

Η δήλωση της βεβαιότητας των γεγονότων μπορεί να γίνει είτε με συντελεστές, οπότε υιοθετείται η συλλογιστική πιθανοτήτων ή με λεκτικούς χαρακτηρισμούς της μορφής "πολύ πιθανό", "πιθανό", "λίγο πιθανό", κτλ., οπότε υιοθετείται η *ασαφής λογική*. Να σημειωθεί ότι αβεβαιότητα μπορεί να υπάρχει και ως προς τα αποτελέσματα των ενεργειών. Γενικά, σε περιπτώσεις ύπαρξης αβεβαιότητας απαιτούνται ειδικοί σχεδιαστές, όπως οι *σχεδιαστές πιθανοτήτων* (*probability planners*) και οι *σχεδιαστές πολλαπλών ενδεχομένων* (*contingency planners*).

Η αναπαράσταση του κόσμου δεν είναι πλήρης επειδή αυτή δεν περιέχει γνώση για όλες τις παραμέτρους του προβλήματος ή του κόσμου γενικότερα αλλά μόνο για αυτές που θεωρείται ότι μπορούν να επηρεάσουν τη λύση του συγκεκριμένου προβλήματος. Για παράδειγμα, στην περιγραφή της αρχικής κατάστασης του προβλήματος των κύβων δεν δηλώνονται τα χρώματα των κύβων ούτε η εξωτερική θερμοκρασία. Τα γεγονότα που συμπεριλαμβάνονται στην αναπαράσταση ενός προβλήματος αποτελούν το *πλαίσιο* του (*frame*). Η ύπαρξη στην αρχική περιγραφή του κόσμου γεγονότων που είναι εντελώς ανεξάρτητα από το πρόβλημα, επιβαρύνει άσκοπα την υπολογιστική προσπάθεια για την επίλυσή του.

15.1.7 Το Πρόβλημα του Πλαισίου και η Αναπαράσταση STRIPS

Τελειώνοντας την ενότητα για την αναπαράσταση των προβλημάτων σχεδιασμού, θα ήταν σκόπιμο να αναφερθεί ο λόγος για τον οποίο επικράτησε η STRIPS αναπαράσταση. Πριν από την εισαγωγή της, η μορφή του σχεδιασμού που επικρατούσε ήταν ο σχεδιασμός με *λογισμό καταστάσεων* (*situation calculus*), σύμφωνα με τον οποίο κάθε ενέργεια έπρεπε να ορίζει με σαφήνεια ολόκληρη την κατάσταση που θα έχει ο κόσμος μετά την εκτέλεσή της. Για παράδειγμα, μια ενέργεια που μετακινεί έναν κύβο, έπρεπε να καθορίζει όχι μόνο τη νέα θέση του μετακινούμενου κύβου, αλλά επίσης και το γεγονός ότι όλοι οι άλλοι κύβοι παρέμειναν στη θέση τους. Για κάθε λοιπόν ενέργεια έπρεπε να γραφούν πάρα πολλοί κανόνες, τα λεγόμενα *αξιώματα του πλαισίου* (*frame axioms*), τα οποία καθόριζαν ποια από τα γεγονότα του προβλήματος παρέμεναν ανεπηρέαστα κατά την εκτέλεση της ενέργειας.

Είναι φανερό ότι όσο μεγαλύτερο ήταν το πλαίσιο ενός προβλήματος, τόσο περισσότερα αξιώματα πλαισίου έπρεπε να γραφούν για τις διάφορες ενέργειες. Σαν αποτέλεσμα, υπήρχε τεράστια δυσκολία στην κωδικοποίηση μεγάλων προβλημάτων. Το πρόβλημα αυτό έμεινε γνωστό σαν το *πρόβλημα του πλαισίου* (*frame problem*) και εξετάζεται αναλυτικότερα σε σχετικό κεφάλαιο.

Το μοντέλο STRIPS παράκαμψε το πρόβλημα του πλαισίου, δηλώνοντας για κάθε ενέργεια μόνο τα γεγονότα που αυτή αλλάζει και κάνοντας την παραδοχή ότι όλα τα υπόλοιπα γεγονότα παραμένουν ανεπηρέαστα. Αυτό όμως μείωσε σημαντικά την εκφραστική δυνατότητα περιγραφής των προβλημάτων σχεδιασμού.

15.1.8 Η γλώσσα αναπαράστασης PDDL

Η γλώσσα PDDL (Planning Domain Description Language) δημιουργήθηκε με σκοπό να υπάρξει ένας ενιαίος συμβολισμός αναπαράστασης προβλημάτων σχεδιασμού, ώστε να διευκολύνει την ανάπτυξη προγραμμάτων σχεδιασμού. Στο παρακάτω παράδειγμα δίνεται μέρος του ορισμού του προβλήματος των κύβων (blocksworld). Ορίζο-

νται τα κατηγορήματα (predicates) που περιγράφουν ιδιότητες μίας κατάστασης και οι ενέργειες (actions) του προβλήματος, με τις παραμέτρους τους (parameters), τις προϋποθέσεις τους (preconditions) και τα αποτελέσματα τους (effects). Οι μεταβλητές (variables) φέρουν το πρόθεμα ?. Οι τελεστές είναι προθεματικοί (prefix), όπως για παράδειγμα ο τελεστής *and*. Στο αρχείο ορισμού (domain file) παρατίθενται τα κατηγορήματα και οι ενέργειες:

```
(define (domain blocksworld)
  (:requirements :strips)

  (:predicates (clear ?x)
               (on-table ?x)
               (holding ?x)
               (on ?x ?y))

  (:action pickup
    :parameters (?ob)
    :precondition (and (clear ?ob) (on-table ?ob))
    :effect (and (holding ?ob) (not (clear ?ob)) (not (on-table ?ob))))

  (:action putdown
    :parameters (?ob)
    :precondition (and (holding ?ob))
    :effect (and (clear ?ob) (on-table ?ob)
                 (not (holding ?ob))))
```

Οι αρχικές και τελικές καταστάσεις ορίζονται στο αρχείο προβλήματος (problem file) ως εξής:

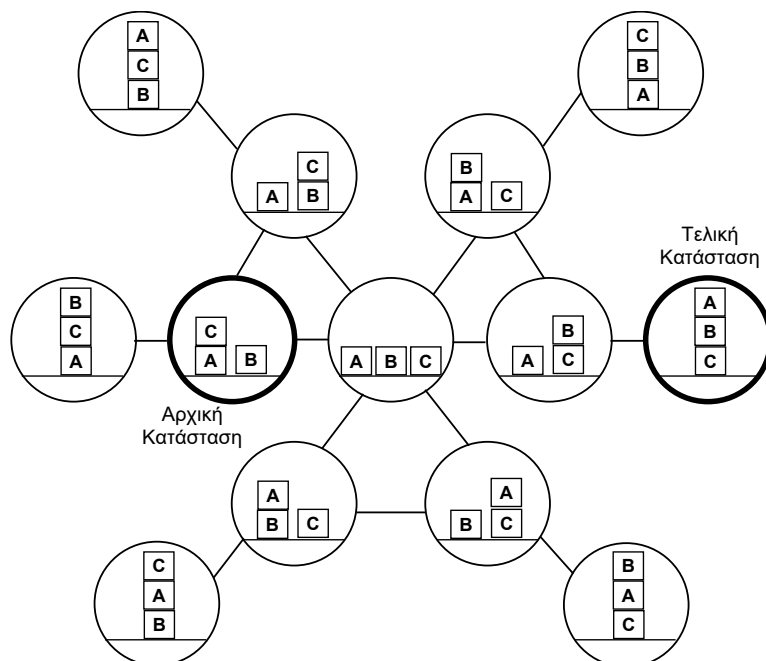
```
(define (problem blocksworld-prob1)
  (:domain blocksworld)
  (:objects a b)
  (:init (on-table a) (on-table b) (clear a) (clear b))
  (:goal (and (on a b))))
```

Η PDDL έχει αρκετές βελτιωμένες εκδόσεις και ακολούθησαν άλλες εξελιγμένες αναπαραστάσεις όπως οι PDDL+, NDDL, MAPL, ADDL, MA-PDDL και άλλες, οι οποίες προσέθεσαν επιπλέον χαρακτηριστικά στην αρχική έκδοση για την πιο αποτελεσματική αντιμετώπιση πολύπλοκων προβλημάτων σχεδιασμού.

15.2 Σχεδιασμός με Αναζήτηση στο Χώρο των Καταστάσεων

Ο πιο απλός τρόπος αντιμετώπισης ενός προβλήματος σχεδιασμού είναι να το αντιμετωπίσουμε ως αναζήτηση στο χώρο των καταστάσεων. Οι αντίστοιχοι σχεδιαστές ονομάζονται *σχεδιαστές χώρου καταστάσεων* (*state-space planners*), γιατί σε κάθε επανάληψη του αλγορίθμου αναζήτησης επιλέγουν μια κατάσταση του χώρου καταστάσεων, από την οποία παράγουν νέες καταστάσεις με την εφαρμογή κάποιων ενεργειών. Στο Σχήμα 15.8 φαίνονται όλες οι καταστάσεις του προβλήματος των τριών

κύβων που παρουσιάστηκε στο Σχήμα 15.1. Κάθε κόμβος του γράφου παριστάνει μια κατάσταση, ενώ οι ακμές μεταξύ των καταστάσεων δηλώνουν ότι η μια κατάσταση μπορεί να επιτευχθεί με εφαρμογή κάποιας ενέργειας στην άλλη κατάσταση. Γενικά, οι ακμές είναι κατευθυνόμενες, ωστόσο στο Σχήμα 15.8 αυτές σχεδιάστηκαν χωρίς βέλη υπονοώντας διπλή κατεύθυνση, επειδή στο συγκεκριμένο πρόβλημα όλες οι ενέργειες είναι αντιστρέψιμες.



Σχήμα 15.8: Ο χώρος των καταστάσεων του προβλήματος των τριών κύβων.

Το πλεονέκτημα της θεώρησης του προβλήματος σχεδιασμού σαν πρόβλημα αναζήτησης στο χώρο των καταστάσεων είναι η δυνατότητα άμεσης εφαρμογής όλων των γνωστών αλγορίθμων τυφλής αλλά και ευρετικής αναζήτησης, οι οποίοι παρουσιάστηκαν σε σχετικά κεφάλαια.

Τα τελευταία χρόνια ο σχεδιασμός στο χώρο των καταστάσεων έχει γνωρίσει ιδιαίτερη άνθηση, με την εμφάνιση ευρετικών συστημάτων σχεδιασμού που χρησιμοποιούν ευρετικές συναρτήσεις λεπτής υφής (fine grain) κατά τρόπο ανεξάρτητο από το πεδίο εφαρμογής. Οι συναρτήσεις αυτές επικεντρώνουν καλύτερα την αναζήτηση προς τους στόχους. Τα πιο γνωστά συστήματα σχεδιασμού αυτής της κατηγορίας είναι τα ASP/HSP, το σύστημα GRT και FF.

15.2.1 Διάσχιση του Χώρου Αναζήτησης

Κατά την παρουσίαση των αλγορίθμων αναζήτησης θεωρήθηκε ότι η αναζήτηση στο χώρο των καταστάσεων πραγματοποιείται ξεκινώντας από την αρχική κατάσταση και προχωρώντας προς μια τελική. Αυτή η κατεύθυνση διάσχισης του χώρου των καταστάσεων ονομάζεται *ορθή διάσχιση (progression)*. Υπάρχει ωστόσο και η δυνατότητα διάσχισης του χώρου των καταστάσεων από την τελική κατάσταση προς την αρχική η οποία ονομάζεται *ανάστροφη διάσχιση (regression)*.