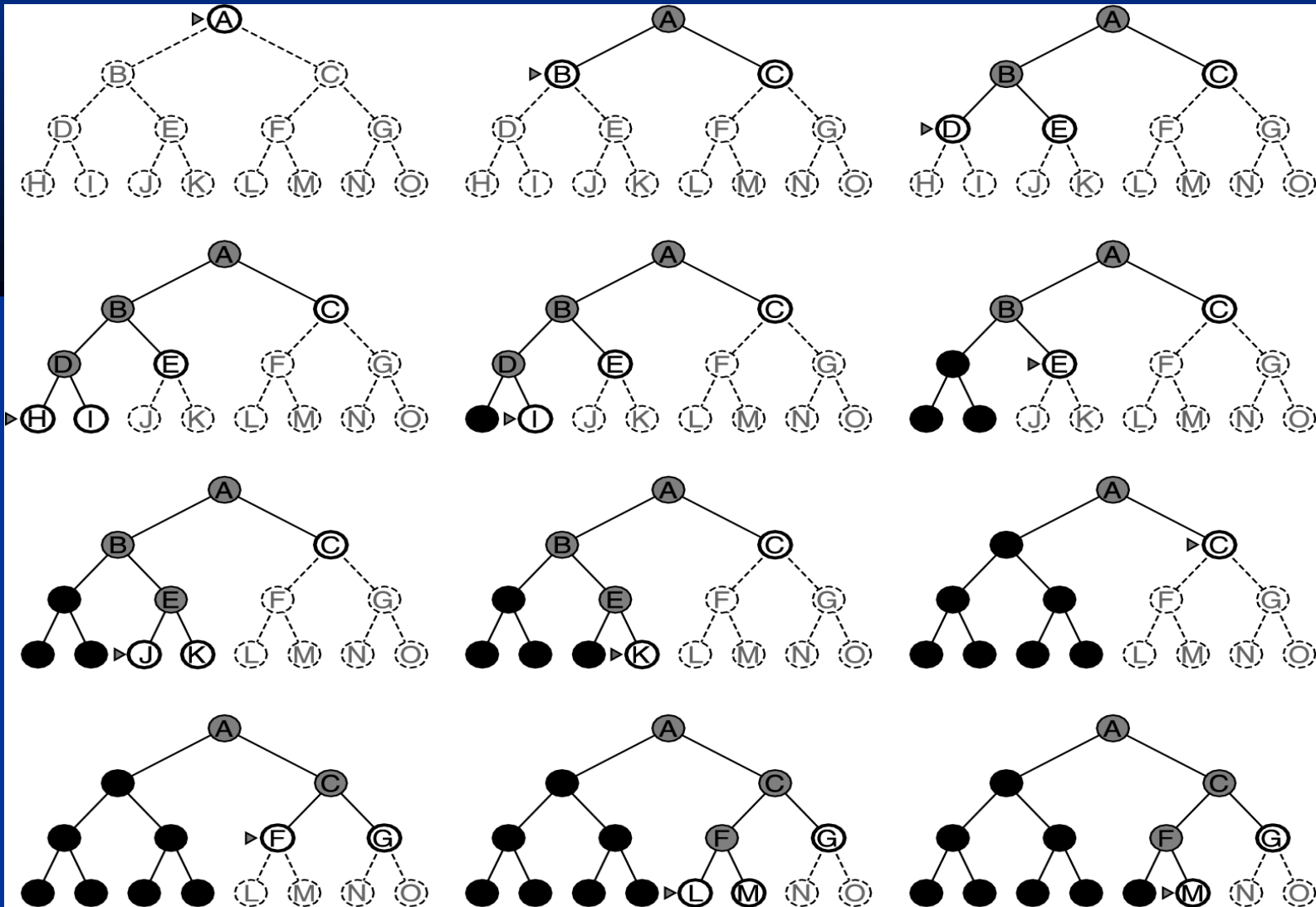
Αναζήτηση ομοιόμορφου κόστους (Uniform-cost search)

- Παραλλαγή της αναζήτησης πρώτα κατά πλάτος:
 - Επειτύνει πρώτα τους κόμβους με το μικρότερο κόστος διαδρομής
 - Δεν ενδιαφέρεται για τον αριθμό των βημάτων μιας διαδρομής παρά μόνο για το ολικό κόστος της
- Προϋπόθεση πληρότητας: Κόστος βήματος > 0
 - Διαφορετικά θα εισέλθουμε σε ατέρμονο βρόγχο
- Εγγυάται ότι θα βρει τη βέλτιστη λύση:
 - Ο έλεγχος στόχου γίνεται κατά την έξοδο ενός κόμβου από το σύνορο

Αναζήτηση πρώτα κατά βάθος (Depth-first search – DFS)

- Επεκτείνει πάντα το βαθύτερο κόμβο του τρέχοντος συνόρου του δέντρου αναζήτησης
- Η αναζήτηση προχωρά αμέσως στο βαθύτερο επίπεδο του δέντρου αναζήτησης όπου οι κόμβοι δεν έχουν διαδόχους
- Καθώς αυτοί επεκτείνονται, αφαιρούνται από το σύνορο, και μετά η αναζήτηση «οπισθοχωρεί» στον αμέσως ρηχότερο κόμβο που έχει ακόμα ανεξερεύνητους διαδόχους

Αναζήτηση πρώτα κατά βάθος (Depth-first search – DFS)



Αναζήτηση πρώτα κατά βάθος (Depth-first search – DFS)

- Ελάχιστες απαιτήσεις σε χώρο: $b \cdot m + 1$
- Δεν είναι πλήρης
- Δεν είναι βέλτιστη
- Πολυπλοκότητα χρόνου: $O(b^m)$
 - Προσοχή: $m \geq d$ (περίπτωση ύπαρξης άπειρης διαδρομής)

Αναζήτηση περιορισμένου βάθους (Depth-limited search)

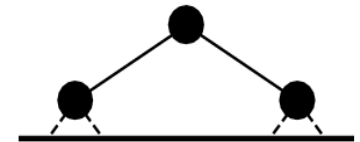
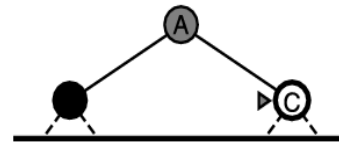
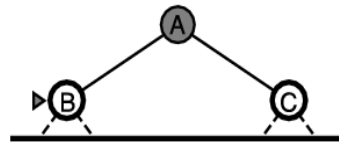
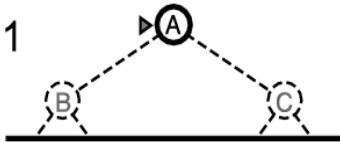
- Τίθεται όριο βάθους $L < m$.
- Λύνεται το πρόβλημα των άπειρων διαδρομών
- Απώλεια πληρότητας αν $L < d$.
- Όχι βέλτιστες λύσεις αν $L > d$
- Χρονική πολυπλοκότητα: $O(b^L)$
- Χωρική πολυπλοκότητα: $O(b \cdot L)$
- Επιλογή L με γνώση του προβλήματος
 - (π.χ. διάμετρος του χώρου καταστάσεων)

Επαναληπτική εκβάθυνση (Iterative-deepening search)

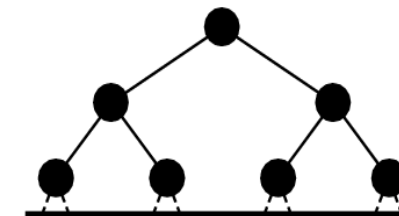
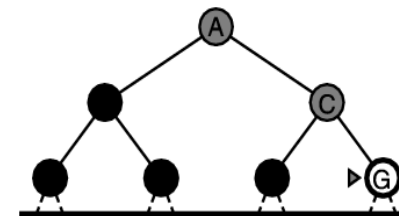
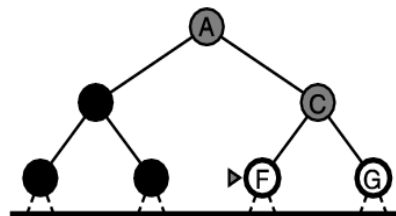
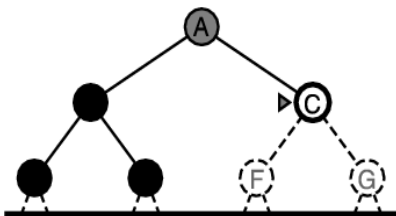
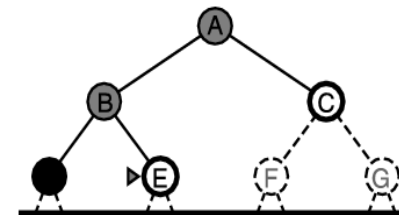
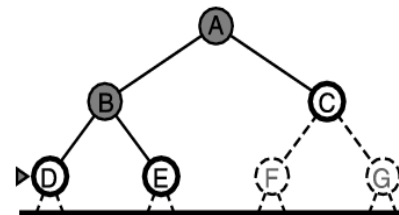
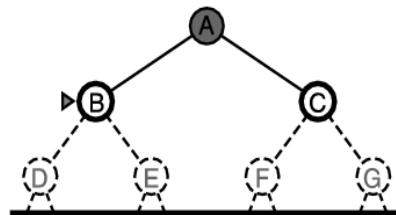
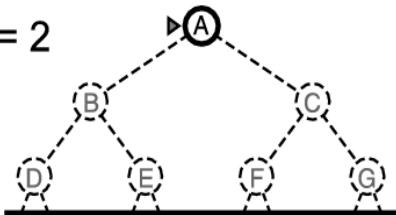
Όριο = 0



Όριο = 1

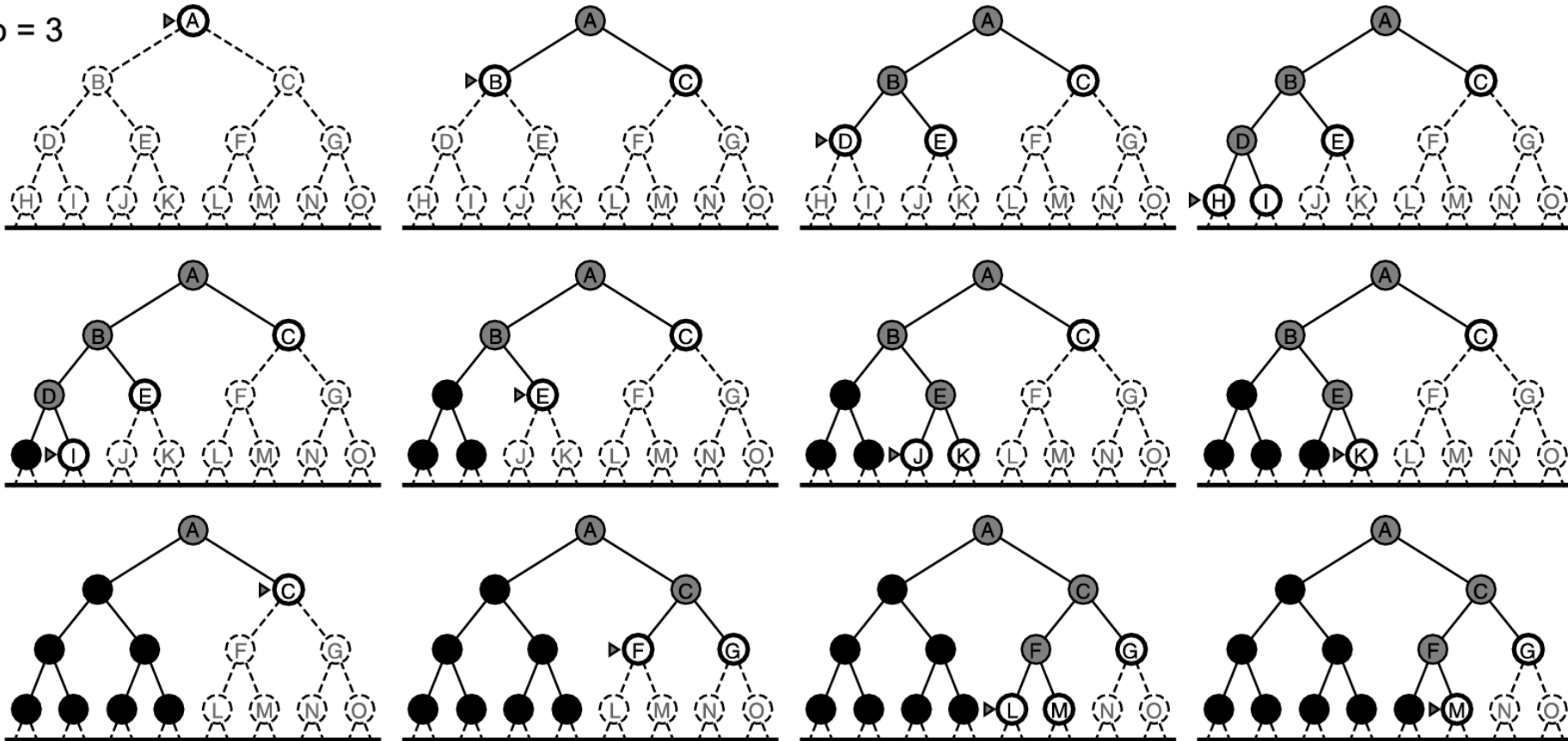


Όριο = 2



Επαναληπτική εκβάθυνση (Iterative-deepening search)

Όριο = 3



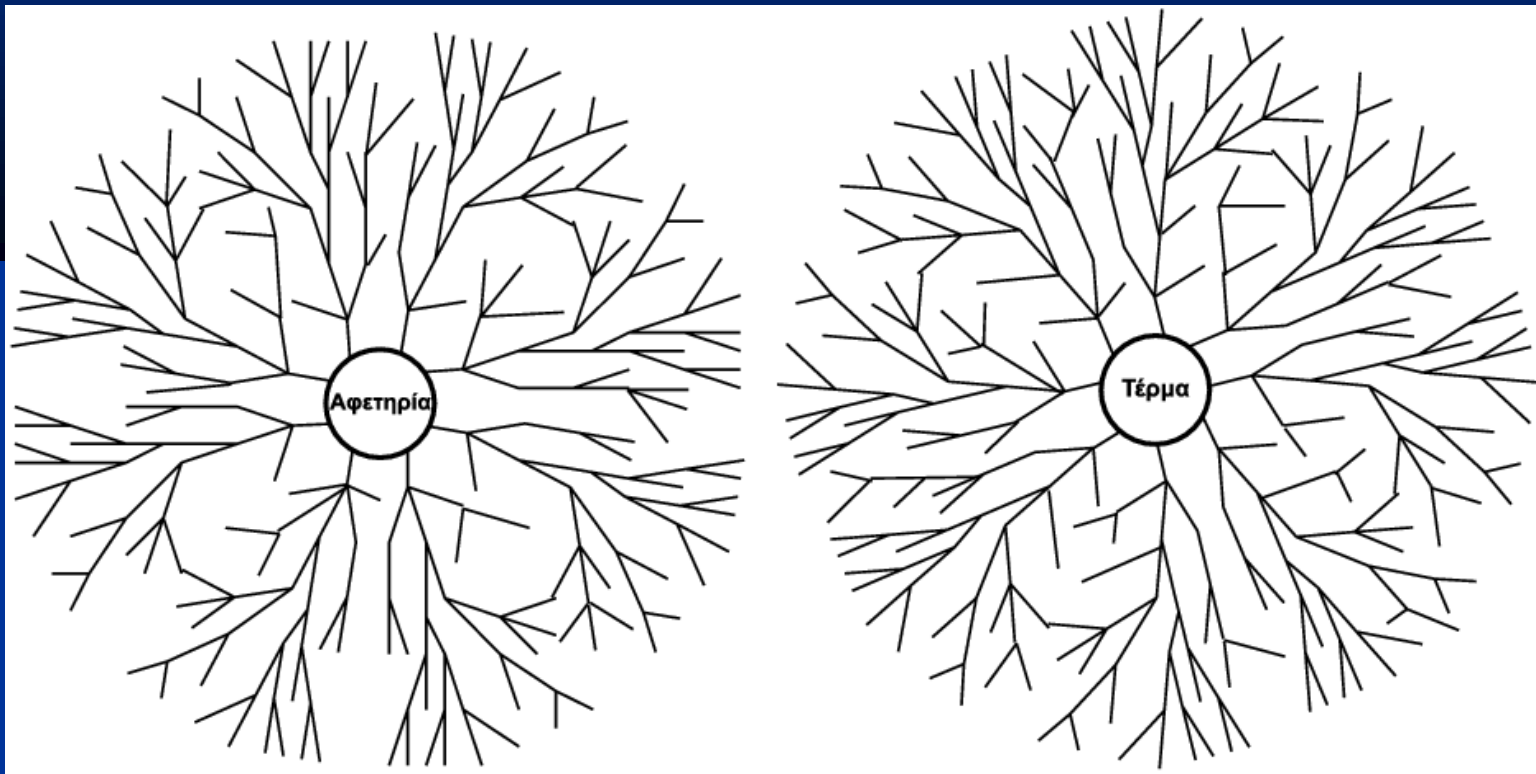
Επαναληπτική εκβάθυνση (Iterative-deepening search)

- Είναι πλήρης
 - Λόγω BFS
- Χωρική πολυπλοκότητα: $O(b \cdot d)$
 - Λόγω DFS
- Χρονική πολυπλοκότητα:
 - $N(\text{IDS}) = d \cdot b + (d - 1) \cdot b^2 + \dots + 1 \cdot b^d = O(b^d)$!!!!
 - Μικρότερη από τη χρονική πολυπλοκότητα της αναζήτησης DFS.

Επαναληπτική εκβάθυνση (Iterative-deepening search)

- Η επαναληπτική εκβάθυνση είναι η προτιμότερη μέθοδος απληροφόρητης αναζήτησης όταν ο χώρος αναζήτησης είναι μεγάλος και το βάθος της λύσης δεν είναι γνωστό
- Αναζήτηση επαναληπτικής επιμήκυνσης
 - Παραλλαγή της επαναληπτικής εκβάθυνσης με κόστη αντί για βήματα

Αμφίδρομη αναζήτηση



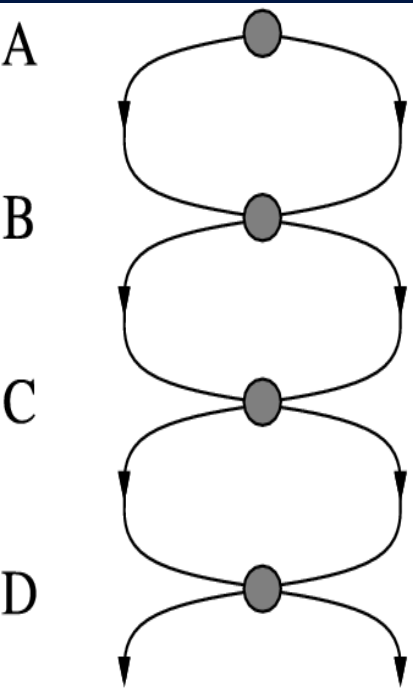
■ $b^{d/2} + b^{d/2} \ll b^d$

Αμφίδρομη αναζήτηση

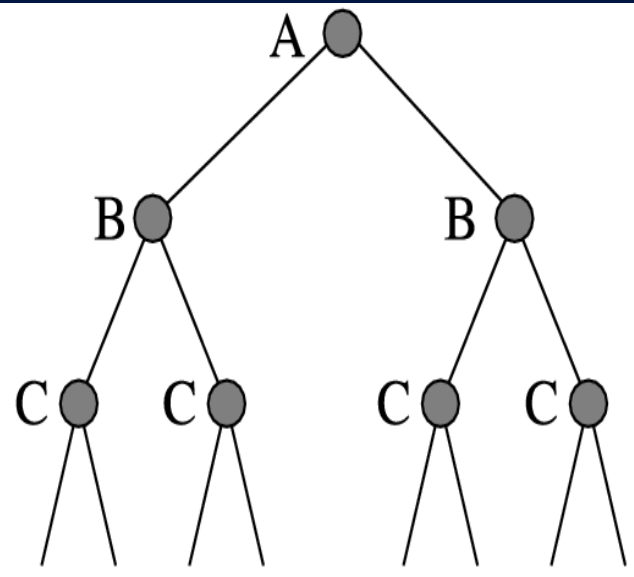
- Χρονική πολυπλοκότητα: $O(b^{d/2})$
- Χωρική πολυπλοκότητα: $O(b^{d/2})$ (μειονέκτημα)
- Πλήρης
 - Αν το b είναι πεπερασμένο
 - Αν και οι δύο αναζητήσεις είναι πρώτα σε πλάτος
- Βέλτιστη
 - Αν τα κόστη των βημάτων είναι όλα ίδια
 - Αν και οι δύο αναζητήσεις είναι πρώτα σε πλάτος
- Προβλήματα:
 - Εύρεση προιατόχων καταστάσεων
 - Έλλειψη καλά καθορισμένων στόχων (π.χ. σιάι)

Αποφυγή επαναλαμβανόμενων καταστάσεων

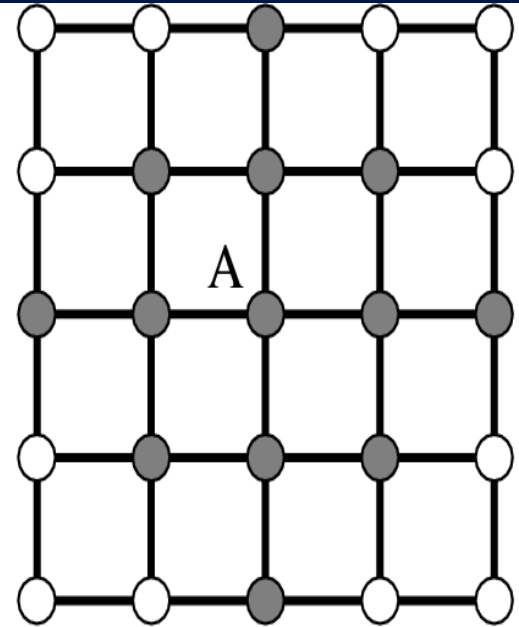
- Καταστάσεις στις οποίες μπορούμε να φτάσουμε με περισσότερες από μία διαδρομές



(α)



(β)



(γ)

Αποφυγή επαναλαμβανόμενων καταστάσεων

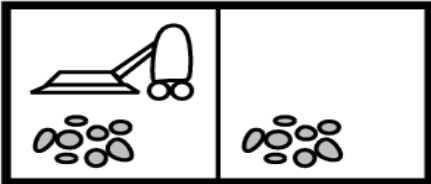
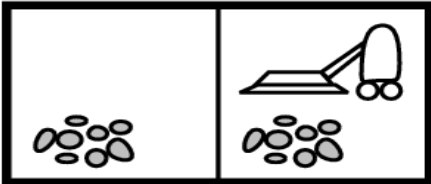
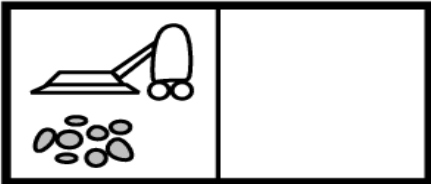
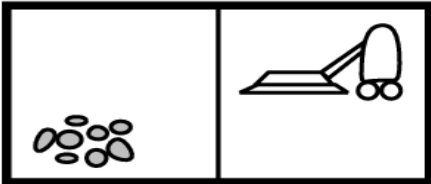
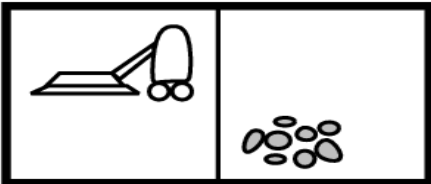
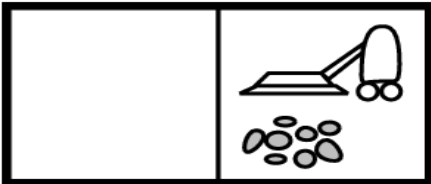
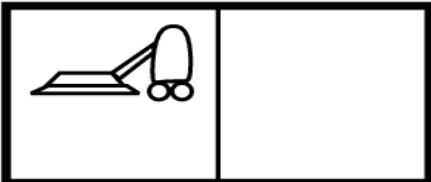
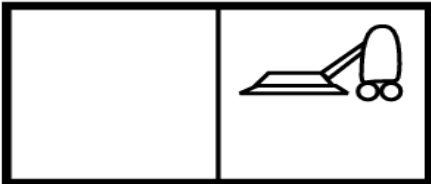
- Οι αλγόριθμοι που ξεχνούν την ιστορία τους είναι καταδικασμένοι να την επαναλαμβάνουν!
- Έλεγχος μόνο τρέχουσας διαδρομής
 - π.χ. Αλγόριθμος DFS
- Έλεγχος όλων των καταστάσεων
 - **Κλειστή λίστα**
 - Οι κόμβοι που έχουν επεκταθεί
 - **Ανοικτή λίστα**
 - Το σύνολο των κόμβων που δεν έχουν επεκταθεί
- Πρόβλημα: Ποια από τις δύο καταστάσεις κρατάμε;

Αναζήτηση με μερική πληροφόρηση

Κατηγορίες προβλημάτων

1. Προβλήματα χωρίς αισθητήρες (sensorless problems) ή σύμμορφα προβλήματα (conformant problems)
 - Ο πράκτορας θα μπορούσε να βρίσκεται σε μία από πολλές δυνατές καταστάσεις
 - Κάθε ενέργεια θα μπορούσε να οδηγεί σε μία από πολλές δυνατές καταστάσεις
2. Προβλήματα ενδεχομένων (contingency problems)
 - Οι αντιλήψεις του πράκτορα παρέχουν νέες πληροφορίες μετά από κάθε ενέργεια
 - **Προβλήματα αντιπαλότητας**
3. Προβλήματα εξερεύνησης (exploration problems)
 - Οι καταστάσεις και οι ενέργειες του περιβάλλοντος είναι άγνωστες

Κατηγορίες προβλημάτων

1		2	
3		4	
5		6	
7		8	

Προβλήματα χωρίς αισθητήρες

- Αρχική $\rightarrow \{1, 2, 3, 4, 5, 6, 7, 8\}$
- [Δεξιά] $\rightarrow \{2, 4, 6, 8\}$
- [Δεξιά, Αναρρόφηση] $\rightarrow \{4, 8\}$
- [Δεξιά, Αναρρόφηση, Αριστερά] $\rightarrow \{3, 7\}$
- [Δεξιά, Αναρρόφηση, Αριστερά, Αναρρόφηση] $\rightarrow \{7\}$

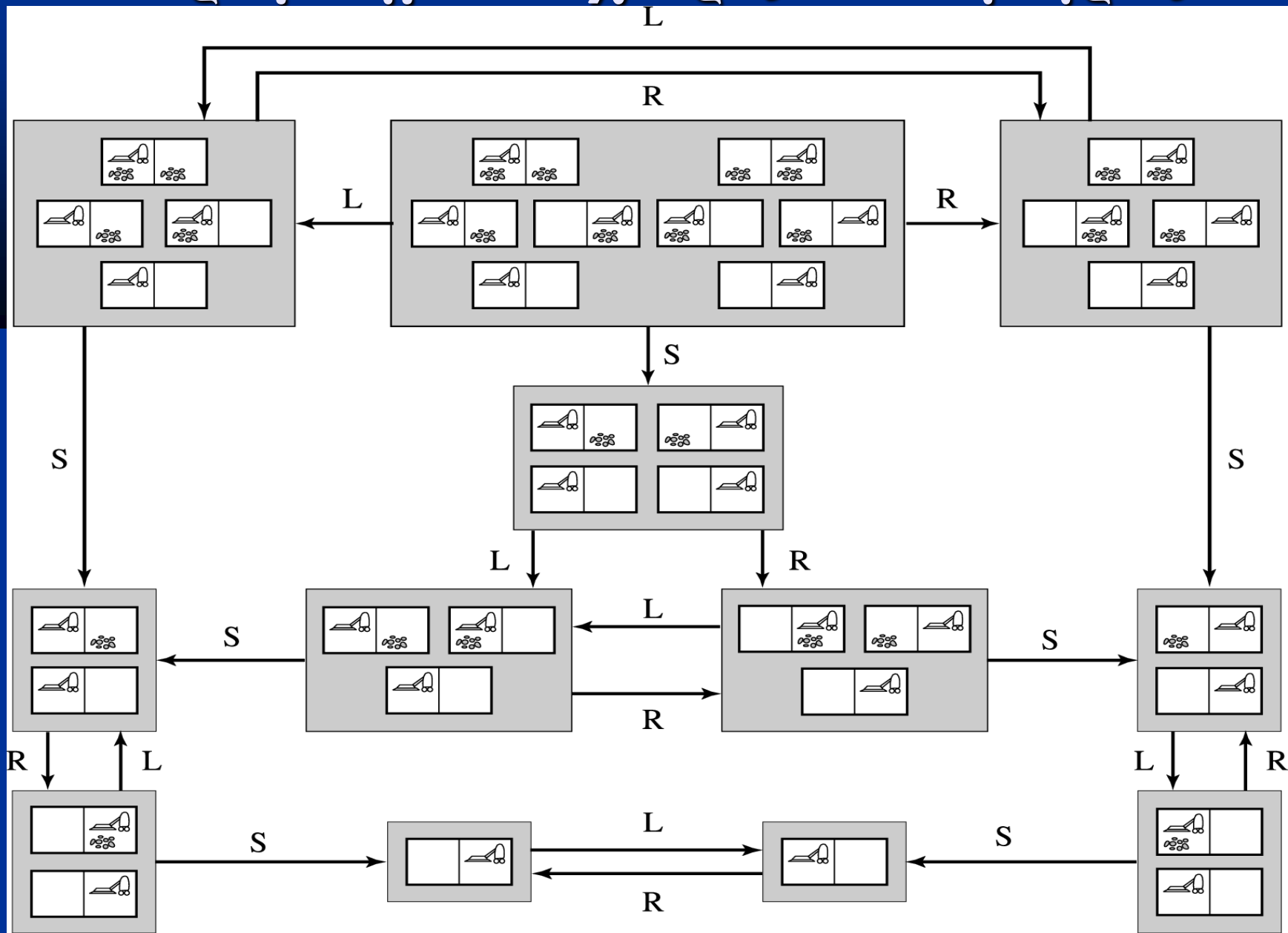
Προβλήματα χωρίς αισθητήρες

- Κατάσταση πεποίθησης
 - Σύνολο καταστάσεων στα οποία μπορεί να μεταβεί
 - Αναπαριστά την τρέχουσα πεποίθηση του πράκτορα για τις δυνατές φυσικές καταστάσεις στις οποίες μπορεί να βρίσκεται
- Εξαναγκασμός
 - Ο πράκτορας μπορεί να εξαναγκάσει τον κόσμο να μεταβεί σε κάποια συγκεκριμένη κατάσταση αιόμα και όταν δε γνωρίζει από που ξεκίνησε

Προβλήματα χωρίς αισθητήρες

- Χώρος καταστάσεων πεποιθήσης
 - Δυναμοσύνολο του χώρου καταστάσεων.
 - Δεν είναι πάντα όλες προσπελάσιμες.
- Δεν έχουν πάντα λύση αυτά τα προβλήματα.

Προβλήματα χωρίς αισθητήρες



Προβλήματα ενδεχομένων

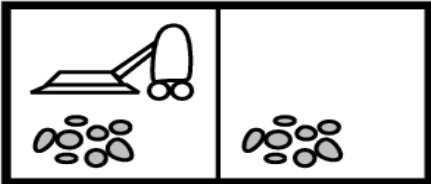
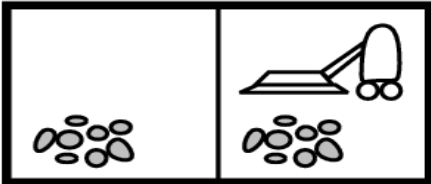
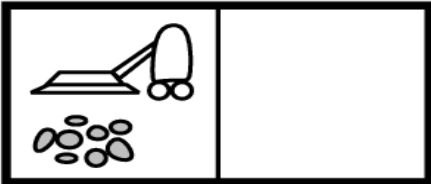
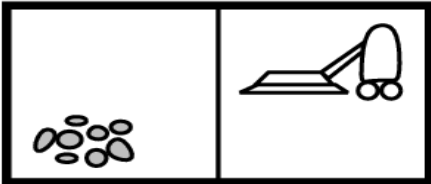
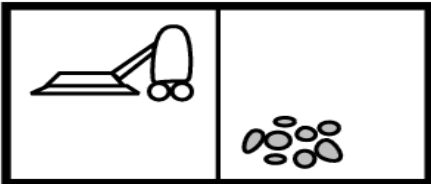
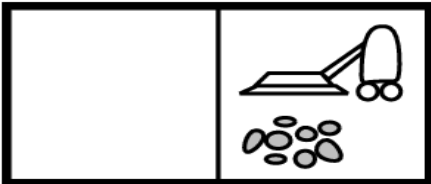
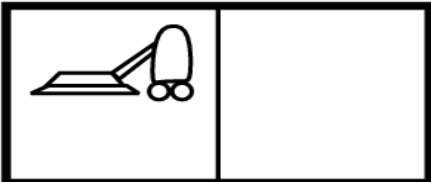
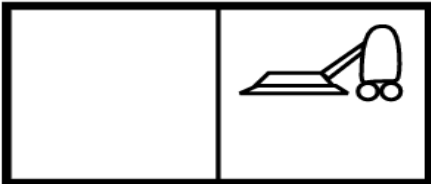
- Όταν έχουμε:
 - Μη πλήρως παρατηρήσιμο περιβάλλον, ή/και
 - Στοχαστικές ενέργειες

- Ενέργειες αίσθησης χρησιμοποιούνται για συλλογή πληροφοριών κατά την εκτέλεση της λύσης.

Προβλήματα ενδεχομένων

- Π.χ. Έστω ο κόσμος της σιούπας, όπου:
 - ο πράκτορας έχει τοπικό αισθητήρα σιόνης
 - η ενέργεια Αναρρόφηση δεν πετυχαίνει πάντα
 - μερικές φορές αποθέτει σιόνη στο χαλί, αν δεν υπάρχει αρχικά σιόνη εκεί
 - Ο πράκτορας ξεκινά με την αντίληψη
 - [Αριστερό, Σιονισμένο]
 - Η ακολουθία ενεργειών
 - [Αναρρόφηση, Δεξιά, Αναρρόφηση]
- θα μας οδηγήσει στο στόχο; (κατάσταση 8)

Κατηγορίες προβλημάτων

1		2	
3		4	
5		6	
7		8	

Προβλήματα ενδεχομένων

- Λύση:
 - [Αναρρόφηση, Δεξιά, **if** [Δεξιό, Σκονισμένο] **then** Αναρρόφηση]
- Δεν είναι απαραίτητο το να έχουμε πλήρη λύση εξαρχής
 - Συνδυασμός (interleaving) αναζήτησης και εκτέλεσης.

Άσκηση 3.2

- Γιατί η διατύπωση του προβλήματος πρέπει να έπεται της διατύπωσης του στόχου;

Άσκηση 3.7

- Βρείτε την αρχική κατάσταση, τον έλεγχο στόχου, τη συνάρτηση διαδόχων και τη συνάρτηση κόστους για το παρακάτω πρόβλημα:
- Πρέπει να χρωματίσετε έναν επίπεδο χάρτη χρησιμοποιώντας μόνο τέσσερα χρώματα, με τέτοιο τρόπο ώστε δύο γειτονικές περιοχές να μην έχουν ποτέ το ίδιο χρώμα

Άσκηση 3.7

- Βρείτε την αρχική κατάσταση, τον έλεγχο στόχου, τη συνάρτηση διαδόχων και τη συνάρτηση κόστους για το παρακάτω πρόβλημα:
- Έχετε τρία δοχεία που χωρούν 12 λίτρα, 8 λίτρα και 3 λίτρα νερό, και μία βρύση. Μπορείτε να γεμίζετε τα δοχεία ή να τα αδειάζετε από το ένα στο άλλο ή στο έδαφος. Πρέπει να μετρήσετε ακριβώς ένα λίτρο

Άσκηση 3.8

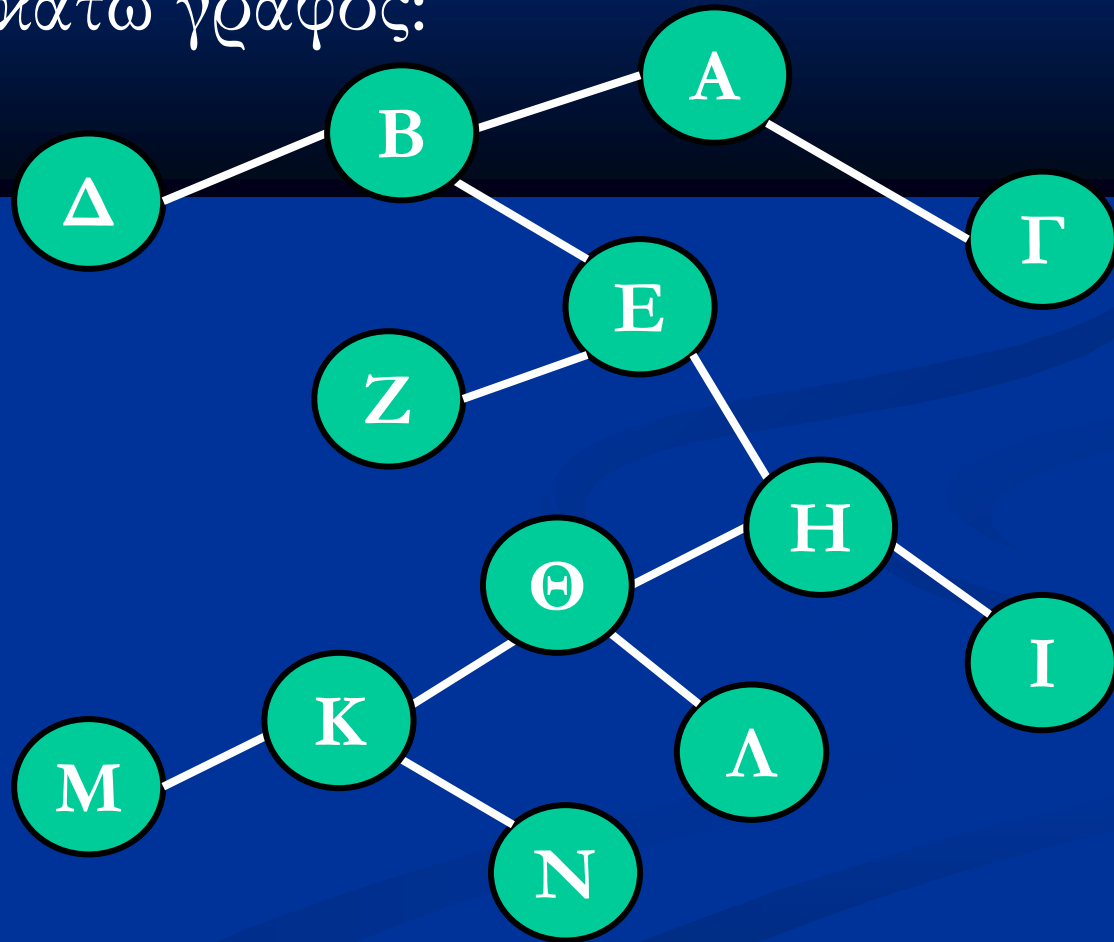
- Έστω ένας χώρος καταστάσεων όπου η αρχική κατάσταση είναι ο αριθμός 1 και η συνάρτηση διαδόχων για την κατάσταση n επιστρέφει δύο καταστάσεις, τους αριθμούς $2n$ και $2n+1$
 1. Σχεδιάστε το τμήμα του χώρου καταστάσεων για τις καταστάσεις 1 μέχρι 15
 2. Ας υποθέσουμε ότι η κατάσταση στόχου είναι η 11. Βρείτε τη σειρά με την οποία θα επισκεφτεί τους κόμβους (μέχρι να βρει τη λύση):
 - μια αναζήτηση πρώτα σε πλάτος,
 - μια αναζήτηση περιορισμένου βάθους με όριο 3
 - μια αναζήτηση επαναληπτικής ειβάθυνσης

Άσκηση 3.8

- Έστω ένας χώρος καταστάσεων όπου η αρχική κατάσταση είναι ο αριθμός 1 και η συνάρτηση διαδόχων για την κατάσταση n επιστρέφει δύο καταστάσεις, τους αριθμούς $2n$ και $2n+1$
 1. Θα ήταν κατάλληλη η αμφίδρομη αναζήτηση γι' αυτό το πρόβλημα; Αν ναι, περιγράψτε αναλυτικά πως θα λειτουργούσε
 2. Ποιος είναι ο παράγοντας διακλάδωσης προς την κάθε κατεύθυνση στην αμφίδρομη αναζήτηση;
 3. Μήπως η απάντηση στο ερώτημα 3 υποδεικνύει μια επαναδιατύπωση του προβλήματος που επιτρέπει τη μετάβαση από τον κόμβο 1 σε μια δεδομένη κατάσταση στόχου σχεδόν χωρίς αναζήτηση;

Άσκηση BFS – DFS

Έστω ο παρακάτω γράφος:



Άσκηση BFS – DFS

- Με ποια σειρά θα πρωτοεπισκεφτεί τους κόμβους του παραπάνω γράφου ο αλγόριθμος αναζήτησης πρώτα κατά πλάτος (Breadth First Search);
- Με ποια σειρά θα πρωτοεπισκεφτεί τους κόμβους του παραπάνω γράφου ο αλγόριθμος αναζήτησης πρώτα κατά βάθος (Depth First Search);

Ο κόσμος της σιούπας

Έστω ο «γνωστός» κόσμος της σιούπας που αποτελείται από 8 διαφορετικές καταστάσεις. Η σιούπα βρίσκεται είτε στην αριστερή είτε στη δεξιά αίθουσα. Κάθε αίθουσα μπορεί να είναι είτε καθαρή είτε με σιόνη.

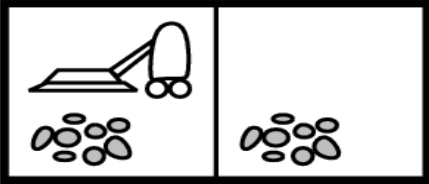
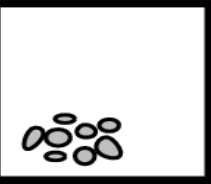
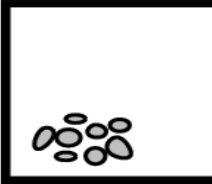

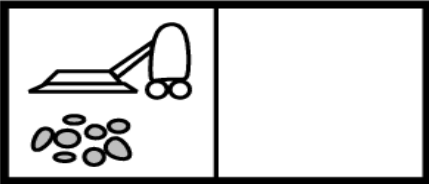
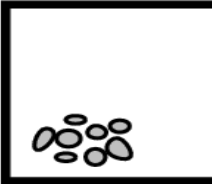
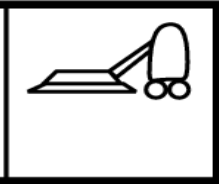
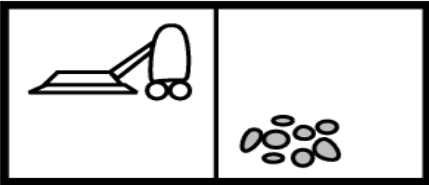
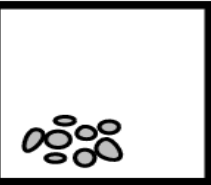
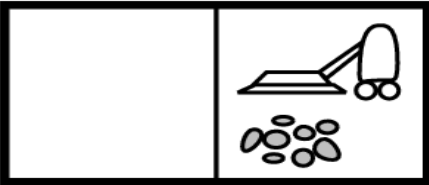
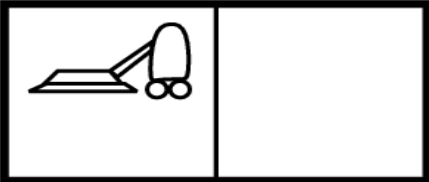
1. Θεωρώντας ως αρχική κατάσταση την κατάσταση (1) του κόσμου της σιούπας (σιόνη και στις δύο αίθουσες – η σιούπα στην αριστερή αίθουσα), βρείτε την πρώτη διαδρομή που επιστρέφει ο αλγόριθμος **breadth-first search** από την **κατάσταση 1** προς κάποια κατάσταση της οποίας το δεξιό δωμάτιο είναι καθαρό, δηλαδή προς την κατάσταση (3), (4), (7) ή (8).

Ο κόσμος της σιούπας

2. Θεωρώντας ως αρχική κατάσταση την κατάσταση (1) του κόσμου της σιούπας (σιόνη και στις δύο αίθουσες – η σιούπα στην αριστερή αίθουσα), βρείτε την πρώτη διαδρομή που επιστρέφει ο αλγόριθμος **depth-first search** από την **κατάσταση 1** προς κάποια κατάσταση της οποίας το δεξιό δωμάτιο είναι καθαρό, δηλαδή προς την κατάσταση (3), (4), (7) ή (8).

Δώστε προτεραιότητα στις ενέργειες με την εξής σειρά:
Σιούπισμα (S), Αριστερή μετακίνηση (L), Δεξιά μετακίνηση (R).

Ο κόσμος της σκούπας

1		
2		
3		
4		
5		
6		
7		
8		