



Πολυτεχνείο
Κρήτης

Σχολή Μηχανικών Παραγωγής και Διοίκησης

*Εισαγωγή στην Τεχνητή
Νοημοσύνη*

Επίλυση προβλημάτων με αναζήτηση

Τσαφαράκης Στέλιος

Περιγραφή Προβλημάτων

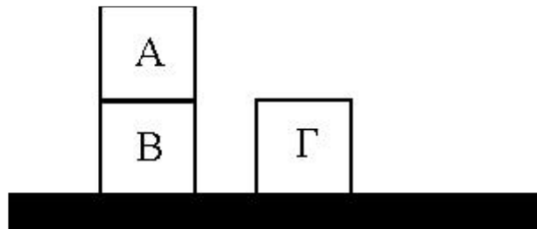
- Διαισθητικά η έννοια του προβλήματος:
 - υπάρχει μία δεδομένη κατάσταση (αρχική),
 - υπάρχει μία επιθυμητή κατάσταση (τελική) και
 - διαθέσιμες ενέργειες που πρέπει να γίνουν ώστε να φτάσουμε στην επιθυμητή κατάσταση.
- Η επίλυση προβλημάτων που επιδιώκεται από την ΤΝ απαιτεί τον τυποποιημένο και σαφή ορισμό τους.
- Ο ορισμός ή περιγραφή ενός προβλήματος είναι ανεξάρτητος από την πολυπλοκότητα επίλυσής του και καθορίζεται από την αναπαράστασή του.
- Η πολυπλοκότητα επίλυσής του καθορίζεται από την πολυπλοκότητα του αλγόριθμο αναζήτησης που εφαρμόζεται για την επίλυσή του.

Κατηγορίες Προβλημάτων

- Απλά προβλήματα (toy problems)
 - κύβοι (*blocks*),
 - *N-puzzle*,
 - τρίλιζα (*tic-tac-toe*),
 - λαβύρινθος (*maze*),
 - πύργοι του Ανόι (*Hanoi towers*),
 - κανίβαλοι και ιεραπόστολοι (*missionaries and cannibals*),
 - ποτήρια (*water glass*)
- Πραγματικά πολύπλοκα προβλήματα (*real world problems*):
 - σκάκι (*chess*),
 - πλανόδιος πωλητής (*traveling salesman*),
 - *N-βασίλισσες* (*N-queens*),
 - σάκος (*knapsack*)
- Η μεθοδολογία που εφαρμόζεται στην αναπαράσταση & την επίλυση είναι παρόμοια & στις 2 περιπτώσεις προβλημάτων.

Παραδείγματα Απλών Προβλημάτων (1/2)

κύβοι (blocks)



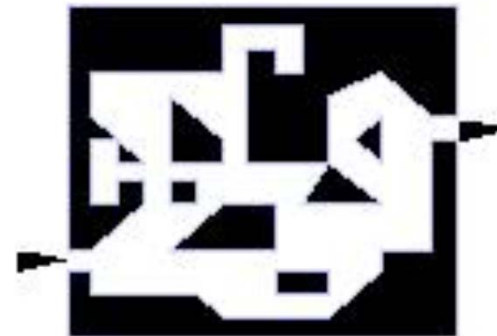
N-puzzle

8	3	5
4	1	7
2		6

τρίλιζα (tic-tac-toe)

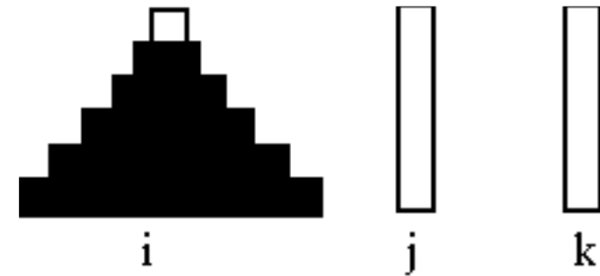
		Ο
Ο	Χ	
	Χ	

λαβύρινθος (maze)

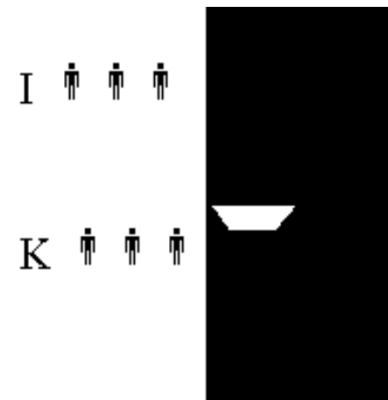


Παραδείγματα Απλών Προβλημάτων (2/2)

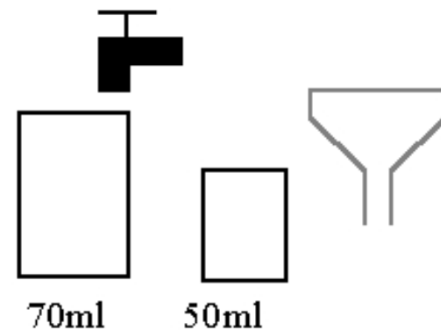
πύργοι του Ανόι (Hanoi towers)



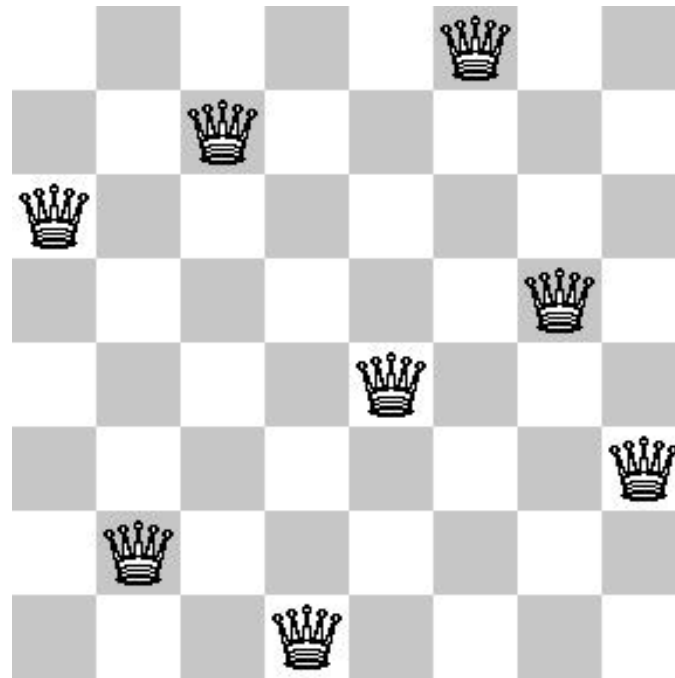
*κανίβαλοι και ιεραπόστολοι
(missionaries and cannibals)*



ποτήρια (water glass)



Το πρόβλημα των 8 βασιλισσών



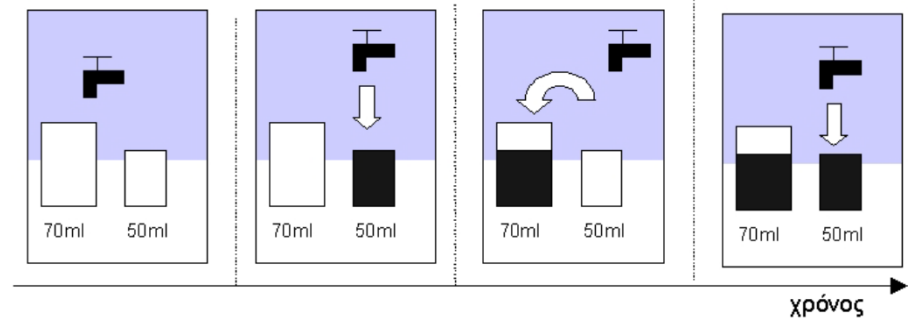
- Δύο εναλλακτικές διατυπώσεις:
 - Αυξητική
 - Πλήρεις καταστάσεις

Περιγραφή προβλημάτων

- Η περιγραφή ενός προβλήματος απαιτεί:
 - την αναπαράσταση του κόσμου του.
 - τα συγκεκριμένα χαρακτηριστικά του (αρχική κατάσταση, τελική κατάσταση, τελεστές μετάβασης)
- Ο *κόσμος* ενός προβλήματος αποτελείται από τα αντικείμενα, τις ιδιότητες των αντικειμένων και τις σχέσεις που τα συνδέουν.
- Ο κόσμος ενός προβλήματος είναι υποσύνολο του πραγματικού κόσμου περιέχοντας μόνο όσα αντικείμενα έχουν άμεση σχέση με το πρόβλημα.
- Θα εξετάσουμε *κλειστούς κόσμους* όπου κανένα νέο αντικείμενο, ιδιότητα ή σχέση δεν εισάγεται ή εξάγεται.
- Η περιγραφή ενός προβλήματος μπορεί να γίνει με *Χώρο Καταστάσεων (State Space)* ή με *Αναγωγή (Reduction)*.

Περιγραφή με Χώρο Καταστάσεων

- Κατάσταση ενός κόσμου είναι ένα στιγμιότυπο (*instance*) ή φωτογραφία (*snapshot*) μίας συγκεκριμένης χρονικής στιγμής της εξέλιξης του κόσμου.



- Αναπαριστούμε μια κατάσταση περιγράφοντας τα χαρακτηριστικά της, με τρόπο αποδεκτό & αξιοποιήσιμο από ένα σύστημα.
- Κατάσταση (*state*) ενός κόσμου είναι μία επαρκής αναπαράσταση του κόσμου σε μία δεδομένη χρονική στιγμή.
- "Επαρκής": οι ιδιότητες μιας κατάστασης πρέπει να επιλεγούν με τέτοιο τρόπο ώστε διαφορετικές τιμές των ιδιοτήτων να αντικατοπτρίζουν τις πραγματικές διαφορές των στιγμιότυπων του κόσμου.
- Για να περιγραφεί μια κατάσταση αφαιρούνται όλες οι λεπτομέρειες που δεν είναι απαραίτητες για την επίλυση του προβλήματος (*abstraction*).

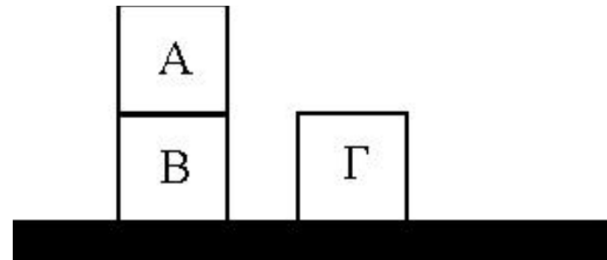
Παράδειγμα: ο κόσμος των κύβων

- Κόσμος του προβλήματος: Τρεις κύβοι και ένα τραπέζι.

Αντικείμενα	Ιδιότητες	Σχέσεις
Κύβος Α	Κύβος Α είναι ελεύθερος	Κύβος Α πάνω στον κύβο Β
Κύβος Β	Κύβος Γ είναι ελεύθερος	Κύβος Β πάνω στο Τ
Κύβος Γ	Τ έχει αρκετό ελεύθερο χώρο	Κύβος Γ πάνω στο Τ
Τ είναι Τραπέζι	Κύβος Β δεν είναι ελεύθερος	

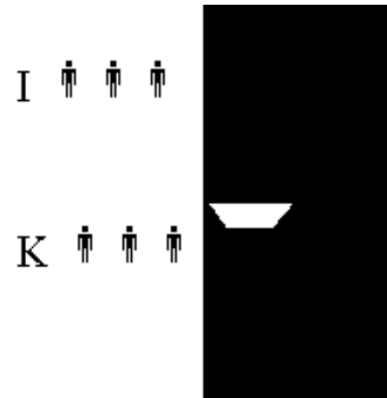
- Μια Κατάσταση:

Κύβος Α πάνω στον κύβο Β
Κύβος Β πάνω στο Τ
Κύβος Γ πάνω στο Τ
Κύβος Α ελεύθερος
Κύβος Γ ελεύθερος



- Άλλα χαρακτηριστικά (π.χ. χρώμα, βάρος, κλπ) είναι αδιάφορα.

Παράδειγμα: Ιεραπόστολοι και Καννίβαλοι



Αντικείμενα	Ιδιότητες	Σχέσεις
3 Ιεραπόστολοι 3 Καννίβαλοι Βάρκα Αριστερή Όχθη Δεξιά Όχθη	Βάρκα δύο ατόμων	Ιεραπόστολοι στην αριστερή όχθη Καννίβαλοι στην αριστερή όχθη Βάρκα στην αριστερή όχθη

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

- 181440 δυνατές καταστάσεις
- Παζλ 15 πλακιδίων: $1,3 \cdot 10^{12}$ Λύνεται εύκολα βέλτιστα
- Παζλ 24 πλακιδίων: 10^{25} Δεν λύνεται βέλτιστα

Τελεστές Μετάβασης (1/2)

- Οι καταστάσεις συνδέονται μεταξύ τους:
 - Από μια κατάσταση μπορεί να προκύψει μια νέα κατάσταση
 - Αυτό υλοποιείται με την εφαρμογή ενός *τελεστή* στην αρχική κατάσταση
- Τελεστής μετάβασης (*transition operator*) ή ενέργεια (*action*) είναι μια αντιστοίχιση μίας κατάστασης του κόσμου σε νέες καταστάσεις.
- Π.χ. στον κόσμο των κύβων τελεστές μετάβασης είναι:
 - *Βάλε τον κύβο A πάνω στον κύβο Γ.*
 - *Βάλε τον κύβο A πάνω στον κύβο B κλπ*
- Στους τελεστές χρησιμοποιούμε και μεταβλητές, π.χ:
 - *Βάλε κάποιον κύβο X πάνω σε κάποιον κύβο Y.*
- Οι τελεστές αλλάζουν την κατάσταση του κόσμου, αλλάζοντας την περιγραφή της υφιστάμενης κατάστασης.

Τελεστές Μετάβασης (2/2)

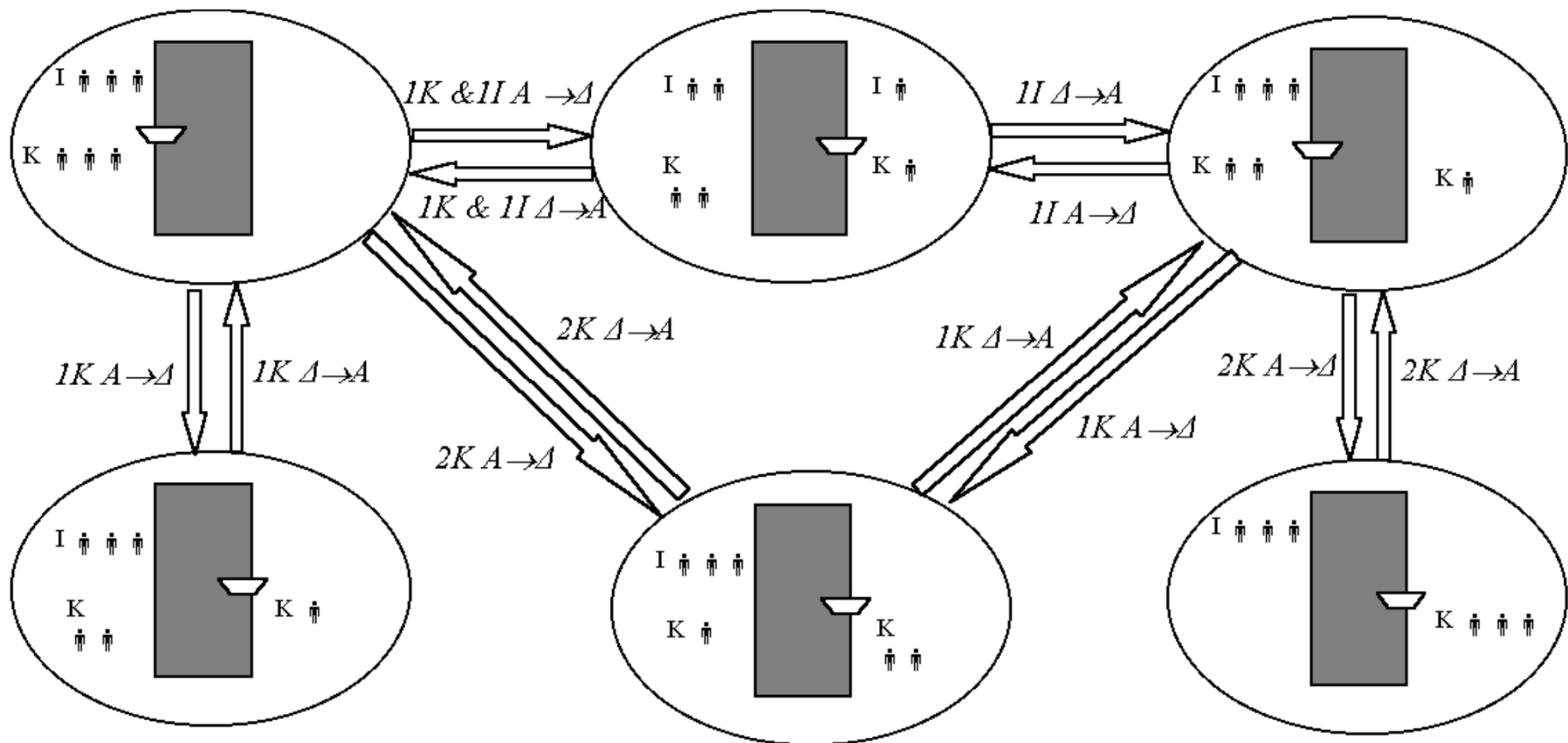
- Η κατάσταση που προκύπτει πρέπει να είναι Έγκυρη (*valid*)
- Ο τελεστής έχει συνήθως *προϋποθέσεις εφαρμογής* (*preconditions*) που πρέπει να ισχύουν για να εφαρμοστεί.

π.χ:

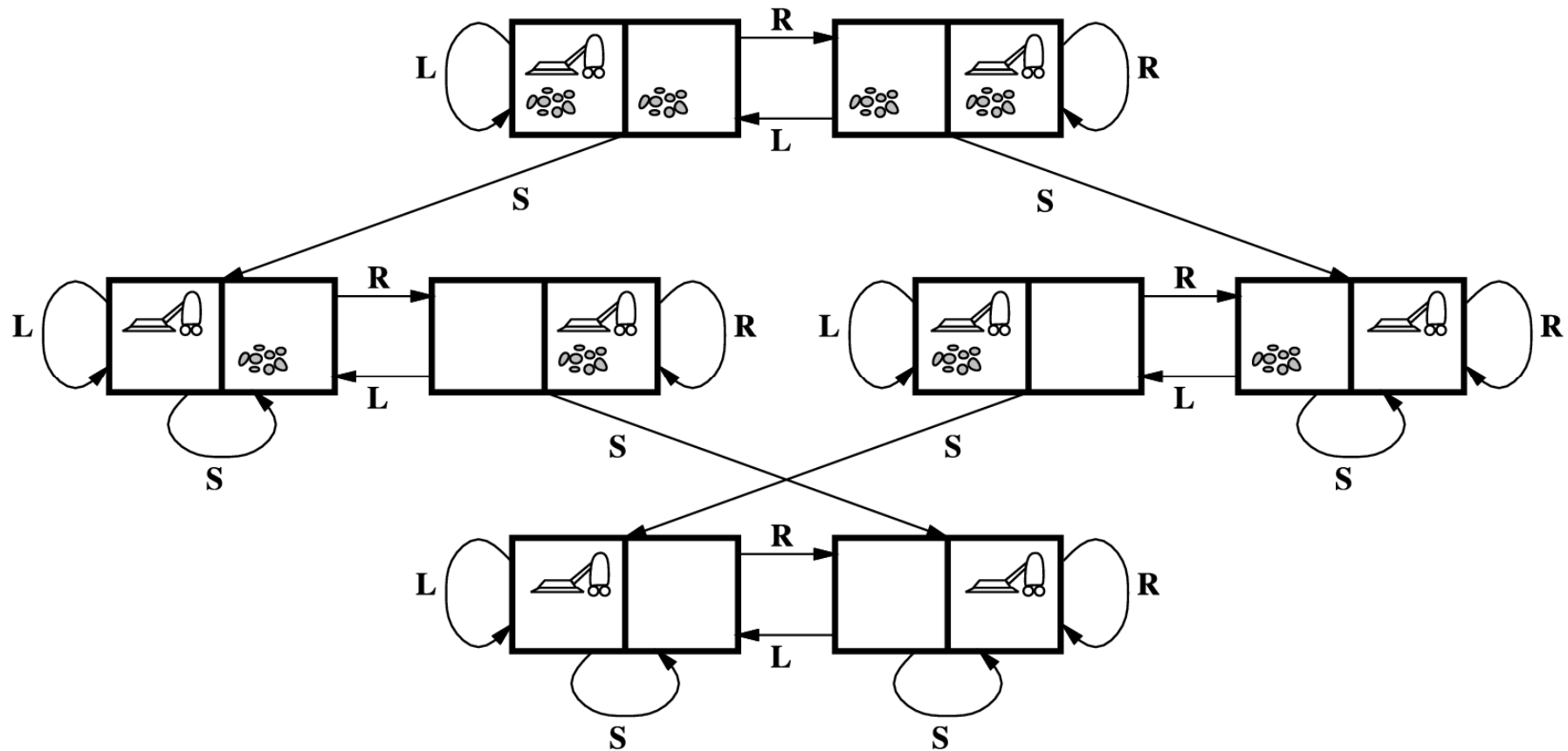
Τελεστής:
Μετέφερε δύο ιεραπόστολους από την αριστερή όχθη στη δεξιά
Προϋποθέσεις:
Υπάρχουν τουλάχιστον 2 ιεραπόστολοι στην αριστερή όχθη. Η βάρκα είναι στην αριστερή όχθη. Ο αριθμός των ιεραποστόλων που θα προκύψει στην αριστερή όχθη να μην είναι μικρότερος από τον αριθμό των κανιβάλων ή να μην υπάρχει άλλος ιεραπόστολος στην αριστερή όχθη.
Αποτελέσματα:
Ο αριθμός των ιεραποστόλων στην αριστερή όχθη μειώνεται κατά 2. Ο αριθμός των ιεραποστόλων στην δεξιά όχθη αυξάνεται κατά 2. Η βάρκα είναι πλέον δεξιά και όχι αριστερά

Χώρος Καταστάσεων (state ή domain space)

- Χώρος καταστάσεων ενός προβλήματος ονομάζεται το σύνολο όλων των έγκυρων καταστάσεων.
- Συνήθως απεικονίζεται με ένα γράφο (κόμβος: κατάσταση, ακμή: τελεστής)



Κόσμος της ηλεκτρικής σκούπας



L=Αριστερά, R=Δεξιά, S=Αναρρόφηση

Αρχικές και Τελικές καταστάσεις

- Η αρχική (*initial state*) και τελική (*final* ή *goal state*) εκφράζουν το δεδομένο και το ζητούμενο αντίστοιχα.
- Σε κάποια προβλήματα η τελική κατάσταση είναι πλήρως γνωστή (puzzle), σε άλλα είναι γνωστές κάποιες επιθυμητές ιδιότητές της (σκάκι).

8	3	5
4	1	7
2		6

Αρχική Κατάσταση

1	2	3
4	5	6
7	8	

Τελική Κατάσταση

- Ένα πρόβλημα (*Problem*) ορίζεται ως η τετράδα $P = (I, G, T, S)$ όπου:
 - S είναι ο χώρος καταστάσεων.
 - I είναι η αρχική κατάσταση, $I \in S$
 - G είναι το σύνολο των τελικών καταστάσεων, $G \subseteq S$
 - T είναι το σύνολο των τελεστών μετάβασης, $T: S \rightarrow S$

Λύση προβλήματος

- Λύση (*Solution*) σε ένα πρόβλημα (I, G, T, S), είναι μία ακολουθία από τελεστές μετάβασης

$t_1, t_2, \dots, t_n \in T$ με την ιδιότητα $g = t_n(\dots(t_2(t_1(I))))$, όπου $g \in G$

- δηλαδή η ακολουθία τελεστών που εφαρμόζονται στην αρχική κατάσταση για να προκύψει η τελική κατάσταση.

Παράδειγμα:

- Μετέφερε 1 ιεραπόστολο και 1 κανίβαλο από την αριστερή στη δεξιά όχθη
- Μετέφερε 1 ιεραπόστολο από τη δεξιά στην αριστερή όχθη
- Μετέφερε 2 κανίβαλους από την αριστερή στη δεξιά όχθη
- Μετέφερε 1 κανίβαλο από τη δεξιά στην αριστερή όχθη
- Μετέφερε 2 ιεραπόστολους από την αριστερή στη δεξιά όχθη
- Μετέφερε 1 ιεραπόστολο και 1 κανίβαλο από τη δεξιά στην αριστερή όχθη
- Μετέφερε 2 ιεραπόστολους από την αριστερή στη δεξιά όχθη
- Μετέφερε 1 κανίβαλο από τη δεξιά στην αριστερή όχθη
- Μετέφερε 2 κανίβαλους από την αριστερή στη δεξιά όχθη
- Μετέφερε 1 ιεραπόστολο από τη δεξιά στην αριστερή όχθη
- Μετέφερε 1 ιεραπόστολο και 1 κανίβαλο από την αριστερή στη δεξιά όχθη

Κατηγορίες προβλημάτων

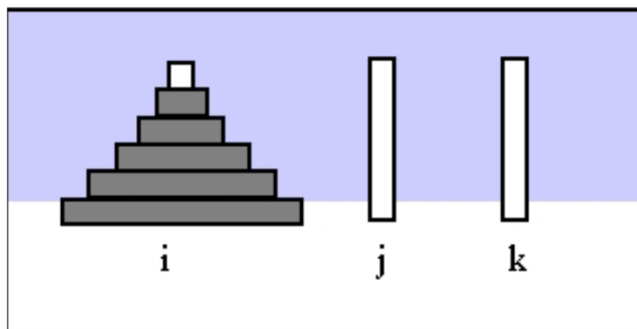
- Προβλήματα όπου είναι πλήρως γνωστές οι τελικές καταστάσεις και επιδιώκεται η εύρεση μίας σειράς ενεργειών:
 - *προβλήματα σχεδιασμού ενεργειών (planning), πλοήγησης, στρατηγικής, εφοδιαστικής κ.α.*
- Προβλήματα όπου είναι γνωστές κάποιες ιδιότητες μόνο της τελικής κατάστασης και επιδιώκεται η εύρεση ενός πλήρους στιγμιότυπου της τελικής κατάστασης
 - *προβλήματα χρονοπρογραμματισμού (scheduling), σταυρόλεξα, κρυπτογραφικά, γνωστά και ως προβλήματα ικανοποίησης περιορισμών (constraint satisfaction problems).*
- Προβλήματα όπου είναι σχετικά εύκολο να βρεθούν λύσεις, αλλά το ζητούμενο είναι η βέλτιστη από αυτές.
 - *προβλήματα βελτιστοποίησης: η τελική κατάσταση δεν είναι πλήρως γνωστή - είναι γνωστά κάποια χαρακτηριστικά της.*

Περιγραφή προβλημάτων με Αναγωγή (1/2)

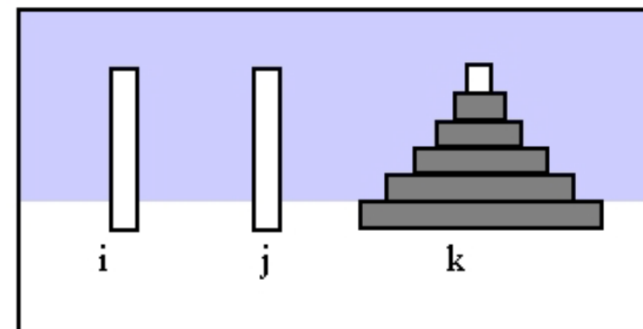
- Βασική δομή: η ίδια η περιγραφή του προβλήματος.
- Γενική ιδέα: αν δοθεί μια αρχική περιγραφή του προβλήματος τότε μια ακολουθία από τελεστές ανάγουν την περιγραφή σε υποπροβλήματα (αρχέγονα) τα οποία είναι άμεσα επιλύσιμα.

π.χ. οι πύργοι του Ανόι:

- Για να μεταφερθούν $n > 1$ δίσκοι από τον στύλο i στο στύλο k , πρέπει:
 - να μεταφερθούν $n-1$ δίσκοι από το i στο j ,
 - να μεταφερθεί 1 δίσκος από το i στο k ,
 - να μεταφερθούν $n-1$ δίσκοι από το j στο k .



Αρχική Περιγραφή



Τελική Περιγραφή

Περιγραφή προβλημάτων με Αναγωγή (2/2)

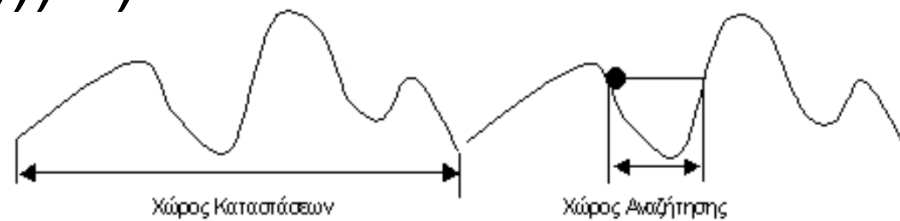
- Ένας *Τελεστής Αναγωγής* (reduction operator) ανάγει ένα πρόβλημα σε υποπροβλήματα:
 - Για να λυθεί το πρόβλημα πρέπει να λυθούν όλα τα υποπροβλήματα
- Μπορεί να υπάρχουν πολλοί διαφορετικοί τελεστές αναγωγής που να εφαρμόζονται σε ένα πρόβλημα.
- Στην περιγραφή με αναγωγή ένα *πρόβλημα* ορίζεται τυπικά ως η τετράδα $P = (ID, GD, TR, PP)$, όπου:
 - ID είναι η *αρχική περιγραφή*,
 - GD είναι ένα σύνολο από *τελικές περιγραφές*,
 - TR είναι ένα σύνολο *τελεστών αναγωγής*
 - PP είναι ένα σύνολο από *αρχέγονα προβλήματα*.
- Προβλήματα με *χωρική διάσπαση* λύνονται με αναγωγή.
- π.χ. εύρεση διαδρομής σε λαβύρινθο που αποτελείται από ένα αριθμό διαμερισμάτων

Αλγόριθμοι Αναζήτησης (1/2)

- Οι αλγόριθμοι που αναζητούν τη λύση σε ένα πρόβλημα ονομάζονται *αλγόριθμοι αναζήτησης* (*search algorithms*)
- Η επιλογή ενός αλγορίθμου αναζήτησης για ένα συγκεκριμένο πρόβλημα είναι σημαντική, διότι οι αλγόριθμοι αυτοί διαφέρουν μεταξύ τους σε αρκετά χαρακτηριστικά.
- Δοθέντος ενός προβλήματος (I, G, T, S) , *χώρος αναζήτησης* (*search space*) SP είναι το σύνολο όλων των καταστάσεων που είναι προσβάσιμες από την αρχική κατάσταση.

Αλγόριθμοι Αναζήτησης (2/2)

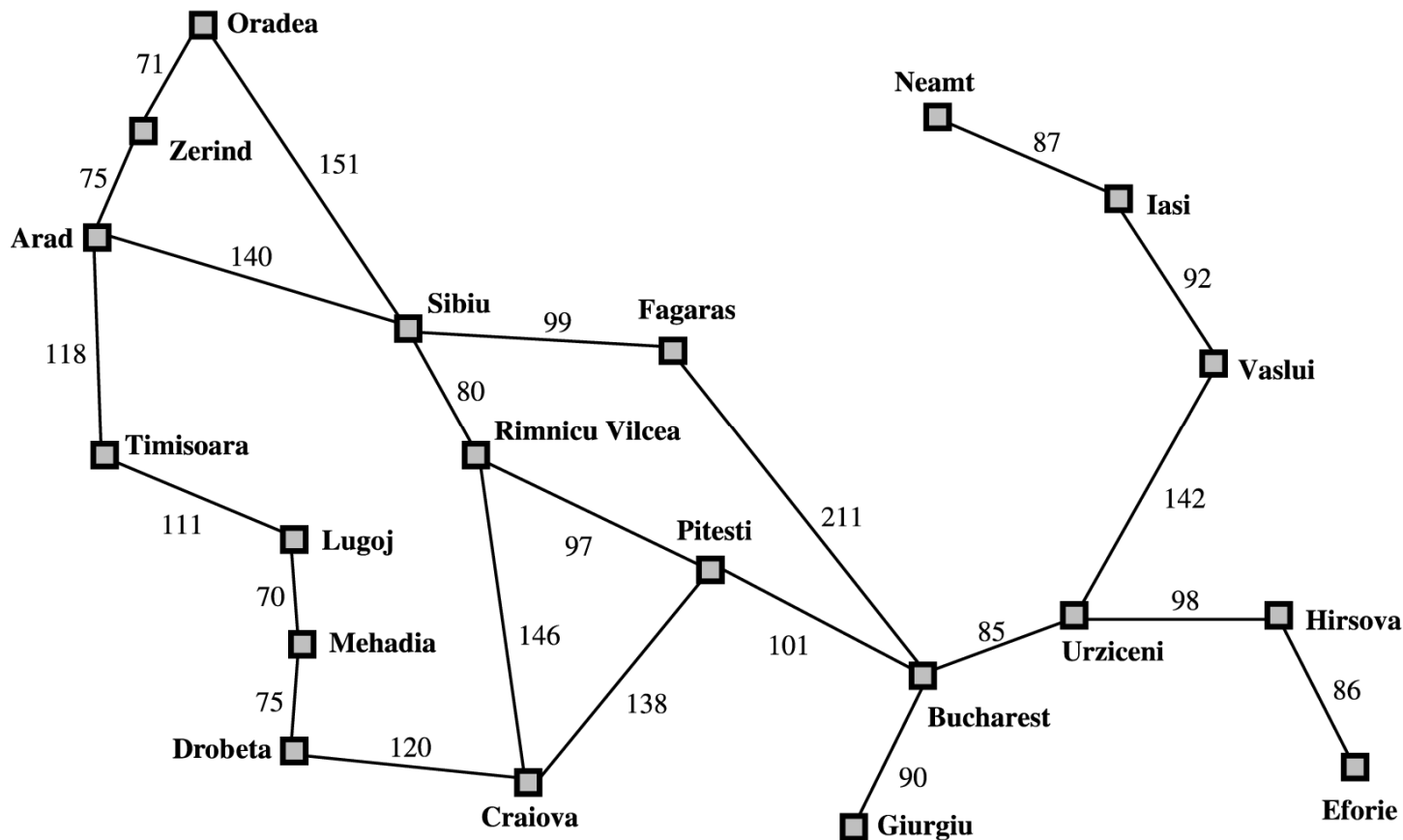
- Μία κατάσταση s ονομάζεται *προσβάσιμη* (*accessible*) αν υπάρχει μια ακολουθία τελεστών μετάβασης $t_1, t_2, \dots, t_k \in T$ τέτοια ώστε $s = t_k(\dots(t_2(t_1(I)))\dots)$.



- Ο χώρος αναζήτησης είναι υποσύνολο του χώρου καταστάσεων, δηλαδή $SP \subseteq S$.
- Ένας αλγόριθμος αναζήτησης δεν μειώνει τον χώρο αναζήτησης (που είναι δεδομένος) αλλά καθορίζει τον αριθμό των καταστάσεων που επισκέπτεται.

Πράκτορες επίλυσης προβλημάτων (1/2)

- Διατύπωση στόχου: Σύνολο επιθυμητών καταστάσεων του κόσμου
- Διατύπωση προβλήματος: ποιες ενέργειες & καταστάσεις εξετάζονται
 - Επιλογή επιπέδου λεπτομέρειας (αφαίρεση)



Πράκτορες επίλυσης προβλημάτων (2/2)

- 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.
- Η διαδικασία εύρεσης μιας τέτοιας ακολουθίας ονομάζεται *Αναζήτηση*
- Ένας αλγόριθμος αναζήτησης παίρνει ως είσοδο ένα πρόβλημα και επιστρέφει μια Λύση (ακολουθία ενεργειών)
- Διατύπωση – Αναζήτηση – Λύση - Εκτέλεση

Απλός πράκτορας επίλυσης προβλημάτων

- **function** Simple-Problem-Solving-Agent(*αντίληψη*) **returns** μια ενέργεια
 inputs: *αντίληψη*, μια αντίληψη
 static: *ακολουθία*, ακολουθία αντιλήψεων, αρχικά κενή
 κατάσταση, περιγραφή της τρέχουσας κατάστασης του
 κόσμου
 στόχος, ένας στόχος, αρχικά κενός
 πρόβλημα, μια διατύπωση προβλήματος

 κατάσταση $\leftarrow$ Update-State(*κατάσταση*, *αντίληψη*)
 if *ακολουθία* είναι κενή **then do**
 στόχος $\leftarrow$ Formulate-Goal(*κατάσταση*)
 πρόβλημα $\leftarrow$ Formulate-Problem(*κατάσταση*, *στόχος*)
 ακολουθία $\leftarrow$ Search(*πρόβλημα*)
 ενέργεια $\leftarrow$ First(*ακολουθία*)
 ακολουθία $\leftarrow$ Rest(*ακολουθία*)
 return *ενέργεια*
- Παραδοχές: Στατικό, παρατηρήσιμο, διακριτό, αιτιοκρατικό περιβάλλον

Ορισμός Προβλήματος

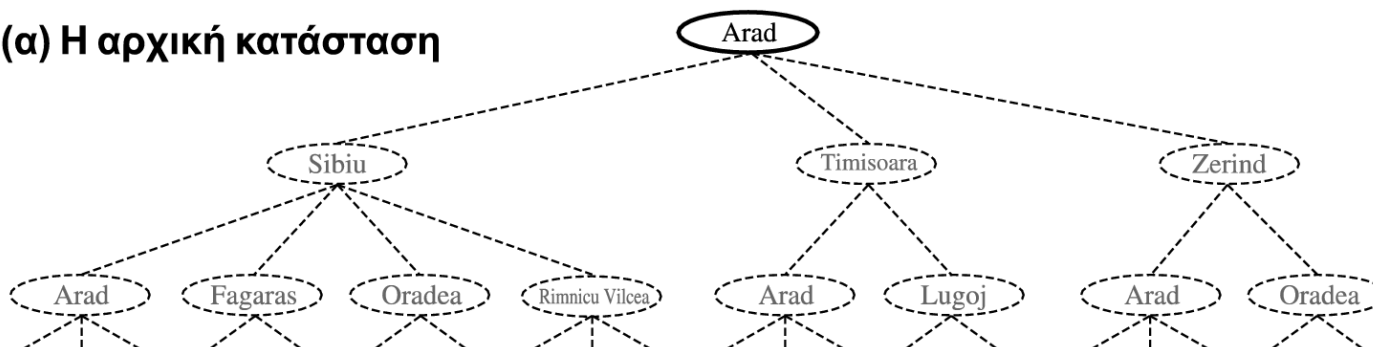
- Συνιστώσες προβλήματος:
 - Αρχική κατάσταση
 - π.χ. *Εντός(Arad)*
 - Ενέργειες, συνάρτηση διαδόχων $\text{Successor-Fn}(x)$
 - π.χ. $\text{Successor-Fn}(\text{Arad}) = \{ \langle \text{Μετάβαση}(\text{Sibiu}), \text{Εντός}(\text{Sibiu}) \rangle, \langle \text{Μετάβαση}(\text{Timisoara}), \text{Εντός}(\text{Timisoara}) \rangle, \langle \text{Μετάβαση}(\text{Zerind}), \text{Εντός}(\text{Zerind}) \rangle \}$
 - Χώρος καταστάσεων, αναπαράσταση με γράφημα, διαδρομές
 - Έλεγχος στόχου
 - Ρητή απαρίθμηση καταστάσεων ή περιγραφή ιδιοτήτων
 - Κόστος διαδρομής
 - Κόστος βήματος, $c(x,a,y)$
- Λύση: Διαδρομή από αρχική κατάσταση σε κατάσταση στόχου (Βέλτιστη λύση)

Χώρος Αναζήτησης ως Δένδρο Αναζήτησης

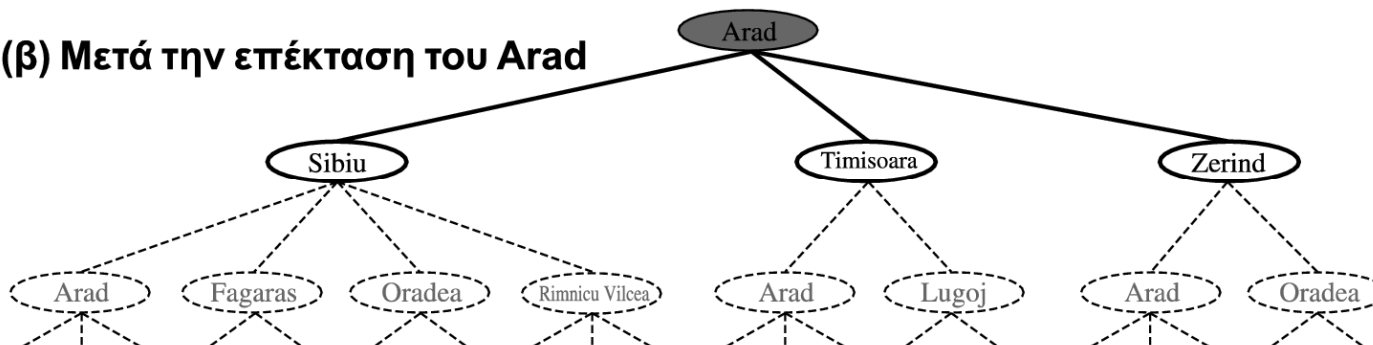
Τμήμα Δένδρου	Αναπαράσταση
Κόμβος (Node)	Κατάσταση
Ρίζα (Root)	Αρχική Κατάσταση
Φύλλο (Tip, Leaf) ή Τερματικός κόμβος	Τελική Κατάσταση ή Αδιέξοδο (Dead Node), δηλαδή κατάσταση στην οποία δεν μπορεί να εφαρμοστεί κανένας τελεστής μετάβασης
Κλαδί (Branch)	Τελεστής Μετάβασης που μετατρέπει μια κατάσταση-Γονέα (Parent State) σε μία άλλη κατάσταση-Παιδί (Child State).
Λύση (Solution)	Μονοπάτι (Path) που ενώνει την αρχική με μία τελική κατάσταση
Επέκταση (Expansion)	Η διαδικασία παραγωγής όλων των καταστάσεων-παιδιών ενός κόμβου.
Παράγοντας Διακλάδωσης (Branching Factor)	Ο αριθμός των καταστάσεων-Παιδιών που προκύπτουν από την επέκταση μιας κατάστασης. Επειδή δεν είναι σταθερός αριθμός, αναφέρεται & ως Μέσος Παράγοντας Διακλάδωσης.

Δένδρο αναζήτησης

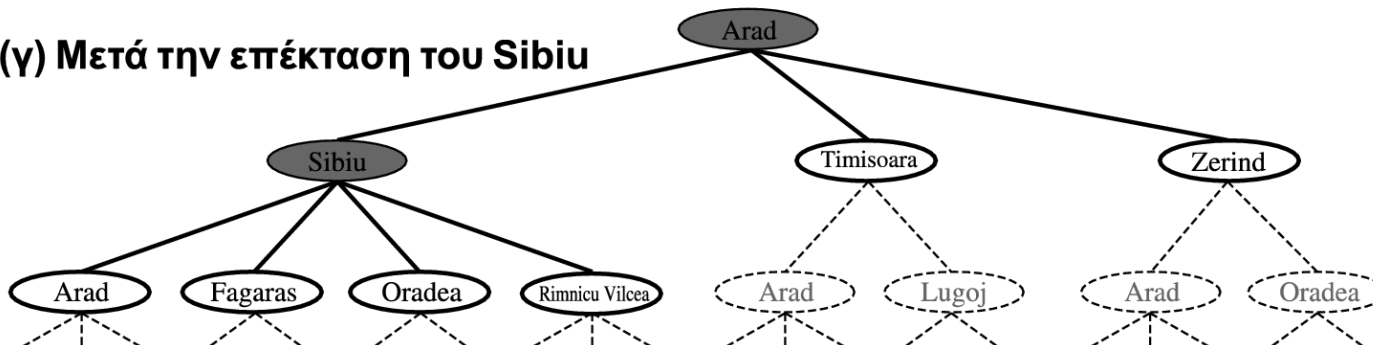
(α) Η αρχική κατάσταση



(β) Μετά την επέκταση του Arad



(γ) Μετά την επέκταση του Sibiu



Άτυπος αλγόριθμος αναζήτησης

function Tree-Search(*πρόβλημα, στρατηγική*) **returns** μια λύση
ή αποτυχία

αρχικοποίηση του δένδρου αναζήτησης με χρήση
της αρχικής κατάστασης του *προβλήματος*

loop do

if δεν υπάρχουν υποψήφιοι για επέκταση

then return αποτυχία

else επιλογή ενός κόμβου-φύλλου για να
επεκταθεί, σύμφωνα με τη *στρατηγική*

if ο κόμβος περιέχει μια κατάσταση στόχου

then return την αντίστοιχη λύση

else ο κόμβος επεκτείνεται και οι κόμβοι που
προκύπτουν προστίθενται στο δένδρο αναζήτησης

Μέτρηση απόδοσης αλγορίθμων αναζήτησης

- Πληρότητα (Completeness)
- Βέλτιστη συμπεριφορά (Optimality)
- Χρονική πολυπλοκότητα (Time complexity)
- Χωρική πολυπλοκότητα (Space complexity)
- Μέγεθος προβλήματος:
 - Παράγοντας διακλάδωσης, b (branching factor)
 - Βάθος d (depth) του πιο ρηχού κόμβου στόχου
 - Μέγιστο μήκος, m , οποιασδήποτε διαδρομής του χώρου καταστάσεων.