

3.7 Παραδείγματα Εφαρμογής Τυφλής Αναζήτησης

3.7.1 Αεροπορική Εταιρία Αναθέτει Πτήσεις στους Πιλότους της

Στο πρόβλημα της ανάθεσης πτήσεων σε πιλότους που ορίστηκε στο προηγούμενο κεφάλαιο, ο χώρος αναζήτησης έχει τα εξής χαρακτηριστικά:

- Είναι πεπερασμένος γιατί δεν υπάρχει μονοπάτι (αλληλουχία διαδοχικών εφαρμογών του τελεστή μετάβασης) που να έχει άπειρο μήκος.
- Ο παράγοντας διακλάδωσης στην αρχική κατάσταση είναι ίσος με τον αριθμό των πιλότων N επί τον αριθμό των διαφορετικών πτήσεων που μπορεί να τους ανατεθούν $N/2$, δηλαδή όλοι οι πιθανοί συνδυασμοί πρώτων αναθέσεων.
- Το μεγαλύτερο μονοπάτι (αυτό της λύσης) έχει μήκος όσο ο αριθμός των πιλότων, δηλαδή N .
- Ο παράγοντας διακλάδωσης στα παρακάτω επίπεδα είναι μικρότερος από τον παραπάνω και μάλιστα μειώνεται σε κάθε επίπεδο εφόσον οι επιλογές ανάθεσης πτήσεων σε διαθέσιμους πιλότους λιγοστεύουν.
- Δεν υπάρχει καλύτερη ή χειρότερη λύση με την έννοια του μήκους της λύσης εφόσον όλες οι λύσεις περιέχουν τον ίδιο αριθμό τελεστών μετάβασης.

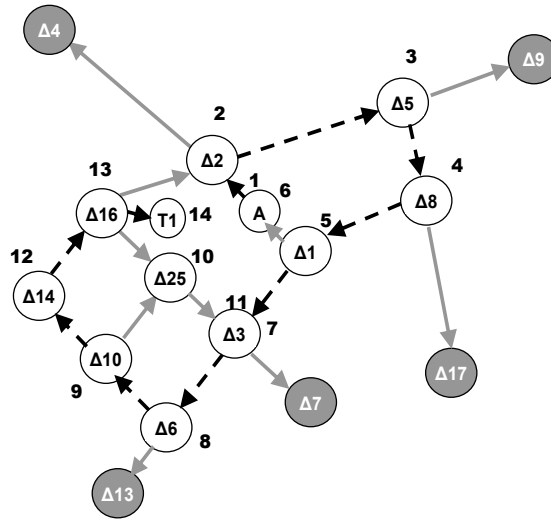
Από τα παραπάνω φαίνεται ότι η μορφή του δένδρου αναζήτησης είναι ένα πλατύ δένδρο που έχει μικρό ρυθμό ανάπτυξης και πεπερασμένο βάθος. Αν θέλουμε να επιλέξουμε έναν αλγόριθμο αναζήτησης, θα αποκλείαμε τον BFS, καθώς το πλάτος του δένδρου είναι μεγάλο αλλά και δεν υπάρχει καλύτερη (ως προς το μήκος) λύση στο πρόβλημα, καθώς όλες οι λύσεις βρίσκονται στο ίδιο βάθος. Αν ο αριθμός N είναι σχετικά μικρός, ο αλγόριθμος DFS θα ήταν μία καλή επιλογή γιατί το δένδρο αναζήτησης είναι πεπερασμένο και το μέτωπο της αναζήτησης δεν μεγαλώνει πολύ. Επίσης, ο αλγόριθμος ID δεν έχει νόημα γιατί εκλείπουν οι λόγοι χρησιμοποίησής του. Ο αλγόριθμος BiS δεν θα μπορούσε να εφαρμοστεί γιατί η τελική κατάσταση δεν είναι πλήρως γνωστή, δηλαδή δεν ξέρουμε την πλήρη ανάθεση πιλότων σε πτήσεις.

Το παραπάνω πρόβλημα μπορεί να επεκταθεί με την επιπλέον υπόθεση ότι κάθε πιλότος κοστίζει στην εταιρία διαφορετικά. Το συνολικό κόστος για την εταιρεία σε μισθούς ανά ημέρα είναι το άθροισμα όλων των ημερομισθίων. Συνεπώς η εταιρία θα επιθυμούσε να βρει μία λύση με το μικρότερο συνολικό ημερήσιο κόστος.

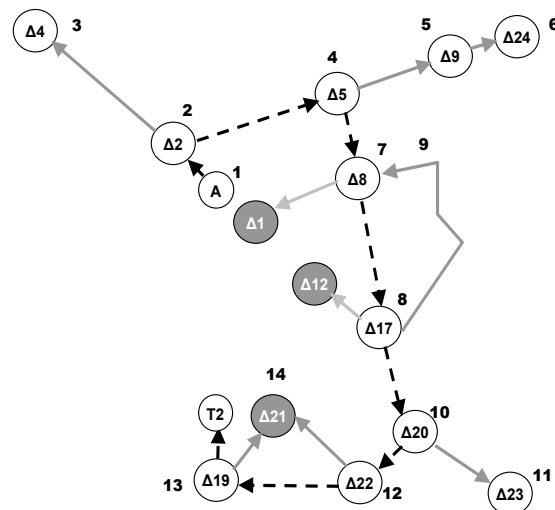
Στην περίπτωση αυτή η αναπαράσταση του προβλήματος θα άλλαζε ώστε να περιλαμβάνει και το κόστος κάθε πιλότου. Για να βρούμε τη βέλτιστη λύση στο πρόβλημα, θα μπορούσαμε να εφαρμόσουμε τον αλγόριθμο B&B. Η βασική του ιδέα είναι να κλαδευτούν μονοπάτια του δένδρου αναζήτησης που δεν οδηγούν σε καλύτερη λύση. Ο αλγόριθμος, αναζητώντας με αναζήτηση κατά βάθος θα έφτανε σε μία λύση με κάποιο συνολικό κόστος για την εταιρία, που είναι το συνολικό κόστος των ημερομισθίων των πιλότων. Ο αλγόριθμος θα συνέχιζε την αναζήτηση και σε άλλα μονοπάτια. Αν το μερικό κόστος σε κάποια κατάσταση ξεπερνούσε αυτό της μέχρι τώρα λύσης τότε δεν θα συνέχιζε σε αυτό το μονοπάτι. Αν βρισκόταν καλύτερη λύση, με μικρότερο συνολικό κόστος, θα κρατούσε τη νέα λύση ως κριτήριο για περαιτέρω κλάδεμα. Η αναζήτηση θα τελείωνε όταν όλα τα μονοπάτια έχουν εξερευνηθεί ή έχουν κλαδευτεί.

3.7.2 Εύρεση Διαδρομής

Έστω ότι εφαρμόζεται ο αλγόριθμος DFS στο χώρο αναζήτησης του προβλήματος εύρεσης διαδρομής σε μία πόλη που ορίστηκε στο ΚΕΦΑΛΑΙΟ 2. Για να υπάρχει συγκεκριμένος τρόπος με τον οποίο αναπτύσσεται ο χώρος αναζήτησης πρέπει οι τελεστές που εφαρμόζονται σε μία κατάσταση να είναι διατεταγμένοι. Αν για παράδειγμα η διάταξη είναι <δεξιά, ευθεία, αριστερά> όπως κατευθύνεται το αυτοκίνητο, η αναζήτηση προχωρά με τη σειρά που φαίνεται στο Σχήμα 3.11. Στο ίδιο σχήμα φαίνονται η λύση (διακεκομμένη γραμμή), οι καταστάσεις που έχουν επεκταθεί (λευκό φόντο) και οι καταστάσεις που έχουν απομείνει στο μέτωπο της αναζήτησης μετά την εύρεση της τελικής κατάστασης.



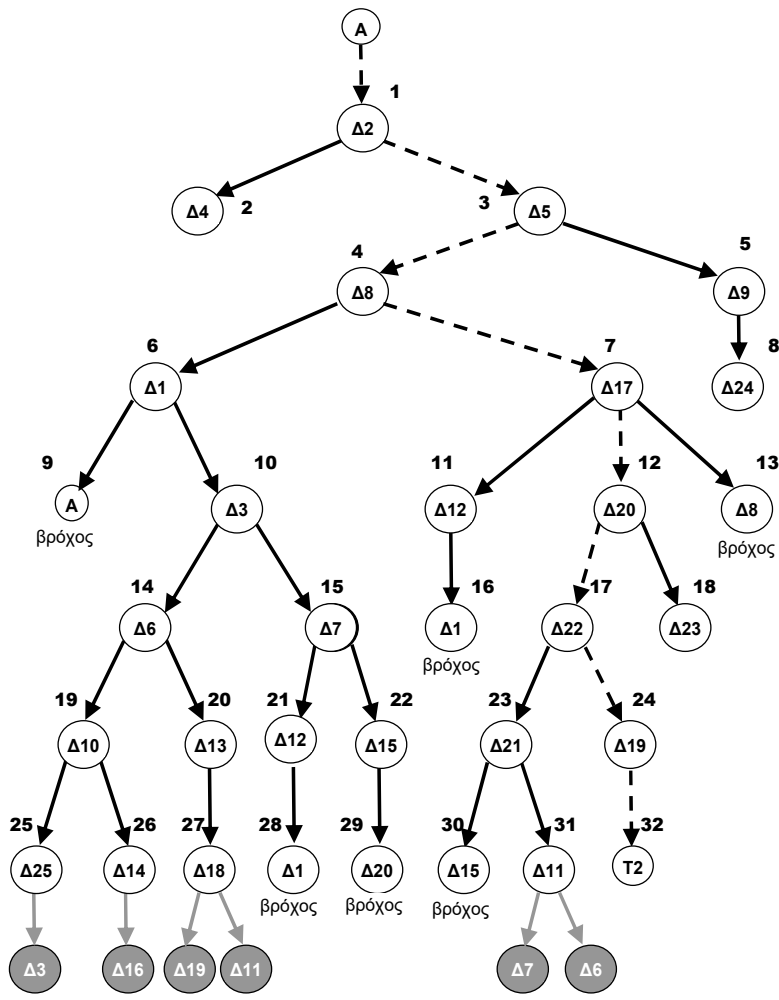
Σχήμα 3.11: Ο χώρος αναζήτησης και η λύση του προβλήματος μετά την εφαρμογή του αλγόριθμου DFS και με επιλογή τον δεξιότερο δρόμο που ξεκινά από μία διασταύρωση.



Σχήμα 3.12: Ο χώρος αναζήτησης και η λύση του προβλήματος με εφαρμογή του DFS και επιλογή του αριστερότερου δρόμου σε κάθε διασταύρωση.

Αν επιλέξουμε ότι σε κάθε διασταύρωση οι επιλογές διατάσσονται αντίθετα από την παραπάνω περίπτωση, δηλαδή η διάταξη είναι <αριστερά, ευθεία, δεξιά> έτσι όπως κατευθύνεται το αυτοκίνητο, η αναζήτηση προχωρά με τη σειρά που φαίνεται στο Σχήμα 3.12. Είναι ενδιαφέρον να παρατηρήσει κανείς ότι ο αλγόριθμος βρίσκει διαφορετική λύση (αυτή που οδηγεί στην τελική T2) από αυτήν της παραπάνω περίπτωσης (τελική T1).

Στο Σχήμα 3.13 απεικονίζεται η εφαρμογή του αλγορίθμου BFS στον ίδιο χώρο. Στο σχήμα αυτό, για λόγους κατανόησης, ο χώρος απεικονίζεται ως δένδρο. Ο αλγόριθμος βρίσκει την καλύτερη (μικρότερος αριθμός διασταυρώσεων) λύση, αυτή που οδηγεί στην τελική κατάσταση T2. Όπου υπάρχει βρόχος, η κατάσταση δεν επεκτείνεται περαιτέρω. Στη συγκεκριμένη περίπτωση αν ο χώρος δεν ήταν πεπερασμένος ο αλγόριθμος θα κατέληγε και πάλι σε λύση χωρίς δυσκολία.



Σχήμα 3.13: Ο χώρος αναζήτησης και η λύση του προβλήματος με εφαρμογή του BFS και επιλογή του δεξιότερου δρόμου σε κάθε διασταύρωση.

Στο Σχήμα 3.14 απεικονίζεται διαγραμματικά μέρος της αναζήτησης κατά την εφαρμογή του αλγορίθμου ID. Προφανώς η αναζήτηση θα συνεχιστεί μέχρι το επίπεδο