Εθνικο και Καποδιστριακο Πανεπιστημιο Αθηνων
Σχολη Θετικων Επιστημων

Τμημα Πληροφορικης και Τηλεπικοινωνιων
Προγραμμα Προπτυχιακων Σπουδων

Πτυχιακή Εργασία:

Ενισχυτική Μάθηση με Χρήση Συνδυαστικής Αναζήτησης και Εφαρμογή σε Παιχνίδι Ενός Παίκτη

Φοιτητής
Διώχνος Δημήτρης
Αριθμός Μητρώου: 1098

Επιβλέπων Καθηγητής
Σταματόπουλος Παναγιώτης

Αθήνα
Φεβρουάριος 2004

Στους γονείς μου.

Ευχαριστίες

Στο σημείο αυτό θέλω να ευχαριστήσω ορισμένους ανθρώπους οι οποίοι με βοήθησαν σε διάφορα στάδια της πτυχιακής μου εργασίας. Πρώτ' απ' όλα, ευχαριστώ τον επιβλέποντα καθηγητή μου, τον κ. Παναγιώτη Σταματόπουλο. Υπήρξε ο άνθρωπος που μου έδινε έμπνευση σε όλη τη διάρκεια των προπτυχιακών μου σπουδών και ήταν πάντοτε διαθέσιμος να βοηθήσει οποιονδήποτε φοιτητή είχε κάποια απορία σχετική με τα μαθήματα. Ιδιαίτερα όσον αφορά την συνεργασία μας στην πτυχιακή, θέλω να τον ευχαριστήσω για μια ακόμη φορά για την ταχύτητα με την οποία μου έδινε απαντήσεις σε διάφορα θέματα, καθώς επίσης και για την ποιότητα των απαντήσεων, η οποία σε κάθε περίπτωση ήταν κάτι παραπάνω από πλήρης.

Επίσης, θέλω να ευχαριστήσω τον Γιώργο Μπουκέα, διδακτορικό φοιτητή του τμήματός μας. Με το Γιώργο είχαμε πολλές συζητήσεις γύρω από το αντικείμενο της πτυχιακής και πάντοτε οι επεξηγήσεις του σε ορισμένα θέματα υπήρξαν καθοριστικής σημασίας. Φυσικά, δεν μπορώ να παραλείψω τις καίριες παρατηρήσεις του σε θέματα στοιχειοθεσίας και παρουσίασης της πτυχιακής, χωρίς τις οποίες η συγκεκριμένη πτυχιακή θα ήταν πολύ χαμηλότερης ποιότητας.

Τέλος, θέλω να ευχαριστήσω όλα τα παιδιά του $\Theta(\pi^2)$, που μέσα από τις παρουσιάσεις τους και τις συζητήσεις που είχα μαζί τους, μου έδωσαν ιδέες και με βοήθησαν να εμβαθύνω πολύ πιο εύκολα στο συγκεκριμένο αντικείμενο. Ειδικότερα, θέλω να πω ένα επιπλέον «ευχαριστώ» στο Γιώργο Χριστοδούλου και στον Βασίλη Στούμπο. Με το Γιώργο, είχαμε κάποιες συζητήσεις το καλοκαίρι γύρω από το αντικείμενο, όπου εξηγώντας του ορισμένα ζητήματα με βοήθησε να κατανοήσω καλύτερα διάφορες μεθόδους. Ο Βασίλης από την άλλη, με βοήθησε να εκμεταλλευτώ στο έπακρο τις δυνατότητες του LaTeX και χωρίς την βοήθειά του πραγματικά δεν γνωρίζω αν θα μπορούσα να κάνω ορισμένα πράγματα που τώρα πια μου φαίνονται τετριμμένα.

Αθήνα, 8 Δεκεμβρίου 2003.

Περίληψη

Στην εργασία αυτή ασχολούμαστε με μια περιοχή της Τεχνητής Νοημοσύνης η οποία έχει γνωρίσει ιδιαίτερη άνθιση τις τελευταίες τρεις δεκαετίες και είναι γνωστή με το όνομα «Ενισχυτική Μάθηση (Reinforcement Learning)». Οι ιδέες οι οποίες απορρέουν από τον συγκεκριμένο κλάδο έχουν συνδυαστεί με εξαιρετική επιτυχία με ιδέες από άλλους κλάδους σε πραγματικές εφαρμογές και φαίνεται πως οι ιδέες αυτές πρέπει να είναι απαραίτητο συστατικό μεθόδων που αντιμετωπίζουν προβλήματα μεγάλης κλίμακας. Αρχικά παραθέτουμε κάποια σύντομα εισαγωγικά στοιχεία για τον αναγνώστη και περιγράφουμε βασικές έννοιες γύρω από διαδικασίες μάθησης. Στη συνέχεια, παρουσιάζονται όλες οι βασικές μέθοδοι οι οποίες αντιμετωπίζουν προβλήματα Ενισχυτικής Μάθησης. Ακολούθως, δίνουμε τους κύριους άξονες επέκτασης των βασικών μεθόδων με τεχνικές κατάστρωσης σχεδίων και αναζήτησης καθώς και τις δύο προτάσεις αλγορίθμων της συγκεκριμένης πτυχιακής οι οποίοι εντάσσονται στο συγκεκριμένο πεδίο. Εκτός από τους αλγορίθμους αυτούς, παρουσιάζουμε συνοπτικά τις κυριότερες ιδέες οι οποίες υπάρχουν μέχρι σήμερα στη συγκεκριμένη περιοχή της Ενισχυτικής Μάθησης καθώς επίσης και την εφαρμογή εκείνη η οποία είχε την μεγαλύτερη επιτυχία ενσωμάτωσης των βασικών μεθόδων Ενισχυτικής Μάθησης. Τέλος, παραθέτουμε μια εφαρμογή των αλγορίθμων που προτείναμε στο παιχνίδι SOLO.

Περιεχόμενα

Κατάλογος Σχημάτων

Κατάλογος Πινάκων

Κεφάλαιο 1

Εισαγωγή

Η συγκεκριμένη πτυχιακή έχει να κάνει με την περιοχή της Τεχνητής Νοημοσύνης η οποία ονομάζεται «Μηχανική Μάθηση (Machine Learning)» και ειδικότερα με έναν συγκεκριμένο κλάδο αυτής της περιοχής ο οποίος ονομάζεται «Ενισχυτική Μάθηση (Reinforcement Learning)». Εκτός από αυτόν τον όρο, ορισμένες φορές χρησιμοποιείται και ο όρος «Νευρο-Δυναμικός Προγραμματισμός (Neuro-Dynamic Programming»), ο οποίος εισήχθηκε μετά τη δημοσίευση του βιβλίου [7] στο συγκεκριμένο χώρο.

Η κεντρική ιδέα της συγκεκριμένης περιοχής είναι η *μάθηση* μέσα από *αλληλεπιδράσεις με το περιβάλλον*. Με τον όρο αλληλεπιδράσεις με το περιβάλλον εννοούμε οποιασδήποτε μορφής αλληλεπίδραση μπορεί να έχει κάποιος με τις πράξεις του μέσα σε ένα περιβάλλον. Με τον όρο μάθηση εννοούμε τον τρόπο με τον οποίο μαθαίνει κάποιος *πώς* και *κατά πόσο* οι πράξεις του επηρεάζουν το περιβάλλον, τους γύρω του και πολύ περισσότερο τον ίδιο του τον εαυτό.

Για παράδειγμα, όταν ένα παιδί μαθαίνει να περπατάει εξερευνεί ουσιαστικά τις διαθέσιμες κινήσεις των μελών του προκειμένου να επιτύχει σωστό βάδισμα. Τις κινήσεις τις οποίες κάνει μπορούμε να τις διαχωρίσουμε σε δύο μεγάλες κατηγορίες. Σε αυτές που έχουν καλά αποτελέσματα και σε αυτές που δεν έχουν καλά αποτελέσματα. Προφανώς, κινήσεις οι οποίες έχουν σαν αποτέλεσμα την ασφαλή μετακίνηση του παιδιού μέσα στο περιβάλλον είναι κινήσεις καλές. Μάλιστα, αυτό το καταλαβαίνει το ίδιο το παιδί, αφού με αυτόν τον τρόπο επιτυγχάνει το στόχο του ο οποίος δεν είναι άλλος από τη μετακίνηση της ίδιας του της οντότητας σε κάποια περιοχή που προτιμάει εκείνη τη στιγμή. Επιπλέον, τα πρώτα αυτά βήματα του παιδιού τα συνοδεύει *η επιβράβευση* από τους γονείς με χαμόγελα, χαρά κ.τ.λ. τα οποία ενθαρρύνουν το παιδί στο να προτιμάει *τη βάδιση* για τη μετακίνηση του από μια θέση σε κάποια άλλη στο μέλλον. Από την άλλη, η παραπάνω διαδικασία έχει και τις άσχημες στιγμές της. Αρκετές φορές το παιδί χάνει τον έλεγχο και πέφτει. Αυτό έχει σαν αποτέλεσμα το παιδί να *πονάει* κι έτσι σιγά-σιγά μαθαίνει να αποφεύγει συγκεκριμένες κινήσεις για τις οποίες ελλοχεύει ο κίνδυνος να πέσει κι άρα να πονέσει. Με τον καιρό η διαδικασία αυτή τελειοποιείται και τελικά μετά από κάποιο διάστημα το παιδί

ξέρει να περπατάει αρκετά επιτυχημένα χωρίς να πέφτει στη διαδρομή.

Η παραπάνω απλή παρατήρηση του τρόπου συμπεριφοράς του παιδιού κατά τη διάρκεια μάθησης βάδισης οδήγησε ([21]) στη δημιουργία ρομπότ τα οποία μαθαίνουν μόνα τους να περπατάνε στο χώρο χωρίς να τα καθοδηγεί κανείς προκειμένου να επιτευχθεί κάτι τέτοιο. Η διαδικασία με την οποία πραγματοποιήθηκε αυτό ήταν αρκετά απλή. Κάθε ρομπότ είχε 6 άκρα και αυτό το οποίο μπορούσε να κάνει ήταν να σηκώσει όσα από αυτά τα άκρα επιθυμούσε ή να κατεβάσει όσα από αυτά τα άκρα επιθυμούσε. Στις περιπτώσεις τώρα που οι κινήσεις των άκρων ήταν τέτοιες ώστε το ρομπότ να περπατάει με ευστάθεια, ένας αισθητήρας το καταλάβαινε αυτό και μετέδιδε μια *αίσθηση επιβράβευσης* στο ρομπότ. Από την άλλη, υπήρχαν κινήσεις τις οποίες προσπαθούσε να κάνει ένα ρομπότ και δεν ήταν εναρμονισμένες μεταξύ τους, με αποτέλεσμα το ρομπότ να πέφτει. Στις περιπτώσεις αυτές ένας αισθητήρας καταλάβαινε το γεγονός αυτό και μετέφερε στο ρομπότ μια *αίσθηση τιμωρίας*, όπως ακριβώς τα μικρά παιδιά πονάνε όταν πέφτουν, στην προσπάθειά τους να μάθουν να περπατούν.

1.1 Γενικά Χαρακτηριστικά Ενισχυτικής Μάθησης

Έτσι λοιπόν, η Ενισχυτική Μάθηση (ΕΜ) έχει να κάνει με το να μαθαίνει κάποιος τι πρέπει να κάνει. Η συνήθης τακτική είναι η δημιουργία ενός πράκτορα (agent) ο οποίος ζει μέσα σε ένα περιβάλλον. Όπως είπαμε, η αλληλεπίδραση κάποιου με το περιβάλλον τον καθοδηγεί στη μάθηση και ουσιαστικά το κύριο οδηγό στοιχείο προς αυτή την κατεύθυνση είναι η *ανταμοιβή* την οποία λαμβάνει μέσα από τις διάφορες ενέργειές του. Με τον όρο περιβάλλον ουσιαστικά εννοούμε οτιδήποτε δεν βρίσκεται υπό τον έλεγχο του πράκτορα. Από την άλλη, ο πράκτορας βρίσκεται σε διαρκή αλληλεπίδραση με το περιβάλλον και *σκοπός του είναι να μεγιστοποιεί την ανταμοιβή την οποία λαμβάνει* μέσα από τις διάφορες *ενέργειές* του. Προκειμένου όμως να επιτυγχάνεται αυτού του είδους η μεγιστοποίηση, ο πράκτορας στην πραγματικότητα μαθαίνει να συσχετίζει καταστάσεις κι ενέργειες με τις ανταμοιβές οι οποίες προκύπτουν από αυτές, όπως κάνουμε κι εμείς οι άνθρωποι αρκετές φορές.

1.1.1 Κύρια Χαρακτηριστικά

Τα δύο πιο σημαντικά όμως χαρακτηριστικά μέσα από αυτή τη διεργασία μάθησης είναι τα ακόλουθα:

- Στον πράκτορα δεν λέγεται ποια ενέργεια θα ήταν η βέλτιστη σε μια κατάσταση, όπως στις περισσότερες μορφές Μηχανικής Μάθησης, αλλά αντίθετα ο πράκτορας πρέπει *να ανακαλύψει μόνος του* ποιες ενέργειες αποφέρουν τη μέγιστη ανταμοιβή σε μια κατάσταση δοκιμάζοντάς τες.

- Ακόμη, κάποιες ενέργειες μπορεί να επηρεάζουν όχι μόνο την άμεση ανταμοιβή την οποίο αποκομίζει λόγω αλλαγής κατάστασης, αλλά επιπλέον

> και την επόμενη κατάσταση και μέσω αυτής *όλα τις πιθανές επόμενες ανταμοιβές* που μπορεί να αποκομίσει ο πράκτορας από 'κει και πέρα.

Αυτά λοιπόν τα δύο χαρακτηριστικά: *δοκιμή-και-λάθος* καθώς και *καθυστερημένη ενίσχυση* (μακροπρόθεσμη ανταμοιβή - delayed reward) είναι τα δύο πιο σημαντικά ξεχωριστά χαρακτηριστικά της ΕΜ.

Κριτική

Βάσει αυτών των χαρακτηριστικών της ΕΜ γίνεται φανερό πως υπάρχει μεγάλη διαφορά μεταξύ της Ενισχυτικής Μάθησης και της Επιβλεπόμενης Μάθησης (Supervised Learning) η οποία χρησιμοποιείται στις περισσότερες ερευνητικές περιοχές της Μηχανικής Μάθησης, της Αναγνώρισης Προτύπων και των Νευρωνικών Δικτύων. Στην περίπτωση της *Επιβλεπόμενης Μάθησης* ο πράκτορας *μαθαίνει από παραδείγματα* ορθής (βέλτιστης) συμπεριφοράς σε διάφορες καταστάσεις τα οποία του παρέχονται από έναν *επιβλέποντα*. Σε διαδραστικά προβλήματα είναι συχνά ανέφικτο να υπάρχουν παραδείγματα της προσδοκώμενης συμπεριφοράς τα οποία να είναι και βέλτιστα και να περιέχουν όλη την απαιτούμενη πληροφορία για όλες τις πιθανές καταστάσεις στις οποίες μπορεί να βρεθεί ο πράκτορας. Έτσι, σε ανεξερεύνητες περιοχές, γίνεται φανερό πως ο πράκτορας θα πρέπει να βασίζει τις εκτιμήσεις του βάσει της προγενέστερης εμπειρίας του με το περιβάλλον. Δυστυχώς όμως η Επιβλεπόμενη Μάθηση δεν είναι από μόνη της επαρκής για μάθηση μέσα από αλληλεπίδραση. Παρ'όλ'αυτά, αυτή είναι η κεντρική ιδέα γύρω από την Ενισχυτική Μάθηση.

Μια από τις προκλήσεις που υπάρχει στην ΕΜ είναι η *ισορροπία μεταξύ εξερεύνησης και εκμετάλλευσης προγενέστερης γνώσης*. Ουσιαστικά, για να αποκτήσει αρκετή ανταμοιβή (reward) ένας πράκτορας πρέπει να προτιμά ενέργειες τις οποίες έχει δοκιμάσει στο παρελθόν και οι οποίες φαίνεται να έχουν, βάσει των παλαιών παρατηρήσεων, καλά αποτελέσματα (μεγάλες τιμές ανταμοιβών). Από την άλλη όμως πως μπορεί να βρίσκει ένας πράκτορας τέτοιες καλές κινήσεις; Ο μόνος τρόπος είναι μέσω δοκιμών, δηλαδή μέσω εξερεύνησης εναλλακτικών επιλογών οι οποίες δεν φαίνονται να είναι βέλτιστες με βάση τις τρέχουσες εκτιμήσεις του πράκτορα. Με άλλα λόγια ο πράκτορας πρέπει να χρησιμοποιεί την προγενέστερή του γνώση προκειμένου να επιλέγει ενέργειες (κινήσεις) οι οποίες του προσφέρουν καλές τιμές ανταμοιβών, αλλά από την άλλη οφείλει να εξερευνά το χώρο προκειμένου να μπορεί να έχει καλύτερες εκτιμήσεις των αποτελεσμάτων των διαφόρων ενεργειών στο μέλλον. Έτσι δυστυχώς δεν γίνεται παρά η συμπεριφορά του να μην είναι διαρκώς η βέλτιστη. Προκειμένου όμως η συμπεριφορά να γίνει βέλτιστη, θα πρέπει ο πράκτορας να δοκιμάζει διάφορες ενέργειες και *σταδιακά* να προτιμάει όλο και περισσότερο αυτές που φαίνονται καλύτερες.

Ένα ακόμη χαρακτηριστικό-κλειδί της ΕΜ είναι ότι αντιμετωπίζει προβλήματα στη γενικότερή τους μορφή: Ο πράκτορας καθοδηγείται από στόχους τους οποίους πρέπει να ικανοποιήσει αλληλεπιδρώντας πολλές φορές με ένα

αβέβαιο περιβάλλον. Αντίθετα, πολλές άλλες προσεγγίσεις ασχολούνται με υποπροβλήματα ή και ειδικές περιπτώσεις ενός γενικότερου προβλήματος χωρίς να αναφέρουν πως οι τεχνικές αυτές μπορούν να ταιριάξουν στο γενικότερο πρόβλημα. Έτσι, αν και υπάρχουν πολλά καλά αποτελέσματα μέσω αυτών των τεχνικών, δυστυχώς ο περιορισμός της ενασχόλησης με μικρότερα υποπροβλήματα είναι πολύ σημαντικός. Όμως στην ΕΜ γίνεται ακριβώς το αντίθετο, ο πράκτορας αντιμετωπίζει το γενικότερο πρόβλημα και μάλιστα υποτίθεται ότι ο πράκτορας θα πρέπει να λειτουργεί καλά (αν όχι βέλτιστα) παρά το ενδεχόμενο ενός αβέβαιου (μη-ντετερμινιστικού) περιβάλλοντος. Γενικά, οποιοδήποτε παράδειγμα προβλήματος μπορεί να σκεφτεί κανείς και το οποίο περιέχει αλληλεπίδραση μεταξύ ενός πράκτορα ο οποίος παίρνει αποφάσεις και ενός περιβάλλοντος μέσα στο οποίο ο πράκτορας προσπαθεί να επιτύχει ένα στόχο, παρ΄ όλη την αβεβαιότητα η οποία ενδεχομένως υπάρχει μέσα στο περιβάλλον, είναι ένα πρόβλημα το οποίο καλείται να αντιμετωπίσει η ΕΜ.

1.2 Αναζήτηση

Όπως αναφέραμε νωρίτερα, οι βασικές ιδέες της Ενισχυτικής Μάθησης δεν είναι από μόνες τους επαρκείς για μάθηση μέσα από αλληλεπίδραση. Ένα κρίσιμο ζήτημα επομένως είναι πως μπορούμε να βοηθήσουμε τον πράκτορα προκειμένου να καλύψουμε αυτά τα κενά. Παρατηρώντας πάλι τη λειτουργία των ανθρώπων μπορούμε να εξάγουμε ένα σημαντικό χαρακτηριστικό και να το ενσωματώσουμε σε διάφορες διαδικασίες μάθησης.

Οι άνθρωποι, σχεδόν πάντα, *σκέφτονται* πριν πραγματοποιήσουν διάφορες ενέργειες και σχηματίζουν διάφορες εκτιμήσεις για τα αποτελέσματα διαφόρων ενεργειών. Μέσα από αυτή τη διαδικασία αναλύονται διάφορες καταστάσεις και συνήθως επιλέγονται ενέργειες οι οποίες φαίνεται να έχουν καλά αποτελέσματα είτε στο άμεσο μέλλον είτε στο απώτερο. Επιπλέον, μετά τη λήψη των όποιων ενεργειών, οι άνθρωποι *κρίνουν* κατά πόσο τα αποτελέσματα ήταν κοντά στις προβλέψεις τους κι επιπλέον *συσχετίζουν* την εμπειρία που αποκομίζουν με παρεμφερείς καταστάσεις. Γίνεται επομένως φανερό, πως οι άνθρωποι *αναζητούν* τρόπους συμπεριφοράς με τη βοήθεια της κριτικής τους ικανότητας και δεν υπακούουν τυφλά στα ένστικτά τους ή σε παλαιότερη εμπειρία που ενδεχομένως έχουν για κάποιες καταστάσεις.

Έτσι, φαίνεται να είναι ιδιαίτερα σημαντικό να ενσωματωθεί η παραπάνω ιδέα στους διαφόρους αλγορίθμους μάθησης. Πράγματι, διάφορες μέθοδοι εμπεριέχουν την ιδέα αυτή, η οποία μεταφράζεται σε μια διαδικασία *αναζήτησης* και είναι η πλέον διαδεδομένη ιδέα σε όλο το φάσμα της Τεχνητής Νοημοσύνης. Ουσιαστικά, ο πράκτορας «εφοδιάζεται» με μια διαδικασία σκέψης και κριτικής ικανότητας προκειμένου να μπορεί να κρίνει καλύτερα διάφορες καταστάσεις και να βγάζει καλύτερα συμπεράσματα από τα διάφορα αποτελέσματα των ενεργειών του. Επιπλέον, με το εφόδιο αυτό, ο πράκτορας κατευθύνει τις σκέψεις σε ζητήματα τα οποία φαίνεται να έχουν μεγαλύτερο ενδιαφέρον από

άλλα κι επομένως να είναι πιο αποτελεσματικός.

1.3 Σύνοψη

Αυτοί είναι οι δύο κύριοι άξονες γύρω από τους οποίους περιστρέφεται η συγκεκριμένη πτυχιακή. Από τη μια η μάθηση μέσω ανταμοιβών κι από την άλλη η αναζήτηση και κρίση διαφόρων ενεργειών ώστε να επιταχύνεται η μάθηση. Τα χαρακτηριστικά αυτά θα τα συναντήσουμε κατ' επανάληψη στα κεφάλαια που ακολουθούν και θα προσπαθήσουμε να αναδείξουμε τις ιδιαιτερότητες που μπορούν να έχουν διάφορες προσεγγίσεις. Μέσα από αυτή την παρουσίαση θα τονίσουμε ιδιαίτερα κάποιες ιδέες οι οποίες μας οδήγησαν στη δημιουργία δύο νέων μεθόδων στη συγκεκριμένη περιοχή. Επιπλέον, θα προσπαθήσουμε να παρουσιάσουμε σύντομα και περιεκτικά τις κυριότερες προσπάθειες κι άλλων ανθρώπων στη συγκεκριμένη περιοχή. Τελικά, θα δείξουμε ένα επιτυχημένο, κατά τη γνώμη μας, παράδειγμα εφαρμογής των μεθόδων που προτείναμε σε ένα παιχνίδι ενός παίκτη, το SOLO.

Κεφάλαιο 2

Βασικές Έννοιες

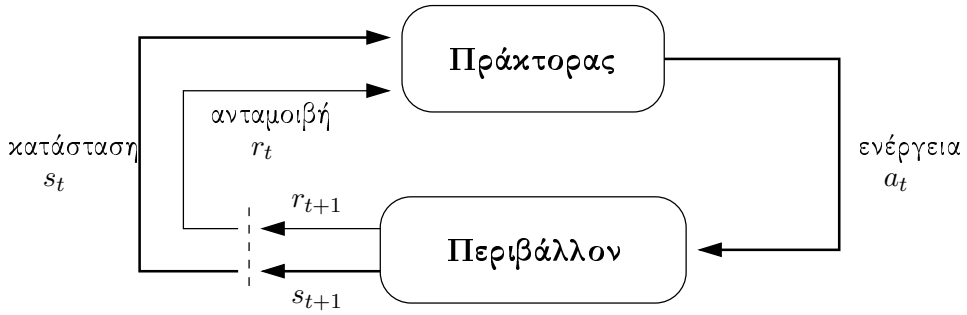
Πριν προχωρήσουμε στην περιγραφή των διαφόρων βασικών αλγορίθμων μάθησης που υπάρχουν, είναι αναγκαίο να γίνει μια σύντομη περιγραφή των βασικών εννοιών που διέπουν αυτούς τους αλγορίθμους. Η πιο βασική έννοια η οποία διακρίνει την Ενισχυτική Μάθηση απ'όλες τις άλλες μορφές μάθησης είναι το γεγονός ότι χρησιμοποιείται πληροφορία για την εκπαίδευση της μαθητευόμενης οντότητας προς μια κατεύθυνση η οποία *αποτιμά τις ενέργειες* οι οποίες έγιναν, αντί να την *καθοδηγούν* δίνοντάς της μια λίστα ορθών (βέλτιστων) ενεργειών. Δηλαδή γίνεται μια κατάταξη των ενεργειών βάσει των εμπειρικών αποτελεσμάτων τους κι όχι βάσει της σειράς ορθότητάς τους η οποία έτσι κι αλλιώς είναι άγνωστη τις περισσότερες φορές.

2.1 Πράκτορας και Περιβάλλον

Η ενισχυτική μάθηση έχει να κάνει με το να μαθαίνει κάποιος τι πρέπει να κάνει μέσω *ανταμοιβών (rewards)*. Η τακτική που ακολουθείται είναι η δημιουργία ενός μαθητή ο οποίος παίρνει αποφάσεις και καλείται *πράκτορας (agent)*. Το αντικείμενο με το οποίο αλληλεπιδρά ο πράκτορας, συμπεριλαμβάνει οτιδήποτε έξω από τον πράκτορα και καλείται *περιβάλλον (environment)*.

Ο πράκτορας και το περιβάλλον αλληλεπιδρούν κάθε *διακριτή*[1] χρονική στιγμή. Κάθε χρονική στιγμή t, ο πράκτορας γνωρίζει μέσω κάποιων σημάτων από το περιβάλλον σε ποια κατάσταση $s \in S$ βρίσκεται, όπου S το σύνολο των πιθανών καταστάσεων στις οποίες μπορεί να βρεθεί. Δεδομένου τώρα ότι ο πράκτορας βρίσκεται σε μια κατάσταση s τη χρονική στιγμή t αποφασίζει ποια ενέργεια θέλει να εκτελέσει. Έτσι, επιλέγει να εκτελέσει μια ενέργεια $a_t \in A(s)$, όπου το σύνολο $A(s)$ αποτελεί το σύνολο των διαθέσιμων ενερ-

[1] Η πρόταση αυτή δεν είναι αληθής γενικά. Επειδή όμως η πτυχιακή είχε χαρακτήρα με προσανατολισμό κάποια εφαρμογή σε παιχνίδια και (τα περισσότερα από) αυτά μπορούν να χαρακτηριστούν από διακριτές μονάδες χρόνου, έτσι και η προσοχή μας στράφηκε σε αυτή την κατηγορία της Ενισχυτικής Μάθησης. Ο αναγνώστης ο οποίος ενδιαφέρεται για περισσότερες πληροφορίες μπορεί να κοιτάξει στο [7].



Σχήμα 2.1: Αλληλεπίδραση Πράκτορα-Περιβάλλοντος

γειών του πράκτορα στην κατάσταση s. Την επόμενη ακριβώς χρονική στιγμή ο πράκτορας θα λάβει ένα *σήμα ενίσχυσης* $r_{t+1} \in \mathbb{R}$ λόγω της ενέργειας a_t που επέλεξε την προηγούμενη χρονική στιγμή και θα βρεθεί σε μια νέα κατάσταση s_{t+1}[2]. Όλη η προηγούμενη περιγραφή φαίνεται παραστατικά στο δημοφιλέστερο ίσως σχήμα της Ενισχυτικής Μάθησης το οποίο εικονίζεται στο σχήμα 2.1.

Έτσι, στη διαδικασία αλληλεπίδρασης πράκτορα-περιβάλλοντος μπορούμε να διακρίνουμε δύο βασικές συνιστώσες. *Πρώτον,* μετά από κάθε *ενέργεια (action)* του πράκτορα, το περιβάλλον τον οδηγεί σε μια «νέα» *κατάσταση (state).* Η λέξη νέα είναι σε εισαγωγικά γιατί ενδέχεται ο πράκτορας με κάποια ενέργεια του να μην αλλάξει κατάσταση, το μόνο που γίνεται είναι να περνάει μια χρονική μονάδα. *Δεύτερον,* αυτή τη μετάβαση κατάστασης τη συνοδεύει πάντοτε ένα σήμα ενίσχυσης από το περιβάλλον προς τον πράκτορα. Το σήμα ενίσχυσης καλείται *ανταμοιβή* και είναι πάντοτε ένας αριθμός $r \in \mathbb{R}$. Η συνάρτηση μέσω της οποίας επιλέγεται το αριθμητικό αυτό σήμα το οποίο αποστέλλεται στον πράκτορα καλείται *συνάρτηση ανταμοιβής*.

Ο μοναδικός σκοπός που έχει ο πράκτορας κατά τη διάρκεια αλληλεπίδρασής του με το περιβάλλον είναι να νιώθει όσο το δυνατόν πιο *ευχάριστα*, ανεξάρτητα από το πόσο αντίξοο μπορεί να είναι το περιβάλλον. Η ευχαρίστηση που νιώθει ο πράκτορας καθορίζεται από τις ανταμοιβές τις οποίες λαμβάνει για τις διάφορες ενέργειές του. Επομένως, αυτό το οποίο προσπαθεί να επιτύχει ένας πράκτορας Ενισχυτικής Μάθησης είναι η *μεγιστοποίηση της ανταμοιβής που θα λάβει από το περιβάλλον*. Το περιβάλλον τώρα μπορεί να είναι το φυσικό περιβάλλον στο οποίο ζούμε όλοι μας ή ένα ανθρώπινο δημιούργημα το οποίο θέλουμε να εκφράζει ένα πρόβλημα βελτιστοποίησης. Τις περισσότερες φορές βέβαια, το περιβάλλον μέσα στο οποίο ζει και αλληλεπιδρά ο πράκτορας είναι τεχνητό. Μια πλήρης περιγραφή (μοντελοποίηση) ενός περιβάλλοντος καθορίζει μια *εργασία (task)*, η οποία είναι μια έκφανση του *γενικότερου προβλήματος*

[2]Στο σημείο αυτό ακολουθώ κατά γράμμα τους συμβολισμούς οι οποίοι υπάρχουν στο [26] μιας και με αυτόν τον τρόπο δίνεται έμφαση στην αλληλουχία των διαφόρων γεγονότων που λαμβάνουν μέρος.

ενισχυτικής μάθησης. Με τον όρο μοντελοποίηση του περιβάλλοντος εννοούμε την περιγραφή όλων των πιθανών διαφορετικών καταστάσεων που μπορεί να συναντήσει ένας πράκτορας κατά τη διάρκεια αλληλεπίδρασής του με αυτό. Επιπλέον, εκτός από τις διάφορες καταστάσεις οφείλουμε να περιγράφουμε όλες τις πιθανές ενέργειες που είναι διαθέσιμες σε μια κατάσταση, την κατανομή πιθανοτήτων πάνω στις διάφορες ενέργειες σχετικά με το που μας οδηγούν αυτές και τέλος για κάθε μια τέτοια μετάβαση την ανταμοιβή η οποία θα δίνεται στον πράκτορα για την εφαρμογή της συγκεκριμένης ενέργειας. Ένας άλλος όρος ο οποίος αναφέρθηκε πιο πάνω είναι αυτός του *γενικότερου προβλήματος ενισχυτικής μάθησης*. Το γενικότερο πρόβλημα ενισχυτικής μάθησης ουσιαστικά είναι το πιο δύσκολο πρόβλημα το οποίο μπορεί να τεθεί σε έναν πράκτορα και τα ιδιαίτερα χαρακτηριστικά του θα φανερωθούν σταδιακά κατά τη διάρκεια της ανάγνωσης.

Με κατάλληλους μετασχηματισμούς μπορούμε να απεικονίζουμε ένα πρόβλημα βελτιστοποίησης σε περιβάλλον και στη συνέχεια να «γεννάμε» έναν πράκτορα μέσα στο περιβάλλον αυτό ο οποίος να αλληλεπιδρά μαζί του. Έτσι, παρακολουθώντας τις ενέργειες του πράκτορα μπορούμε να βρούμε λύσεις σε προβλήματα βελτιστοποίησης. Οι ενέργειες οι οποίες επιλέγονται από τον πράκτορα καθορίζουν την *πολιτική (policy)* του στην αλληλεπίδρασή του με το περιβάλλον.

Η *πολιτική* η οποία ακολουθείται από τον πράκτορα συμβολίζεται με π_t και με αυτόν τον τρόπο δηλώνουμε ότι η *πιθανότητα* να διαλέξει ο πράκτορας την ενέργεια a στην κατάσταση s τη χρονική στιγμή t είναι $\pi_t(s, a)$. Δηλαδή, η πολιτική είναι μια *εσωτερική απεικόνιση* που διαθέτει ο πράκτορας από καταστάσεις σε πιθανότητες επιλογής όλων των διαθέσιμων ενεργειών στις καταστάσεις αυτές. Έτσι, οι μέθοδοι Ενισχυτικής Μάθησης, καθορίζουν τον *τρόπο* με τον οποίο ο πράκτορας αλλάζει την πολιτική του - σαν αποτέλεσμα των εμπειριών του (ανταμοιβών) - *με σκοπό να μεγιστοποιήσει το συνολικό άθροισμα ανταμοιβών που θα λάβει από το περιβάλλον*.

Επομένως, οποιοδήποτε εργασία ενισχυτικής μάθησης (η οποία καθορίζεται από στόχους) μπορεί να απλοποιηθεί σε τρία σήματα τα οποία ανταλλάσσονται μεταξύ του πράκτορα και του περιβάλλοντος:

I. Ένα σήμα το οποίο αναπαριστά τις *ενέργειες* (επιλογές) του πράκτορα.

II. Ένα σήμα το οποίο αναπαριστά τις *καταστάσεις* στις οποίες μπορεί να βρεθεί ο πράκτορας. Ουσιαστικά το σήμα αυτό καθορίζει το πλήθος και το είδος των διαθέσιμων ενεργειών του πράκτορα[3].

III. Τέλος, ένα ακόμη σήμα το οποίο να καθορίζει τους *στόχους* που πρέπει να έχει ένας πράκτορας και μεταφέρεται μέσω των διαφόρων *ανταμοιβών* που δέχεται ο πράκτορας από το περιβάλλον.

[3]Για το λόγο αυτό, αρκετές φορές στη βιβλιογραφία χρησιμοποιείται μια συνάρτηση απεικόνισης μ από καταστάσεις σε ενέργειες.

2.1.1 Στόχοι και Ενισχύσεις

Ας περάσουμε τώρα στο θέμα των στόχων που πρέπει να έχει ένας πράκτορας και πως αυτοί μπορούν να καθοριστούν μέσα από τις ανταμοιβές τις οποίες λαμβάνει από το περιβάλλον. Όπως έχει ήδη αναφερθεί, σκοπός του πράκτορα είναι η μεγιστοποίηση της συνολικής ανταμοιβής που θα λάβει από το περιβάλλον. Αυτό σημαίνει *μεγιστοποίηση* του αθροίσματος των ανταμοιβών *μακροπρόθεσμα* και *όχι άμεσα*. Το θέμα επομένως είναι, πως μπορούμε να εξαρτήσουμε αυτόν το γενικότερο στόχο του πράκτορα με τον πραγματικό στόχο της εκάστοτε εργασίας (προβλήματος).

Κάθε χρονική στιγμή, ο πράκτορας, λόγω αλληλεπίδρασης με το περιβάλλον λαμβάνει μια ενίσχυση η οποία είναι ένας αριθμός $r_t \in \mathbb{R}$. Έτσι λοιπόν, προκειμένου ο πράκτορας να μαθαίνει να κάνει κάτι που εμείς θέλουμε, οφείλουμε να του δίνουμε τέτοιες ανταμοιβές ώστε στην προσπάθειά του να επιτύχει μακροπρόθεσμη μεγιστοποίηση των ανταμοιβών που λαμβάνει, να επιτυγχάνει ταυτόχρονα τους στόχους μας, του στόχους δηλαδή του εκάστοτε προβλήματος βελτιστοποίησης. Προκειμένου να γίνουν τα προηγούμενα πιο κατανοητά, παρατίθεται το ακόλουθο παράδειγμα:

> Φανταστείτε ότι θέλουμε ένα ρομπότ να μάθει να βγαίνει από έναν λαβύρινθο. Μια λύση είναι να δίνουμε στο ρομπότ σε κάθε κίνηση που κάνει μέσα στο λαβύρινθο μια ανταμοιβή με τιμή μηδέν (0). Μόλις τώρα, το ρομπότ καταφέρει να βγει από το λαβύρινθο του δίνουμε μια ανταμοιβή με τιμή ένα (1). Αυτό φαίνεται αρκετά ελκυστικό, αλλά δεν είναι σίγουρο ότι το ρομπότ θα μάθει να βγαίνει από τον λαβύρινθο όσο πιο γρήγορα γίνεται. Μπορεί για παράδειγμα να βρίσκει μεγάλα «δωμάτια» στη διαδρομή, να κάνει μερικές βόλτες μέσα σε αυτά, και στη συνέχεια να πηγαίνει να βρει την έξοδο. Έτσι, μια λύση είναι να δίνουμε στο ρομπότ μια αρνητική ανταμοιβή κάθε επιπλέον χρονική στιγμή που την περνάει μέσα στο λαβύρινθο, π.χ. -1, και όταν τελικά καταφέρει να βγει από το λαβύρινθο να παίρνει μια ανταμοιβή 0. Αυτή η τεχνική θα το οδηγήσει πολύ πιο γρήγορα στο να βρει το δρόμο για την έξοδο.

Επομένως, είναι κρίσιμο οι ανταμοιβές τις οποίες θα επιστρέφει το περιβάλλον να αντικατοπτρίζουν επακριβώς αυτό που θέλουμε να επιτύχει ο πράκτορας. Έτσι, τα σήματα ενίσχυσης τα οποία θα στέλνει το περιβάλλον στον πράκτορα, είναι το μέσο επικοινωνίας που διαθέτουμε προκειμένου να ενημερώνουμε τον πράκτορα για το *τι* θέλουμε να επιτύχει και όχι για το *πως* θέλουμε αυτό να επιτευχθεί.

2.1.2 Επιστροφές

Σκοπός του πράκτορα είναι η συνολική μεγιστοποίηση των ανταμοιβών τις οποίες θα λάβει από την αλληλεπίδρασή του με το περιβάλλον. Αν υποθέσουμε

ότι με $r_{t+1}, r_{t+2}, r_{t+3}, \ldots$, δίνεται η ακολουθία των ανταμοιβών τις οποίες λαμβάνει ο πράκτορας μετά τη χρονική στιγμή t, τότε ο πράκτορας προσπαθεί να μεγιστοποιήσει την *προσδοκώμενη επιστροφή (Expected Return)*. Η επιστροφή R_t ορίζεται σαν μια συνάρτηση πάνω στην ακολουθία των ανταμοιβών τις οποίες λαμβάνει ο πράκτορας μετά τη χρονική στιγμή t.

Έτσι, στην απλούστερή της μορφή, η επιστροφή είναι το άθροισμα όλων των ανταμοιβών:

$$R_t = r_{t+1} + r_{t+2} + \ldots + r_T, \tag{2.1}$$

όπου T είναι ένα τερματικό βήμα. Αυτή η προσέγγιση έχει νόημα βέβαια στις περιπτώσεις εκείνες που η ακολουθία αλληλεπιδράσεων πράκτορα-περιβάλλοντος μπορεί να διαιρεθεί εννοιολογικά σε υπακολουθίες αλληλεπιδράσεων, οι οποίες καλούνται *επεισόδια ή δοκιμές*. Εργασίες οι οποίες εντάσσονται σε αυτήν την κατηγορία καλούνται *επεισοδιακές εργασίες (episodic tasks)*. Παραδείγματα τέτοιων περιπτώσεων είναι παρτίδες σκακιού, παρτίδες από τάβλι, διαδρομές μέσα από λαβυρίνθους, κ.τ.λ. Κάθε επεισόδιο τελειώνει σε μια ειδική κατάσταση η οποία καλείται *τερματική*. Μετά τη λήξη του επεισοδίου, ο πράκτορας οδηγείται πάλι στην αρχική κατάσταση της εργασίας και η διαδικασία μάθησης συνεχίζεται. Στην περίπτωση που έχουμε περισσότερες από μια αρχικές καταστάσεις, επιλέγεται μια από αυτές στην τύχη ή βάσει κάποιου άλλου κριτηρίου. Τέλος, σε επεισοδιακές εργασίες συχνά έχει νόημα να διαχωρίζουμε όλες τις μη-τερματικές καταστάσεις από την τερματική. Έτσι, με S δηλώνουμε τις μη-τερματικές καταστάσεις, ενώ το σύνολο όλων των πιθανών καταστάσεων (περιλαμβάνοντας και την τελική κατάσταση) δηλώνεται με S^+.

Από την άλλη μεριά, η ακολουθία αλληλεπιδράσεων πράκτορα-περιβάλλοντος δεν μπορεί να διαχωριστεί πάντοτε σε ξεχωριστά επεισόδια. Για παράδειγμα, φανταστείτε έναν πράκτορα ο οποίος είναι ένα πρόγραμμα σε κάποιον εξυπηρέτη και η λειτουργία του είναι να αποδέχεται ή να απορρίπτει αιτήσεις προς εξυπηρέτηση. Εργασίες σαν κι αυτή, τις ονομάζουμε *συνεχιζόμενες εργασίες (continuing tasks)*. Σε αυτές τις περιπτώσεις όμως, δεν μπορούμε να χρησιμοποιήσουμε τη σχέση (2.1), αφού η τελική χρονική στιγμή T ταυτίζεται με το άπειρο και δεν υπάρχει καμία εγγύηση ότι το άθροισμα αυτό συγκλίνει. Για το λόγο αυτό, στις περιπτώσεις αυτές χρησιμοποιείται μια λίγο διαφορετική έννοια του όρου «επιστροφή».

Εισάγεται λοιπόν η έννοια της *έκπτωσης (discounting)*. Σύμφωνα με αυτή την προσέγγιση, ο πράκτορας προσπαθεί να διαλέγει ενέργειες ώστε το άθροισμα των *εκπιπτόμενων ανταμοιβών (discounted rewards)* τις οποίες θα λάβει στο μέλλον να μεγιστοποιείται. Πιο συγκεκριμένα, τη χρονική στιγμή t διαλέγει μια ενέργεια a_t τέτοια ώστε να μεγιστοποιείται η *προσδοκώμενη επιστροφή η οποία έχει υποστεί έκπτωση (expected discounted return)*:

$$R_t = r_{t+1} + \gamma \cdot r_{t+2} + \gamma^2 \cdot r_{t+3} + \ldots = \sum_{k=0}^{\infty} \gamma^k \cdot r_{t+k+1}, \tag{2.2}$$

όπου γ μια παράμετρος, με $0 \leq \gamma \leq 1$, η οποία καλείται *ρυθμός έκπτωσης*

(discount rate).

Ο ρυθμός έκπτωσης καθορίζει την τρέχουσα αξία μιας μελλοντικής ανταμοιβής: Μια ανταμοιβή δηλαδή η οποία πρόκειται να ληφθεί k χρονικά βήματα αργότερα στην πραγματικότητα αξίζει $(1-\gamma^{k-1})$ φορές λιγότερο απ'ότι αν η ανταμοιβή αυτή λαμβανόταν αμέσως. Για παράδειγμα, φανταστείτε ότι κερδίζετε ένα λαχείο το οποίο αν σας το ξεπληρώσουν σήμερα θα πάρετε 100.000€, ενώ μπορούν να σας δώσουν 1.000.000€ αν δεχθείτε να γίνει η αποπληρωμή μετά από ακριβώς 100 χρόνια. Τι θα προτιμήσετε;

Σχετικά με τις τιμές τώρα που μπορεί να λάβει η παράμετρος γ μπορούμε να παρατηρήσουμε τα ακόλουθα:

- Αν $\gamma < 1$, τότε το άπειρο άθροισμα έχει μια πεπερασμένη τιμή υπό την προϋπόθεση ότι η ακολουθία $\{r_k\}$ φράσσεται.

- Αν $\gamma = 0$, τότε ο πράκτορας λειτουργεί «μυωπικά» προσπαθώντας να μεγιστοποιήσει μονάχα το άμεσο κέρδος που μπορεί να έχει από την αλληλεπίδρασή του με το περιβάλλον.

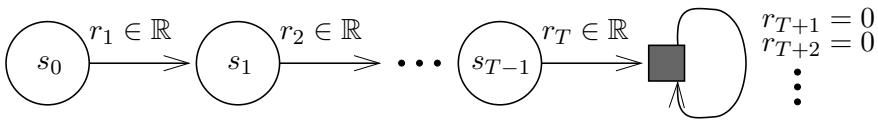
- Καθώς $\gamma \to 1$, ο πράκτορας λαμβάνει υπ'όψιν του τις μελλοντικές ανταμοιβές όλο και περισσότερο.

Ενιαίος Συμβολισμός

Από τις προηγούμενες παρατηρήσεις γίνεται φανερό πως πρέπει να εργαζόμαστε με δύο διαφορετικούς τύπους προκειμένου να μεταχειριζόμαστε τις προσδοκώμενες επιστροφές του πράκτορα. Επιπλέον, στις επεισοδιακές εργασίες, γίνεται φανερό πως απαιτείται η εισαγωγή ενός επιπλέον δείκτη σε όλες τις μεταβλητές που έχουν οριστεί μέχρι στιγμής προκειμένου με αυτόν τον τρόπο να δηλώνουμε σε ποιο επεισόδιο αναφερόμαστε.

Παρ'όλ'αυτά, προκύπτει ότι δεν χρειάζεται κάποια ιδιαίτερη αναφορά σε συγκεκριμένα επεισόδια στις όποιες αναλύσεις. Αυτό το οποίο γίνεται, είναι να βγαίνουν γενικά συμπεράσματα για τον πράκτορα και τη συμπεριφορά του, εξετάζοντας τη διαδικασία μάθησης σαν μια ολότητα. Ακόμη, προκειμένου να μην χρησιμοποιούνται διαφορετικοί τύποι για επεισοδιακές εργασίες και για συνεχιζόμενες εργασίες, θεωρούμε μια λίγο διαφορετική μοντελοποίηση των επεισοδιακών εργασιών. Έτσι, σε επεισοδιακές εργασίες, όταν ο πράκτορας εισέρχεται στην τελική κατάσταση, επιτρέπουμε την ύπαρξη μεταβάσεων από την κατάσταση αυτή στην ίδια κατάσταση, με ανταμοιβή κάθε φορά που πραγματοποιείται μια τέτοια μετάβαση ίση με το μηδέν (0). Κάτι τέτοιο φαίνεται παραστατικά στο σχήμα 2.2.

Έτσι, μπορούμε να ορίσουμε την επιστροφή σύμφωνα με τη σχέση (2.2), παραλείποντας τον αριθμό επεισοδίου όταν αυτός δεν είναι αναγκαίος και περιλαμβάνοντας την πιθανότητα ο ρυθμός έκπτωσης γ να έχει τιμή 1 στις περιπτώσεις επεισοδιακών εργασιών. Με άλλα λόγια, μπορούμε να γράψουμε την



Σχήμα 2.2: Αναπαράσταση επεισοδιακής εργασίας για ενιαίο συμβολισμό.

επιστροφή ως

$$R_t = \sum_{k=0}^{T} \gamma^k \cdot r_{t+k+1}, \tag{2.3}$$

περιλαμβάνοντας την πιθανότητα είτε $T = \infty$ και $\gamma < 1$ ή $\gamma = 1$ και $T =$ πεπερασμένο.

2.1.3 Ιδιότητα Markov

Ο πράκτορας λαμβάνει αποφάσεις συναρτήσει της κατάστασης στην οποία βρίσκεται. Την κατάσταση βέβαια, την εκλαμβάνει σαν ένα σήμα το οποίο έρχεται από το περιβάλλον. Επομένως, είναι ανάγκη να οριστεί *το είδος της πληροφορίας* το οποίο περιέχεται σε αυτό το σήμα.

Γενικά, το σήμα αυτό πρέπει να περιέχει πληροφορία για την κατάσταση στην οποία βρίσκεται ένας πράκτορας χωρίς όμως αυτό να σημαίνει πως ο πράκτορας γνωρίζει τα πάντα για το περιβάλλον στην κατάσταση στην οποία βρίσκεται. Για παράδειγμα, αν ο πράκτορας παίζει πόκερ, δεν μπορούμε να περιμένουμε να γνωρίζει τα χαρτιά των υπολοίπων παικτών. Έτσι, γίνεται φανερό πως σε ορισμένες περιπτώσεις υπάρχει μια «κρυμμένη» πληροφορία πίσω από τις καταστάσεις την οποία αν την ήξερε ο πράκτορας θα ήταν πολύ χρήσιμο, αλλά δεν μπορεί να γνωρίζει κάτι τέτοιο μιας και ποτέ δεν έλαβε κάποιο σχετικό ερέθισμα. Δηλαδή, δεν μπορούμε να κατηγορήσουμε τον πράκτορα επειδή δεν γνωρίζει κάτι που έχει σημασία, αλλά μόνο στην περίπτωση που έμαθε κάτι και μετά το ξέχασε. Ιδεατά, αυτό το οποίο θα θέλαμε να περιέχεται μέσα στο σήμα κατάστασης, είναι όλη η πληροφορία από προηγούμενες ενέργειες. Ένα σήμα το οποίο έχει αυτή την ιδιότητα ονομάζεται *Markov*, ή λέμε ότι το σήμα αυτό έχει την *ιδιότητα Markov (Markov property)*.

Φανταστείτε τώρα ένα περιβάλλον το οποίο αποκρίνεται σε κάποια ενέργεια του πράκτορα τη χρονική στιγμή $(t+1)$. Τότε, στην πιο γενική της μορφή, η απόκριση μπορεί να εξαρτάται από οτιδήποτε έχει προηγηθεί νωρίτερα. Επομένως, η δυναμική του περιβάλλοντος μπορεί να οριστεί καθορίζοντας την πλήρη κατανομή πιθανότητας:

$$Pr\Big\{s_{t+1} = s', r_{t+1} = r \mid s_t, a_t, r_t, s_{t-1}, a_{t-1}, r_{t-1}, \ldots, r_1, s_0, a_0\Big\}, \tag{2.4}$$

για όλες τις καταστάσεις s', τις πιθανές ανταμοιβές r και βάσει όλων των παλαιότερων συμβάντων: $s_t, a_t, r_t, s_{t-1}, a_{t-1}, r_{t-1}, \ldots, r_1, s_0, a_0$. Από την άλλη,

αν ένα σήμα έχει την *ιδιότητα Markov*, τότε η απόκριση του περιβάλλοντος τη χρονική στιγμή $(t+1)$ εξαρτάται μόνο από την κατάσταση στην οποία βρισκόταν τη χρονική στιγμή t. Έτσι, η δυναμική του περιβάλλοντος σε αυτή την περίπτωση μπορεί να καθοριστεί ορίζοντας τις πιθανότητες:

$$Pr\Big\{s_{t+1} = s', r_{t+1} = r \mid s_t, a_t\Big\}, \tag{2.5}$$

για όλες τις καταστάσεις s', τις πιθανές ανταμοιβές r και βάσει της προηγούμενης κατάστασης s_t καθώς και της ενέργειας a_t την οποία πραγματοποίησε ο πράκτορας στην κατάσταση s_t.

Πιο απλά, ένα σήμα έχει την *ιδιότητα Markov*, αν και μόνο αν η (2.5) είναι ίση με την (2.4), για όλα τα s', r και τα παλαιότερα συμβάντα. Σε αυτή την περίπτωση, το περιβάλλον και η εργασία λέγεται ότι έχουν την *ιδιότητα Markov*. Επίσης μπορεί να δειχθεί ότι η βέλτιστη πολιτική για επιλογή ενεργειών σαν συνάρτηση μιας Markov κατάστασης είναι το ίδιο καλή με την βέλτιστη πολιτική για επιλογή ενεργειών σαν συνάρτηση όλης της προηγούμενης ιστορίας. Τέλος, ακόμη και στις περιπτώσεις που το περιβάλλον δεν έχει την ιδιότητα Markov είναι καλό να σκεφτόμαστε τις καταστάσεις σαν μια προσέγγιση καταστάσεων Markov. Μάλιστα, η υπόθεση αυτή δεν γίνεται μόνο στον τομέα της Ενισχυτικής Μάθησης αλλά σχεδόν παντού στον τομέα της Τεχνητής Νοημοσύνης εξαιτίας των ιδιαίτερα καλών χαρακτηριστικών που μπορεί να προσδώσει αυτή η ιδιότητα σε ένα πρόβλημα.

2.1.4 Διαδικασίες Απόφασης Markov

Μια *εργασία* Ενισχυτικής Μάθησης η οποία ικανοποιεί την ιδιότητα Markov καλείται *Διαδικασία Απόφασης Markov (Markov Decision Process - MDP)*. Αν επιπλέον ο χώρος των καταστάσεων και των ενεργειών είναι πεπερασμένοι, τότε καλείται *Πεπερασμένη Διαδικασία Απόφασης Markov (finite Markov Decision Process - finite MDP)*.

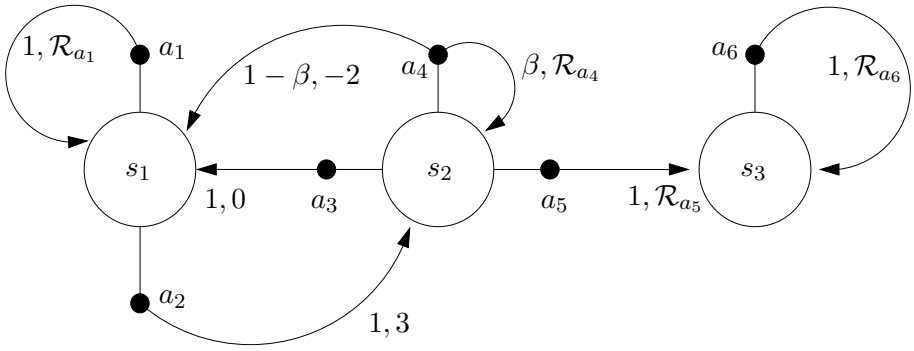
Μια πεπερασμένη Διαδικασία Απόφασης Markov καθορίζεται από τα σύνολα καταστάσεων και ενεργειών καθώς και από την ενός-βήματος δυναμική του περιβάλλοντος. Έτσι, δοθέντων μιας οποιαδήποτε κατάστασης s και ενέργειας a, η πιθανότητα κάθε πιθανής επόμενης κατάστασης s' δίνεται από την:

$$\mathcal{P}^a_{s\to s'} = \mathcal{P}^a_{ss'} = Pr\Big\{s_{t+1} = s' \mid s_t = s, a_t = a\Big\}. \tag{2.6}$$

Η παραπάνω ποσότητα καλείται *πιθανότητα μετάβασης*. Όμοια, δοθέντων μιας οποιαδήποτε κατάστασης s, ενέργειας a μαζί με οποιαδήποτε επόμενη κατάσταση s', η προσδοκώμενη τιμή για την επόμενη ανταμοιβή είναι:

$$\mathcal{R}^a_{s\to s'} = \mathcal{R}^a_{ss'} = E\Big\{r_{t+1} \mid s_t = s, a_t = a, s_{t+1} = s'\Big\}. \tag{2.7}$$

Οι προηγούμενες ποσότητες, $\mathcal{P}^a_{ss'}$ και $\mathcal{R}^a_{ss'}$, καθορίζουν τα πιο σημαντικά χαρακτηριστικά της δυναμικής μιας πεπερασμένης Διαδικασίας Απόφασης Markov. Η μόνη πληροφορία η οποία χάνεται είναι η κατανομή των ενισχύσεων γύρω από την προσδοκώμενη (μέση) τιμή.



Σχήμα 2.3: Παράδειγμα Γράφου Μετάβασης Καταστάσεων.

Γράφοι Μετάβασης Καταστάσεων

Ένας γράφος μετάβασης καταστάσεων είναι ένας χρήσιμο εργαλείο για να περιγράψουμε τη δυναμική μιας πεπερασμένης Διαδικασίας Απόφασης Markov. Σε αυτόν το γράφο υπάρχουν δύο είδη κόμβων: *κόμβοι καταστάσεων* και *κόμβοι ενεργειών*.

Κάθε πιθανή κατάσταση την απεικονίζουμε με έναν κόμβο κατάστασης (ένας μεγάλος κύκλος με το όνομα της κατάστασης στο εσωτερικό του) και κάθε πιθανή ενέργεια για κάθε ζευγάρι κατάστασης-ενέργειας με έναν κόμβο ενέργειας (ένας συμπαγής κύκλος ο οποίος έχει σαν ετικέτα το όνομα της ενέργειας). Τα τόξα τα οποία ξεκινούν από έναν κόμβο κατάστασης s, περνάνε μέσα από έναν κόμβο ενέργειας a και καταλήγουν σε μια κατάσταση s' δηλώνουν ότι αν κάποιος βρίσκεται σε μια κατάσταση s και εφαρμόσει την ενέργεια a, τότε η επόμενη κατάσταση στην οποία θα βρεθεί είναι η s'. Επιπλέον, σε κάθε τόξο αντιστοιχούν δύο ετικέτες. Η μια αντικατοπτρίζει την πιθανότητα το ζευγάρι $\langle s, a\rangle$ να μας οδηγήσει στην κατάσταση s'. Αυτό γίνεται γιατί δεν υπάρχει αναγκαστικά μια μονοσήμαντη αντιστοιχία μεταξύ ζευγαριών $\langle s, a\rangle$ και τριάδων $\langle s, a, s'\rangle$. Η άλλη ετικέτα, αντικατοπτρίζει την προσδοκώμενη ανταμοιβή από μια συγκεκριμένη μετάβαση. Δηλαδή, οι ετικέτες που υπάρχουν στα τόξα έχουν τις τιμές που δίνονται από τις εξισώσεις (2.6) και (2.7). Στο σχήμα 2.3 μπορούμε να δούμε ένα παράδειγμα ενός γράφου μετάβασης καταστάσεων. Παρατηρήστε επιπλέον ότι το άθροισμα των πιθανοτήτων μετάβασης από τα τόξα τα οποία εξέρχονται από μια συγκεκριμένη ενέργεια έχουν άθροισμα 1.

2.2 Συναρτήσεις Αποτίμησης

Σχεδόν όλοι οι αλγόριθμοι Ενισχυτικής Μάθησης βασίζονται σε *συναρτήσεις αποτίμησης (Value Functions, Q-Factors)*, συναρτήσεις δηλαδή οι οποίες υπολογίζουν *πόσο καλή* είναι μια κατάσταση ή ένα ζευγάρι κατάστασης-ενέργειας για έναν πράκτορα. Ο όρος «πόσο καλή είναι μια κατάσταση» ο-

ρίζεται συναρτήσει της *προσδοκώμενης επιστροφής* που θα έχει ο πράκτορας από τη συγκεκριμένη κατάσταση (ή ζευγάρι κατάστασης-ενέργειας) κι έπειτα. Φυσικά, οι ανταμοιβές που περιμένει να πάρει ο πράκτορας στο μέλλον καθορίζονται από τις επιλογές τις οποίες θα κάνει. Επομένως, οι συναρτήσεις αποτίμησης ορίζονται παράλληλα με συγκεκριμένες πολιτικές.

Γενικά, η *αποτίμηση (value)* μιας κατάστασης s ενώ ακολουθείται μια πολιτική π, γράφεται $V^\pi(s)$ και εκφράζει την προσδοκώμενη επιστροφή του πράκτορα όταν αυτός ξεκινήσει από την κατάσταση s και ακολουθήσει την πολιτική π στη συνέχεια. Για Διαδικασίες Απόφασης Markov, ορίζεται αυστηρά η $V^\pi(s)$ ως:

$$V^\pi(s) = E_\pi\{R_t \mid s_t = s\} = E_\pi\left\{\sum_{k=0}^{\infty} \gamma^k \cdot r_{t+k+1} \;\middle|\; s_t = s\right\}, \tag{2.8}$$

όπου $E_\pi\{\cdots\}$ δηλώνει την προσδοκώμενη τιμή δοθέντος ότι ο πράκτορας θα ακολουθήσει την πολιτική π. Στο σημείο αυτό πρέπει να σημειώσουμε πως στην περίπτωση που υπάρχει τερματική κατάσταση η τιμή αποτίμησής της είναι μηδέν (0). Η V^π ονομάζεται *συνάρτηση αποτίμησης-κατάστασης για την πολιτική π (state-value function for policy π).*

Όμοια, μπορούμε να ορίσουμε την αποτίμηση της λήψης μιας ενέργειας a σε μια κατάσταση s ενώ ακολουθείται μια πολιτική π, ως την προσδοκώμενη επιστροφή του πράκτορα ο οποίος ξεκινάει από μια κατάσταση s, εφαρμόζει την ενέργεια a και στη συνέχεια ακολουθεί την πολιτική π και τη συμβολίζουμε $Q^\pi(s,a)$. Έτσι λοιπόν, έχουμε:

$$Q^\pi(s,a) = E_\pi\{R_t \mid s_t = s, a_t = a\} = E_\pi\left\{\sum_{k=0}^{\infty} \gamma^k \cdot r_{t+k+1} \;\middle|\; s_t = s, a_t = a\right\}. \tag{2.9}$$

Η $Q^\pi(s,a)$ ονομάζεται *συνάρτηση αποτίμησης-ενέργειας για την πολιτική π (action-value function for policy π).*

2.2.1 Αναδρομικοί Συσχετισμοί

Μια πολύ σημαντική ιδιότητα των συναρτήσεων αποτίμησης η οποία χρησιμοποιείται σε όλο το φάσμα του Νευρο-Δυναμικού Προγραμματισμού είναι ότι ικανοποιούν συγκεκριμένες αναδρομικές σχέσεις. Για οποιαδήποτε πολιτική π και οποιαδήποτε κατάσταση s ισχύει η ακόλουθη σχέση μεταξύ της αποτίμησης (εκτίμησης) για την κατάσταση s και τις αποτιμήσεις (εκτιμήσεις) για όλες τις πιθανές επόμενες καταστάσεις:

$$\begin{aligned} V^\pi(s) &= E_\pi\{R_t \mid s_t = s\} \\ &= E_\pi\left\{\sum_{k=0}^{\infty} \gamma^k \cdot r_{t+k+1} \;\middle|\; s_t = s\right\} \end{aligned}$$

$$
\begin{aligned}
&= E_\pi\left\{ r_{t+1} + \gamma \cdot \sum_{k=0}^{\infty} \gamma^k \cdot r_{t+k+2} \;\middle|\; s_t = s \right\} \\
&= \sum_a \pi(s,a) \cdot \sum_{s'} \mathcal{P}^a_{ss'} \cdot \left[\mathcal{R}^a_{ss'} + \gamma \cdot \Pi_1 \right] \\
&\quad \text{όπου: } \Pi_1 = E_\pi\left\{ \sum_{k=0}^{\infty} \gamma^k \cdot r_{t+k+2} \;\middle|\; s_{t+1} = s' \right\} \\
&= \sum_a \pi(s,a) \cdot \sum_{s'} \mathcal{P}^a_{ss'} \cdot [\mathcal{R}^a_{ss'} + \gamma \cdot V^\pi(s')], \qquad (2.10)
\end{aligned}
$$

όπου οι ενέργειες a επιλέγονται από το σύνολο $A(s)$ και οι επόμενες καταστάσεις από το σύνολο S ή S^+ στην περίπτωση επεισοδιακών εργασιών. Η εξίσωση (2.10) είναι η *εξίσωση Bellman για τη συνάρτηση* V^π και εκφράζει τη σχέση η οποία υπάρχει μεταξύ της αποτίμησης μιας κατάστασης και των αποτιμήσεων για τις επόμενες καταστάσεις. Η εξίσωση (2.10) σταθμίζει βάσει της πιθανότητας εμφάνισής τους όλες τις επόμενες καταστάσεις. Έτσι εκφράζει πως η αποτίμηση μιας αρχικής κατάστασης είναι ίση με την αποτίμηση (που υπόκειται σε έκπτωση γ) της προσδοκώμενης επόμενης κατάστασης συν την ανταμοιβή η οποία θα ληφθεί κατά τη μετάβαση αυτή.

Όμοια, μπορούμε να πάρουμε την αντίστοιχη *εξίσωση Bellman για τη συνάρτηση* $Q^\pi(s,a)$:

$$
\begin{aligned}
Q^\pi(s,a) &= E_\pi\{R_t \mid s_t = s, a_t = a\} \\
&= E_\pi\left\{ \sum_{k=0}^{\infty} \gamma^k \cdot r_{t+k+1} \;\middle|\; s_t = s, a_t = a \right\} \\
&= E_\pi\left\{ r_{t+1} + \gamma \cdot \sum_{k=0}^{\infty} \gamma^k \cdot r_{t+k+2} \;\middle|\; s_t = s, a_t = a \right\} \\
&= \sum_{s'} \mathcal{P}^a_{ss'} \cdot \sum_{a' \in A(s')} \pi(s',a') \cdot \left[\mathcal{R}^a_{ss'} + \gamma \cdot \Pi_2 \right] \\
&\quad \text{όπου: } \Pi_2 = E_\pi\left\{ \sum_{k=0}^{\infty} \gamma^k \cdot r_{t+k+2} \;\middle|\; s_{t+1} = s', a_{t+1} = a' \right\} \\
&= \sum_{s'} \mathcal{P}^a_{ss'} \cdot \sum_{a' \in A(s')} \pi(s',a') \cdot [\mathcal{R}^a_{ss'} + \gamma \cdot Q^\pi(s',a')] \qquad (2.11)
\end{aligned}
$$

Οι συναρτήσεις V^π και Q^π είναι οι μοναδικές λύσεις των εξισώσεων Bellman (2.10) και (2.11) αντίστοιχα.

2.2.2 Βέλτιστες Συναρτήσεις Αποτίμησης

Η επίλυση μιας εργασίας Ενισχυτικής Μάθησης έγκειται στην εύρεση μιας πολιτικής η οποία επιτυγχάνει μακροπρόθεσμα μεγάλες τιμές ανταμοιβών. Για

Διαδικασίες Απόφασης Markov είναι δυνατό να οριστεί επακριβώς μια βέλτιστη πολιτική με τον ακόλουθο τρόπο. Οι συναρτήσεις αποτίμησης καθορίζουν μια μερική διάταξη πάνω στις πολιτικές. Μια πολιτική π ορίζεται να είναι καλύτερη από ή ίση μιας πολιτικής π' εάν η προσδοκώμενη επιστροφή από κάθε κατάσταση με εφαρμογή της πολιτικής π, είναι μεγαλύτερη ή ίση από την αντίστοιχη επιστροφή στην αντίστοιχη κατάσταση με εφαρμογή της πολιτικής π'. Πιο απλά, $\pi \geq \pi'$ *αν και μόνο αν* $V^{\pi}(s) \geq V^{\pi'}(s), \forall s \in S$. Επιπλέον, πάντα υπάρχει τουλάχιστον μια πολιτική η οποία είναι καλύτερη ή ίση από κάθε άλλη πολιτική. Αυτή ονομάζεται *βέλτιστη πολιτική*. Όλες οι βέλτιστες πολιτικές συμβολίζονται π^*. Όλες μοιράζονται την ίδια συνάρτηση αποτίμησης-κατάστασης, η οποία καλείται *βέλτιστη συνάρτηση αποτίμησης-κατάστασης*, συμβολίζεται με $V^*(s)$ και ορίζεται ως:

$$V^*(s) = \max_{\pi} \Big\{ V^{\pi}(s) \Big\}, \ \forall s \in S. \tag{2.12}$$

Στο μοναδικό σημείο που μπορούν να διαφέρουν δύο βέλτιστες πολιτικές είναι στην κατανομή των πιθανοτήτων επιλογής των διαφόρων ενεργειών στις καταστάσεις εκείνες όπου υπάρχουν τουλάχιστον δύο διαφορετικές ενέργειες για τις οποίες η προσδοκώμενη τιμή είναι ίδια (και μάλιστα μεγαλύτερη από κάθε άλλη βάσει του ορισμού της βέλτιστης πολιτικής).

Σε αντιστοιχία με τα προηγούμενα, οι βέλτιστες πολιτικές μοιράζονται την ίδια *βέλτιστη συνάρτηση αποτίμησης-ενέργειας*, η οποία συμβολίζεται Q^* και ορίζεται ως:

$$Q^*(s,a) = \max_{\pi} \Big\{ Q^{\pi}(s,a) \Big\}, \ \forall s \in S \ \text{ και } \ \forall a \in A(s). \tag{2.13}$$

Για το ζευγάρι κατάστασης-ενέργειας $\langle s, a \rangle$, η παραπάνω συνάρτηση δίνει την προσδοκώμενη επιστροφή αν επιλέξει κάποιος την ενέργεια a στην κατάσταση s και από εκεί κι έπειτα ακολουθήσει μια βέλτιστη πολιτική. Έτσι, μπορούμε να εκφράσουμε την Q^* συναρτήσει της V^*:

$$Q^*(s,a) = E\{r_{t+1} + \gamma \cdot V^*(s_{t+1}) \mid s_t = s, a_t = a\} \tag{2.14}$$

Επειδή η V^* είναι η συνάρτηση αποτίμησης για μια πολιτική, θα πρέπει κι αυτή να ικανοποιεί τη συνάρτηση Bellman (2.10). Επειδή όμως είναι η βέλτιστη πολιτική, η εξίσωση αυτή μπορεί να πάρει μια ειδική μορφή χωρίς αναφορά σε καμία συγκεκριμένη πολιτική. Αυτή η μορφή είναι η εξίσωση Bellman για τη V^*, ή *η βέλτιστη εξίσωση Bellman (Bellman optimality equation)*. Διαισθητικά, η βέλτιστη εξίσωση Bellman εκφράζει το γεγονός πως η *αποτίμηση* μιας κατάστασης κάτω από βέλτιστη πολιτική πρέπει να ισούται με την προσδοκώμενη επιστροφή της καλύτερης ενέργειας από αυτή την κατάσταση:

$$\begin{aligned} V^*(s) &= \max_{a \in A(s)} Q^{\pi^*}(s,a) \\ &= \max_a E_{\pi^*} \Big\{ R_t \mid s_t = s, a_t = a \Big\} \end{aligned}$$

$$
\begin{aligned}
&= \max_a E_{\pi^*}\left\{\sum_{k=0}^{\infty}\gamma^k \cdot r_{t+k+1} \;\middle|\; s_t = s, a_t = a\right\} \\
&= \max_a E_{\pi^*}\left\{r_{t+1} + \gamma \cdot \sum_{k=0}^{\infty}\gamma^k \cdot r_{t+k+2} \;\middle|\; s_t = s, a_t = a\right\} \\
&= \max_a E_{\pi^*}\Big\{r_{t+1} + \gamma \cdot V^*(s_{t+1}) \mid s_t = s, a_t = a\Big\} \qquad (2.15) \\
&= \max_a \sum_{s'} \mathcal{P}^a_{ss'} \cdot \Big[\mathcal{R}^a_{ss'} + \gamma \cdot V^*(s')\Big]. \qquad (2.16)
\end{aligned}
$$

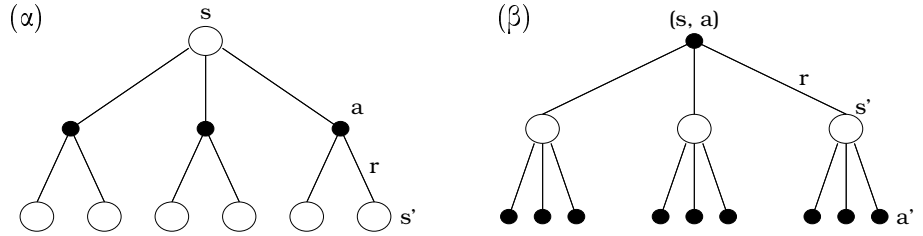
Οι τελευταίες δύο εξισώσεις είναι δύο μορφές της *βέλτιστης εξίσωσης Bellman*. Με όμοιο τρόπο, για τη συνάρτηση Q^*, μπορούμε να πάρουμε δύο παρόμοιες μορφές για τη *βέλτιστη εξίσωση Bellman*:

$$
\begin{aligned}
Q^*(s,a) &= E\Big\{r_{t+1} + \gamma \cdot \max_{a'} Q^*(s_{t+1}, a') \mid s_t = s, a_t = a\Big\} \qquad (2.17) \\
&= \sum_{s'} \mathcal{P}^a_{ss'} \cdot \Big[\mathcal{R}^a_{ss'} + \gamma \cdot \max_{a'} Q^*(s', a')\Big]. \qquad (2.18)
\end{aligned}
$$

Για πεπερασμένες Διαδικασίες Απόφασης Markov, οι εξισώσεις Bellman (2.16) και (2.18) έχουν μοναδικές λύσεις ανεξαρτήτως πολιτικής. Ουσιαστικά, υπάρχει μια εξίσωση για κάθε κατάσταση (ή ζευγάρι κατάστασης-ενέργειας), οπότε αυτό που αντιμετωπίζει κάποιος είναι ένα σύστημα N εξισώσεων με N αγνώστους. Αν η δυναμική του περιβάλλοντος είναι γνωστή ($\mathcal{P}^a_{ss'}$ και $\mathcal{R}^a_{ss'}$), τότε κατά κανόνα το προηγούμενο σύστημα μπορεί να λυθεί με οποιαδήποτε μέθοδο επίλυσης μη-γραμμικών συστημάτων.

Από τη στιγμή που κάποιος έχει τις ακριβείς τιμές της V^* είναι αρκετά εύκολο να καθοριστεί μια βέλτιστη πολιτική. Για κάθε κατάσταση s, θα υπάρχουν μία ή περισσότερες ενέργειες για τις οποίες θα ικανοποιείται η συνθήκη μεγίστου στην εξίσωση Bellman. Έτσι, οποιαδήποτε πολιτική αναθέτει μηδενικές πιθανότητες σε όλες τις υπόλοιπες ενέργειες και οποιεσδήποτε τιμές πιθανότητας επιλογής (που να έχουν άθροισμα 1) στις ενέργειες που αποφέρουν το μέγιστο, είναι μια βέλτιστη πολιτική. Με άλλα λόγια, οποιαδήποτε πολιτική λειτουργεί *άπληστα* σε σχέση με τις βέλτιστες τιμές της συνάρτησης V^* είναι μια βέλτιστη πολιτική. Η ομορφιά της V^* έγκειται στο γεγονός ότι εκτιμώντας τις συνέπειες ενός μόνο βήματος μπορούμε να αποφανθούμε για όλη την παραπέρα πορεία, επειδή η V^* δημιουργήθηκε λαμβάνοντας υπ'όψιν της όλες τις πιθανές ανταμοιβές από οποιαδήποτε μελλοντική συμπεριφορά. Επομένως με μια αναζήτηση ενός μόνο βήματος μπορούμε να ανακαλύψουμε τις ενέργειες εκείνες οι οποίες είναι βέλτιστες.

Από την άλλη, έχοντας κανείς τις τιμές της Q^* η επιλογή ενεργειών είναι ακόμη πιο εύκολη. Σε αυτή την περίπτωση δεν χρειάζεται καν να κάνουμε αναζήτηση, αφού όλη η πληροφορία που χρειαζόμαστε είναι μέσα στην Q^*. Έτσι επιλέγουμε την ενέργεια εκείνη η οποία μεγιστοποιεί την Q^* για τη συγκεκριμένη κατάσταση. Επομένως, το επιπλέον κόστος το οποίο απαιτείται για τη



Σχήμα 2.4: Παράδειγμα Διαγράμματος Ενημέρωσης
α) Για τη συνάρτηση V^π και β) για τη συνάρτηση Q^π.

φύλαξη τιμών σε ζευγάρια ⟨κατάσταση, ενέργεια⟩ δικαιολογείται από το γεγονός ότι στην περίπτωση αυτή δεν χρειάζεται να γνωρίζουμε τίποτα για τις μελλοντικές καταστάσεις (και τις αποτιμήσεις τους) που μπορούν να προκύψουν. Δηλαδή, σε αυτές τις περιπτώσεις δεν χρειάζεται να ξέρουμε τίποτα από τη δυναμική του περιβάλλοντος.

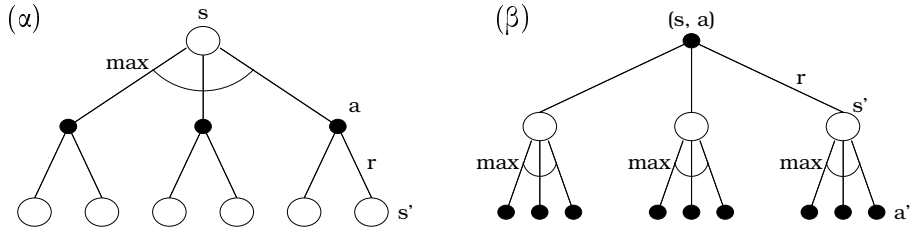
Η ακριβής επίλυση των βέλτιστων συναρτήσεων Bellman είναι ένας τρόπος για την εύρεση μιας βέλτιστης πολιτικής και επομένως επίλυσης μιας *εργασίας* Ενισχυτικής Μάθησης. Παρ'όλ'αυτά, αυτή η προσέγγιση είναι σπάνια χρήσιμη, αφού βασίζεται σε τρεις υποθέσεις οι οποίες είναι σπάνια χρήσιμες στην πράξη:

1. πρέπει να γνωρίζουμε επακριβώς τη δυναμική του περιβάλλοντος,
2. έχουμε αρκετή υπολογιστική ισχύ προκειμένου να βρούμε τη λύση και
3. την ιδιότητα Markov.

Το συνηθέστερο πρόβλημα είναι η υπολογιστική ισχύς μιας και η επίλυση των βέλτιστων εξισώσεων Bellman απαιτεί χρόνο πολυωνυμικό ως προς το πλήθος των καταστάσεων αλλά δυστυχώς οι καταστάσεις είναι πάρα πολλές για να περιμένουμε απάντηση σε ένα εύλογο χρονικό διάστημα. Έτσι, τις περισσότερες φορές είμαστε αναγκασμένοι να ψάχνουμε για προσεγγιστικές λύσεις. Τις περισσότερες μεθόδους επίλυσης Ενισχυτικής Μάθησης μπορούμε να τις δούμε σαν προσεγγιστικές λύσεις των βέλτιστων εξισώσεων Bellman οι οποίες χρησιμοποιούν εμπειρία από μεταβάσεις που έχουν γίνει στο παρελθόν στη θέση της δυναμικής του περιβάλλοντος.

2.2.3 Διαγράμματα Ενημέρωσης

Τις σχέσεις (2.10), (2.11) μπορούμε να τις δούμε εποπτικά στα διαγράμματα του σχήματος 2.4. Τα διαγράμματα αυτά ονομάζονται *διαγράμματα ενημέρωσης (backup diagrams)* επειδή δίνουν μια σχηματική περιγραφή των σχέσεων οι οποίες χρησιμοποιούνται σαν βάση στις *ενημερώσεις (updates, backup operations)* οι οποίες γίνονται στην Ενισχυτική Μάθηση. Αυτές οι ενημερώσεις



Σχήμα 2.5: Παράδειγμα Διαγράμματος Ενημέρωσης
α) Για τη συνάρτηση V^* και β) για τη συνάρτηση Q^*.

μεταφέρουν πληροφορία σχετική με τις αποτιμήσεις των προσδοκώμενων ενισχύσεων από κατώτερα επίπεδα σε ανώτερα. *Αντίθετα με τους γράφους μετάβασης καταστάσεων οι κόμβοι των καταστάσεων στα διαγράμματα ενημερώσεων δεν αναπαριστούν κατ'ανάγκη διαφορετικές καταστάσεις.* Αυτό οφείλεται στο γεγονός πως στα διαγράμματα αυτά υπάρχει και η έννοια του χρόνου. Μάλιστα, κατά σύμβαση θεωρούμε ότι ο χρόνος «ρέει» προς τα κάτω, επομένως μια κατάσταση μπορεί να είναι ο επόμενος του εαυτού της.

Τα διαγράμματα του σχήματος 2.5 δείχνουν γραφικά τη σχέση η οποία υπάρχει μεταξύ των μελλοντικών καταστάσεων και των ενεργειών στις εξισώσεις Bellman για τις V^* και Q^*. Τα διαγράμματα αυτά είναι παρόμοια με αυτά που εικονίζονται στο σχήμα 2.4 με τη διαφορά ότι υπάρχουν κάποιες καμάρες στα σημεία επιλογής του πράκτορα ώστε να περιγράφεται η έννοια πως ο πράκτορας διαλέγει την καλύτερη δυνατότητα που του προσφέρεται.

2.2.4 Ενημερώσεις Συναρτήσεων Αποτίμησης

Μέχρι τώρα έχουμε εξετάσει τις αναδρομικές εξαρτήσεις οι οποίες υπάρχουν μεταξύ των συναρτήσεων αποτίμησης για οποιαδήποτε πολιτική. Ένα κρίσιμο ζήτημα είναι πως προκύπτουν οι όποιες αποτιμήσεις έχει ο πράκτορας για τις διάφορες ενέργειες. Δηλαδή, με ποιον τρόπο οι ανταμοιβές τις οποίες δέχεται ο πράκτορας από το περιβάλλον επηρεάζουν τις συναρτήσεις αποτίμησης. Από μια άλλη οπτική γωνιά, θα μπορούσαμε να δούμε όσα ακολουθούν σαν ένα τρόπο έκφρασης της *αυτοκριτικής* που κάνει ο πράκτορας μετά από τη λήψη οποιασδήποτε ανταμοιβής ή οποιασδήποτε επιστροφής.

Ενημερώσεις Μέσου Όρου Δειγμάτων[4]

Ένας φυσικός τρόπος ενημέρωσης της αποτίμησης που έχει ο πράκτορας για μια συγκεκριμένη ενέργεια είναι να κρατάει μέσους όρους των εμπειρικών αποτελεσμάτων (πραγματικές επιστροφές) των διαφόρων ενεργειών. Έτσι λοιπόν, μία χρονική στιγμή t, η εκτίμηση την οποία έχει για το αποτέλεσμα μιας

[4]Ξενόγλωσση Ορολογία: Sample Average Backups

ενέργειας a σε μία κατάσταση s είναι ο μέσος όρος όλων των προηγούμενων παρατηρημένων επιστροφών αυτής της ενέργειας. Επομένως, αν τη συγκεκριμένη ενέργεια a την έχει δοκιμάσει ο πράκτορας k_a φορές στο παρελθόν στη συγκεκριμένη κατάσταση s και οι αντίστοιχες επιστροφές δίνονται από τα σήματα ενίσχυσης $R_1(s,a), R_2(s,a), \ldots, R_{k_a}(s,a)$, τότε η εκτίμηση που έχει ο πράκτορας για τη συγκεκριμένη ενέργεια δίνεται από τη σχέση:

$$Q_t(s,a) = \frac{R_1(s,a) + R_2(s,a) + \ldots + R_{k_a}(s,a)}{k_a} \tag{2.19}$$

Βέβαια, στην περίπτωση που ο πράκτορας δεν έχει προηγούμενη εμπειρία ο παραπάνω τύπος δεν μπορεί να εφαρμοστεί, αλλά τότε η τιμή $Q_t(s,a)$ έχει κάποια συγκεκριμένη αρχική τιμή. Συνήθως η τιμή αυτή είναι το μηδέν (0).

Ένα πρόβλημα το οποίο φαίνεται να υπάρχει με αυτή τη μέθοδο, είναι ότι οι απαιτήσεις της σε μνήμη μεγαλώνουν με το πέρασμα του χρόνου αν ο πράκτορας είναι αναγκασμένος να κρατάει κάθε επιμέρους επιστροφή (ακολουθία ανταμοιβών) από το περιβάλλον. Έτσι φαίνεται πως θα πρέπει να φυλάμε το άθροισμα των επιμέρους επιστροφών σε μια μεταβλητή και σε μια άλλη να φυλάμε το πλήθος των μεταβλητών που αποτελούν το συγκεκριμένο άθροισμα. Δυστυχώς και σε αυτή την περίπτωση ελλοχεύει κίνδυνος, αν και διαφορετικού είδους από πριν. Αν η διαδικασία μάθησης είναι «αργή», τότε ενδέχεται να εμφανιστούν φαινόμενα υπερχείλισης της μνήμης σε προβλήματα με πάρα πολύ μεγάλους χώρους αναζήτησης, μιας και οι διάφορες ενέργειες θα δοκιμασθούν πάρα πολλές φορές.

Ευτυχώς τα παραπάνω προβλήματα μπορούν να λυθούν αν παρατηρήσουμε το εξής:

$$\begin{aligned}
Q_k(s,a) &= \frac{1}{k} \cdot \sum_{i=1}^{k} R_i(s,a) \\
&= \frac{1}{k} \cdot \left(R_k(s,a) + \sum_{i=1}^{k-1} R_i(s,a) \right) \\
&= \frac{1}{k} \cdot \Big(R_k(s,a) + (k-1) \cdot Q_{k-1}(s,a) + Q_{k-1}(s,a) - Q_{k-1}(s,a) \Big) \\
&= \frac{1}{k} \cdot \Big(R_k(s,a) + k \cdot Q_{k-1}(s,a) - Q_{k-1}(s,a) \Big) \\
&= Q_{k-1}(s,a) + \frac{1}{k} \cdot \Big[R_k(s,a) - Q_{k-1}(s,a) \Big],
\end{aligned} \tag{2.20}$$

όπου $Q_{k-1}(s,a)$ συμβολίζει το μέσο όρο ο οποίος προκύπτει από $(k-1)$ παλαιότερα δείγματα και $R_k(s,a)$ συμβολίζει την επιστροφή (ακολουθία ανταμοιβών) που δέχθηκε ο πράκτορας την $k^{στη}$ φορά που επέλεξε την ενέργεια a στην κατάσταση s. Να σημειωθεί ότι η παραπάνω εξίσωση (2.20) ισχύει ακόμη και στην περίπτωση όπου $k = 0$, δίνοντας έτσι $Q_1(s,a) = R_1(s,a)$ για οποιαδήποτε αρχική τιμή $Q_0(s,a)$.

Η μορφή ενημέρωσης που έχει η εξίσωση (2.20) είναι η δημοφιλέστερη μορφή ενημέρωσης μιας μεταβλητής σε όλο το πεδίο του Νευρο-Δυναμικού Προγραμματισμού. Η γενική περιγραφή ενημέρωσης μιας μεταβλητής έχει τη μορφή:

$$\text{Νέα Πρόβλεψη} \longleftarrow \text{Παλαιά Πρόβλεψη} + \alpha \cdot \Big[\text{Στόχος} - \text{Παλαιά Πρόβλεψη}\Big] \tag{2.21}$$

Η έκφραση $\Big[\text{Στόχος} - \text{Παλαιά Πρόβλεψη}\Big]$ είναι ένα *σφάλμα* στην εκτίμηση και μειώνεται κάνοντας ένα βήμα προ τον "Στόχο". Ο στόχος υποθέτουμε ότι συνήθως δείχνει προς την σωστή κατεύθυνση προς την οποία πρέπει να γίνει η ενημέρωση αν και βέβαια ενδέχεται να περιέχει *θόρυβο*. Για παράδειγμα, στην εξίσωση (2.20), ο στόχος είναι η $k^{στη}$ επιστροφή που λαμβάνει ο πράκτορας. Επιπλέον, σε περιβάλλοντα στα οποία οι επιμέρους ανταμοιβές τις οποίες λαμβάνει ο πράκτορας προέρχονται από μια κατανομή, και άρα περιέχουν θόρυβο, τα επιμέρους δείγματα R_k δεν είναι *σχεδόν ποτέ* ακριβή. Όμως, αν μπορούμε να εξασφαλίσουμε ότι ο πράκτορας θα βρεθεί σε κάθε πιθανή θέση του χώρου άπειρες φορές και θα δοκιμάσει κάθε πιθανή ενέργεια από κάθε κατάσταση άπειρες φορές, τότε από το Νόμο των Μεγάλων Αριθμών, ο παραπάνω μέσος όρος είναι καταδικασμένος να συγκλίνει στις αντίστοιχες μέσες τιμές των κατανομών οι οποίες βρίσκονται πίσω από κάθε ζευγάρι $\langle \text{κατάσταση}\, s, \text{ενέργεια} a\rangle$.

Ακόμη, να σημειώσω εδώ, ότι η παράμετρος που καθορίζει το μέγεθος του βήματος που κάνουμε στην εκτίμησή μας προς την εκτίμηση-στόχο θα συμβολίζεται από τώρα και στο εξής ως α. Επίσης, η παράμετρος αυτή ονομάζεται πολλές φορές και *παράγοντας μάθησης*. Έτσι με το νέο συμβολισμό για τον παράγοντα μάθησης και χρησιμοποιώντας το γενικό μοτίβο που προτείνει η εξίσωση (2.21) η αντίστοιχη σχέση (2.20) μπορεί να γραφεί:

$$Q_k(s,a) = Q_{k-1}(s,a) + \alpha \cdot \Big[r_k(s,a) - Q_{k-1}(s,a)\Big] \tag{2.22}$$

Ενημερώσεις Εκθετικά Πρόσφατου-Σταθμισμένου Αθροίσματος[5]

Από τις εξισώσεις (2.20) και (2.22) μπορούμε να παρατηρήσουμε ότι προκειμένου ο πράκτορας να κρατάει τους μέσους όρους όλων των παρατηρούμενων δειγμάτων επιστροφής, πρέπει ο *παράγοντας μάθησης* α να είναι *κυμαινόμενος* κατά τη διάρκεια μάθησης - για την ακρίβεια *να ελαττώνεται*. Ένα ενδιαφέρον θέμα προκύπτει όταν ο παράγοντας μάθησης είναι *σταθερός* $(0 < \alpha \leq 1)$. Στην περίπτωση αυτή μπορούμε να παρατηρήσουμε τα ακόλουθα:

$$Q_k(s,a) \;\; = \;\; Q_{k-1}(s,a) + \alpha \cdot [R_k(s,a) - Q_{k-1}(s,a)]$$

[5] Ο αντίστοιχος όρος στα αγγλικά είναι: exponential recency-weighted average. Ένας άλλος όρος ο οποίος χρησιμοποιείται πιο συχνά από αυτόν της επικεφαλίδας είναι: «Ενημερώσεις Παραμέτρου Σταθερού Μεγέθους Βήματος» με αντίστοιχη ορολογία στα αγγλικά: Constant step-size parameter backup ή Constant-α backup. Ο λόγος για τον οποίο εκλέχθηκε η συγκεκριμένη ονομασία για επικεφαλίδα της ενότητας φαίνεται από την ανάλυση που ακολουθεί.

$$
\begin{aligned}
&= \alpha \cdot R_k(s,a) + (1-\alpha) \cdot Q_{k-1}(s,a) \\
&= \alpha \cdot R_k(s,a) + (1-\alpha) \cdot \alpha \cdot R_{k-1}(s,a) + (1-\alpha)^2 \cdot Q_{k-2}(s,a) \\
&= \dots \\
&= (1-\alpha)^k \cdot Q_0(s,a) + \sum_{i=1}^{k} \alpha \cdot (1-\alpha)^{k-i} \cdot R_i(s,a) \qquad (2.23)
\end{aligned}
$$

Το άθροισμα στο οποίο καταλήγουμε στην προηγούμενη εξίσωση είναι ένα *σταθμισμένο* άθροισμα μιας και το άθροισμα των βαρών $(1-\alpha)^k + \sum_{i=1}^{k} \alpha \cdot (1-\alpha)^{k-i}$ ισούται με τη μονάδα (1). Επίσης, το βάρος $\alpha \cdot (1-\alpha)^{k-i}$ το οποίο αντιστοιχεί στην επιστροφή $R_i(s,a)$, εξαρτάται από το *πόσο παλιά* την παρατήρησε ο πράκτορας. Μάλιστα, το βάρος *φθίνει εκθετικά* με το πέρασμα του χρόνου σύμφωνα με έναν παράγοντα $(1-\alpha)$. Οι δύο αυτές παρατηρήσεις έχουν οδηγήσει στη συγκεκριμένη ονομασία της μεθόδου η οποία δίνεται και σαν επικεφαλίδα αυτής της ενότητας.

Μια ακόμη ενδιαφέρουσα παρατήρηση η οποία μπορεί να γίνει από την τελευταία εξίσωση, είναι πως ο παράγοντας $Q_0(s,a)$ δεν εξαλείφεται τελείως όπως γινόταν στην προηγούμενη μορφή ενημέρωσης. Με άλλα λόγια, ο πράκτορας *θυμάται* πάντοτε την αρχική του εκτίμηση για κάθε πιθανή ενέργεια. Βέβαια, όσο περνάει ο καιρός, η επιρροή αυτή για κάθε πιθανή ενέργεια μειώνεται δραματικά και εξαλείφεται μόνο στο όριο καθώς η πραγματοποίηση των διαφόρων ενεργειών γίνεται άπειρες φορές, οπότε και ο πράκτορας στηρίζεται πλέον μόνο στις εμπειρικές παρατηρήσεις του. Όμως το φαινόμενο αυτό, δεν ισχύει για το συγκεκριμένο μόνο παράγοντα $Q_0(s,a)$, αλλά η βαρύτητα η οποία δίνεται σε κάθε μια εκτίμηση $Q_i(s,a)$ φθίνει εκθετικά με το πέρασμα του χρόνου. Συνεπώς, ο πράκτορας *θυμάται περισσότερο τις πλέον πρόσφατες ενισχύσεις* που δέχθηκε από το περιβάλλον. Οπότε, ακόμη κι αν το περιβάλλον δεν είναι στατικό, δηλαδή αποκρίνεται διαφορετικά στα ίδια δεδομένα, ο πράκτορας μπορεί να παρακολουθήσει πολύ πιο εύκολα αυτές τις μεταβολές σε σχέση με τις ενημερώσεις μέσου όρου. Έτσι, σε τέτοια περιβάλλοντα, έχει πολύ μεγαλύτερο νόημα να χρησιμοποιούμε τη συγκεκριμένη μορφή ενημέρωσης σταθερού παράγοντα μάθησης, παρά να κρατάμε το μέσο όρο όλων των παρατηρούμενων αποτελεσμάτων.

Σύγκλιση: Στις ενημερώσεις μέσου όρου οι τιμές των $Q_i(s,a)$ είναι αναγκασμένες να συγκλίνουν στις μέσες τιμές των εκάστοτε κατανομών που κρύβονται πίσω από τα ζευγάρια ⟨s, a⟩. Βέβαια, εκεί είχαμε μια συγκεκριμένη ακολουθία για τον παράγοντα μάθησης $\left(\alpha_k(s,a) = \frac{1}{k_{(s,a)}}\right)$. Ένα εύλογο ερώτημα το οποίο γεννιέται είναι για το τι γίνεται για μια *οποιαδήποτε ακολουθία* $\alpha_k(s,a)$*;* Για το ερώτημα αυτό υπάρχουν οι δύο παρακάτω συνθήκες, οι οποίες όταν ικανοποιούνται εξασφαλίζουν σύγκλιση με πιθανότητα 1:

$$
\sum_{k=1}^{\infty} \alpha_k(s,a) = \infty \qquad \text{και} \qquad \sum_{k=1}^{\infty} \alpha_k^2(s,a) < \infty \qquad (2.24)
$$

- Η πρώτη συνθήκη απαιτείται για να εξασφαλιστεί ότι τα «βήματα» είναι αρκετά μεγάλα ώστε τελικά να ξεπεραστούν οι όποιες αρχικές συνθήκες ή τα φαινόμενα του θορύβου.

- Η δεύτερη συνθήκη εξασφαλίζει ότι τελικά τα «βήματα» γίνονται αρκετά μικρά ώστε να γίνεται βέβαιη η σύγκλιση.

Έτσι, μπορούμε να παρατηρήσουμε πως και οι δύο συνθήκες σύγκλισης ικανοποιούνται από την περίπτωση ενημερώσεων μέσου όρου, αλλά αντίθετα δεν ικανοποιούνται στην περίπτωση που ο παράγοντας μάθησης παραμένει σταθερός. Στην τελευταία περίπτωση οι τιμές δεν συγκλίνουν, αλλά είναι πιο κοντά στα πρόσφατα δείγματα ενισχύσεων που δέχθηκε ο πράκτορας από το περιβάλλον. Βέβαια, όπως αναφέρθηκε και νωρίτερα, αυτό είναι ιδιαίτερα θεμιτό σε περιβάλλοντα τα οποία δεν είναι στατικά.

2.3 Εξερεύνηση

Η πιο βασική έννοια γύρω από τους αλγορίθμους Ενισχυτικής Μάθησης, είναι ο τρόπος με τον οποίο ο πράκτορας μαθαίνει να *προβλέπει* το αποτέλεσμα των διαφόρων ενεργειών. Αυτή η έννοια συναντάται ευρύτατα σε όλα τα εισαγωγικά εγχειρίδια στον συγκεκριμένο τομέα, όπως είναι για παράδειγμα τα [18] και [17]. Τα μόνα μέσα που διαθέτει ο πράκτορας προκειμένου να προβλέψει το αποτέλεσμα μιας μεμονωμένης ενέργειας ή μιας ακολουθίας ενεργειών είναι οι συναρτήσεις αποτίμησης.

Η συνάρτηση αποτίμησης παρέχει στον πράκτορα μια πρόβλεψη για την *προσδοκώμενη επιστροφή*. Στην προηγούμενη παράγραφο αναφέρθηκε ο όρος συναρτήσεις αποτίμησης και όχι συνάρτηση αποτίμησης γιατί αρκετές φορές ο πράκτορας καλείται να προβλέψει *και* την «άμεση» ανταμοιβή (reward) σε μια μετάβαση. Μια συνάρτηση με αυτή τη λειτουργικότητα σχεδόν ποτέ δεν υλοποιείται σε πραγματικά συστήματα, αφού όλη η πληροφορία που χρειαζόμαστε για να εξάγουμε τα αποτελέσματά της κρύβεται μέσα στη συνάρτηση αποτίμησης του πράκτορα. Παρ'όλ'αυτά, για λόγους πληρότητας χρησιμοποιείται πληθυντικός αριθμός υπονοώντας κι αυτή τη δεύτερη συνάρτηση. Ακόμη, χρειάζεται να διαχωρίσουμε πλήρως τη δεύτερη αυτή συνάρτηση *«άμεσης» αποτίμησης* από τη *συνάρτηση ανταμοιβής*. Κι αυτό γιατί η συνάρτηση ανταμοιβής είναι κάτι εξωτερικό από τον πράκτορα και δεν μπορεί να μεταβληθεί από αυτόν. Είναι μια συνάρτηση η οποία εξαρτάται από τα ιδιαίτερα χαρακτηριστικά του περιβάλλοντος. Αντίθετα, η συνάρτηση «άμεσης» αποτίμησης παρέχει στον πράκτορα μια *πρόβλεψη* για την προσδοκώμενη άμεση ανταμοιβή πριν πραγματοποιήσει κάποια συγκεκριμένη ενέργεια (μετάβαση κατάστασης) και είναι μια συνάρτηση η οποία *μεταβάλλεται* από τον ίδιο τον πράκτορα κατά τη διάρκεια της μάθησης. Η πραγματική όμως ανταμοιβή έρχεται σαν ένα σήμα ενίσχυσης από το περιβάλλον μέσω της συνάρτησης ανταμοιβής κατά τη διάρκεια μιας ενέργειας του πράκτορα ή μετά την ολοκλήρωσή της.

Όπως έχει γίνει ήδη φανερό, οι συναρτήσεις αποτίμησης αλλάζουν με το πέρασμα του χρόνου. Μάλιστα, οι προβλέψεις οι οποίες γίνονται μέσω αυτών των συναρτήσεων, πλησιάζουν με τον καιρό όλο και περισσότερο τα παρατηρούμενα σήματα ενίσχυσης τα οποία λαμβάνει ο πράκτορας από το περιβάλλον ή αν θέλετε, οι προβλέψεις των αποτελεσμάτων διαφόρων ενεργειών γίνονται με μεγαλύτερη ακρίβεια. Όμως προκειμένου να έχει ο πράκτορας πιο καλές εκτιμήσεις των αποτελεσμάτων συγκεκριμένων ενεργειών είναι αναγκαίο να δοκιμάζει και ενέργειες οι οποίες δεν φαίνονται να είναι οι καλύτερες δυνατές. Κι αυτό, γιατί πολύ απλά, για συγκεκριμένες ενέργειες ενδέχεται ο πράκτορας να έχει πολύ εσφαλμένη εντύπωση του τι πρόκειται να συμβεί στο μέλλον, με αποτέλεσμα να ενεργεί «υπο-βέλτιστα». Είναι επομένως φανερό πως η εξερεύνηση όλων των διαθέσιμων ενεργειών του πράκτορα είναι αναγκαία. Από την άλλη πλευρά βέβαια, μέσα από αυτή τη διαδικασία εξερεύνησης ο πράκτορας «μοιραία» θα δοκιμάζει την τύχη του σε καταστάσεις που θεωρεί ότι δεν είναι καλό να τις επισκέπτεται και πράγματι δεν είναι, αλλά προκειμένου να μην πέφτει σε «παγίδες» (τοπικά μέγιστα) είναι αναγκασμένος να περάσει και από εκεί.

Επομένως είναι ανάγκη, η εξερεύνηση νέων περιοχών ή περιοχών οι οποίες προς στιγμήν δείχνουν χειρότερες να γίνεται όσο το δυνατόν πιο εποικοδομητικά. Βέβαια, στην Ενισχυτική Μάθηση δεν υπάρχει κάποιος επιβλέποντας ο οποίος να λέει στον πράκτορα τι ήταν το καλύτερο να κάνει σε κάποια προηγούμενη κατάσταση. Ο πράκτορας από μόνος του πρέπει να πάρει την απόφαση να εξερευνήσει περισσότερο τις δυνατότητες και τη δυναμική του περιβάλλοντος. Συνεπώς, πρέπει να βρεθεί κάποιος τρόπος προκειμένου οι συναρτήσεις αποτίμησης - που είναι «πνευματική ιδιοκτησία» του πράκτορα - να μπορούν να καθοδηγήσουν τον πράκτορα *πότε* και *προς ποια κατεύθυνση* είναι η κατάλληλη στιγμή για εξερεύνηση. Οι τρόποι αυτοί θα παρουσιαστούν στη συνέχεια.

2.3.1 Μέθοδοι Επιλογής Ενεργειών

Άπληστη και ϵ-Άπληστη Επιλογή

Ο απλούστερος κανόνας βάσει του οποίου ο πράκτορας επιλέγει ποια ενέργεια θα πράξει σε μια συγκεκριμένη κατάσταση είναι να επιλέξει την ενέργεια (ή κάποια από τις ενέργειες) για την οποία η πρόβλεψη προσδοκώμενης επιστροφής από τις συναρτήσεις αποτίμησης είναι μέγιστη. Δηλαδή, αν ο πράκτορας βρίσκεται σε μια κατάσταση s τη χρονική στιγμή t, τότε επιλέγει μια ενέργεια a^* για την οποία ισχύει $Q_t(s,a^*) = \max_a Q_t(s,a)$. Η μέθοδος αυτή καλείται *μέθοδος Άπληστης Επιλογής (Greedy Selection)*. Όπως γίνεται φανερό, με αυτή τη μέθοδο, ο πράκτορας *διαρκώς* εκμεταλλεύεται τις γνώσεις που έχει για το περιβάλλον προκειμένου να μεγιστοποιεί την συνολική ανταμοιβή την οποία θα λάβει.

Μια μικρή παραλλαγή της προηγούμενης μεθόδου είναι ο πράκτορας να ενεργεί «άπληστα» τον περισσότερο καιρό, αλλά πότε-πότε, με μια μικρή πι-

θανότητα ϵ, να επιλέγει μια από τις διαθέσιμες κινήσεις του εντελώς τυχαία (ομοιόμορφη κατανομή). Αυτή η σχεδόν άπληστη μέθοδος επιλογής ονομάζεται *μέθοδος ϵ-Άπληστης Επιλογής (ϵ-Greedy Selection)*.

Ποια από τις δύο μεθόδους είναι καλύτερη έναντι της άλλης εξαρτάται από την εφαρμογή. Έτσι σε στατικά περιβάλλοντα, όπου η μετάβαση από μια κατάσταση s_1 σε μια κατάσταση s_2 μέσω μιας ενέργειας a πραγματοποιείται πάντοτε και δίνει πάντα την ίδια ανταμοιβή στον πράκτορα, οι Άπληστες Μέθοδοι έχουν κατά κανόνα καλύτερα αποτελέσματα από τις ϵ-Άπληστες Μεθόδους. Αντίθετα, αν το περιβάλλον είναι δεν είναι στατικό - και με αυτό εννοείται πως υπάρχει αβεβαιότητα για το που μπορεί να οδηγήσει μια ενέργεια τον πράκτορα ή/και οι ανταμοιβές τις οποίες λαμβάνει ο πράκτορας κατά τις διάφορες μεταβάσεις να αλλάζουν (π.χ. η μέση τιμή της κατανομής από την οποία προκύπτει κάποια ανταμοιβή μεταβάλλεται) - έχει ιδιαίτερη σημασία να εξερευνήσει ο πράκτορας όσο καλύτερα μπορεί το χώρο καταστάσεων (ή ζευγαριών καταστάσεων-ενεργειών).

SoftMax Επιλογή

Αν και οι ϵ-Άπληστες μέθοδοι είναι πολύ δημοφιλείς και τις περισσότερες φορές πολύ αποτελεσματικές, εντούτοις όταν ο πράκτορας αποφασίζει να κάνει μια κίνηση εξερεύνησης δίνει την ίδια πιθανότητα επιλογής $(1/\parallel A(s) \parallel)$ σε όλες τις διαθέσιμες κινήσεις. Κάτι τέτοιο μπορεί να μην είναι ιδιαίτερα επιθυμητό αν για ορισμένες ενέργειες του πράκτορα είναι πολύ πιθανό να ακολουθήσει μια πολύ άσχημη ανταμοιβή από το περιβάλλον. Για την αποφυγή λοιπόν τέτοιων περιπτώσεων, μια λύση είναι ο πράκτορας να δημιουργήσει μια κατανομή πιθανοτήτων για τις διάφορες διαθέσιμες ενέργειες κι έτσι η πιθανότητα εκλογής φαινομενικά άσχημων ενεργειών να είναι μικρότερη απ'ότι αυτών που φαίνονται να είναι καλύτερες. Όλες αυτές οι μέθοδοι καλούνται *μέθοδοι SoftMax επιλογών*. Η δημοφιλέστερη ίσως μέθοδος μεταξύ όλων αυτών, είναι εκείνη η οποία χρησιμοποιεί μια κατανομή Boltzmann. Έτσι, σύμφωνα με αυτή τη μέθοδο, μια ενέργεια a επιλέγεται με πιθανότητα:

$$\frac{e^{Q_t(s,a)/T}}{\sum_{b=1}^{n} e^{Q_t(s,b)/T}}, \tag{2.25}$$

όπου T μια θετική παράμετρος η οποία καλείται *θερμοκρασία*. Από την προηγούμενη σχέση γίνεται φανερό, πως όσο υψηλότερη είναι η θερμοκρασία, αυτό έχει σαν επίπτωση οι διάφορες ενέργειες που έχει στη διάθεσή του ο πράκτορας να είναι σχεδόν ισοπίθανες. Από την άλλη, χαμηλές θερμοκρασίες έχουν σαν αποτέλεσμα μεγαλύτερη διαφορά στην πιθανότητα εκλογής διαφόρων ενεργειών για τις οποίες διαφέρουν οι τιμές που έχει ο πράκτορας από τις συναρτήσεις αποτίμησης. Τελικά, στο όριο, καθώς $T \to 0$, η SoftMax Επιλογή γίνεται το ίδιο με την Άπληστη Επιλογή.

- Το ποια μέθοδος επιλογής είναι προτιμότερη είναι αβέβαιο, και τις περισσότερες φορές εξαρτάται από το πρόβλημα που έχουμε να αντιμετω-

πίσουμε. Ακόμη, εξαρτάται πως νιώθει και αυτός ο οποίος υλοποιεί την μέθοδο. Και οι δύο μέθοδοι (ϵ-Άπληστη Επιλογή και SoftMax Επιλογή) έχουν μόνο μια παράμετρο που πρέπει να οριστεί. Έτσι φαίνεται πιο εύκολο - και είναι υπολογιστικά πιο φθηνό - να χρησιμοποιήσει κανείς μια ϵ-Άπληστη Μέθοδο, μιας και οι SoftMax Μέθοδοι απαιτούν γνώσεις των δυνάμεων του e.

- Μία κεντρική ιδέα γύρω από την Ενισχυτική Μάθηση είναι οι ενέργειες του πράκτορα οι οποίες ακολουθούνται από μεγάλες επιστροφές να γίνονται με τον καιρό όλο και πιο πιθανές στην εκλογή τους. Έτσι, προς την κατεύθυνση αυτή, η λύση για τις παραπάνω μεθόδους είναι η σταδιακή μείωση της παραμέτρου ϵ για τις ϵ-Άπληστες μεθόδους και η μείωση της θερμοκρασίας T για τις SoftMax μεθόδους. Επομένως, με το πέρασμα του χρόνου ο πράκτορας όλο και λιγότερο θα κάνει κινήσεις εξερεύνησης με αποτέλεσμα όλο και περισσότερο να εκμεταλλεύεται παλαιότερη γνώση.

Επιλογή Ενισχυτικής Σύγκρισης[6]

Η προηγούμενη ιδέα είναι το κύριο χαρακτηριστικό των μεθόδων που περιγράφονται σε αυτή την ενότητα. Έτσι, σύμφωνα με αυτές τις μεθόδους επιλογής ενεργειών, ο πράκτορας κρίνει κάθε στιγμή αν η επιστροφή η οποία έλαβε από το περιβάλλον είναι «μικρή» ή «μεγάλη». Βέβαια, εδώ τίθεται το θέμα ποιες επιστροφές είναι «μικρές» και ποιες είναι «μεγάλες». Προκειμένου επομένως να μπορεί να λάβει μια τέτοια απόφαση ο πράκτορας, συγκρίνει την επιστροφή την οποία έλαβε από το περιβάλλον με μια καθορισμένη τιμή η οποία ονομάζεται *επιστροφή αναφοράς*. Επομένως, αν η επιστροφή από το περιβάλλον είναι μεγαλύτερη απ'ότι η επιστροφή αναφοράς, τότε η επιστροφή θεωρείται ότι είναι μεγάλη. Σε αντίθετη περίπτωση, θεωρείται ότι είναι μικρή. Οι μέθοδοι επιλογής αυτοί ονομάζονται *μέθοδοι επιλογής Ενισχυτικής Σύγκρισης (Reinforcement Comparison Selection Methods)*.

Στις μεθόδους Ενισχυτικής Σύγκρισης δεν διατηρούμε τιμές για κάθε πιθανή ενέργεια του πράκτορα. Αντίθετα, διατηρείται μια μόνο μεταβλητή, η επιστροφή αναφοράς. Προκειμένου όμως να μπορεί ο πράκτορας να επιλέγει ανάμεσα στις διάφορες ενέργειες, φυλάσσεται ένα ξεχωριστό *μέτρο προτίμησης* για την κάθε ενέργεια. Συνήθως[7], το μέτρο αυτό προτίμησης μιας ενέργειας a σε μια κατάσταση s μια χρονική στιγμή t το συμβολίζουμε με $p_t(s,a)$. Οι προτιμήσεις αυτές τώρα μπορούν να χρησιμοποιηθούν προκειμένου να καθορίσουν τις πιθανότητες επιλογής ενεργειών σύμφωνα με SoftMax συσχετισμούς, π.χ.

$$\pi_t(s,a) = \frac{e^{p_t(s,a)}}{\sum_{b=1}^{n} e^{p_t(s,b)}}, \tag{2.26}$$

[7]βλ. βιβλίο [26]

όπου $\pi_t(s,a)$ δηλώνει την πιθανότητα εκλογής της ενέργειας a στην κατάσταση s, τη χρονική στιγμή t. Φυσικά, αν οι πιθανότητες εκλογής παρέμεναν στατικές, τότε ο πράκτορας δεν θα μπορούσε να έχει πρόοδο στη διαδικασία μάθησης. Χρειάζεται επομένως να γίνεται *κάποιου είδους ενημέρωση* η οποία να αντικατοπτρίζει το πόσο καλή ή πόσο άσχημη ήταν μια συγκεκριμένη επιλογή από τη μεριά του πράκτορα. Έτσι, αυτό το οποίο γίνεται, είναι να ενημερώνουμε τις προτιμήσεις που έχουμε για τις διάφορες επιλογές. Και φυσικά προς αυτή την κατεύθυνση μας βοηθάει η *επιστροφή αναφοράς*. Τελικά, οι ενημερώσεις τις οποίες κάνουμε έχουν τη μορφή:

$$p_{t+1}(s,a) = p_t(s,a) + \beta \cdot [R_t(s) - \tilde{R}_t(s)], \tag{2.27}$$

όπου β μια θετική παράμετρος, $R_t(s)$ η πραγματική επιστροφή από το περιβάλλον και $\tilde{R}_t(s)$ η τιμή που έχει η επιστροφή αναφοράς τη χρονική στιγμή t για την κατάσταση s στην οποία βρισκόταν ο πράκτορας. Η προηγούμενη εξίσωση εφαρμόζει την ιδέα ότι για ενέργειες οι οποίες οδηγούν σε υψηλές τιμές επιστροφών θα πρέπει να αυξάνουμε την πιθανότητα εκλογής τους όταν βρισκόμαστε στη συγκεκριμένη κατάσταση s, ενώ χαμηλές τιμές επιστροφών θα πρέπει να μειώνουν αυτή την πιθανότητα.

Μια ακόμη πολύ δημοφιλής ενημέρωση στη συγκεκριμένη κατηγορία μεθόδων επιλογής δίνεται από την παρακάτω εξίσωση:

$$p_{t+1}(s,a) = p_t(s,a) + \beta \cdot \left(\frac{\sum_{b=1, b\neq a}^{n} e^{p_t(s,b)}}{\sum_{b=1}^{n} e^{p_t(s,b)}} \right) \cdot [R_t(s) - \tilde{R}_t(s)] \tag{2.28}$$

ή πιο απλά από την:

$$p_{t+1}(s,a) = p_t(s,a) + \beta \cdot \Big(1 - \pi_t(s,a)\Big) \cdot [R_t(s) - \tilde{R}_t(s)] \tag{2.29}$$

Ο λόγος για τον οποίο ορισμένες φορές εισάγεται ο παράγοντας $\Big(1-\pi_t(s,a)\Big)$ στην ενημέρωση της προτίμησης του πράκτορα για μια συγκεκριμένη ενέργεια θα εξηγηθεί αμέσως. Φανταστείτε τον πράκτορα να έχει πολύ άσχημες εκτιμήσεις για το πως θα εξελιχθούν τα πράγματα δεδομένου ότι θα ενεργήσει κατά συγκεκριμένο τρόπο - δηλαδή η επιστροφή αναφοράς $\tilde{R}_t(s)$ έχει μικρότερη τιμή απ'οποιαδήποτε ενίσχυση θα λάβει στην πραγματικότητα ο πράκτορας ανεξάρτητα από το ποια ενέργεια θα επιλέξει. Με άλλα λόγια, φανταστείτε έναν πράκτορα ο οποίος θέλει να μάθει κάτι αλλά είναι πολύ *απαισιόδοξος* στις προβλέψεις του. Τότε σύμφωνα με την ενημέρωση (2.27) ο πράκτορας θα μεταβάλλει κατά μεγάλο βαθμό την προτίμησή του προς μια ενέργεια a χωρίς απαραίτητα η συγκεκριμένη ενέργεια να ήταν καλή. Το πρόβλημα είναι πως ήταν τόσο απαισιόδοξος ώστε να μην μπορεί να διαχωρίσει τις καλές κινήσεις από τις άσχημες! Επομένως με την εισαγωγή του συγκεκριμένου παράγοντα η επίδραση του φαινομένου αυτού μειώνεται.

Επιλογή Επιδίωξης[8]

Η τελευταία αυτή κατηγορία μεθόδων επιλογής ενεργειών προσπαθεί να κατευθύνει τον πράκτορα γεφυρώνοντας δύο διαφορετικούς κόσμους. Έτσι, στις μεθόδους αυτής της κατηγορίας φυλάσσονται τόσο εκτιμήσεις των προσδοκώμενων επιστροφών από το περιβάλλον, όσο και προτιμήσεις προς τις διάφορες επιλογές, με τις προτιμήσεις συνεχώς να *επιδιώκουν* την ενέργεια η οποία είναι άπληστη σύμφωνα με τις εκτιμήσεις των προσδοκώμενων επιστροφών.

Οι πιθανότητες επομένως ενημερώνονται έτσι ώστε η άπληστη ενέργεια να είναι πιο πιθανό να εκλεχθεί. Έστω λοιπόν ότι με $a^*_{t+1} = \arg\max_a Q_{t+1}(s,a)$ δηλώνουμε την άπληστη ενέργεια (ή αν είναι περισσότερες από μια, κάποια από αυτές στην τύχη) για τη χρονική στιγμή $(t+1)$. Τότε η πιθανότητα να εκλέξουμε την ενέργεια $a_{t+1} = a^*_{t+1}$ αυξάνεται κατά ένα ποσοστό β προς το 1:

$$\pi_{t+1}(s, a^*_{t+1}) = \pi_t(s, a^*_{t+1}) + \beta \cdot \Big[1 - \pi_t(s, a^*_{t+1})\Big], \tag{2.30}$$

ενώ αντίθετα οι πιθανότητες επιλογής των υπολοίπων ενεργειών μειώνονται προς το 0:

$$\pi_{t+1}(s,a) = \pi_t(s,a) + \beta \cdot \Big[1 - \pi_t(s,a)\Big] \tag{2.31}$$

Ενημερώσεις Εκτιμήσεων των Μεθόδων Επιλογής Ενεργειών: Όπως αναφέρθηκε και στην αρχή της ενότητας, οι διάφορες μέθοδοι επιλογής ενεργειών οφείλουν να περιέχουν στο μηχανισμό τους ενημερώσεις πάνω στις συναρτήσεις αποτίμησης που διαθέτει ο πράκτορας. Έτσι, οποιαδήποτε από τις δύο μεθόδους ενημέρωσης των συναρτήσεων αποτίμησης μπορεί να ενσωματωθεί σε αυτές. Για τις περιπτώσεις των μεθόδων που χρησιμοποιούν αποκλειστικά συναρτήσεις αποτίμησης για τα διάφορα ζευγάρια $\langle$κατάσταση, ενέργεια$\rangle$ οι παραπάνω τεχνικές ενημέρωσής τους μεταφέρονται άμεσα, δηλαδή χρησιμοποιείται ο γενικός κανόνας ενημέρωσης που περιγράφεται από τη σχέση (2.22).

Η μόνη περίπτωση που χρειάζεται ιδιαίτερη μνεία, είναι αυτή των μεθόδων επιλογής Ενισχυτικής Σύγκρισης. Στην περίπτωση αυτή, δεν φυλάσσονται τιμές $Q(s,a)$, για κάθε ξεχωριστό ζευγάρι κατάστασης-τιμής, αλλά μία μόνο μεταβλητή, η *επιστροφή αναφοράς*. Έτσι, με ακριβώς όμοιο τρόπο, οι ενημερώσεις οι οποίες γίνονται στη συγκεκριμένη μεταβλητή έχουν τη μορφή:

$$\tilde{R}_{t+1}(s) = \tilde{R}_t(s) + \alpha \cdot [R_t(s) - \tilde{R}_t(s)] \tag{2.32}$$

Αισιόδοξες Αρχικές Τιμές

Βρίσκοντας αφορμή από τις τελευταίες δύο παραγράφους, να αναφερθούμε σε ένα ακόμη τελευταίο ζήτημα για τις τιμές των συναρτήσεων αποτίμησης. Όπως έχει γίνει φανερό, οι μέθοδοι επιλογής ενεργειών και οι μέθοδοι ενημέρωσης εκτιμήσεων λειτουργούν βοηθητικά η μία με την άλλη. Παρ'όλ'αυτά, οι

[8] Ξενόγλωσση ορολογία: Pursuit Selection Methods

αρχικές τιμές οι οποίες δίνουμε στις μεταβλητές $Q(s,a)$ επηρεάζουν τη συμπεριφορά του πράκτορα *ανεξάρτητα* από το ποια μέθοδο ενημέρωσης τιμών των συναρτήσεων αποτίμησης χρησιμοποιούμε. Στην περίπτωση της σταθερής παραμέτρου μάθησης, αυτό είναι φανερό αφού ο πράκτορας θυμάται πάντα την αρχική του εκτίμηση Q_0. Από την άλλη, ακόμα και στην περίπτωση της ενημέρωσης τιμών με μέσο όρο, η εκάστοτε τιμή $Q_0(s_i,a_j)$ καθορίζει την πιθανότητα με την οποία ο πράκτορας θα επιλέξει την ενέργεια a_j στην κατάσταση s_i την πρώτη φορά επιλογής της a_j, έστω κι αν μετά από αυτή την επιλογή η τιμή Q_0 θα «ξεχαστεί».

Μια χρήσιμη λοιπόν τεχνική που χρησιμοποιείται σε διάφορα *στατικά* περιβάλλοντα, είναι να θέτουμε τις αρχικές τιμές $Q_0(s_i,a_j), \forall s_i, \forall a_j \in A(s_i)$ σε τιμές οι οποίες είναι υψηλότερες απ'οποιαδήποτε επιστροφή μπορεί να προκύψει. Έτσι, σε αυτές τις περιπτώσεις, ο πράκτορας ξεκινάει τη διαδικασία μάθησης και είναι *αισιόδοξος* για τις διάφορες πιθανές εκβάσεις των διαφόρων ενεργειών. Το αποτέλεσμα αυτής της αισιοδοξίας, αναγκάζει τον πράκτορα να εξερευνήσει περισσότερο το χώρο, αφού οποιαδήποτε ενέργεια κι αν πράξει θα «απογοητευθεί» επειδή το αποτέλεσμα δεν ήταν αυτό που περίμενε! Συνεπώς, πριν ο πράκτορας αρχίσει να κατασταλάζει σε κάποιες συγκεκριμένες ενέργειες, θα είναι αναγκασμένος να δοκιμάσει πρώτα όλες τις υπόλοιπες (συνήθως περισσότερες από μια φορές). Στη συνέχεια, οι όποιες αποφάσεις του θα είναι καλύτερα τεκμηριωμένες, αφού θα αποτελούν απόρροια μιας εξονυχιστικής εξερευνητικής διαδικασίας η οποία έχει προηγηθεί στο χώρο καταστάσεων-ενεργειών. Επομένως, στις περιπτώσεις εκείνες που χρησιμοποιούμε *αισιόδοξες αρχικές τιμές*, δεν χρειάζεται η μέθοδος επιλογής ενεργειών του πράκτορα να έχει κάποια πιθανότητα εξερεύνησης. *Η εξερεύνηση είναι αναγκαστική λόγω των αρχικών τιμών. Ο πράκτορας επομένως οφείλει να συμπεριφέρεται άπληστα συνέχεια.*

Η εξερεύνηση αυτή βέβαια, πραγματοποιείται στα πρώιμα στάδια μάθησης του πράκτορα. Συνεπώς, δεν είναι μια τεχνική η οποία μπορεί να επεκταθεί και σε μη-στατικά περιβάλλοντα όπου το περιβάλλον διαρκώς αλλάζει τον τρόπο απόκρισής του στις διάφορες ενέργειες του πράκτορα. Παρ'ολ'αυτά, δεν είναι λίγες οι περιπτώσεις κατά τις οποίες το περιβάλλον που έχει να αντιμετωπίσει ο πράκτορας είναι στατικό. Μάλιστα, στις περιπτώσεις αυτές, όταν χρησιμοποιείται η τεχνική των *αισιόδοξων αρχικών τιμών*, τα αποτελέσματα είναι εντυπωσιακά.

Κεφάλαιο 3

Βασικές Μέθοδοι Μάθησης

Στο συγκεκριμένο κεφάλαιο θα αναφερθούν οι βασικότερες μέθοδοι μάθησης στην περιοχή της Ενισχυτικής Μάθησης. Με τις μεθόδους αυτές θα παρουσιαστούν οι πρώτες προσπάθειες οι οποίες έγιναν προκειμένου να έχουμε νοήμονες πράκτορες. Κάποιοι από τους πράκτορες αυτούς απαιτούν τη γνώση της δυναμικής του περιβάλλοντος και κάποιοι άλλοι όχι. Προς το τέλος του κεφαλαίου θα αναφερθούν επιγραμματικά και οι επεκτάσεις οι οποίες έχουν γίνει πάνω σε αυτούς τους πρώιμους αλγορίθμους, προκειμένου ο αναγνώστης να έχει μια σφαιρικότερη άποψη για τους διάφορους αλγορίθμους.

3.1 Δυναμικός Προγραμματισμός

Σε αυτή την ενότητα θα παρουσιάσουμε μια ομάδα αλγορίθμων οι οποίοι χρησιμοποιούνται για τον υπολογισμό βέλτιστων πολιτικών και περιγράφονται στη βιβλιογραφία με τον όρο *Δυναμικός Προγραμματισμός*. Οι αλγόριθμοι αυτοί απαιτούν ένα πλήρες μοντέλο της δυναμικής του περιβάλλοντος το οποίο ταυτόχρονα υποτίθεται ότι είναι μια πεπερασμένη Διαδικασία Απόφασης Markov. Οι ιδέες βέβαια μπορούν να επεκταθούν και σε συνεχείς χώρους καταστάσεων και ενεργειών, αλλά ακριβείς λύσεις είναι δυνατές μόνο σε ορισμένες περιπτώσεις. Μια συνήθης τακτική σε συνεχείς χώρους είναι η κβάντιση των χώρων αυτών και στη συνέχεια η εφαρμογή μεθόδων Δυναμικού Προγραμματισμού για πεπερασμένους χώρους.

Η χρήση συναρτήσεων αποτίμησης για την εύρεση καλών πολιτικών είναι η κεντρική ιδέα στην Ενισχυτική Μάθηση και δεν θα μπορούσε να λείπει από τις μεθόδους Δυναμικού Προγραμματισμού. Η εύρεση βέλτιστων πολιτικών είναι απλή από τη στιγμή που κάποιος υπολογίσει τις βέλτιστες συναρτήσεις αποτίμησης V^* ή Q^* οι οποίες ικανοποιούν τις βέλτιστες εξισώσεις Bellman (2.16) και (2.18) αντίστοιχα. Διαισθητικά, οι μέθοδοι Δυναμικού Προγραμματισμού μετατρέπουν τις προηγούμενες εξισώσεις σε αναθέσεις τιμών, δηλαδή σε κανόνες ενημέρωσης οι οποίοι βελτιώνουν τις προσεγγίσεις των επιθυμητών συναρτήσεων αποτίμησης.

3.1.1 Πολιτική

Εκτίμηση Πολιτικής

Προκειμένου να μπορούμε να συγκρίνουμε δύο διαφορετικές πολιτικές π_1, π_2, θα πρέπει να υπάρχει κάποιο μέτρο σύγκρισης για να συγκρίνουμε τις δύο αυτές πολιτικές. Το μέτρο αυτό δεν είναι άλλο από τις συναρτήσεις V^{π_i} και Q^{π_i}. Επομένως το κρίσιμο ζήτημα είναι ο υπολογισμός των συναρτήσεων αυτών. Ο υπολογισμός της συνάρτησης V^π για μια τυχαία πολιτική π καλείται *εκτίμηση πολιτικής* ή αναφέρεται συχνά και ως *το πρόβλημα πρόβλεψης*. Από το προηγούμενο κεφάλαιο έχουμε δείξει ότι για κάθε κατάσταση $s \in S$ ισχύει:

$$V^\pi = \sum_a \pi(s,a) \cdot \sum_{s'} \mathcal{P}^a_{ss'} \cdot [\mathcal{R}^a_{ss'} + \gamma \cdot V^\pi(s')]. \tag{3.1}$$

Η ύπαρξη και η μοναδικότητα της V^π εξασφαλίζεται αν είτε $\gamma < 1$ είτε ο τερματισμός είναι εξασφαλισμένος για όλες τις καταστάσεις κάτω από την πολιτική π.

Ξεκινάμε με μια τυχαία προσέγγιση V_0 για τη συνάρτηση V^π και υπολογίζουμε την πραγματική τιμή της συνάρτησης αυτής με κάποια επαναληπτική μέθοδο επίλυσης του παραπάνω συστήματος $|S|$ εξισώσεων. Στο σημείο αυτό πρέπει να σημειώσουμε πως κάθε τερματική κατάσταση πρέπει να έχει αρχική προσέγγιση ίση με το μηδέν. Έτσι, κάθε νέα προσέγγιση υπολογίζεται χρησιμοποιώντας την εξίσωση του Bellman για τη V^π σύμφωνα με την ανάθεση:

$$V_{k+1}(s) = \sum_a \pi(s,a) \cdot \sum_{s'} \mathcal{P}^a_{ss'} \cdot [\mathcal{R}^a_{ss'} + \gamma \cdot V_k(s')], \ \forall s \in S. \tag{3.2}$$

Προφανώς στην παραπάνω εξίσωση το $V_k = V^\pi$ είναι ένα σταθερό σημείο γι'αυτόν τον αναδρομικό κανόνα, αφού η εξίσωση του Bellman μας εξασφαλίζει την ισότητα σε αυτή την περίπτωση. Στην πραγματικότητα η ακολουθία $\{V_k\}$ συγκλίνει στην V^π καθώς $k \to \infty$ κάτω από τις ίδιες συνθήκες οι οποίες εξασφαλίζουν την ύπαρξη της V^π. Ο αλγόριθμος αυτός ονομάζεται *επαναληπτική εκτίμηση πολιτικής (iterative policy evaluation)*.

Προκειμένου να παραχθεί κάθε διαδοχική προσέγγιση V_{k+1} από την προηγούμενη V_k, η επαναληπτική εκτίμηση πολιτικής εφαρμόζει τη ίδια διαδικασία σε κάθε κατάσταση s: αντικαθιστά την παλαιά τιμή της s με μια νέα τιμή η οποία λαμβάνεται από τις παλαιές τιμές των επόμενων καταστάσεων της s καθώς και τις προσδοκώμενες άμεσες ενισχύσεις που μπορούν να προκύψουν από όλες αυτές τις πιθανές μεταβάσεις ενός βήματος. Η διαδικασία αυτή ονομάζεται *πλήρης ενημέρωση (full backup)* επειδή λαμβάνει υπ'όψιν της όλες τις πιθανές μεταβάσεις ενός βήματος.

Η σειρά με την οποία θα γίνουν τα προηγούμενα *περάσματα (sweeps)* από το χώρο των καταστάσεων (η ζευγαριών καταστάσεων-ενεργειών) έχει ιδιαίτερη σημασία στο ρυθμό σύγκλισης της μεθόδου. Συνήθως, χρησιμοποιείται ο ίδιος πίνακας για τις παλαιές και νέες τιμές των συναρτήσεων που προσεγγίζονται

μιας και με αυτόν τον τρόπο χρησιμοποιούνται όσο το δυνατόν πιο γρήγορα νέα δεδομένα (περισσότερη πληροφορία). Τέλος, ένα συνηθισμένο κριτήριο τερματισμού της μεθόδου είναι να ελέγχεται μετά από κάθε πέρασμα η ποσότητα $\max_{s\in S}|V_{k+1}(s)-V_k(s)|$ και να διακόπτεται η λειτουργία του αλγορίθμου μόλις η ποσότητα αυτή γίνει πολύ μικρή.

Βελτίωση Πολιτικής

Η εκτίμηση πολιτικής σίγουρα είναι χρήσιμη προκειμένου να συγκρίνουμε δύο διαφορετικές πολιτικές, όμως χρειαζόμαστε κι ένα εργαλείο προκειμένου να μπορούμε να βελτιώσουμε μια πολιτική. Προς την κατεύθυνση αυτή μας βοηθάει το επόμενο θεώρημα:

Θεώρημα 3.1.1 ***Θεώρημα Βελτίωσης Πολιτικής:*** *Έστω π και π' ένα ζευγάρι από ντετερμινιστικές πολιτικές, τέτοιο ώστε $\forall s \in S$ να ισχύει:*

$$Q^{\pi}(s,\pi'(s)) \geq V^{\pi}(s). \tag{3.3}$$

Τότε η πολιτική π' είναι καλύτερη ή ίση από την πολιτική π. Δηλαδή, η προσδοκώμενη επιστροφή ακολουθώντας την πολιτική π' είναι μεγαλύτερη ή ίση για όλες τις καταστάσεις $s \in S$:

$$V^{\pi}(s) \geq V^{\pi}(s). \tag{3.4}$$

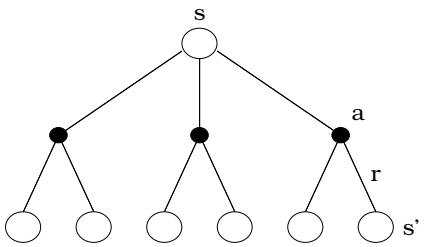
Μάλιστα, στην περίπτωση που υπάρχει αυστηρή ανισότητα στην (3.3) σε οποιαδήποτε κατάσταση, τότε πρέπει να υπάρχει αυστηρή ανισότητα της (3.4) σε μια ή περισσότερες καταστάσεις.

Επομένως, δοθέντος μιας πολιτικής π και της αντίστοιχης συνάρτησης αποτίμησης που συνοδεύει την πολιτική αυτή, αρκεί να βρούμε μια ενέργεια a, η οποία δεν καθορίζεται από την πολιτική π, σε μια οποιαδήποτε κατάσταση s, η εφαρμογή της οποίας να έχει καλύτερη προσδοκώμενη επιστροφή απ'ότι η εφαρμογή της ενέργειας που μας υπαγορεύει η πολιτική π για τη συγκεκριμένη κατάσταση s. Σε αυτή την περίπτωση μπορούμε να δημιουργήσουμε μια καλύτερη πολιτική ως εξής: στην κατάσταση s εφαρμόζουμε την ενέργεια a και σε όλες τις άλλες καταστάσεις ακολουθούμε την πολιτική π.

Μια φυσική προέκταση της προηγούμενης παρατήρησης είναι να πραγματοποιούνται αλλαγές σε όλες τις ενέργειες όλων των καταστάσεων. Δηλαδή, η νέα βελτιωμένη πολιτική π' μπορεί να οριστεί ως:

$$\begin{aligned}
\pi'(s) &= \arg\max_a Q^{\pi}(s,a) \\
&= \arg\max_a E\{r_{t+1} + \gamma \cdot V^{\pi}(s_{t+1}) \mid s_t = s, a_t = a\} \\
&= \arg\max_a \sum_{s'} \mathcal{P}^a_{ss'} \cdot \Big[\mathcal{R}^a_{ss'} + \gamma \cdot V^{\pi}(s')\Big],
\end{aligned} \tag{3.5}$$

όπου $\arg\max_a$ καθορίζει την ενέργεια a για την οποία η παράσταση που ακολουθεί μεγιστοποιείται. Εκ κατασκευής, η παραπάνω *άπληστη* πολιτική π'



Σχήμα 3.1: Παράδειγμα Διαγράμματος Ενημέρωσης Επανάληψης Πολιτικής.

ικανοποιεί τα κριτήρια του θεωρήματος 3.1.1, επομένως παράγει μια νέα καλύτερη πολιτική. Η διαδικασία αυτή ονομάζεται *βελτίωση πολιτικής (policy improvement)*. Η μόνη περίπτωση κατά την οποία δεν προκύπτει μια καλύτερη πολιτική π' από μια πολιτική π, είναι όταν η π είναι βέλτιστη.

3.1.2 Βέλτιστη Πολιτική

Επανάληψη Πολιτικής

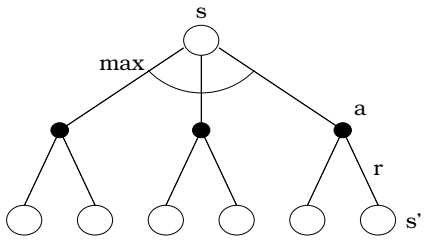
Ακολουθώντας τη διαδικασία εκτίμησης και βελτίωσης πολιτικής επαναληπτικά μπορούμε να πάρουμε μια ακολουθία βελτιώνουσων πολιτικών και αντίστοιχων συναρτήσεων αποτίμησης:

$$\pi_0 \xrightarrow{E} V^{\pi_0} \xrightarrow{B} \pi_1 \xrightarrow{E} V^{\pi_1} \xrightarrow{B} \cdots \xrightarrow{B} \pi_i \xrightarrow{E} V^{\pi_i} \xrightarrow{B} \cdots \xrightarrow{B} \pi^* \xrightarrow{E} V^*,$$

όπου $\xrightarrow{E}$ δηλώνει ένα βήμα εκτίμησης πολιτικής και $\xrightarrow{B}$ δηλώνει ένα βήμα βελτίωσης πολιτικής. Κάθε βήμα βελτίωσης πολιτικής σε αυτή τη διαδικασία είναι εξασφαλισμένο ότι θα δίνει μια πολιτική αυστηρά καλύτερη από την παλαιότερη με εξαίρεση την περίπτωση μιας βέλτιστης πολιτικής. Επίσης, επειδή οι πεπερασμένες Διαδικασίες Απόφασης Markov έχουν πεπερασμένο αριθμό διαφορετικών πολιτικών, η προηγούμενη διαδικασία πρέπει να συγκλίνει σε μια βέλτιστη πολιτική και μια βέλτιστη συνάρτηση αποτίμησης μέσα σε πεπερασμένο αριθμό επαναλήψεων. Ο τρόπος αυτός εύρεσης μιας βέλτιστης πολιτικής καλείται *επανάληψη πολιτικής (policy iteration)*. Το διάγραμμα ενημέρωσης του αλγορίθμου είναι αυτό που εικονίζεται στο σχήμα 3.1.

Επανάληψη Αποτίμησης

Ένα μειονέκτημα της επανάληψης πολιτικής είναι ότι σε κάθε επανάληψη εμπεριέχεται μια πλήρης εκτίμηση πολιτικής η οποία μπορεί να απαιτεί αρκετή υπολογιστική ισχύ. Παρ'όλ'αυτά, την υπολογιστική ισχύ την οποία απαιτεί το βήμα εκτίμησης πολιτικής μπορούμε να την περιορίσουμε με αρκετούς τρόπους χωρίς να χάνουμε τις εγγυήσεις σύγκλισης της επανάληψης πολιτικής. Μια σημαντική ειδική περίπτωση είναι στο βήμα εκτίμησης πολιτικής να κάνουμε



Σχήμα 3.2: Παράδειγμα Διαγράμματος Ενημέρωσης Επανάληψης Αποτίμησης.

μονάχα ένα πέρασμα από το χώρο των καταστάσεων. Ο αλγόριθμος αυτός καλείται *επανάληψη αποτίμησης (value iteration)* και περιγράφεται από μια απλή ενημέρωση η οποία συνδυάζει τη βελτίωση πολιτικής και λιγότερα βήματα εκτίμησης πολιτικής:

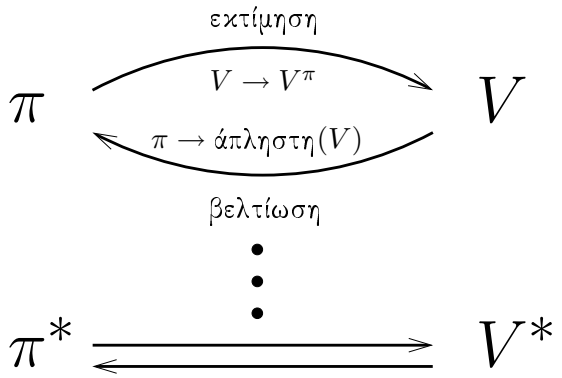
$$\begin{aligned} V_{k+1}(s) &= \max_a E\Big\{r_{t+1} + \gamma \cdot V_k(s_{t+1}) \mid s_t = s, a_t = a\Big\} \qquad (3.6) \\ &= \max_a \sum_{s'} \mathcal{P}^a_{ss'} \cdot \Big[\mathcal{R}^a_{ss'} + \gamma \cdot V_k(s')\Big] \forall s \in S. \end{aligned}$$

Πάλι, για τυχαία αρχική προσέγγιση της συνάρτησης αξίας V_0, η ακολουθία $\{V_k\}$ μπορεί να δειχθεί οτί συγκλίνει στη V^* κάτω από τις ίδιες συνθήκες οι οποίες εξασφαλίζουν την ύπαρξη της V^*. Το κριτήριο τερματισμού αυτής της μεθόδου είναι το ίδιο με το κριτήριο τερματισμού της εκτίμησης πολιτικής. Το αντίστοιχο διάγραμμα ενημέρωσης για την επανάληψη αποτίμησης είναι αυτό που εικονίζεται στο σχήμα 3.2.

Αξίζει να παρατηρήσουμε ότι η διαφορά πράξεων στην εκτίμηση πολιτικής που υπάρχει στους δύο τελευταίους αλγόριθμους εκφράζεται και γραφικά από τα αντίστοιχα διαγράμματα ενημερώσεων. Τέλος, να αναφέρουμε πως ορισμένες φορές, προκειμένου να έχουμε ταχύτερη σύγκλιση της μεθόδου, επιτρέπουμε περισσότερα από ένα περάσματα εκτίμησης πολιτικής και στη συνέχεια εφαρμόζουμε το βήμα βελτίωσης πολιτικής.

3.1.3 Γενικευμένη Επανάληψη Πολιτικής

Η επανάληψη πολιτικής καθώς και η επανάληψη αποτίμησης αποτελούνται από δύο ταυτόχρονες διεργασίες οι οποίες αλληλεπιδρούν μεταξύ τους. Η μία από αυτές κάνει τη συνάρτηση αποτίμησης συνεπή με την τρέχουσα πολιτική (εκτίμηση πολιτικής) και η άλλη (βελτίωση πολιτικής) κάνει την πολιτική άπληστη σε σχέση με την τρέχουσα συνάρτηση αποτίμησης. Για το λόγο αυτό, χρησιμοποιείται ο όρος *γενικευμένη επανάληψη πολιτικής (Generalized Policy Iteration - GPI)* προκειμένου να αναφερόμαστε στη γενική ιδέα της αλληλεπίδρασης της εκτίμησης πολιτικής και της βελτίωσης πολιτικής. Όλες σχεδόν οι μέθοδοι Ενισχυτικής Μάθησης μπορούν να περιγραφούν σαν γενικευμένες



Σχήμα 3.3: Γενικευμένη Επανάληψη Πολιτικής.
Η συνάρτηση αποτίμησης και η πολιτική αλληλεπιδρούν μέχρι να γίνουν βέλτιστες και άρα συνεπείς η μία ως προς την άλλη.

επαναλήψεις πολιτικής. Στο σχήμα 3.3 φαίνεται ένα παραστατικό διάγραμμα της διεργασίας της γενικευμένης επανάληψης πολιτικής.

Όταν και οι δύο επιμέρους διεργασίες που λαμβάνουν χώρα στη γενικευμένη επανάληψη πολιτικής σταθεροποιηθούν, τότε η συνάρτηση αποτίμησης και η πολιτική πρέπει να είναι βέλτιστες. Η συνάρτηση αποτίμησης σταθεροποιείται μόνο όταν είναι συνεπής με την τρέχουσα πολιτική και η πολιτική σταθεροποιείται μόνο όταν είναι άπληστη σε σχέση με την τρέχουσα συνάρτηση αποτίμησης. Επομένως, και οι δύο αυτές διεργασίες σταθεροποιούνται μόνο όταν η πολιτική η οποία έχει βρεθεί είναι άπληστη σε σχέση με τη συνάρτηση αποτίμησης που της αντιστοιχεί. Αυτό συνεπάγεται ότι η βέλτιστη εξίσωση Bellman (2.16) ή (2.18) ικανοποιείται και επομένως η πολιτική και η συνάρτηση αποτίμησης είναι βέλτιστες.

Τα βήματα εκτίμησης και βελτίωσης της πολιτικής στη γενικευμένη επανάληψη πολιτικής μπορούμε να τα δούμε σαν δύο εργασίες οι οποίες είναι ταυτόχρονα ανταγωνιστικές και συνεργαζόμενες. Είναι ανταγωνιστικές επειδή ακολουθούν διαφορετικές κατευθύνσεις. Από τη μια, κάνοντας την πολιτική άπληστη σε σχέση με τη συνάρτηση αποτίμησης ουσιαστικά η συνάρτηση αποτίμησης είναι εσφαλμένη για τη νέα πολιτική που προέκυψε και από την άλλη, κάνοντας τη συνάρτηση αποτίμησης συνεπή με την τρέχουσα πολιτική συνεπάγεται ότι η πολιτική δεν είναι πλέον βέλτιστη. Παρ'όλ'αυτά, αυτές οι δύο διεργασίες μακροπρόθεσμα συνεργάζονται προκειμένου να βρεθεί ένας κοινός παρονομαστής: μια βέλτιστη συνάρτηση αποτίμησης και μια βέλτιστη πολιτική.

3.1.4 Χαρακτηριστικά Δυναμικού Προγραμματισμού

Οι μέθοδοι Δυναμικού Προγραμματισμού εγγυώνται την εύρεση μιας βέλτιστης πολιτικής σε πολυωνυμικό χρόνο ως προς το πλήθος των καταστάσεων $|S|$

και ενεργειών $|A|$ μιας εργασίας. Το γεγονός αυτό είναι εκπληκτικό μιας και το πλήθος των ντετερμινιστικών πολιτικών που είναι διαθέσιμες για ένα τέτοιο πρόβλημα είναι $|S|^{|A|}$. Με άλλα λόγια, οι μέθοδοι Δυναμικού Προγραμματισμού είναι εκθετικά ταχύτεροι από οποιαδήποτε τυφλή αναζήτηση στο χώρο των πολιτικών της εκάστοτε εργασίας, επειδή οι τυφλές μέθοδοι αναζήτησης πρέπει να εξετάσουν εξαντλητικά όλες τις πιθανές πολιτικές προκειμένου να μπορούν να δώσουν τις ίδιες εγγυήσεις.

Παρ'όλ'αυτά, ο Δυναμικός Προγραμματισμός πολλές φορές θεωρείται ότι έχει περιορισμένες εφαρμογές εξαιτίας της *κατάρας της διαστασιμότητας του Bellman (Bellman's curse of dimensionality)*. Η κατάρα αυτή έχει να κάνει με το γεγονός ότι το πλήθος των καταστάσεων σε μια εργασία συχνά αυξάνει εκθετικά σε σχέση με το πλήθος των μεταβλητών κατάστασης. Όμως η δυσκολία αυτή δεν είναι ένα πρόβλημα για το οποίο ευθύνονται οι τεχνικές που χρησιμοποιούνται στο Δυναμικό Προγραμματισμό. Είναι μια «έμφυτη» δυσκολία του προβλήματος και στην πραγματικότητα ο Δυναμικός Προγραμματισμός είναι μια πολύ καλύτερη μέθοδος επίλυσης για μεγάλους χώρους καταστάσεων σε σχέση με μεθόδους τυφλής αναζήτησης ή μεθόδους Γραμμικού Προγραμματισμού (Linear Programming).

Με τα σημερινά δεδομένα υπολογιστικής ισχύς, οι μέθοδοι Δυναμικού Προγραμματισμού είναι δυνατόν να χρησιμοποιηθούν για Διαδικασίες Απόφασης Markov οι οποίες έχουν εκατομμύρια καταστάσεις. Μάλιστα το πρόβλημα το οποίο αντιμετωπίζεται στο τέλος της πτυχιακής ανήκει σε αυτή την κατηγορία εργασιών και προκειμένου να έχουμε κάποια μέτρα σύγκρισης για τις άλλες μεθόδους μάθησης, χρησιμοποιήσαμε Δυναμικό Προγραμματισμό για την επίλυσή του. Γενικά πάντως, οι μέθοδοι *επανάληψης πολιτικής* και *επανάληψης αποτίμησης* στην πράξη συγκλίνουν πολύ πιο γρήγορα από τους θεωρητικά χειρότερους αναμενόμενους χρόνους, ιδιαίτερα αν εκκινηθούν με καλές αρχικές προσεγγίσεις για τις συναρτήσεις αποτίμησης ή τις πολιτικές.

Μια τελευταία σημαντική ιδιότητα των μεθόδων Δυναμικού Προγραμματισμού είναι ότι οι ενημερώσεις των συναρτήσεων αποτίμησης γίνονται με τη βοήθεια *προσεγγίσεων* των συναρτήσεων αποτίμησης από επόμενες καταστάσεις. Δηλαδή, ενημερώνουν προσεγγίσεις βασισμένοι σε άλλες προσεγγίσεις. Αυτή η γενική ιδέα καλείται *bootstrapping*.

3.2 Monte Carlo Μέθοδοι

Σε αντίθεση με τις μεθόδους Δυναμικού Προγραμματισμού, οι μέθοδοι Monte Carlo δεν απαιτούν πλήρη γνώση του περιβάλλοντος. Αντίθετα, απαιτούν μόνο *εμπειρία*, αλληλεπίδραση δηλαδή με το περιβάλλον. Ο τρόπος με τον οποίο επιλύουν προβλήματα Ενισχυτικής Μάθησης βασίζεται σε μέσους όρους πάνω στις διάφορες ακολουθίες ανταμοιβών (πραγματικές επιστροφές) που επιστρέφονται από το περιβάλλον στον πράκτορα. Για το λόγο αυτό και προκειμένου οι επιστροφές να είναι καλά ορισμένες θεωρούμε πάντα πως οι μέ-

θοδοι εφαρμόζονται σε επεισοδιακές εργασίες. Οι όποιες ενημερώσεις γίνονται, λαμβάνουν χώρα μόνο στο τέλος κάθε επεισοδίου.

Η αποτίμηση μιας κατάστασης ή ενός ζευγαριού ⟨κατάστασης, ενέργειας⟩ δείχνει την προσδοκώμενη επιστροφή του πράκτορα για τη συγκεκριμένη κατάσταση ή ζευγάρι ⟨κατάστασης, ενέργειας⟩. Ένας προφανής τρόπος για να υπολογίζεται κάτι τέτοιο μέσω εμπειρίας είναι να κρατάει κανείς μέσους όρους από τις επιστροφές οι οποίες παρατηρούνται από το περιβάλλον από τη στιγμή που κάποιος επισκεφθεί τη συγκεκριμένη κατάσταση s ή ζευγάρι $\langle s, a\rangle$. Έτσι, καθώς οι παρατηρούμενες επιστροφές είναι όλο και περισσότερες, ο μέσος όρος πρέπει να συγκλίνει στην πραγματική προσδοκώμενη επιστροφή. Αυτή είναι και η κεντρική ιδέα όλων των μεθόδων Monte Carlo.

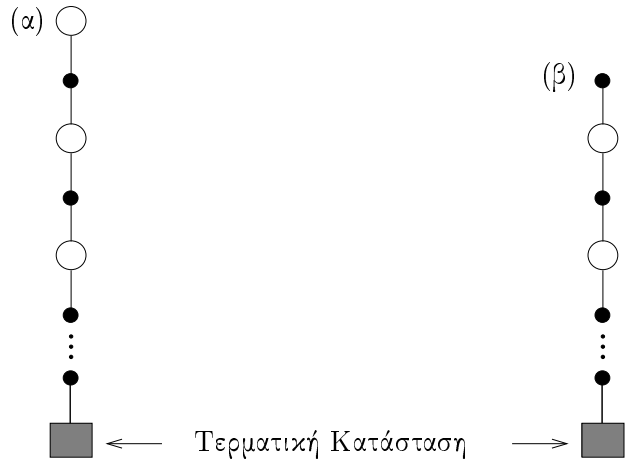
3.2.1 Πολιτική

Εκτίμηση Πολιτικής

Έχοντας κανείς την παραπάνω ιδέα σα γνώμονα μπορεί εύκολα να κάνει εκτίμηση μιας συγκεκριμένης πολιτικής π. Παρ'όλ'αυτά, ο τρόπος με τον οποίο γίνεται ο υπολογισμός της αποτίμησης μιας κατάστασης s ή ενός ζευγαριού $\langle s, a\rangle$ δεν είναι αρκετά ξεκάθαρος. Κι αυτό, γιατί για τα περισσότερα περιβάλλοντα που καλείται να αντιμετωπίσει ένας πράκτορας ενισχυτικής μάθησης είναι δυνατόν μέσα σε κάθε επεισόδιο να εμφανίζεται περισσότερες από μια φορές η ίδια κατάσταση s ή ζευγάρι $\langle s, a\rangle$. Κάθε εμφάνιση της s ή του $\langle s, a\rangle$ μέσα σε κάθε επεισόδιο καλείται *επίσκεψη* στην s ή στο $\langle s, a\rangle$ αντίστοιχα. Το παραπάνω φαινόμενο των πολλαπλών επισκέψεων σε ένα επεισόδιο έχει σαν αποτέλεσμα οι αλγόριθμοι Monte Carlo να διαχωρίζονται σε δύο κατηγορίες:

- Σε αυτούς που λαμβάνουν υπ'όψιν τους κάθε επίσκεψη και καλούνται *μέθοδοι Monte Carlo κάθε επίσκεψης (every-visit Monte Carlo methods)*, και
- Σε αυτούς που λαμβάνουν υπ'όψιν τους μόνο την πρώτη επίσκεψη του πράκτορα στη συγκεκριμένη κατάσταση s ή ζευγάρι $\langle s, a\rangle$ μέσα σε κάθε επεισόδιο και καλούνται *μέθοδοι Monte Carlo πρώτης επίσκεψης (first-visit Monte Carlo methods)*.

Οι *μέθοδοι Monte Carlo κάθε επίσκεψης* υπολογίζουν τις τιμές αποτίμησης $V^\pi(s)$ ή $Q^\pi(s, a)$ παίρνοντας τον μέσο όρο των επιστροφών κάθε επίσκεψης της s ή του $\langle s, a\rangle$ μέσα από ένα σύνολο επεισοδίων τα οποία έχουν δημιουργηθεί ακολουθώντας την πολιτική π. Αντίθετα, οι *μέθοδοι Monte Carlo πρώτης επίσκεψης* δημιουργούν τους αντίστοιχους μέσους όρους για τις συναρτήσεις αποτίμησης λαμβάνοντας υπ'όψιν τους τις επιστροφές οι οποίες ακολουθούν από το περιβάλλον μετά από την πρώτη επίσκεψη στην s ή στο $\langle s, a\rangle$ μέσα σε κάθε επεισόδιο το οποίο έχει δημιουργηθεί ακολουθώντας την πολιτική π. Έτσι, όσο περισσότερα επεισόδια έχει σαν εμπειρία ένας πράκτορας για μια συγκεκριμένη κατάσταση s ή ζευγάρι $\langle s, a\rangle$, τόσο καλύτερη θα είναι και η εκτίμησή του για



Σχήμα 3.4: Παράδειγμα Διαγράμματος Ενημέρωσης μεθόδων Monte Carlo. α) Για τη συνάρτηση V και β) για τη συνάρτηση Q.

τη συνάρτηση αποτίμησης που χρησιμοποιεί (για τη συγκεκριμένη πολιτική π). Τα αντίστοιχα διαγράμματα ενημερώσεων των συναρτήσεων V^π και Q^π για τους αλγορίθμους πρώτης επίσκεψης φαίνονται στη συνέχεια:

Βελτίωση Πολιτικής

Το σχήμα το οποίο ακολουθείται για την βελτίωση πολιτικής και στους συγκεκριμένου αλγορίθμους είναι αυτό της γενικευμένης επανάληψης πολιτικής. Το μόνο πρόβλημα το οποίο φαίνεται να υπάρχει με τους συγκεκριμένους αλγορίθμους είναι το γεγονός ότι για συγκεκριμένες ντετερμινιστικές πολιτικές κάποιες ενέργειες ενδεχομένως να μην εκλεχθούν ποτέ σε κανένα επεισόδιο. Επομένως, δεν είναι δυνατόν να βελτιώνουμε την πολιτική για συγκεκριμένες εργασίες αν δεν διαθέτουμε εμπειρία από τις συγκεκριμένες αυτές καταστάσεις/ενέργειες.

Η παραπάνω παρατήρηση έχει οδηγήσει στην *υπόθεση των εκκινήσεων εξερεύνησης (assumption of exploring starts)*. Σύμφωνα με την υπόθεση αυτή, πρέπει κάθε σύνολο επεισοδίων το οποίο δημιουργείται ακολουθώντας την πολιτική π να περιλαμβάνει επισκέψεις για όλες τις πιθανές καταστάσεις $s \in S$ ή πιθανών ζευγαριών $\langle s, a\rangle, s \in S \bigwedge a \in A(s)$. Έτσι, ένας εύκολος τρόπος για να επιτευχθεί κάτι τέτοιο είναι να δίνεται σε κάθε πιθανή κατάσταση s ή ζευγάρι $\langle s, a\rangle$ μη-μηδενική πιθανότητα για να ξεκινήσει από εκεί ένα επεισόδιο.

Ένα ακόμη σημείο στο πρέπει να σταθεί κανείς είναι το γεγονός πως οι μέσοι όροι από τις παρατηρούμενες επιστροφές προσεγγίζουν επακριβώς τις προσδοκώμενες επιστροφές του πράκτορα για τις συγκεκριμένες καταστάσεις ή ζευγάρια καταστάσεων-ενεργειών καθώς η εμπειρία την οποία διαθέτει ο πράκτορας για αυτές/αυτά πλησιάζει το άπειρο. Βέβαια, το ίδιο πρόβλημα υπήρχε

και στο δυναμικό προγραμματισμό. Μια από τις λύσεις που είχαν προταθεί εκεί και χρησιμοποιείται ευρύτατα στην πράξη είναι η επανάληψη αποτίμησης. Έτσι, το λογικό αντίστοιχο για τους αλγορίθμους Monte Carlo, είναι να γίνεται βελτίωση πολιτικής μετά από κάθε επεισόδιο χωρίς να χρειάζεται να περιμένει κανείς να ολοκληρωθεί η εκτίμηση πολιτικής για τη συγκεκριμένη πολιτική την οποία ακολουθεί ο πράκτορας.

Βέλτιστη Πολιτική: Ακολουθώντας την προηγούμενη τακτική και κάτω από το πρίσμα της γενικευμένης επανάληψης πολιτικής μπορούμε να βρούμε μια βέλτιστη πολιτική για μια συγκεκριμένη εργασία. Παρ'όλ'αυτά, κάτι τέτοιο δεν έχει αποδειχθεί αυστηρά ακόμη, αλλά όλα τα πειραματικά αποτελέσματα μέχρι σήμερα επιβεβαιώνουν αυτόν τον κανόνα.

Άμεση και Έμμεση Εκτίμηση Πολιτικής

Ένα άλλο σημαντικό ζήτημα είναι ότι πέραν των μεθόδων Δυναμικού Προγραμματισμού, σχεδόν όλες οι υπόλοιπες μέθοδοι ενισχυτικής μάθησης - μεταξύ αυτών και οι μέθοδοι Monte Carlo - είναι δυνατόν να υπολογίζουν (βέλτιστες) πολιτικές με δύο διαφορετικούς τρόπους.

Όπως έχει γίνει φανερό, αυτό το οποίο χρειάζεται ένας πράκτορας προκειμένου να υπολογίσει τις συναρτήσεις αποτίμησης για μια εργασία (εκτίμηση πολιτικής) στις μεθόδους Monte Carlo είναι επεισόδια τα οποία θα δίνουν αρκετή εμπειρία στον πράκτορα για κάθε πιθανή θέση. Δεν έχει γίνει όμως κανένας λόγος σχετικά με το πως αυτά τα επεισόδια δημιουργούνται. Σε όλη την προηγούμενη ανάλυση υποθέσαμε ότι τα επεισόδια αυτά δημιουργούνται ακολουθώντας την πολιτική π για την οποία θέλουμε να υπολογίσουμε κάποια συνάρτηση αποτίμησης. Όμως κάτι τέτοιο δεν είναι απαραίτητο. Μπορούμε να υπολογίσουμε τις συναρτήσεις αποτίμησης V^{π} και Q^{π} έχοντας σαν δεδομένα επεισόδια τα οποία δημιουργήθηκαν με αλληλεπίδραση με το περιβάλλον ακολουθώντας μια πολιτική $\pi' \neq \pi$. Ο μόνος περιορισμός προκειμένου κάτι τέτοιο να είναι εφικτό, είναι κάθε πιθανή ενέργεια την οποία ενδεχομένως να ακολουθήσουμε σε ένα επεισόδιο ακολουθώντας την πολιτική π να μπορούμε να την ακολουθήσουμε και σε κάποιο επεισόδιο ακολουθώντας την πολιτική π'. Με άλλα λόγια, σε αυτές τις περιπτώσεις απαιτούμε η πιθανότητα $\pi(s,a) > 0$ να υπονοεί την πιθανότητα $\pi'(s,a) > 0$. Στις περιπτώσεις τώρα που τα επεισόδια δημιουργούνται από την ίδια την πολιτική της οποίας τη συνάρτηση αποτίμησης θέλουμε να υπολογίσουμε λέμε ότι έχουμε αλγορίθμους *άμεσης εκτίμησης (on-policy algorithms)*, ενώ στις περιπτώσεις εκείνες που χρησιμοποιείται διαφορετική πολιτική για τη δημιουργία επεισοδίων από αυτή για τον υπολογισμό της συνάρτησης αποτίμησης λέμε ότι έχουμε αλγορίθμους *έμμεσης εκτίμησης (off-policy algorithms)*.

3.2.2 Χαρακτηριστικά Μεθόδων Monte Carlo

Ένα σημαντικό χαρακτηριστικό των μεθόδων Monte Carlo είναι το γεγονός ότι ο πράκτορας μαθαίνει άμεσα από την αλληλεπίδρασή του με το περιβάλλον στο τέλος κάθε επεισοδίου. Αντίθετα, στις μεθόδους Δυναμικού Προγραμματισμού δεν μπορούμε να παρατηρήσουμε κάτι τέτοιο. Εκεί, ο πράκτορας δεν μπορεί να αποφανθεί για καμία κατάσταση εάν δεν ολοκληρωθεί η διαδικασία μάθησης. Βέβαια, από τη στιγμή που η διαδικασία ολοκληρωθεί, ο πράκτορας γνωρίζει τη βέλτιστη πολιτική για κάθε πιθανό ενδεχόμενο. Όμως, στις περιπτώσεις που θέλουμε ο πράκτορας να μαθαίνει από την αλληλεπίδρασή του με το περιβάλλον και κατά τη διάρκεια της μάθησης, τότε είναι φανερό πως δεν μπορούμε να χρησιμοποιήσουμε Δυναμικό Προγραμματισμό ενώ οι μέθοδοι Monte Carlo επιτρέπουν κάτι τέτοιο. Ένα ακόμη σημαντικό στοιχείο το οποίο μπορεί να μας οδηγήσει προς αυτή την κατεύθυνση είναι το γεγονός ότι δεν χρειάζεται να γνωρίζει ο πράκτορας τη δυναμική του περιβάλλοντος προκειμένου να αποφανθεί για μια κατάσταση. Η εμπειρία του από παλαιότερες αλληλεπιδράσεις μπορεί να τον καθοδηγήσει στις επιλογές του.

Επίσης, ένα άλλο σημαντικό συστατικό το οποίο πρέπει να αναγνωρίσει κανείς στις μεθόδους Monte Carlo είναι το γεγονός ότι η εκτίμηση που έχει ο πράκτορας για κάθε κατάσταση ή ζευγάρι κατάστασης-ενέργειας είναι ανεξάρτητη του συνολικού πλήθους των καταστάσεων ή των ζευγαριών καταστάσεων-ενεργειών αντίστοιχα. Η συνάρτηση αποτίμησης για μια δεδομένη κατάσταση s ή ένα δεδομένο ζευγάρι $\langle s, a\rangle$ δεν δημιουργείται με τη βοήθεια άλλων προσεγγιστικών τιμών από τη συνάρτηση αποτίμησης για άλλες καταστάσεις ή ζευγάρια καταστάσεων-ενεργειών όπως γίνεται με τους αλγορίθμους Δυναμικού Προγραμματισμού. Με άλλα λόγια, στους αλγορίθμους αυτούς δεν εμφανίζεται η έννοια του bootstrapping.

Πιο συγκεκριμένα, το υπολογιστικό κόστος για την εκτίμηση της συνάρτησης αποτίμησης για μια δεδομένη κατάσταση s είναι ανεξάρτητο του πλήθους των καταστάσεων $|S|$ της συγκεκριμένης εργασίας. Το ίδιο ισχύει και για την εκτίμηση της συνάρτησης αποτίμησης ενός ζευγαριού $\langle s, a\rangle$ η οποία είναι ανεξάρτητη του συνολικού πλήθους των ενεργειών $\sum_{s\in S}|A(s)|$ για τη συγκεκριμένη εργασία. Όμως η ιδιότητα αυτή καθιστά τις μεθόδους Monte Carlo ιδιαίτερα ελκυστικές όταν κάποιος θέλει να υπολογίσει τη συνάρτηση αποτίμησης για ένα υποσύνολο του χώρου καταστάσεων ή ζευγαριών καταστάσεων-ενεργειών. Έτσι, ξεκινώντας κάποιος επεισόδια μόνο από τις καταστάσεις που τον ενδιαφέρουν μπορεί να υπολογίσει τη συνάρτηση αποτίμησης για αυτές τις καταστάσεις αγνοώντας όλες τις υπόλοιπες.

Τέλος, να σημειωθεί ότι με τις ενημερώσεις μέσου όρου οι οποίες γίνονται στις Monte Carlo μεθόδους αυτό το οποίο επιτυγχάνεται είναι ελαχιστοποίηση του RMS σφάλματος για τις συναρτήσεις αποτίμησης που αντιπροσωπεύουν το δείγμα των επεισοδίων που χρησιμοποιήθηκαν κατά τη διάρκεια της μάθησης.

3.3 Μάθηση Χρονικών Διαφορών

Η πιο καινοτόμος ιδέα σε όλο το φάσμα της Ενισχυτικής Μάθησης είναι η *Μάθηση Χρονικών Διαφορών (Temporal Difference Learning)*. Η μέθοδος αυτή συνδυάζει τις ιδέες του Δυναμικού Προγραμματισμού και των μεθόδων Monte Carlo. Όπως γίνεται στο Δυναμικό Προγραμματισμό, έτσι και στις μεθόδους Χρονικών Διαφορών οι ενημερώσεις οι οποίες γίνονται υπολογίζουν τιμές της συνάρτησης αποτίμησης βασισμένες σε τιμές άλλων καταστάσεων των συναρτήσεων αποτίμησης. Δηλαδή εμφανίζεται το φαινόμενο του bootstrapping. Από την άλλη, δεν απαιτείται γνώση της δυναμικής του περιβάλλοντος και μπορούν να μάθουν *άμεσα* από την εμπειρία τους με την αλληλεπίδραση που έχουν με αυτό όπως γίνεται και στις μεθόδους Monte Carlo.

3.3.1 Πολιτική

Εκτίμηση Πολιτικής

Ένα σημείο στο οποίο διαφέρουν οι μέθοδοι Χρονικών Διαφορών από τις μεθόδους Monte Carlo είναι το γεγονός ότι για να μάθουν από την εμπειρία τους δεν απαιτείται τερματισμός του επεισοδίου. Μαθαίνουν *κάθε χρονική στιγμή* από την αλληλεπίδραση την οποία έχουν με το περιβάλλον. Η απλούστερη μέθοδος μάθησης Χρονικών Διαφορών είναι γνωστή ως TD(0) και χρησιμοποιεί σαν κανόνα μάθησης την ενημέρωση:

$$V(s_t) \leftarrow V(s_t) + \alpha \cdot \Big[r_{t+1} + \gamma \cdot V(s_{t+1}) - V(s_t) \Big]. \tag{3.7}$$

Η παραπάνω ενημέρωση γίνεται όταν ο πράκτορας εργάζεται πάνω στο χώρο των καταστάσεων, δηλαδή υπολογίζει τη συνάρτηση αποτίμησης V. Στην περίπτωση που ο πράκτορας εργάζεται πάνω σε ενέργειες, τότε η αντίστοιχη ενημέρωση είναι:

$$Q(s_t, a_t) \leftarrow Q(s_t, a_t) + \alpha \cdot \Big[r_{t+1} + \gamma \cdot Q(s_{t+1}, a_{t+1}) - Q(s_t, a_t) \Big] \tag{3.8}$$

Από τις παραπάνω εξισώσεις γίνεται φανερό πως ο υπολογισμός της εκάστοτε συνάρτησης αποτίμησης βασίζεται σε προσεγγίσεις για δύο λόγους:

- Αφ'ενός, γιατί η τιμή που χρησιμοποιείται στην παραπάνω ενημέρωση r_{t+1} για την άμεση ανταμοιβή που δέχεται ο πράκτορας από το περιβάλλον είναι απλά ένα *δείγμα* της ανταμοιβής που θα δέχεται ο πράκτορας σε αυτή τη μετάβαση κι όχι η πραγματική μέση τιμή της κατανομής από την οποία προέρχεται αυτό το δείγμα.

- Κι αφ'ετέρου, γιατί όπως και στον Δυναμικό Προγραμματισμό, η τιμή της συνάρτησης αποτίμησης $V(s_{t+1})$ ή $Q(s_{t+1}, a_{t+1})$ αποτελεί μια προσέγγιση κι όχι την πραγματική προσδοκώμενη επιστροφή για τον πράκτορα.

Επομένως, ακολουθώντας μια πολιτική π και εφαρμόζοντας έναν από τους παραπάνω τύπους μπορούμε να υπολογίσουμε τη συνάρτηση αποτίμησης για τη συγκεκριμένη πολιτική π.

Βελτίωση Πολιτικής

Ακολουθώντας το γενικό σχήμα το οποίο προτείνεται από την γενικευμένη επανάληψη πολιτικής ο πράκτορας μπορεί να βελτιώνει με το πέρασμα του χρόνου την πολιτική του για μια εργασία. Μάλιστα, η βελτίωση αυτή γίνεται κάθε χρονική στιγμή αφού οι ενημερώσεις λαμβάνουν χώρα μετά από κάθε ενέργεια του πράκτορα.

Έτσι, για τον παραπάνω αλγόριθμο αποδεικνύεται ότι υπάρχει σύγκλιση στην μέση τιμή της V^{π} αν ο παράγοντας μάθησης είναι σταθερός και αρκετά μικρός καθώς επίσης και με πιθανότητα 1 αν ο παράγοντας μάθησης ικανοποιεί τα κριτήρια σύγκλισης (2.24).

Ακολουθώντας το σχήμα της γενικευμένης επανάληψης πολιτικής υπάρχει πάντοτε το κρίσιμο ζήτημα της ανταλλαγής μεταξύ ενεργειών εξερεύνησης και ενεργειών οι οποίες εκμεταλλεύονται παλαιότερη γνώση. Έτσι, οι προσεγγίσεις χωρίζονται σε δύο μεγάλες κατηγορίες: σε αυτές που ο πράκτορας χρησιμοποιεί ενημερώσεις από ενέργειες οι οποίες υπαγορεύονται από την πολιτική που ακολουθεί *(on-policy)* και σε αυτές που χρησιμοποιεί ενημερώσεις από ενέργειες που υπαγορεύονται από μια άλλη πολιτική *(off-policy)*.

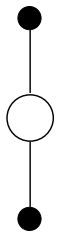
Αλγόριθμος Sarsa

Στο συγκεκριμένο αλγόριθμο όλες οι ενημερώσεις στηρίζονται στην πολιτική του πράκτορα για την επιλογή ενεργειών και μόνο σε αυτή. Έτσι, κατά τη διάρκεια ενός επεισοδίου χρησιμοποιούνται ενημερώσεις της μορφής:

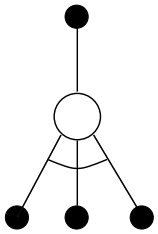
$$Q(s_t, a_t) \leftarrow Q(s_t, a_t) + \alpha \cdot \Big[r_{t+1} + \gamma \cdot Q(s_{t+1}, a_{t+1}) - Q(s_t, a_t)\Big], \qquad (3.9)$$

όπου s_{t+1} η παρατηρούμενη επόμενη κατάσταση του πράκτορα μέσα στο περιβάλλον και οι ενέργειες a_t, a_{t+1} εκλέγονται στις αντίστοιχες θέσεις βάσει της πολιτικής π που ακολουθεί ο πράκτορας (π.χ. ϵ-Άπληστη). Το γεγονός ότι εμφανίζεται η πλειάδα των μεταβλητών $\langle s_t, a_t, r_{t+1}, s_{t+1}, a_{t+1}\rangle$ μέσα στην ενημέρωση που πραγματοποιεί ο αλγόριθμος και οι οποίες εξαρτώνται αποκλειστικά από την πολιτική που ακολουθεί ο πράκτορας έχει οδηγήσει στην ονομασία του συγκεκριμένου αλγορίθμου. Το διάγραμμα ενημέρωσης που αντιστοιχεί στον συγκεκριμένο αλγόριθμο φαίνεται στο σχήμα 3.5.

Οι ιδιότητες σύγκλισης του αλγορίθμου εξαρτώνται από τη φύση της εξάρτησης της πολιτικής που ακολουθείται με τη συνάρτηση αποτίμησης Q. Ο Sarsa συγκλίνει με πιθανότητα 1 σε μια βέλτιστη πολιτική και συνάρτηση αποτίμησης Q αρκεί όλα τα ζευγάρια καταστάσεων-ενεργειών να επισκεφθούν άπειρες φορές και η πολιτική να συγκλίνει στο όριο στην Άπληστη πολιτική. Κάτι τέτοιο είναι εύκολο να γίνει αν κάποιος να ακολουθεί ϵ-Άπληστη πολιτική



Σχήμα 3.5: Παράδειγμα Διαγράμματος Ενημέρωσης για τον αλγόριθμο Sarsa.



Σχήμα 3.6: Παράδειγμα Διαγράμματος Ενημέρωσης για τον αλγόριθμο Q-Learning.

ή SoftMax πολιτική και σταδιακά μειώνει την πιθανότητα εξερεύνησης η οποία στο όριο πρέπει να τείνει στο μηδέν.

Αλγόριθμος Q-Learning

Στο συγκεκριμένο αλγόριθμο, ο πράκτορας ακολουθεί μια πολιτική π (π.χ. ϵ-Άπληστη) αλλά οι ενημερώσεις οι οποίες γίνονται βασίζονται εν μέρει σε αυτή την πολιτική που ακολουθεί ο πράκτορας και εν μέρει σε μια Άπληστη πολιτική. Έτσι, οι ενημερώσεις οι οποίες γίνονται κάθε χρονική στιγμή έχουν τη μορφή:

$$Q(s_t, a_t) \leftarrow Q(s_t, a_t) + \alpha \cdot \Big[r_{t+1} + \gamma \cdot \max_a Q(s_{t+1}, a) - Q(s_t, a_t)\Big]. \quad (3.10)$$

Έτσι, σε αυτή την περίπτωση, η συνάρτηση αποτίμησης Q προσεγγίζει άμεσα την Q^*, την βέλτιστη συνάρτηση αποτίμησης, ανεξάρτητα από το ποια πολιτική ακολουθείται από τον πράκτορα. Από την άλλη, η πολιτική που ακολουθεί ο πράκτορας έχει επιρροή πάνω στα ζευγάρια καταστάσεων-ενεργειών που επισκέπτεται ο πράκτορας και άρα στις ενημερώσεις που πραγματοποιούνται πάνω σε αυτά. Παρ'όλ'αυτά, το μόνο που χρειάζεται προκειμένου να εξασφαλίζεται η σύγκλιση της μεθόδου είναι όλα τα ζεύγη καταστάσεων-ενεργειών να επισκεφθούν άπειρες φορές και η παράμετρος α να ικανοποιεί τα κριτήρια σύγκλισης (2.24). Το αντίστοιχο διάγραμμα ενημέρωσης του αλγορίθμου Q-Learning φαίνεται στο σχήμα 3.6.

Από το παραπάνω διάγραμμα γίνεται φανερό, πως η διαφορά των ενημερώσεων των (3.9) και (3.10) έχει άμεσο «οπτικό» αντίκτυπο στα αντίστοιχα διαγράμματα ενημερώσεων και δείχνει ξεκάθαρα τη λογική η οποία ακολουθείται στον αλγόριθμο Q-Learning προκειμένου να βρει κανείς την βέλτιστη συνάρτηση αποτίμησης Q^*.

Διαισθητικά, οι δύο αλγόριθμοι (Sarsa και Q-Learning) αντιμετωπίζουν διαφορετικά τις καταστάσεις κατά τη διάρκεια της μάθησης. Μάλιστα, ο Sarsa λαμβάνει υπ'όψιν του το γεγονός ότι οι ενημερώσεις οι οποίες γίνονται οφείλονται *εν μέρει* και σε ενέργειες εξερεύνησης ενώ αντίθετα ο αλγόριθμος Q-Learning προσπαθεί να βρει τις βέλτιστες τιμές για τη συνάρτηση αποτίμησης ανεξάρτητα από τη συμπεριφορά που ακολουθεί ως προς την επιλογή ενεργειών. Αυτό έχει σαν αποτέλεσμα τις περισσότερες φορές οι αλγόριθμοι Sarsa να λαμβάνουν πολύ καλύτερες ανταμοιβές κατά την αλληλεπίδρασή τους με το περιβάλλον. Κι αυτό γιατί «θυμούνται» το γεγονός ότι ακόμη γίνονται κινήσεις εξερεύνησης κι έτσι μαθαίνουν να αποφεύγουν καταστάσεις που οδηγούν σε πολύ άσχημες ανταμοιβές - έστω κι αν αυτές οι καταστάσεις είναι πάνω σε βέλτιστες διαδρομές - σε αντίθεση με τους αλγορίθμους Q-Learning οι οποίοι ακολουθούν τη βέλτιστη διαδρομή χωρίς να τους ενδιαφέρει τίποτε άλλο.

3.3.2 Χαρακτηριστικά μεθόδων Χρονικών Διαφορών

Εκτός από το γεγονός ότι οι μέθοδοι Χρονικών Διαφορών βασίζουν τις εκτιμήσεις τους πάνω σε άλλες εκτιμήσεις καθώς επίσης και το γεγονός ότι δεν απαιτείται η γνώση της δυναμικής του περιβάλλοντος από τον πράκτορα, υπάρχει ένα πολύ σημαντικό χαρακτηριστικό το οποίο υποβόσκει σε όλη την προηγούμενη ανάλυση. Οι αλγόριθμοι μάθησης Χρονικών Διαφορών μαθαίνουν *διαρκώς* κατά την εξέλιξη των επεισοδίων.

Αυτό οφείλεται στο γεγονός πως οι ενημερώσεις λαμβάνουν χώρα μετά από *κάθε ενέργεια* του πράκτορα. Έτσι, ο πράκτορας μπορεί πιο εύκολα να καταλάβει ότι κάποια πολιτική δεν είναι καλή. Στο παράδειγμα που είχε δοθεί νωρίτερα με το ρομπότ που προσπαθούσε να βγει από έναν λαβύρινθο είπαμε ότι αν του δίνουμε μια αρνητική ανταμοιβή κάθε χρονική στιγμή τότε αυτό θα μάθει να βγαίνει από το λαβύρινθο όσο το δυνατόν πιο γρήγορα. Εφαρμόζοντας μεθόδους Monte Carlo αυτό θα είναι σωστό αν εξασφαλίζεται το γεγονός ότι η πολιτική που ακολουθεί το ρομπότ είναι τέτοια που να του εξασφαλίζει τελικά την έξοδο από το λαβύρινθο. Από την άλλη, οι μέθοδοι Χρονικών Διαφορών δεν χρειάζεται να περιμένουν μέχρι το τέλος του επεισοδίου προκειμένου ο πράκτορας να καταλάβει ότι δεν έχει νόημα να κάνει παλινδρομικές κινήσεις μέσα στο λαβύρινθο. Καθώς ο πράκτορας κάνει κινήσεις παλινδρομικές δέχεται άσχημες ανταμοιβές (γιατί είναι ακόμη μέσα στον λαβύρινθο) οπότε κρίνει «άμεσα» ότι τέτοιες ενέργειες έχουν πολύ φτωχά αποτελέσματα (ανταμοιβές) κι άρα δοκιμάζει την τύχη του σε νέες ενέργειες. Αυτό βέβαια έχει σαν αποτέλεσμα να βγαίνει πολύ γρήγορα ένας πράκτορας που μαθαίνει με χρήση Χρονικών Διαφορών από το πρώτο κιόλας επεισόδιο και διαισθητικά είναι πιο κοντά στον

τρόπο σκέψης που υιοθετούμε εμείς οι άνθρωποι.

Επίσης, αυτή η *διαρκής* μάθηση η οποία λαμβάνει χώρα στις μεθόδους Χρονικών Διαφορών, δίνει τη δυνατότητα σε ένα πράκτορα να αντιμετωπίζει *συνεχιζόμενες* εργασίες κι όχι κατ'ανάγκη *επεισοδιακές* όπως οι μέθοδοι Monte Carlo. Παράδειγμα τέτοιας μεθόδου είναι ο αλγόριθμος R-Learning ο οποίος αντιμετωπίζει προβλήματα ενισχυτικής μάθησης στα οποία δεν υπάρχει η έννοια της έκπτωσης αλλά ούτε και η εμπειρία του πράκτορα μπορεί να διαχωριστεί σε διακριτά επεισόδια. Ο ενδιαφερόμενος αναγνώστης μπορεί να βρει μια διαισθητική εισαγωγή στον συγκεκριμένο αλγόριθμο στο [26] καθώς επίσης και μια πλήρη ανάλυση του αλγορίθμου στο [11].

3.4 Επεκτάσεις

Όλες οι προηγούμενες μέθοδοι αποτελούν τη βάση των σύγχρονων μεθόδων Ενισχυτικής Μάθησης. Είναι αλγόριθμοι που έχουν εξεταστεί ευρύτατα στην πράξη και με αξιοσημείωτη επιτυχία σε πολλές εφαρμογές. Υπάρχουν όμως δύο προβλήματα:

1. Όλοι οι προηγούμενοι αλγόριθμοι «πάσχουν» από την κατάρα της διαστασιμότητας του Bellman. Αυτό σημαίνει, ότι με τα σημερινά δεδομένα υπολογιστικής ισχύος και μνήμης μπορούν να αντιμετωπίζουν προβλήματα με αρκετά εκατομμύρια καταστάσεις (Δυναμικός Προγραμματισμός) αλλά όχι παραπάνω. Για προβλήματα όπως είναι για παράδειγμα το σκάκι δεν θα μπορέσουν να βρουν εφαρμογή ποτέ, κι αυτό γιατί το πλήθος των δυνατών καταστάσεων του συγκεκριμένου παιχνιδιού ξεπερνά κατά πολύ το πλήθος των ατόμων στο σύμπαν κι επομένως δεν γίνεται να έχουμε αναπαράσταση των διαφόρων θέσεων στη μνήμη.

2. Οι αλγόριθμοι Monte Carlo και μάθησης Χρονικών Διαφορών μπορεί μεν να μαθαίνουν μέσω των ανταμοιβών από το περιβάλλον και να βελτιώνουν την πολιτική που ακολουθούν διαρκώς - σε αντίθεση με τους αλγορίθμους Δυναμικού Προγραμματισμού οι οποίοι μαθαίνουν μόνο στο τέλος της διαδικασίας μάθησης - αλλά παρ'όλ'αυτά, ο ρυθμός μάθησης είναι αρκετά αργός.

Τα παραπάνω προβλήματα έχουν οδηγήσει στη δημιουργία μεθόδων οι οποίες αφενός έχουν πολύ καλύτερους ρυθμούς μάθησης και αφετέρου προσπαθούν να καταπολεμήσουν την κατάρα της διαστασιμότητας του Bellman. Οι ιδέες όμως όλων αυτών των μεθόδων πηγάζουν από τις μεθόδους που έχουν παρουσιαστεί μέχρι τώρα.

3.4.1 TD(λ) Μέθοδοι

Οι μέθοδοι αυτής της κατηγορίας αποτελούν επέκταση των μεθόδων μάθησης Χρονικών Διαφορών οι οποίες παρουσιάστηκαν πιο πάνω. Η γενική ιδέα

των μεθόδων περιστρέφεται γύρω από την κατανομή επιρροής της εκάστοτε ανταμοιβής που δέχεται ο πράκτορας. Έτσι, οι μέθοδοι αυτής της κατηγορίας κατανέμουν την ανταμοιβή σε περισσότερες από μια ενέργειες τις οποίες έχει πραγματοποιήσει ο πράκτορας. Διαισθητικά, οι μέθοδοι αυτοί στηρίζονται στην ιδέα πως η ανταμοιβή που δέχεται κάποια στιγμή ο πράκτορας από το περιβάλλον δεν οφείλεται μόνο στην τελευταία ενέργεια του πράκτορα αλλά σε ολόκληρη την ακολουθία ενεργειών του πράκτορα.

Έτσι, κάθε χρονική στιγμή που ο πράκτορας λαμβάνει μια ανταμοιβή από το περιβάλλον, δεν πραγματοποιείται μονάχα μια ενημέρωση αλλά ένα σύνολο ενημερώσεων - μια ενημέρωση για κάθε ζεύγος $\langle s, a \rangle$ το οποίο έχει ακολουθήσει στο παρελθόν ο πράκτορας. Ένα κρίσιμο ζήτημα επομένως είναι η *βαρύτητα* η οποία θα δοθεί σε κάθε ενέργεια από την εκάστοτε ανταμοιβή την οποία λαμβάνει ο πράκτορας. Το πρόβλημα αυτό της βαρύτητας που πρέπει να δοθεί σε κάποια ενέργεια λύνεται με τη χρήση μιας παραμέτρου που ονομάζεται *ίχνος*. Η παράμετρος αυτή φθίνει εκθετικά με το πέρασμα του χρόνου με έναν ρυθμό λ απ'όπου πήρε και το όνομα η συγκεκριμένη μέθοδος. Μάλιστα, αποδεικνύεται ότι η μέθοδος Χρονικών Διαφορών που περιγράφηκε στην προηγούμενη ενότητα είναι μια ειδική κατηγορία TD(λ) μεθόδου, για $\lambda = 0$ γι'αυτό και πολλές φορές αναφέρεται στη βιβλιογραφία ως TD(0).

Ο ενδιαφερόμενος αναγνώστης μπορεί να βρει μια πολύ καλή εισαγωγή στο συγκεκριμένο ζήτημα στα [24], [26], [7], [6] και [4]. Τέλος, μεγάλο ενδιαφέρον έχει η επέκταση του Q-Learning στις TD(λ) μεθόδους και η πρόταση του Peng για αντίστοιχη μεταφορά των ιδεών η οποία μπορεί να βρεθεί στο [23].

3.4.2 Rollout Μέθοδοι

Αυτή η κατηγορία μεθόδων προσφέρει μια εναλλακτική προσέγγιση στο θέμα επιλογής ενεργειών του πράκτορα. Οι μέθοδοι αυτοί απαιτούν γνώση του περιβάλλοντος και εκμεταλλεύονται λίγο διαφορετικά την ιδέα της συνάρτησης Q. Μιας και η συνάρτηση Q αποτελεί στην ουσία μια πρόβλεψη για την προσδοκώμενη επιστροφή του πράκτορα από μια δεδομένη κατάσταση s στην οποία εφαρμόζει μια ενέργεια a, τότε είναι δυνατόν αυτή την επιστροφή να την προσεγγίσουμε με το μέσο όρο επιστροφών μιας πληθώρας επεισοδίων τα οποία ξεκινούν από την κατάσταση η οποία προκύπτει μετά από εφαρμογή της ενέργειας a στην κατάσταση s.

Έτσι, κατά τη διάρκεια μάθησης, κάθε φορά που ο πράκτορας επισκέπτεται μια κατάσταση αναγνωρίζει τις διαθέσιμες κινήσεις του. Για κάθε μια από αυτές παίρνει μια προσεγγιστική τιμή για την προσδοκώμενη επιστροφή που θα έχει αν ακολουθήσει κάθε μια από αυτές μέσω *προσομοιωμένων επεισοδίων (simulated trajectories)*. Τελικά, επιλέγει να ακολουθήσει εκείνη την ενέργεια για την οποία η προσδοκώμενη επιστροφή είναι μέγιστη ή αν υπάρχουν περισσότερες από μια τέτοιες ενέργειες κάποια στην τύχη.

Ένα σοβαρό μειονέκτημα της συγκεκριμένης μεθόδου είναι το ακριβό υπολογιστικό τίμημα το οποίο πληρώνει ο πράκτορας προκειμένου να επιλέξει

κάποια ενέργεια. Από την άλλη βέβαια, το υπολογιστικό αυτό κόστος αντισταθμίζεται από πολύ καλύτερη συμπεριφορά του πράκτορα μέσα στο περιβάλλον (λαμβάνει πολύ καλύτερες ανταμοιβές). Να σημειωθεί επίσης ότι για ντετερμινιστικά περιβάλλοντα είναι απαραίτητη η εφαρμογή ενός μόνο προσομοιωμένου επεισοδίου για κάθε διαθέσιμη ενέργεια μιας και δεν υπάρχει η έννοια του θορύβου στις ανταμοιβές. Το γεγονός αυτό έχει επιτρέψει στη συγκεκριμένη μέθοδο να έχει ένα πλήθος καλών αποτελεσμάτων σε εφαρμογές σε τέτοιους χώρους, όπως είναι για παράδειγμα τα παιχνίδια.

Ο ενδιαφερόμενος αναγνώστης μπορεί να βρει περισσότερες πληροφορίες για τους συγκεκριμένους αλγορίθμους στα [8], [7] και [5].

3.4.3 Γενίκευση και Προσέγγιση Συναρτήσεων

Για όλες τις μεθόδους που έχουν περιγραφεί μέχρι τώρα έχουμε υποθέσει ότι ο χώρος καταστάσεων και ενεργειών είναι αρκετά μικρός ώστε να χωράει ολόκληρος στη μνήμη του υπολογιστή. Η συγκεκριμένη κατηγορία μεθόδων έρχεται να αντιμετωπίσει αυτό το πρόβλημα. Έτσι, η κύρια ιδέα η οποία υπάρχει στις μεθόδους αυτής της κατηγορίας είναι αυτή της *γενίκευσης (generalization)*. Δηλαδή, με τις μεθόδους αυτές γίνεται μια προσπάθεια *γενίκευσης της εμπειρίας* που έχει ένας πράκτορας από ένα υποσύνολο των πιθανών καταστάσεων της εργασίας που αντιμετωπίζει σε ένα σύνολο πολύ ευρύτερο.

Η γενίκευση αυτή συχνά καλείται *προσέγγιση συνάρτησης (function approximation)* επειδή χρησιμοποιούνται παραδείγματα από μια επιθυμητή συνάρτηση (συνάρτηση αποτίμησης) και γίνεται προσπάθεια γενίκευσης αυτών των παραδειγμάτων προκειμένου να κατασκευαστεί μια συνάρτηση η οποία προσεγγίζει καλά τη ζητούμενη συνάρτηση. Η γενίκευση πραγματοποιείται με τη βοήθεια ενός διανύσματος παραμέτρων $\{\overrightarrow{\theta}_t\}$. Το διάνυσμα αυτό περιέχει σε κάθε θέση του αριθμητικές τιμές *(χαρακτηριστικά)* οι οποίες εξαρτώνται από την κατάσταση στην οποία αναφερόμαστε κάθε στιγμή. Έτσι, μέσω των N χαρακτηριστικών του διανύσματος $\overrightarrow{\theta}$ προσπαθούμε να εξαρτήσουμε τη συνάρτηση αποτίμησης η οποία στην πραγματικότητα έχει $|S|$ μεταβλητές (αν πρόκειται για τη συνάρτηση V) ή $\sum_{s \in S} A(s)$ μεταβλητές (αν πρόκειται για τη συνάρτηση Q). Βέβαια, κάτι τέτοιο είναι αρκετά δύσκολο μιας και το πλήθος των καταστάσεων είναι πολύ μεγαλύτερο από το πλήθος των χαρακτηριστικών. Παρ'όλ'αυτά, τα αποτελέσματα στον συγκεκριμένο τομέα είναι πολλές φορές εντυπωσιακά.

Συνήθως η διαχείριση των χαρακτηριστικών γίνεται με τη βοήθεια ενός νευρωνικού δικτύου. Όμως, η διαδικασίες μάθησης δεν αλλάζουν. Έτσι, όλοι οι προηγούμενοι αλγόριθμοι μάθησης μπορούν να μεταφερθούν εύκολα κάτω από το πρίσμα της γενίκευσης και να επηρεάζουν το διάνυσμα $\overrightarrow{\theta}$ και μέσω αυτού τις τιμές της συνάρτησης V ή Q. Επομένως, μπορεί κανείς μέσω απλών παραδειγμάτων να δει ποιοι αλγόριθμοι έχουν καλά αποτελέσματα σε μικρά προβλήματα και στη συνέχεια να αποφασίσει και να εφαρμόσει τους αλγορίθμους αυτούς στο κύριο πρόβλημα που τον ενδιαφέρει και το οποίο πρέπει να αντιμετωπίσει

με προσέγγιση συνάρτησης.

Μια διαισθητική εισαγωγή στη συγκεκριμένη κατηγορία αλγορίθμων μπορεί να βρει κανείς στο [26], ενώ μια αυστηρά μαθηματική - και πιο εκτενή - περιγραφή μπορεί να βρεθεί στο [7]. Επίσης, ένα ενδιαφέρον άρθρο σχετικό με γενίκευση εμπειρίας είναι το [25] και τέλος ένα άρθρο ομαδοποιήσης αλγορίθμων στη συγκεκριμένη περιοχή είναι το [2].

Κεφάλαιο 4

Σχέδια και Αναζήτηση

Μέχρι τώρα έχουμε συναντήσει αλγορίθμους οι οποίοι είτε απαιτούν τη γνώση της δυναμικής του περιβάλλοντος (Δυναμικός Προγραμματισμός) προκειμένου ο πράκτορας να μάθει μια βέλτιστη πολιτική είτε όχι (μέθοδοι Monte Carlo, Χρονικών Διαφορών) γιατί ο πράκτορας μπορεί να μάθει απ'ευθείας από την αλληλεπίδρασή του με το περιβάλλον. Ένα κρίσιμο ζήτημα επομένως είναι αν μπορούμε να συνδυάσουμε αυτές τις δύο κατηγορίες αλγορίθμων και να έχουμε αλγόριθμους με τους οποίους ένας πράκτορας μαθαίνει από την αλληλεπίδρασή του με το περιβάλλον, αλλά επιπλέον μπορεί να εκμεταλλευτεί το γεγονός ότι γνωρίζει τη δυναμική του περιβάλλοντος. Στο συγκεκριμένο κεφάλαιο θα αναφερθούν οι δύο πιο γνωστοί αλγόριθμοι της συγκεκριμένης περιοχής καθώς επίσης και οι δύο προτάσεις αλγορίθμων της συγκεκριμένης πτυχιακής.

4.1 Μοντέλο και Κατάστρωση Σχεδίου

Με τον όρο *μοντέλο (model)* ενός περιβάλλοντος εννοούμε οτιδήποτε μπορεί να χρησιμοποιήσει ένας πράκτορας προκειμένου να προβλέψει την απόκριση του περιβάλλοντος στις ενέργειες τις οποίες πραγματοποιεί. Δοθέντων μιας κατάστασης και μιας ενέργειας, ένα μοντέλο παράγει μια *πρόβλεψη* για την επόμενη κατάσταση στην οποία θα βρεθεί ο πράκτορας καθώς επίσης και την ανταμοιβή η οποία θα συνοδεύει αυτή τη μετάβαση. Τα μοντέλα επομένως μπορούν να διαχωριστούν σε δύο κατηγορίες:

1. Σε αυτά τα οποία επιστρέφουν όλες τις πιθανές επόμενες καταστάσεις μαζί με την πιθανότητα εμφάνισής τους, τα οποία ονομάζονται *μοντέλα κατανομής (distribution models)* και

2. Σε αυτά τα οποία επιστρέφουν μονάχα μια επόμενη κατάσταση επιλεγμένη βάσει της πιθανότητας εμφάνισής της, τα οποία ονομάζονται *μοντέλα δειγμάτων (sample models)*.

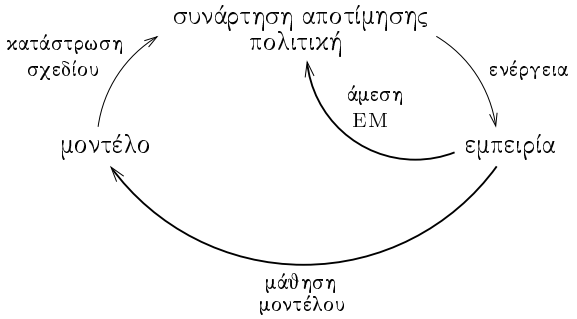
Προφανώς ένα *μοντέλο κατανομής* είναι ένα πιο ισχυρό εργαλείο για τον πράκτορα, μιας και μέσω αυτού μπορεί να προσομοιώσει τη συμπεριφορά ενός *μοντέλου δειγμάτων*. Όπως κι αν είναι όμως τα πράγματα, αν ο πράκτορας έχει στη διάθεσή του ένα μοντέλο για το περιβάλλον με το οποίο αλληλεπιδρά, μπορεί να χρησιμοποιήσει αυτό το μοντέλο για να αποκτήσει *προσομοιωμένη εμπειρία* και στη συνέχεια έχοντας γνώση από αυτή την εμπειρία να αντιμετωπίσει πολύ πιο επιτυχημένα το πραγματικό περιβάλλον με το οποίο φαίνεται να αλληλεπιδρά.

Με τον όρο *κατάστρωση σχεδίου (planning)* εννοείται στην Ενισχυτική Μάθηση μια υπολογιστική διαδικασία η οποία χρησιμοποιεί ένα μοντέλο σαν είσοδο και παράγει ή βελτιώνει μια πολιτική για αλληλεπίδραση με το μοντελοποιημένο περιβάλλον. Συγκεκριμένα, ο τρόπος με τον οποίο γίνεται αυτή η διαδικασία κατάστρωσης σχεδίου περιλαμβάνει δύο ιδέες:

- Υπολογισμό των συναρτήσεων αποτίμησης προκειμένου να γίνεται προσπάθεια βελτίωσης της πολιτικής η οποία ακολουθείται.
- Υπολογισμό των συναρτήσεων αποτίμησης με ενημερώσεις οι οποίες προέρχονται από προσομοιωμένη εμπειρία.

Σίγουρα οι μέθοδοι Δυναμικού Προγραμματισμού περιλαμβάνουν και τις δύο προηγούμενες ιδέες. Όμως δεν είναι οι μόνοι. Κάτω από αυτό το πρίσμα μπορούμε να δούμε μια πληθώρα μεθόδων οι οποίες διαφέρουν μεταξύ τους ως προς το *είδος των ενημερώσεων* που πραγματοποιούν, τη *σειρά* με την οποία πραγματοποιούν τις ενημερώσεις καθώς επίσης τη *διάρκεια* που μεσολαβεί μεταξύ δύο διαφορετικών ενημερώσεων για την ίδια κατάσταση ή ζευγάρι κατάστασης-ενέργειας.

Έχοντας αυτές τις ιδέες σαν γνώμονα μπορούμε να κατασκευάσουμε έναν πράκτορα κατάστρωσης σχεδίων με κάποια επιπλέον χαρακτηριστικά. Για έναν τέτοιο πράκτορα η πραγματική εμπειρία από το περιβάλλον έχει τουλάχιστον δύο ρόλους. Πρώτον, μέσω της εμπειρίας ο πράκτορας μπορεί να βελτιώνει άμεσα τη συνάρτηση αποτίμησης και την πολιτική χρησιμοποιώντας κάποια από τις μεθόδους που παρουσιάστηκαν νωρίτερα. Δεύτερον, η ίδια η εμπειρία μπορεί να χρησιμοποιηθεί προκειμένου ο πράκτορας να βελτιώνει το μοντέλο το οποίο χρησιμοποιεί, δηλαδή το μοντέλο να ανταποκρίνεται καλύτερα στην συμπεριφορά του πραγματικού περιβάλλοντος. Ο πρώτος ρόλος καλείται άμεση ενισχυτική μάθηση (άμεση ΕΜ - direct reinforcement learning) και ο δεύτερος μάθηση του μοντέλου (model learning). Οι αντίστοιχες εξαρτήσεις μεταξύ της εμπειρίας, του μοντέλου, της συνάρτησης αποτίμησης και της πολιτικής σε έναν πράκτορα κατάστρωσης σχεδίων φαίνονται στο σχήμα 4.1. Κάθε βέλος σε αυτό το σχήμα δείχνει μια επιρροή και υπονοεί μια βελτίωση. Έτσι, γίνεται φανερό πως η εμπειρία την οποία αποκομίζει ο πράκτορας από το περιβάλλον επηρεάζει τη συνάρτηση αποτίμησης τόσο *άμεσα* αλλά και *έμμεσα* μέσω του μοντέλου. Για το λόγο αυτό, η τελευταία αυτή επιρροή πολλές φορές καλείται και *έμμεση ενισχυτική μάθηση (indirect reinforcement learning)*. Τόσο οι



Σχήμα 4.1: Συσχετισμοί μεταξύ μάθησης, κατάστρωσης σχεδίων και ενεργειών

άμεσες όσο και οι έμμεσες μέθοδοι έχουν πλεονεκτήματα και μειονεκτήματα. Οι έμμεσες μέθοδοι συχνά κάνουν καλύτερη χρήση περιορισμένων εμπειριών και επομένως επιτυγχάνουν καλύτερη πολιτική με λιγότερες αλληλεπιδράσεις με το περιβάλλον. Από την άλλη, οι άμεσες μέθοδοι είναι πολύ απλούστερες και δεν επηρεάζονται από ενδεχομένως λανθασμένες εκτιμήσεις από το μοντέλο που χρησιμοποιεί ο πράκτορας.

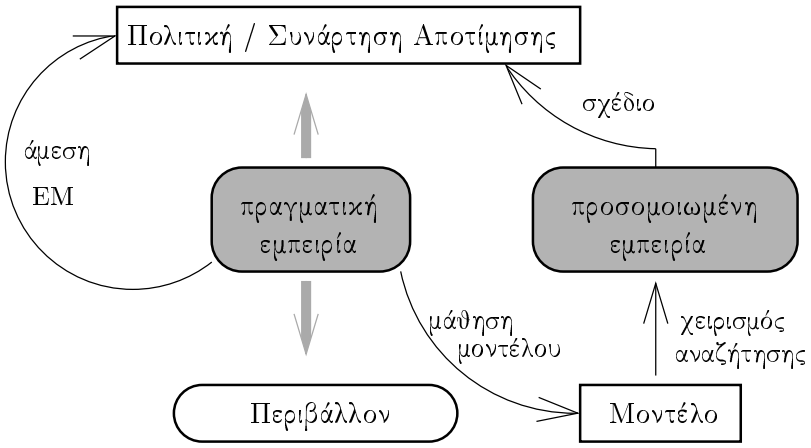
4.1.1 Οικογένεια αλγορίθμων DYNA

Η παλαιότερη οικογένεια αλγορίθμων στη συγκεκριμένη περιοχή είναι η οικογένεια των DYNA αλγορίθμων. Στους αλγόριθμους αυτούς ο πράκτορας αλληλεπιδρά με το περιβάλλον και αρχικά δεν γνωρίζει τίποτα για τη δυναμική του περιβάλλοντος. Μέσω της εμπειρίας του ο πράκτορας βελτιώνει τη συνάρτηση αποτίμησης και την πολιτική του και επίσης διορθώνει ή επεκτείνει το μοντέλο που έχει για το περιβάλλον. Ακόμη, μέσω του μοντέλου ο πράκτορας μπορεί να καταστρώνει σχέδια και να κάνει υπολογισμούς βασιζόμενος σε προσομοιωμένη εμπειρία. Το γενικό σχήμα της αρχιτεκτονικής αυτών των αλγορίθμων φαίνεται στο σχήμα 4.2.

Ο όρος *χειρισμός αναζήτησης* χρησιμοποιείται για να αναφερθούμε στη διαδικασία κατά την οποία ο πράκτορας πρέπει να διαλέξει ορισμένες αρχικές καταστάσεις και ενέργειες για την προσομοιωμένη εμπειρία η οποία θα προκύψει από το μοντέλο. Τελικά, η κατάστρωση σχεδίου επιτυγχάνεται αν διαχειριστούμε την προσομοιωμένη εμπειρία σαν πραγματική εμπειρία και εφαρμόσουμε πάνω σε αυτή άμεσες μεθόδους ενισχυτικής μάθησης. Συνήθως σε αυτούς τους αλγορίθμους η διαδικασία ενισχυτικής μάθησης είναι κοινή τόσο για μάθηση μέσω πραγματικής εμπειρίας όσο και μέσω προσομοιωμένης εμπειρίας.

Ο αλγόριθμος DYNA-Q

Η απλούστερη εκδοχή των αλγορίθμων αυτής της κατηγορίας είναι ο αλγόριθμος DYNA-Q. Τόσο η άμεση μέθοδος μάθησης, όσο και η έμμεση είναι ο αλγόριθμος Q-Learning. Έτσι, ανεξάρτητα από το αν οι μεταβάσεις συμβαίνουν



Σχήμα 4.2: Η γενική αρχιτεκτονική των αλγορίθμων DYNA.

στην πραγματικότητα (πραγματική εμπειρία) ή όχι (προσομοιωμένη εμπειρία), ο πράκτορας βασίζεται σε αυτές προκειμένου να πραγματοποιεί ενημερώσεις πάνω στη συνάρτηση αποτίμησης. Μέσω των πραγματικών μεταβάσεων ο πράκτορας μαθαίνει ή διορθώνει το μοντέλο του περιβάλλοντος. Έτσι, μετά από κάθε μετάβαση $s_t, a_t \rightarrow s_{t+1}, r_{t+1}$, ο πράκτορας καταγράφει στο μοντέλο του για το περιβάλλον πως αν στη θέση s_t εφαρμόσει την ενέργεια a_t τότε η κατάσταση s_{t+1} καθώς και η ανταμοιβή r_{t+1} ακολουθούν ντετερμινιστικά. Τέλος, ο τρόπος χειρισμού αναζήτησης ο οποίος χρησιμοποιείται διαλέγει τυχαία μεταξύ ζευγαριών $\langle s, a \rangle$ για τα οποία ο πράκτορας έχει πραγματική εμπειρία από το παρελθόν. Μάλιστα, ο τρόπος αυτός χειρισμού αναζήτησης είναι χαρακτηριστικός όλων των αλγορίθμων DYNA.

4.1.2 Αλγόριθμος Περασμάτων Προτεραιότητας

Αν και οι αλγόριθμοι DYNA τα πηγαίνουν αρκετά καλά σε ντετερμινιστικά περιβάλλοντα, εντούτοις όλες οι προσομοιωμένες μεταβάσεις διαλέγονται τυχαία από όλα τα ζευγάρια $\langle s, a \rangle$ τα οποία έχει επισκεφθεί στο παρελθόν ο πράκτορας. Παρ'όλ'αυτά, η τυχαία εκλογή συχνά δεν είναι η καλύτερη· η μάθηση μπορεί να είναι πιο αποδοτική αν οι προσομοιωμένες μεταβάσεις και ενημερώσεις επικεντρώνονται σε συγκεκριμένα ζευγάρια καταστάσεων-ενεργειών. Γύρω από αυτή την ιδέα αναπτύχθηκαν οι αλγόριθμοι περασμάτων προτεραιότητας (prioritized sweeping algorithms).

Η γενική ιδέα πηγάζει από την παρατήρηση πως η αναζήτηση χρήσιμων ενημερώσεων μπορεί να είναι ιδιαίτερα αποδοτική αν κανείς δουλεύει αντίστροφα από καταστάσεις των οποίων η αποτίμηση αλλάζει πολύ. Αν κάτι τέτοιο πραγματοποιείται, αυτό σημαίνει πως η προσδοκώμενη επιστροφή του πράκτορα μεταβάλλεται σε μεγάλο βαθμό. Επομένως, ίσως θα πρέπει να μεταβληθεί

αρκετά και η εκτίμηση που έχει ο πράκτορας για την προγενέστερη κατάσταση η οποία τον οδήγησε στη συγκεκριμένη κατάσταση της οποίας η συνάρτηση αποτίμησης μεταβλήθηκε σε μεγάλο βαθμό. Αν όμως πράγματι συμβαίνει κάτι τέτοιο, τότε ίσως θα πρέπει να επανεξετάσει ο πράκτορας και την εκτίμηση που έχει για τη γονική κατάσταση της γονικής κατάστασης της κατάστασης της οποίας μεταβλήθηκε πολύ η συνάρτηση αποτίμησης, κ.ο.κ. Έτσι, καθώς το μέτωπο αυτό ενημερώσεων εξαπλώνεται προς τα πίσω, συχνά μεγαλώνει πολύ γρήγορα. Όμως δεν είναι όλες αυτές οι ενημερώσει το ίδιο χρήσιμες. Για ορισμένες καταστάσεις ενδεχομένως η συνάρτηση αποτίμησης να αλλάζει πολύ ενώ για άλλες όχι. Συνεπώς, είναι φυσικό να μπαίνουν προτεραιότητες στις ενημερώσεις οι οποίες θα πραγματοποιηθούν σύμφωνα με ένα μέτρο που θα καθορίζει πόσο επιτακτικό είναι να γίνει άμεσα η εκάστοτε ενημέρωση και να πραγματοποιούνται σύμφωνα με αυτή την προτεραιότητα. Αυτή είναι και η ιδέα πίσω από τους αλγορίθμους *περασμάτων προτεραιότητας*.

Από πλευράς υλοποίησης διατηρείται μια ουρά προτεραιότητας με στοιχεία ζευγάρια καταστάσεων-ενεργειών των οποίων η εκτίμηση της συνάρτησης αποτίμησης θα άλλαζε περισσότερο από μια μικρή σταθερά ϑ αν πραγματοποιούταν η ενημέρωση. Έτσι, λαμβάνεται κάθε φορά η κεφαλή από την ουρά προτεραιότητας, πραγματοποιείται η ενημέρωση που αντιστοιχεί στο συγκεκριμένο ζευγάρι $\langle s, a\rangle$ και υπολογίζεται η επιρροή της νέας τιμής της εκτίμησης του ζευγαριού $\langle s, a\rangle$ στους προγόνους του. Αν αυτή ξεπερνάει την σταθερά ϑ, τότε αυτό το ζευγάρι $\langle s', a'\rangle$ του προγόνου μπαίνει στην ουρά με προτεραιότητα αυτή που του αντιστοιχεί βάσει της επικείμενης ενημέρωσης σε αυτό. Έτσι, ακόμα κι αν το περιβάλλον δεν είναι στατικό, οι επιρροές των όποιων αλλαγών πραγματοποιούνται διαδίδονται αποδοτικά προς τις περιοχές που επηρεάζουν.

4.2 Συνδυαστική Αναζήτηση

Όπως έχει γίνει φανερό μέχρι τώρα ένα πολύ ενδιαφέρον ζήτημα είναι η σειρά με την οποία πραγματοποιούνται οι ενημερώσεις. Στην κλασική προσέγγιση από τον Δυναμικό Προγραμματισμό πραγματοποιούνται περάσματα σε ολόκληρο το χώρο καταστάσεων ή ζευγαριών καταστάσεων-ενεργειών. Κάτι τέτοιο όμως είναι προβληματικό σε μεγάλες εργασίες επειδή τις περισσότερες φορές δεν υπάρχει χρόνος για την ολοκλήρωση ακόμα κι ενός περάσματος.

Από την άλλη, μπορεί κανείς να κάνει τις ενημερώσεις με όποια σειρά θέλει προκειμένου να πραγματοποιούνται οι πλέον χρήσιμες ενημερώσεις όσο το δυνατόν νωρίτερα. Κάτι τέτοιο το εκμεταλλεύονται συχνά διάφορες μέθοδοι Δυναμικού Προγραμματισμού και επιτυγχάνουν ταχεία σύγκλιση και εύρεση μιας βέλτιστης πολιτικής μέσα σε πολύ λίγα περάσματα ή ακόμη και μέσα σε ένα πέρασμα.

4.2.1 Δειγματολήπτηση Μονοπατιών

Μια πολύ χρήσιμη τεχνική είναι να κατανέμει κανείς τις ενημερώσεις ανάλογα με την πολιτική η οποία ακολουθείται. Δηλαδή η κατανομή των ενημερώσεων να ακολουθεί την κατανομή των παρατηρούμενων καταστάσεων ή ζευγαριών καταστάσεων-ενεργειών. Ένα πλεονέκτημα αυτής της κατανομής είναι το γεγονός ότι μπορεί να δημιουργηθεί πολύ εύκολα αλληλεπιδρώντας με ένα μοντέλο για το περιβάλλον ακολουθώντας την τρέχουσα πολιτική. Έτσι, σε μια επεισοδιακή εργασία μπορεί κανείς να ξεκινήσει από την αρχική θέση και να προσομοιώσει μεταβάσεις μέχρι μια τερματική θέση. Ακόμη και σε συνεχιζόμενες εργασίες μπορεί κανείς να ξεκινήσει από μια οποιαδήποτε θέση και να προσομοιώνει μεταβάσεις για όσο διάστημα επιθυμεί. Με άλλα λόγια, κάποιος μπορεί να προσομοιώνει ξεχωριστά μονοπάτια και να πραγματοποιεί ενημερώσεις πάνω σε καταστάσεις ή ζευγάρια καταστάσεων-ενεργειών τα οποία συναντάει στις διάφορες διαδρομές. Ο τρόπος αυτός γένεσης προσομοιωμένης εμπειρίας και επιλογής των αντίστοιχων ενημερώσεων καλείται *δειγματολήπτηση μονοπατιών (trajectory sampling)*.

Διαισθητικά, η δειγματολήπτηση μονοπατιών φαίνεται να είναι μια καλή επιλογή για κατάστρωση σχεδίων· τουλάχιστον καλύτερη από την ομοιόμορφη κατανομή η οποία χρησιμοποιείται στους αλγόριθμους DYNA. Επίσης η συγκεκριμένη κατανομή ενημερώσεων έχει σημαντικά πλεονεκτήματα όταν χρησιμοποιούνται τεχνικές γενίκευσης και προσέγγισης συναρτήσεων (function approximation). Μάλιστα, προς το παρόν, είναι η μοναδική τεχνική για την οποία μπορεί να εξασφαλιστεί η σύγκλιση όταν χρησιμοποιείται ένα σχήμα γραμμικών προσεγγίσεων συναρτήσεων (linear function approximation). Τέλος, ανεξάρτητα από το αν χρησιμοποιείται ή όχι προσέγγιση συναρτήσεων, η συγκεκριμένη κατανομή των ενημερώσεων μπορούμε να περιμένουμε ότι θα επιταχύνει τη διαδικασία μάθησης μιας και οι ενημερώσεις θα επικεντρώνονται σε καταστάσεις τις οποίες συναντά πιο συχνά ο πράκτορας κι έτσι δεν θα ξοδεύει χρήσιμους υπολογιστικούς πόρους για καταστάσεις άσχημες ή καταστάσεις τις οποίες δεν επισκέπτεται σύμφωνα με την τρέχουσα πολιτική. Την πεποίθηση αυτή μάλιστα την επιβεβαιώνουν όλα τα πειραματικά αποτελέσματα στα οποία είναι ξεκάθαρο ότι η δειγματολήπτηση μονοπατιών επιταχύνει σημαντικά την διαδικασία μάθησης.

4.2.2 Αναζήτηση

Όπως έχει γίνει φανερό, τόσο στα περάσματα προτεραιότητας όσο και στη δειγματολήπτηση μονοπατιών προσπαθεί κανείς να αλλάξει την σειρά με την οποία πραγματοποιούνται οι ενημερώσεις προκειμένου να επιτευχθεί ταχύτερη μάθηση. Αυτή είναι και η κεντρική ιδέα γύρω από τους δύο αλγορίθμους που προτείνουμε στη συνέχεια. Πριν γίνει όμως αυτό, θα υπενθυμίσουμε μια απλή μορφή αναζήτησης η οποία συναντάται σε όλα τα εισαγωγικά εγχειρίδια Τεχνητής Νοημοσύνης, την *αναζήτηση κατά βάθος (depth first search)*.

Αναζήτηση Κατά Βάθος

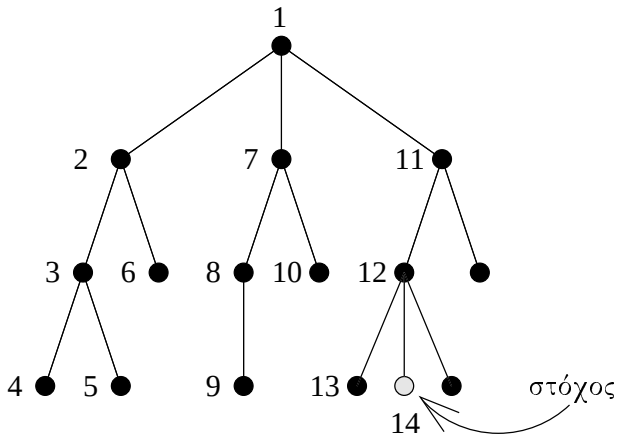
Η μέθοδος αυτή είναι μια εξαντλητική μέθοδος αναζήτησης και χρησιμοποιείται ευρύτατα σε όλα τα πεδία της τεχνητής νοημοσύνης προκειμένου να βρίσκει κανείς καταστάσεις στόχους. Ο απλούστερος τρόπος για να δούμε την κατά βάθος αναζήτηση είναι σαν να εκτείνουμε ένα δέντρο αναζήτησης, όπως για παράδειγμα αυτό του σχήματος 4.3, έτσι ώστε οι τερματικοί κόμβοι να εξετάζονται από αριστερά προς τα δεξιά. Τη σειρά εξερεύνησης των κόμβων του συγκεκριμένου δέντρου του σχήματος 4.3 μπορούμε να την παρατηρήσουμε με τους αριθμούς οι οποίοι συνοδεύουν τους αντίστοιχους κόμβους. Η επιλογή επίσκεψης συνίσταται στην ιδέα να επισκεπτόμαστε πρώτα τα παιδιά του τελευταίου εξερευνημένου κόμβου. Σε περίπτωση που κάποιος κόμβος δεν έχει παιδιά (είναι τερματικός), τότε οπισθοδρομούμε και διαλέγουμε έναν άλλο κόμβο να επισκεφθούμε. Συνήθως η διαδικασία αυτή τερματίζει όταν επισκεφθούμε έναν συγκεκριμένο κόμβο ο οποίος έχει κάποια ιδιαίτερα επιθυμητά χαρακτηριστικά. Για παράδειγμα στο σχήμα 4.3 ένας τέτοιος κόμβος είναι αυτός που επισκεπτόμαστε 14^o κατά σειρά.

Από πλευράς υλοποίησης, η αναζήτηση κατά βάθος μπορεί να υλοποιηθεί με τη βοήθεια μιας στοίβας. Έτσι, προκειμένου να επισκεφθούμε έναν κόμβο, διαλέγουμε αυτόν που βρίσκεται στην κορυφή της στοίβας και στη συνέχεια τοποθετούμε στην κορυφή της στοίβας τα παιδιά αυτού του κόμβου. Επομένως:

1. Αρχικά τοποθετούμε στη στοίβα όλους τους αρχικούς κόμβους (αρχικές καταστάσεις) του προβλήματος.

2. Σε κάθε επανάληψη διαλέγουμε να επισκεφθούμε τον κόμβο x που βρίσκεται στην κορυφή της στοίβας.

   - Αν η στοίβα είναι κενή, τότε η διαδικασία αναζήτησης αποτυγχάνει· δεν βρήκαμε δηλαδή κόμβο ο οποίος να ικανοποιεί τα κριτήρια αναζήτησης που είχαμε θέσει.
   - Διαφορετικά (η στοίβα δεν είναι κενή), επισκεπτόμαστε τον κόμβο x που διαλέξαμε από την κορυφή της στοίβας.
     - Αν ο κόμβος αυτός ικανοποιεί τα κριτήρια αναζήτησης, τότε η διαδικασία αναζήτησης τερματίζει επιτυχώς και επιστρέφει και το μονοπάτι μέσω του οποίου οδηγηθήκαμε σε αυτόν τον κόμβο στόχο.
     - Διαφορετικά, αφαιρούμε τον κόμβο x από την κορυφή της στοίβας και τοποθετούμε στη στοίβα τα παιδιά του, τοποθετώντας σε κάθε ένα από αυτά μια ετικέτα η οποία δείχνει το μονοπάτι το οποίο οδήγησε σε αυτά από τον αρχικό κόμβο.



Σχήμα 4.3: Ένα δέντρο αναζήτησης.

4.2.3 Ιδέες Προτεινόμενων Μεθόδων

Οι αλγόριθμοι οι οποίοι θα προταθούν στη συνέχεια προσπαθούν να συνδυάσουν τρεις κυρίως διαφορετικές ιδέες, με μια νέα προσέγγιση, στη συγκεκριμένη περιοχή της ενισχυτικής μάθησης:

- Κατ'αρχήν, στηρίζονται στην ιδέα της *δειγματοληπτησης μονοπατιών*.
- Η προσομοιωμένη κατανομή ζευγαριών $\langle s, a \rangle$ πάνω στα οποία γίνονται οι ενημερώσεις λόγω κατάστρωσης σχεδίου κατευθύνεται:
  - από τη μια, από το εκάστοτε μονοπάτι το οποίο ακολουθεί ο πράκτορας μέσα σε ένα επεισόδιο·
  - κι από την άλλη, από τη μοναδική ευρετική συνάρτηση την οποία διαθέτει ο πράκτορας κι αυτή είναι η συνάρτηση αποτίμησης.

  Προς αυτή την κατεύθυνση βοηθάει μια *σύνθετη διαδικασία αναζήτησης* η οποία προσπαθεί να συμπεριλάβει ιδέες από τους αλγορίθμους περασμάτων προτεραιότητας.
- Τέλος, όπως και στους αλγορίθμους περασμάτων προτεραιότητας, έτσι κι εδώ γίνεται μια προσπάθεια *διάδοσης χρήσιμης πληροφορίας* όσο γίνεται πιο κοντά στην αρχική θέση του πράκτορα.

4.3 Προτεινόμενοι Αλγόριθμοι

Προκειμένου να συνδυάσουμε τις προηγούμενες ιδέες δημιουργήσαμε αλγορίθμους οι οποίοι προσανατολίζονται στην *αναζήτηση υποδέντρων* και διάδοση των βέλτιστων προσδοκώμενων επιστροφών που υπάρχουν μέσα σε αυτά προς

την αρχική θέση του πράκτορα. Μάλιστα, η ιδέα διάδοσης προσδοκώμενης επιστροφής μέσα από υποδέντρα του χώρου αναζήτησης φάνηκε να είναι αρκετά ελπιδοφόρα. Επιπλέον, η ιδέα αυτή είναι αρκετά ευέλικτη σε επεκτάσεις με αποτέλεσμα να προκύπτουν διαφορετικές παραλλαγές αλγορίθμων. Στην υπόλοιπη ανάλυση θα εξηγήσουμε αναλυτικά τις δύο προτάσεις γενικών αλγορίθμων που προτείνουμε και θα προσπαθήσουμε να αναδείξουμε τις ιδιαιτερότητες και τα πλεονεκτήματα των διαφόρων αλγορίθμων οι οποίοι μπορούν να προκύψουν από αυτούς. Να σημειωθεί μόνο το γεγονός, πως κατά τη δημιουργία αυτών των μεθόδων είχαμε κάνει την παραδοχή πως θα αντιμετωπίσουμε ντετερμινιστικά-στατικά περιβάλλοντα - η συνηθέστερη μορφή περιβάλλοντος για παιχνίδια - στα οποία μια ενέργεια μπορεί να προκαλέσει μετάβαση της κατάστασης στην οποία είναι ο πράκτορας σε μια μόνο επόμενη κατάσταση και πάντοτε σε αυτή τη μετάβαση η «άμεση» ανταμοιβή την οποία λαμβάνει ο πράκτορας είναι μια σταθερή τιμή. Έτσι, στις όποιες περιγραφές ακολουθούν στη συνέχεια η επιλογή ενός ζευγαριού $\langle s, a\rangle$ έχει σαν αποτέλεσμα την μετάβαση του πράκτορα σε μια κατάσταση s' με πιθανότητα ίση με τη μονάδα (1).

4.3.1 Αλγόριθμος TS−CS−Q.

Ο πρώτος αλγόριθμος βρίσκεται πολύ κοντά στη φιλοσοφία των DYNA αλγορίθμων. Μάλιστα, το εννοιολογικό σχήμα το οποίο αντιπροσωπεύει την αρχιτεκτονική του είναι αυτό που εικονίζεται στο σχήμα 4.2. Η διαφορά με τους αλγορίθμους DYNA εντοπίζεται σε διαφορετική φιλοσοφία γύρω από τον τρόπο λειτουργίας του *χειρισμού αναζήτησης*. Έτσι, ο αλγόριθμος αυτός αποτελείται από δύο ξεχωριστές φάσεις:

1. Στην πρώτη φάση ο πράκτορας αλληλεπιδρά με το περιβάλλον μέχρι την ολοκλήρωση του επεισοδίου· είναι η φάση δηλαδή της *πραγματικής εμπειρίας* του πράκτορα. Κατά τη διάρκεια του επεισοδίου μαθαίνει με τον ίδιο ακριβώς τρόπο που μαθαίνει κι ο αλγόριθμος DYNA-Q. Δηλαδή πραγματοποιούνται ενημερώσεις πάνω σε ζευγάρια $\langle s, a\rangle$ ίδιες με αυτές που πραγματοποιούνται σε έναν αλγόριθμο Q-Learning.

2. Στη δεύτερη φάση ο πράκτορας πραγματοποιεί ενημερώσεις βασισμένος σε *προσομοιωμένη εμπειρία*. Ο τρόπος και το πλήθος των ενημερώσεων που πραγματοποιούνται σε κάθε ζευγάρι $\langle s, a\rangle$ βρίσκεται σε άμεση εξάρτηση με τη μέθοδο χειρισμού αναζήτησης χρήσιμων ενημερώσεων.

Στη συνέχεια θα ασχοληθούμε με την περιγραφή της φάσης της προσομοιωμένης εμπειρίας του αλγορίθμου μιας και η περιγραφή των ενημερώσεων και άλλων χαρακτηριστικών για το Q-Learning έχει γίνει στο προηγούμενο κεφάλαιο.

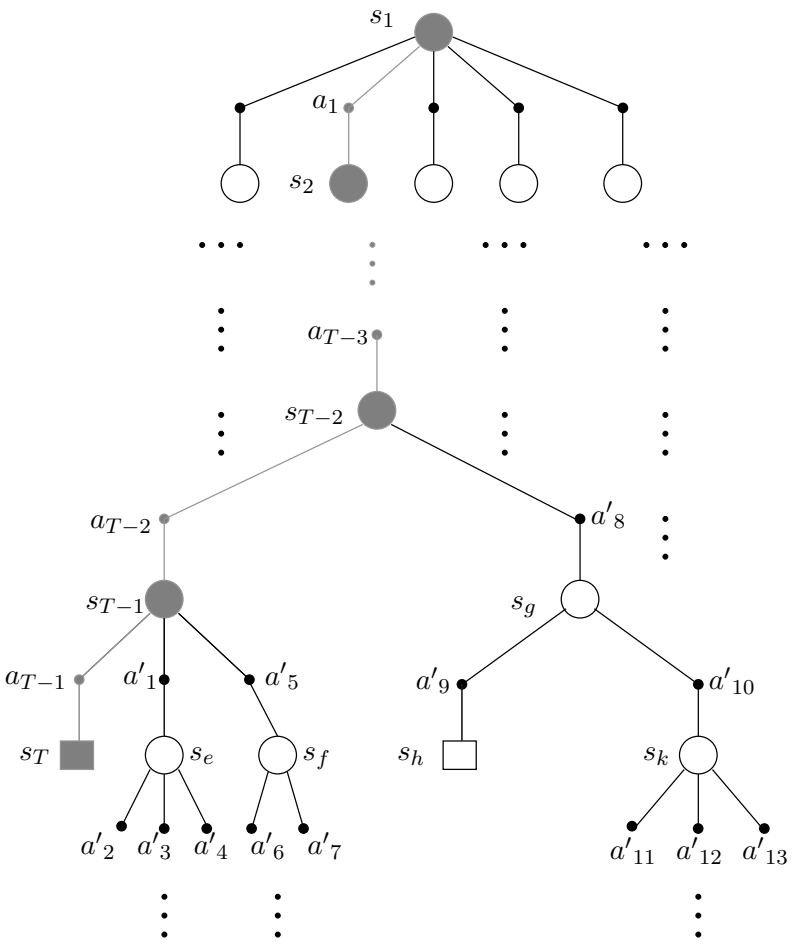
Χειρισμός Αναζήτησης

Ο πιο απλός τρόπος για να παρουσιάσουμε τη μέθοδο χειρισμού αναζήτησης που χρησιμοποιείται σε αυτόν τον αλγόριθμο είναι μέσω ενός παραδείγματος.

Έτσι, ας υποθέσουμε ότι ο πράκτορας κατά την αλληλεπίδρασή του με το περιβάλλον ακολούθησε τα ζευγάρια $\langle s_1, a_1\rangle, \langle s_2, a_2\rangle, \ldots, \langle s_{T-2}a_{T-2}\rangle, \langle s_{T-1}, a_{T-1}\rangle$ και οδηγήθηκε τελικά σε μια τερματική κατάσταση s_T, όπως φαίνεται και με την γκρι διαδρομή του σχήματος 4.4. Τότε, η ενημέρωση η οποία πραγματοποιήθηκε στο τέλος του επεισοδίου, αφορούσε το ζευγάρι $\langle s_{T-1}, a_{T-1}\rangle$, για το οποίο ο πράκτορας διόρθωσε την εκτίμησή του βάσει της ανταμοιβής r_T την οποία έλαβε από το περιβάλλον κατά την τελική του μετάβαση $s_{T-1} \rightarrow s_T$. Να σημειωθεί πως το βάθος στο οποίο βρίσκεται η κατάσταση s_T ορίζει μια μεταβλητή η οποία ονομάζεται *μέγιστο βάθος στοίβας*. Από το σημείο αυτό κι έπειτα εργάζεται η μέθοδος χειρισμού αναζήτησης η οποία εφαρμόζει ενημερώσεις Q-Learning.

Τα πρώτα ζευγάρια καταστάσεων τα οποία θα ελέγξει η μέθοδος χειρισμού αναζήτησης είναι τα $\langle s_{T-1}, a'_1\rangle$ και $\langle s_{T-1}, a'_5\rangle$. Έτσι, για το ζευγάρι $\langle s_{T-1}, a'_1\rangle$ θα πραγματοποιήσει μια ενημέρωση βάσει των εκτιμήσεων που έχει για τα ζευγάρια που προκύπτουν από την επόμενη κατάσταση - $\langle s_e, a'_2\rangle$, $\langle s_e, a'_3\rangle$ και $\langle s_e, a'_4\rangle$ καθώς επίσης και της ανταμοιβής η οποία συνοδεύει αυτή τη μετάβαση. Δηλαδή, θα διαλέξει το ζευγάρι εκείνο για το οποίο η εκτίμηση προσδοκώμενης επιστροφής είναι μέγιστη και βάσει του αθροίσματος της τιμής αυτής και της άμεσης ανταμοιβής θα ενημερώσει τη συνάρτηση αποτίμησης. Όμοια, για το ζευγάρι $\langle s_{T-1}, a'_5\rangle$ θα πραγματοποιήσει μια ενημέρωση βάσει της μέγιστης εκτίμησης των επόμενων ζευγαριών $\langle s_f, a'_6\rangle$, $\langle s_f, a'_7\rangle$. Η σημαντική ιδέα της μεθόδου χειρισμού αναζήτησης εμφανίζεται αυτή τη στιγμή. Στο σημείο αυτό, ο πράκτορας έχει ενημερώσει τη συνάρτηση αποτίμησης για κάθε πιθανή ενέργεια διαθέσιμη στην κατάσταση s_{T-1}. Επομένως, μπορεί να γνωρίζει καλύτερα ποια είναι η προσδοκώμενη επιστροφή που μπορεί να έχει αν βρεθεί ξανά στο μέλλον στην κατάσταση s_{T-1}. Συνεπώς, πραγματοποιεί μια ενημέρωση Q-Learning στο ζευγάρι $\langle s_{T-2}, a_{T-2}\rangle$ χρησιμοποιώντας τις νέες τιμές της συνάρτησης αποτίμησης για τα ζευγάρια $\langle s_{T-1}, a_{T-1}\rangle$, $\langle s_{T-1}, a'_1\rangle$ και $\langle s_{T-1}, a'_5\rangle$.

Συνεχίζοντας με τον ίδιο τρόπο, ο πράκτορας θα πραγματοποιήσει μια ενημέρωση και στο ζευγάρι $\langle s_{T-2}, a'_8\rangle$ βάσει της καλύτερης τιμής εκτίμησης μεταξύ των ζευγαριών $\langle s_g, a'_9\rangle$ και $\langle s_g, a'_{10}\rangle$. Στο σημείο αυτό, ο πράκτορας δεν θα συνεχίσει να ανεβαίνει το δέντρο, αλλά αντίθετα, θα ακολουθήσει την ενέργεια a'_8 η οποία θα τον οδηγήσει στην κατάσταση s_g. Ο λόγος για τον οποίο γίνεται αυτό είναι γιατί η ενέργεια a'_8 τον οδηγεί σε κατάσταση βάθους μικρότερου του μέγιστου βάθους στοίβας. Στην κατάσταση s_g ο πράκτορας παρατηρεί τις διαθέσιμες ενέργειες a'_9 και a'_{10} και εφαρμόζει μια ενημέρωση στα ζευγάρια που αντιστοιχούν αυτές. Όμως, δεν συνεχίζεται η κατάβαση στο δέντρο μέσω της κατάστασης s_k. Ο λόγος είναι γιατί η κατάσταση s_k - όπως και οι καταστάσεις s_e και s_f τις οποίες είχε συναντήσει νωρίτερα ο πράκτορας - βρίσκεται στο ίδιο βάθος με την κατάσταση s_T μέσω της οποίας έληξε το πραγματικό επεισόδιο. Αυτό είναι το ένα από τα δύο κύρια χαρακτηριστικά του αλγορίθμου. Δηλαδή, *ο πράκτορας προσπαθεί να κάνει ενημερώσεις όσο πιο «βαθιά» μπορεί χωρίς όμως να ξεπερνάει το μέγιστο βάθος στοίβας.*



Σχήμα 4.4: Παράδειγμα διαδρομής πράκτορα μέσα σε ένα επεισόδιο.
Με γκρι χρώμα παρατηρούμε τη διαδρομή που ακολούθησε ο πράκτορας, ενώ οι τερματικές καταστάσεις συμβολίζονται με τετράγωνα αντί για κύκλους.

Αφού λοιπόν ο πράκτορας πραγματοποιήσει τις ενημερώσεις στα ζευγάρια $\langle s_g, a'_9 \rangle$ και $\langle s_g, a'_{10} \rangle$, συνεχίζει την «επεξεργασία» ζευγαριών καταστάσεων-ενεργειών με τον ίδιο ακριβώς τρόπο. Δηλαδή, πραγματοποιεί μια νέα ενημέρωση στο ζευγάρι $\langle s_{T-2}, a'_8 \rangle$ αφού όλα τα παιδιά του έχουν ενημερωθεί. Εν συνεχεία, όλες οι ενέργειες οι οποίες είναι διαθέσιμες στην κατάσταση s_{T-2} έχουν ενημερωμένη συνάρτηση αποτίμησης, επομένως ακολουθεί ενημέρωση του ζευγαριού $\langle s_{T-3}, a_{T-3} \rangle$, κ.ο.κ.

Μέχρι εδώ έχουμε παρουσιάσει μια διαφορετική προσέγγιση από τους αλγορίθμους περασμάτων προτεραιότητας προκειμένου να διαδίδεται χρήσιμη πληροφορία τιμών της συνάρτησης αποτίμησης προς την αρχική θέση του πράκτορα. Προκειμένου να μπορούμε να επιταχύνουμε την παραπάνω διαδικασία διάδοσης πληροφορίας, χρησιμοποιείται μια επιπλέον μεταβλητή max-error. Η μεταβλητή αυτή φυλάει το μέγιστο σφάλμα στη συνάρτηση αποτίμησης του πράκτορα κατά τις ενημερώσεις οι οποίες λαμβάνουν χώρα στο μέγιστο βάθος στοίβας. Έτσι, αν η μέγιστη αυτή τιμή είναι κάτω από μια μικρή σταθερά ϑ, τότε μειώνουμε κατά μια μονάδα το μέγιστο βάθος στοίβας. Αυτό διαισθητικά σημαίνει πως ο πράκτορας φαίνεται να έχει αρκετά καλές εκτιμήσεις στο συγκεκριμένο βάθος, άρα θα ήταν καλύτερο να ανέβει πιο ψηλά στο δέντρο και να κάνει πιο χρήσιμες ενημερώσεις όπου αλλού υπάρχει ανάγκη. Ακολουθώντας το σύγχρονο ρεύμα που επικρατεί στο χώρο το οποίο θέλει αλγορίθμους σε διαδικαστική περιγραφή, παραθέτουμε τον πλήρη αλγόριθμο της συγκεκριμένης μεθόδου στην παράγραφο 4.3.4.

Παρατηρήσεις

Πριν συνεχίσουμε με την περιγραφή του επόμενου αλγορίθμου θα ήταν χρήσιμο να τονίσουμε ορισμένα λεπτά ζητήματα γύρω από τον αλγόριθμο που μόλις παρουσιάστηκε.

Στοίβα: Από την παραπάνω ανάλυση φαίνεται πως η αναζήτηση η οποία γίνεται από την πλευρά του αλγορίθμου προκειμένου να βρίσκουμε χρήσιμες ενημερώσεις ακολουθεί πιστά την λογική η οποία υπάρχει σε μια κατά βάθος αναζήτηση. Η διαφορά με τη συγκεκριμένη μέθοδο είναι ότι δεν επιτρέπεται στον αλγόριθμο TS−CS−Q να επεκτείνει υποδένδρα του χώρου αναζήτησης τα οποία βρίσκονται σε βάθος μεγαλύτερο από κάποια μέγιστη επιτρεπόμενη τιμή. Παρ'όλ'αυτά, αν θεωρήσουμε όλους τους κόμβους οι οποίοι βρίσκονται σε αυτό το βάθος σαν τερματικούς, τότε η διαδικασία αναζήτησης είναι μια κατά βάθος αναζήτηση η οποία παρουσιάστηκε στην παράγραφο 4.2.2. Επομένως, η χρησιμοποίηση μιας στοίβας στις για τη φύλαξη των διαφόρων διαδρομών που ακολουθεί ο πράκτορας είναι η πιο φυσιολογική επιλογή. Υπάρχει όμως μια ακόμη μικρή αλλά σημαντική λεπτομέρεια. Όταν ο πράκτορας προχωράει σε μεταγενέστερα επίπεδα του γράφου του χώρου αναζήτησης, τότε· όταν διαλέγει να ακολουθήσει ένα ζευγάρι $\langle s, a \rangle$, αυτό τοποθετείται ξανά στην στοίβα, προκειμένου να γίνει μια επιπλέον χρήσιμη ενημέρωση σε αυτό

όταν έχουν πραγματοποιηθεί όλες οι επιτρεπτές ενημερώσεις σε μεταγενέστερα επίπεδα.

Επιλογή: Αν παρατηρήσει κανείς προσεκτικά την διαδικαστική περιγραφή του αλγορίθμου στην παράγραφο 4.3.4 θα προσέξει πως η επιλογή ενέργειας κατά τη διαδικασία κατάβασης του γράφου αναζήτησης (εντολή 42) δεν γίνεται βάσει της πολιτικής του πράκτορα αλλά μέσω μιας συνάρτησης επιλογής ενέργειας. Αυτό γίνεται, προκειμένου ο πράκτορας να κάνει όσο το δυνατόν πιο χρήσιμες ενημερώσεις είναι δυνατόν. Προφανώς, οι πιο χρήσιμες ενημερώσεις γίνονται όταν εξερευνούμε φαινομενικά άσχημες περιοχές αφού η πιθανότητα να τις επισκεφθεί κανείς αυτές είναι πολύ μικρή. Επομένως, δεν είναι βέβαιο ότι για τις συγκεκριμένες θεωρούμενες «άσχημες» περιοχές ο πράκτορας έχει καλή εκτίμηση του αποτελέσματος. Ένας άλλος τρόπος για να δει κανείς την παραπάνω διαδικασία είναι να σκεφτεί πως αυτή κατευθύνει τον πράκτορα σε ενημερώσεις περιοχών με μεγάλο RMS σφάλμα. Έτσι κι αλλιώς, στις καλύτερες περιοχές - αυτές δηλαδή οι οποίες έχουν καλές τιμές στη συνάρτηση αποτίμησης - ο πράκτορας θα βρεθεί πιο συχνά και άρα το μεγάλο πλήθος επεισοδίων - το οποίο έτσι κι αλλιώς είναι απαραίτητο σε κάθε διαδικασία μάθησης - θα εξασφαλίζει ελαχιστοποίηση του σφάλματος πάνω σε αυτές τις περιοχές. Επομένως, αν ένας πράκτορας χρησιμοποιεί ϵ-Βέλτιστη πολιτική κατά τη διαδικασία της μάθησης έχει νόημα να τοποθετεί στη στοίβα πρώτα τα βέλτιστα ζευγάρια $\langle s, a\rangle$ κι έπειτα να τοποθετεί τυχαία, όλα τα υποβέλτιστα (βάσει της συνάρτησης αποτίμησης). Από την άλλη, αν ακολουθείται μια SoftMax πολιτική υπάρχει μεγάλο νόημα - τουλάχιστον σε πρώιμα στάδια μάθησης - να γίνεται μια κατάταξη των ζευγαριών $\langle s, a\rangle$ πριν αυτά τοποθετηθούν στη στοίβα. Έτσι, μετά την κατάταξη αυτή να τοποθετούνται στη στοίβα τα ζευγάρια με σειρά όμοια της (μερικής) διάταξης που ορίζει πάνω σε αυτά η τρέχουσα συνάρτηση αποτίμησης - δηλαδή πρώτα τα ζευγάρια με μέγιστες προσδοκώμενες επιστροφές, μετά αυτά με μικρότερες, στη συνέχεια αυτά με πιο μικρές, κ.ο.κ.

Ενημερώσεις: Μια κρίσιμη επιλογή του σχεδιαστή του αλγορίθμου βρίσκεται στη συνθήκη η οποία υπάρχει στη γραμμή 28. Σε αυτή τη γραμμή, γίνεται έλεγχος για το αν πρέπει να μειωθεί το μέγιστο βάθος στο οποίο επιτρέπεται να πραγματοποιεί ενημερώσεις ένας πράκτορας. Έτσι όπως είναι γραμμένος ο αλγόριθμος στην παράγραφο 4.3.4 ο πράκτορας έχει κύριο μέλημα τη διάδοση πληροφορίας προς την κορυφή. Αν πραγματοποιείται κάτι τέτοιο, τότε σε μεταγενέστερα στάδια μάθησης ενδέχεται ο πράκτορας να μην χρησιμοποιεί όλο το χρονικό διάστημα το οποίο του διατίθεται για να κάνει MAX-PLANNING-BACKUPS ενημερώσεις αφού θα θεωρεί ότι δεν υπάρχει κάποιος ιδιαίτερος λόγος για να πραγματοποιήσει κάτι τέτοιο. Από την άλλη, μια λύση σε αυτό είναι η προσθήκη (με σύζευξη)

της λογικής συνθήκης στη γραμμή 28:

$$\Big(\text{Πλήθος(stack)} \geq (\text{MAX-PLANNING-BACKUPS} - \text{stack-backups})\Big)$$

έτσι ώστε ο πράκτορας να αναγκάζεται να κάνει «χαμηλά» αναζητήσεις και μόνο όταν εξασφαλίζεται ότι δεν θα «χάσει» ενημερώσεις (μιας και θα υπάρχουν αρκετά ζευγάρια $\langle s, a\rangle$ στη στοίβα) να προχωράει σε διάδοση πληροφορίας. Από την άλλη όμως, με μια τέτοια επιπλέον συνθήκη ένας πράκτορας ενδέχεται να μην καταφέρει να διαδώσει πιο ψηλά χρήσιμη πληροφορία σε περίπτωση που το περιβάλλον δίνει πολλές δυνατότητες σε έναν πράκτορα (πολύ μεγάλο παράγοντα διακλάδωσης). Τελικά, κάτι πιο συνετό φαίνεται να είναι είτε μια βαθμιαία εξασθένιση της παραμέτρου ϑ με το πέρασμα του χρόνου όπως ακριβώς γίνεται και με την πιθανότητα ϵ εξερεύνησης, ή, αν ο παράγοντας διακλάδωσης είναι πολύ μεγάλος να πραγματοποιούνται οι ελάχιστες σε πλήθος απαιτούμενες ενημερώσεις προκειμένου ο πράκτορας να συνεχίσει τη διάδοση τιμών πιο ψηλά χωρίς όμως να «χάνεται» ούτε μια ενημέρωση.

4.3.2 Αλγόριθμος TS−CS−MC.

Ο αλγόριθμος αυτός αποτελείται από τρεις φάσεις σε αντίθεση με τον προηγούμενο. Στην πρώτη φάση ο πράκτορας αλληλεπιδρά με το περιβάλλον δημιουργώντας ένα επεισόδιο, στη δεύτερη φάση πραγματοποιούνται χρήσιμες ενημερώσεις σε ένα υποδένδρο του χώρου αναζήτησης και τέλος στην τρίτη φάση διαδίδεται η καλύτερη τιμή από το υποδένδρο το οποίο εξερευνήθηκε μέχρι την κορυφή. Η ιδέα διάδοσης τιμών αποτίμησης προσεγγίζεται από μια διαφορετική οπτική γωνιά απ'όλους τους αλγορίθμους οι οποίοι χρησιμοποιούν μοντέλο κι έχουν εφαρμοστεί μέχρι σήμερα. Πριν προχωρήσουμε όμως περισσότερο στην περιγραφή του αλγορίθμου, να τονίσουμε τα ακόλουθα:

- Η αλληλεπίδραση του πράκτορα με το περιβάλλον δεν έχει κύριο στόχο την άμεση μάθηση μέσω αλληλεπίδρασης, αλλά την προσήλωση του πράκτορα σε έναν συγκεκριμένο χώρο, στον οποίο θα μάθει μέσω προσομοιωμένων αλληλεπιδράσεων από το μοντέλο που χρησιμοποιεί για το περιβάλλον.

- Όλες οι ενημερώσεις οι οποίες πραγματοποιούνται είναι ενημερώσεις Q-Learning. Όμως, οι ενημερώσεις αυτές πραγματοποιούνται σε μια από τις παρακάτω τρεις περιπτώσεις:

  1. Αν ο πράκτορας επισκέπτεται τερματική κατάσταση, είτε μέσω πραγματικής εμπειρίας είτε μέσω προσομοιωμένης εμπειρίας.
  2. Αν ο πράκτορας κατά τη διαδικασία αναζήτησης χρήσιμων ενημερώσεων δεν επιτρέπεται να επεκτείνει το δέντρο του χώρου σε μεγαλύτερο βάθος· όπως ακριβώς γινόταν και με τον TS−CS−Q αλγόριθμο.

3. Τέλος, αν έχουν πραγματοποιηθεί ενημερώσεις σε όλους τους επόμενους (σε βάθος) κόμβους κατά τη διαδικασία αναζήτησης, τότε πραγματοποιείται μια ενημέρωση και στον υπό εξέταση κόμβο.

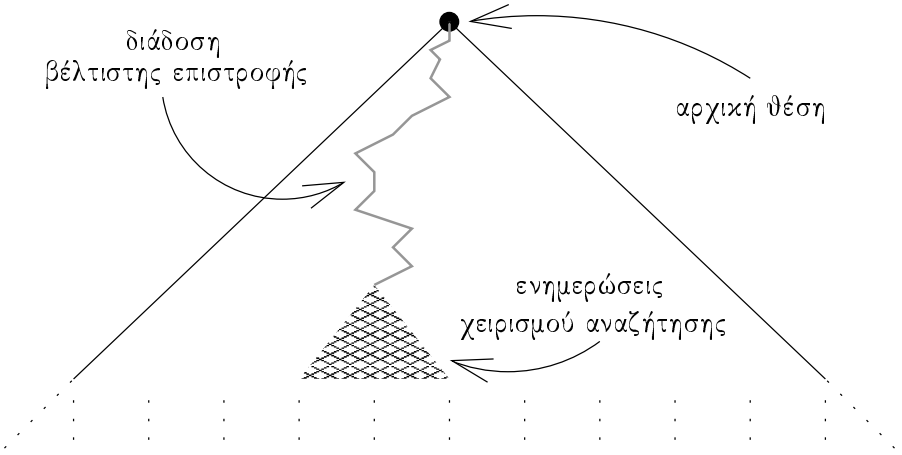
- Όταν ολοκληρωθεί η διαδικασία αναζήτησης και πραγματοποίησης χρήσιμων ενημερώσεων μέσα σε ένα υποδένδρο του χώρου αναζήτησης, τότε ο πράκτορας διαδίδει προς την αρχική θέση την καλύτερη επιστροφή η οποία έχει παρατηρηθεί είτε μέσω πραγματικής εμπειρίας με το περιβάλλον, είτε μέσω προσομοιωμένης εμπειρίας.

Χειρισμός Αναζήτησης

Ο χειρισμός αναζήτησης είναι ταυτόσημος με αυτόν του προηγούμενου αλγορίθμου. Έτσι, αν σκεφτούμε έναν πράκτορα ο οποίος κατά την πραγματική του αλληλεπίδραση με το περιβάλλον ακολούθησε την διαδρομή $\langle s_1, a_1\rangle$, $\langle s_2, a_2\rangle$, ..., $\langle s_{T-2}a_{T-2}\rangle$, $\langle s_{T-1}, a_{T-1}\rangle$ και οδηγήθηκε τελικά στην τερματική κατάσταση s_T, όπως φαίνεται και με την γκρι διαδρομή του σχήματος 4.4, τότε ο χειρισμός αναζήτησης θα επιχειρήσει την ανεύρεση χρήσιμων ενημερώσεων μέσα από την ίδια διαδικασία η οποία ακολουθήθηκε και στον TS−CS−Q αλγόριθμο. Η διαφορά είναι ότι ο πράκτορας δεν πραγματοποιεί ενημερώσεις όταν κατεβαίνει το δένδρο αναζήτησης παρά μόνο εάν ένα ζευγάρι $\langle s, a\rangle$ τον οδηγήσει σε τερματική κατάσταση ή τον οδηγεί σε κατάσταση μέγιστου επιτρεπόμενου βάθους. Από την άλλη, όταν κατά τη διαδικασία αναζήτησης ο πράκτορας επισκέπτεται ένα ζευγάρι $\langle s, a\rangle$ του οποίου όλα τα επόμενα ζευγάρια $\langle s', a'\rangle$ μόλις ενημερώθηκαν, τότε πραγματοποιεί μια ενημέρωση και στο ζευγάρι $\langle s, a\rangle$. Με άλλα λόγια, ενημερώσεις πραγματοποιούνται μέσω του χειρισμού αναζήτησης μόνο όταν ο πράκτορας επισκέπτεται ζευγάρια $\langle s, a\rangle$ καθώς ανεβαίνει το δένδρο του χώρου αναζήτησης.

Διάδοση Βέλτιστων Εμπειρικών Τιμών

Μετά την ολοκλήρωση των ενημερώσεων οι οποίες πραγματοποιούνται μέσω του χειρισμού αναζήτησης, ο πράκτορας διαδίδει σε κάθε ανώτερο επίπεδο τη βέλτιστη επιστροφή η οποία υπολογίστηκε για το συγκεκριμένο υποδένδρο το οποίο εξερεύνησε νωρίτερα. Έτσι, σε κάθε ζευγάρι $\langle s, a\rangle$ το οποίο βρίσκεται σε ανώτερο επίπεδο και το οποίο βρίσκεται πάνω στη διαδρομή η οποία οδήγησε τον πράκτορα σε ενημερώσεις στο συγκεκριμένο υποδένδρο (μέσω της διαδικασίας χειρισμού αναζήτησης), πραγματοποιείται μια ενημέρωση με στόχο την βέλτιστη προσδοκώμενη «εμπειρική» επιστροφή του πράκτορα. Η βέλτιστη προσδοκώμενη εμπειρική επιστροφή που έχει ένας πράκτορας για ένα ζευγάρι $\langle s, a\rangle$ είναι η μέγιστη τιμή επιστροφής την οποία έχει λάβει ακολουθώντας το ζευγάρι $\langle s, a\rangle$ είτε μέσω πραγματικής εμπειρίας είτε μέσω προσομοιωμένης εμπειρίας. Οι ενημερώσεις οι οποίες οφείλονται σε αυτή τη διαδικασία διάδοσης καθώς επίσης και η σχέση τους με τις ενημερώσεις οι οποίες οφείλονται στο χειρισμό αναζήτησης φαίνονται παραστατικά στο σχήμα 4.5.



Σχήμα 4.5: Διάδοση βέλτιστης εμπειρικής επιστροφής στον TS-CS-MC αλγόριθμο.

Παρατηρήσεις

Στη συνέχεια αναφέρουμε κάποια κρίσιμα χαρακτηριστικά σχετικά με τον αλγόριθμο ο οποίος μόλις προτάθηκε και παρουσιάζεται κι αυτός σε διαδικαστική μορφή στην παράγραφο 4.3.4.

Στοίβα: Μιας και ο μηχανισμός της διαδικασίας χειρισμού αναζήτησης δεν έχει αλλάξει σε σχέση με τον προηγούμενο αλγόριθμο, έτσι κι εδώ είναι αναγκαία η ύπαρξη μιας στοίβας. Επιπλέον, η λειτουργικότητά της είναι παρόμοια με αυτή του προηγούμενου αλγορίθμου. Η διαφορά με τον προηγούμενο αλγόριθμο είναι πως ενημερώσεις σε ζευγάρια $\langle s, a \rangle$ πραγματοποιούνται μόνο όταν εξάγονται τα ζευγάρια αυτά από τη στοίβα κι όχι κι όταν εισάγονται.

Επιλογή: Όπως και στον προηγούμενο αλγόριθμο, έτσι κι εδώ (εντολή 32), κατά τη φάση προσομοιωμένης εμπειρίας, ο πράκτορας επεκτείνει υποδέντρα του χώρου αναζήτησης βάσει μιας συνάρτησης επιλογής κι όχι βάσει της πολιτικής που ακολουθεί. Όπως και πριν, ο στόχος μέσα από αυτή τη διαδικασία είναι η ελαχιστοποιήσει του σφάλματος από άσχημες περιοχές. Επιπλέον, ισχύουν κι εδώ τα σχόλια τα οποία έγιναν και στον προηγούμενο αλγόριθμο και αναφέρονται στην όποια πολιτική ακολουθεί ο πράκτορας (ϵ-Άπληστη ή SoftMax).

Ενημερώσεις: Αντίθετα με τον προηγούμενο αλγόριθμο, ο συγκεκριμένος πραγματοποιεί προσομοιωμένες ενημερώσεις όσο το δυνατόν πιο μακριά μπορεί. Δηλαδή, δεν ανεβάζει το μέγιστο βάθος στοίβας προκειμένου να διαδώσει πληροφορία πιο ψηλά. Κάτι τέτοιο επιτυγχάνεται διαφορετικά, μιας και στο τέλος της προσομοιωμένης εμπειρίας, ο πράκτορας

επισκέπτεται τη διαδρομή την οποία ακολούθησε ώστε κατά την πραγματική του αλληλεπίδραση με το περιβάλλον και πραγματοποιεί ενημερώσεις Q-Learning *μεταξύ των διαφόρων ζευγαριών* $\langle s, a \rangle$ *για τα οποία είχε πραγματική εμπειρία στο παρελθόν*. Οι ενημερώσεις δηλαδή οι οποίες πραγματοποιούνται κατά τη φάση διάδοσης βέλτιστης επιστροφής είναι βεβιασμένες αφού δεν λαμβάνουν υπ' όψιν τους τις όποιες άλλες τιμές της συνάρτησης αποτίμησης για περιοχές στις οποίες ο πράκτορας δεν είχε ποτέ ούτε πραγματική ούτε προσομοιωμένη εμπειρία.

Εξερεύνηση: Το τελευταίο αυτό χαρακτηριστικό του αλγορίθμου καθιστά αναγκαία την ύπαρξη κάποιας πιθανότητας εξερεύνησης στην πολιτική την οποία ακολουθεί ο πράκτορας. Προφανώς, όσο δεν έχει επισκεφθεί όλες τις περιοχές ο πράκτορας, οφείλει να πραγματοποιεί εξερευνητικές ενέργειες ώστε να εξασφαλίζεται το γεγονός πως δεν θα κολλάει σε τοπικά μέγιστα της συνάρτησης αποτίμησης. Απ' το σημείο βέβαια που ο πράκτορας έχει επισκεφτεί όλες τις περιοχές, τότε προφανώς κατά τη φάση διάδοσης βέλτιστης εμπειρικής επιστροφής θα πραγματοποιούνται ενημερώσεις Q-Learning οπότε μπορούμε να εκμηδενίσουμε την πιθανότητα εξερεύνησης σε ντετερμινιστικά περιβάλλοντα, ή να πραγματοποιήσουμε βαθμιαία ελάττωση της πιθανότητας αυτής αν το περιβάλλον είναι μόνο στατικό.

4.3.3 Επεκτάσεις Προτεινόμενων Μεθόδων.

Πριν ολοκληρώσουμε την παρουσίαση των νέων μεθόδων που προτείνουμε, είναι σημαντικό να αναφέρουμε πως οι μέθοδοι αυτοί μπορούν να χρησιμοποιηθούν και σε περιβάλλοντα τα οποία δεν είναι κατ' ανάγκη ντετερμινιστικά-στατικά σύμφωνα με την παραδοχή η οποία έγινε στην αρχή της παραγράφου. Νωρίτερα, τονίσαμε τα σημαντικότερα χαρακτηριστικά των συγκεκριμένων αλγορίθμων προκειμένου ο εκάστοτε σχεδιαστής να τα λαμβάνει σοβαρά υπ' όψιν του ώστε να προσαρμόζει τις μεθόδους στο εκάστοτε πρόβλημα το οποίο αντιμετωπίζει.

Για παράδειγμα σε περιβάλλοντα όπου οι ανταμοιβές τις οποίες λαμβάνει ο πράκτορας προκύπτουν μέσα από δειγματοληπτηση κάποιων κατανομών, ή γενικά σε μη-στατικά περιβάλλοντα, θα ήταν χρήσιμο και ο αλγόριθμος $TS-CS-Q$ να περιλαμβάνει κάποια πιθανότητα εξερεύνησης. Σε τέτοιες περιπτώσεις, ή σε περιπτώσεις όπου η αποτελεσματικότητα του πράκτορα παίζει ιδιαίτερο ρόλο, ίσως θα ήταν επίσης χρήσιμο ο σχεδιαστής να πραγματοποιεί ενημερώσεις Sarsa αντί για ενημερώσεις Q-Learning.

Όπως κι αν έχουν όμως τα πράγματα, η συζήτηση σχετικά με τις επιλογές σχεδιασμού του εκάστοτε αλγόριθμου θα περιστρέφεται γύρω από τα χαρακτηριστικά τα οποία αναφέραμε στην προηγούμενη παρουσίαση των μεθόδων. Η συζήτηση όμως αυτή θα συνεχιστεί μετά την παρουσίαση κάποιων πειραματικών αποτελεσμάτων, ώστε ο αναγνώστης να μπορεί να εκτιμήσει καλύτερα

όσα αναφέρονται.

4.3.4 Οι προτεινόμενοι αλγόριθμοι σε διαδικαστική μορφή.

Αλγόριθμος TS−CS−Q

01 Αρχικοποίηση των $Q(s,a)$ για κάθε $s \in S$ και $a \in A(s)$.
02 Επανάληψη για πάντα:
03 $s \leftarrow$ Αρχική θέση
04 stack $\leftarrow$ Αρχικοποίηση στοίβας
05 depth $\leftarrow 0$
06 Επανέλαβε:
07 Λίστα $\leftarrow$ Διαθέσιμες-Ενέργειες (s)
08 $a \leftarrow$ Πολιτική (s, Q)
09 Τοποθέτηση στη στοίβα stack όλων των ζευγαριών $\langle s, a' \rangle, a \neq a' \in$ Λίστα
10 Λήψη ενέργειας a, παρατήρηση επόμενης κατάστασης s' και ανταμοιβής r
11 Αν ($s' =$ τερματική)
12 max-error $\leftarrow \alpha \cdot \Big[r + \gamma \cdot \max_{a'} Q(s', a') - Q(s,a) \Big]$
13 $Q(s,a) \leftarrow Q(s,a) + \alpha \cdot \Big[r + \gamma \cdot \max_{a'} Q(s', a') - Q(s,a) \Big]$
14 Αν ($s' \neq$ τερματική)
15 Τοποθέτηση στη στοίβα stack του ζευγαριού $\langle s, a \rangle$
16 $s \leftarrow s'$
17 depth $\leftarrow$ depth + 1
18 Όσο ($s \neq$ τερματική)
19 stack-backups $\leftarrow 0$
20 max-stack-depth $\leftarrow$ depth $- 1$
21 previous-backup-level $\leftarrow$ max-stack-depth
22 Όσο $\Big(\Big($ Πλήθος (stack) $> 0 \Big)$ && (stack-backups $<$ MAX-PLANNING-BACKUPS$\Big)$
23 $\langle s, a \rangle \leftarrow$ Εξώθηση από τη στοίβα stack
24 $s' \leftarrow$ Εφαρμογή $(\langle s, a \rangle)$
25 stack-depth $\leftarrow$ Βάθος $(s) + 1$
26 Λίστα $\leftarrow$ Διαθέσιμες-Ενέργειες (s')
27 Αν $\Big(\Big($(previous-backup-level $\leq$ stack-depth) && $\Big($ Πλήθος (stack) $= 0 \Big)$ ||
(previous-backup-level $>$ stack-depth) ||
$\Big($ (stack-depth $- 1$) = max-stack-depth$\Big) \Big)$
28 Αν $\Big(\Big($previous-backup-level $>$ (stack-depth $- 1$) $\Big)$ && (max-error $< \vartheta$)$\Big)$
29 max-stack-depth $\leftarrow$ max-stack-depth $- 1$
30 max-error $\leftarrow 0$
31 Αν ($s' =$ τερματική)
32 Σφάλμα $\leftarrow \alpha \cdot \Big[r + \gamma \cdot \max_{a'} Q(s', a') - Q(s,a) \Big]$
33 $Q(s,a) \leftarrow Q(s,a) + \alpha \cdot \Big[r + \gamma \cdot \max_{a'} Q(s', a') - Q(s,a) \Big]$
34 Αν $\Big(\Big($(stack-depth $- 1$) = max-stack-depth $\Big)$ && (Σφάλμα $>$ max-error)$\Big)$
35 max-error $\leftarrow$ Σφάλμα
36 stack-backups $\leftarrow$ stack-backups + 1
37 previous-backup-level $\leftarrow$ stack-depth - 1
38 Αλλιώς
39 $Q(s,a) \leftarrow Q(s,a) + \alpha \cdot \Big[r + \gamma \cdot \max_{a'} Q(s', a') - Q(s,a) \Big]$
40 stack-backups $\leftarrow$ stack-backups + 1
41 Τοποθέτηση στη στοίβα stack του ζευγαριού $\langle s, a \rangle$
42 $a'_2 \leftarrow$ Επιλογή (s', Q)
43 Τοποθέτηση στη στοίβα stack όλων των ζευγαριών $\langle s', a'' \rangle, a'_2 \neq a'' \in$ Λίστα
44 Τοποθέτηση στη στοίβα stack του ζευγαριού $\langle s', a'_2 \rangle$

Αλγόριθμος TS−CS−MC

Αρχικοποίηση των $Q(s,a)$ για κάθε $s \in S$ και $a \in A(s)$.
Επανάληψη για πάντα:
    $s \leftarrow$ Αρχική θέση
    stack $\leftarrow$ Αρχικοποίηση στοίβας
    depth $\leftarrow 0$
    Επανέλαβε:
        Λίστα $\leftarrow$ Διαθέσιμες-Ενέργειες (s)
        $a \leftarrow$ Πολιτική (s, Q)
        Τοποθέτηση στη στοίβα stack όλων των ζευγαριών $\langle s, a' \rangle, a \neq a' \in$ Λίστα
        Λήψη ενέργειας a, παρατήρηση επόμενης κατάστασης s' και ανταμοιβής r
        Αν $(s' =$ τερματική$)$
            $Q(s,a) \leftarrow Q(s,a) + \alpha \cdot \Big[r + \gamma \cdot \max_{a'} Q(s',a') - Q(s,a) \Big]$
        Αλλιώς
            Τοποθέτηση στη στοίβα stack του ζευγαριού $\langle s, a \rangle$
        $s \leftarrow s'$
        depth $\leftarrow$ depth + 1
    Όσο $(s \neq$ τερματική$)$
    stack-backups $\leftarrow 0$
    max-stack-depth $\leftarrow$ depth $- 1$
    previous-backup-level $\leftarrow$ max-stack-depth
    Όσο $\Big(\Big($ Πλήθος (stack) $> 0 \Big)$ && (stack-backups $<$ MAX-PLANNING-BACKUPS$) \Big)$
        $\langle s, a \rangle \leftarrow$ Εξώθηση από τη στοίβα stack
        $s' \leftarrow$ Εφαρμογή $(\langle s, a \rangle)$
        stack-depth $\leftarrow$ Βάθος $(s) + 1$
        Λίστα $\leftarrow$ Διαθέσιμες-Ενέργειες (s')
        Αν $\Big(\Big($(previous-backup-level $\leq$ stack-depth) && $\Big($Πλήθος (stack) $= 0 \Big)$ ||
            (previous-backup-level $>$ stack-depth) ||
            $\Big($ (stack-depth $- 1$) $=$ max-stack-depth$\Big) \Big)$
            $Q(s,a) \leftarrow Q(s,a) + \alpha \cdot \Big[r + \gamma \cdot \max_{a'} Q(s',a') - Q(s,a) \Big]$
            stack-backups $\leftarrow$ stack-backups + 1
            previous-backup-level $\leftarrow$ stack-depth - 1
        Αλλιώς
            Τοποθέτηση στη στοίβα stack του ζευγαριού $\langle s, a \rangle$
            $a' \leftarrow$ Επιλογή (s', Q)
            Τοποθέτηση στη στοίβα stack όλων των ζευγαριών $\langle s', a'' \rangle, a' \neq a'' \in$ Λίστα
            Τοποθέτηση στη στοίβα stack του ζευγαριού $\langle s', a' \rangle$
    Όσο $\Big($Πλήθος (stack) $> 0 \Big)$
        $\langle s, a \rangle \leftarrow$ Εξώθηση από τη στοίβα stack
        Όσο $\Big($Βάθος $(s) =$ previous-backup-level $\Big)$
            $\langle s, a \rangle \leftarrow$ Εξώθηση από τη στοίβα stack
        previous-backup-level $\leftarrow$ Βάθος (s)
        Λίστα $\leftarrow$ Διαθέσιμες-Ενέργειες-για-τις-οποίες-υπάρχει-Εμπειρία (s)
        $Q(s,a) \leftarrow Q(s,a) + \alpha \cdot \Big[r + \gamma \cdot \max_{a'} Q(s',a') - Q(s,a) \Big]$
        stack-backups $\leftarrow$ stack-backups + 1

Κεφάλαιο 5

Άλλες εργασίες στο χώρο.

Πριν προχωρήσουμε σε μια εφαρμογή των δύο προτάσεων της συγκεκριμένης πτυχιακής κρίνουμε αναγκαίο να παρουσιάσουμε τις κυριότερες ιδέες οι οποίες έχουν δημοσιευτεί μέχρι σήμερα στο συγκεκριμένο χώρο της Ενισχυτικής Μάθησης. Οι περισσότερες από αυτές έχουν να κάνουν κατά κύριο λόγο με διακριτούς - ντετερμινιστικούς χώρους αναζήτησης αν και υπάρχουν προτάσεις για προσαρμογή των αλγορίθμων αυτών είτε σε συνεχείς χώρους αναζήτησης είτε σε μη-ντετερμινιστικούς χώρους αναζήτησης. Τέλος, από το συγκεκριμένο κεφάλαιο δεν θα μπορούσε να λείπει μια σύντομη περιγραφή για το μεγαλύτερο καμάρι της Ενισχυτικής Μάθησης, ένα πρόγραμμα το οποίο παίζει τάβλι και κυριαρχεί στον παγκόσμιο χώρο τόσο μεταξύ μηχανών όσο και μεταξύ ανθρώπων.

5.1 Επεκτάσεις Δυναμικού Προγραμματισμού.

Το μεγαλύτερο πρόβλημα των μεθόδων Δυναμικού Προγραμματισμού είναι η λεγόμενη «κατάρα της διαστασιμότητας» του Bellman. Τα προβλήματα τα οποία καλείται να αντιμετωπίσει ένας πράκτορας Ενισχυτικής Μάθησης συνήθως έχουν χώρους αναζήτησης οι οποίοι αυξάνουν εκθετικά καθώς το μέγεθος του εκάστοτε προβλήματος αυξάνει γραμμικά. Αυτό έχει σαν αποτέλεσμα οι μέθοδοι Δυναμικού Προγραμματισμού να περιορίζονται σε προβλήματα με χώρους αναζήτησης οι οποίοι δεν ξεπερνούν τις μερικές δεκάδες εκατομμύρια καταστάσεις. Από την άλλη, οι μέθοδοι Δυναμικού Προγραμματισμού βρίσκουν μια βέλτιστη πολιτική για έναν πράκτορα για οποιαδήποτε πιθανή κατάσταση στο χώρο αναζήτησης. Το μειονέκτημα της συγκεκριμένης διαδικασίας είναι το γεγονός πως ενδέχεται να σπαταλούνται υπολογιστικοί πόροι για καταστάσεις τις οποίες έτσι κι αλλιώς ένας πράκτορας δεν πρόκειται να τις επισκεφθεί ποτέ κι επομένως δεν χρειάζεται να γνωρίζει κανείς μια βέλτιστη πολιτική για τις καταστάσεις αυτές.

Έτσι, ενώ για την «κατάρα της διαστασιμότητας» του Bellman δεν μπορεί να γίνει κάτι (εκτός βέβαια κι αν $P = NP$), εντούτοις έχουν γίνει κάποιες προ-

σπάθειες προκειμένου να αντιμετωπισθεί το δεύτερο μειονέκτημα των μεθόδων Δυναμικού Προγραμματισμού. Η ιδέα στην οποία στηρίζονται οι προσπάθειες αυτές, βασίζεται στο γεγονός πως για πολλά προβλήματα Διαδικασιών Απόφασης Markov ο πράκτορας έχει να βρει μια βέλτιστη πολιτική για μια συγκεκριμένη αρχική κατάσταση του προβλήματος. Επομένως υπάρχει μεγάλο νόημα να εξεταστούν μόνο οι καταστάσεις εκείνες οι οποίες μπορούν να προκύψουν από τη συγκεκριμένη αρχική κατάσταση του προβλήματος στην οποία βρίσκεται ο πράκτορας.

Οι δύο σημαντικότερες συνεισφορές προς την κατεύθυνση αυτή είναι γνωστές με τα ονόματα Δυναμικός Προγραμματισμός Πραγματικού Χρόνου (Real-Time Dynamic Programming - RTDP) και αλγόριθμος LAO*. Η βασική ιδέα και των δύο αλγορίθμων έχει να κάνει με την επαναληπτική εφαρμογή μεθόδων Δυναμικού Προγραμματισμού σε ένα υποσύνολο των συνολικών καταστάσεων του εκάστοτε προβλήματος. Έτσι, μέθοδοι όπως η επανάληψη πολιτικής και η επανάληψη αποτίμησης εφαρμόζονται μόνο σε ένα ενδιαφέρον υποσύνολο του συνολικού χώρου του εκάστοτε προβλήματος με αποτέλεσμα να έχουμε μειωμένο υπολογιστικό κόστος προκειμένου να υπολογίσουμε τη συνάρτηση αποτίμησης για τις πραγματικά ενδιαφέρουσες περιοχές του χώρου αναζήτησης.

Η τακτική η οποία ακολουθείται και στις δύο περιπτώσεις είναι η εναλλαγή ενός βήματος ελέγχου (εφαρμογή μιας ενέργειας a) με την εφαρμογή μιας μεθόδου Δυναμικού Προγραμματισμού για μια περιοχή καταστάσεων η οποία περιέχει πάντοτε την θέση του πράκτορα πριν την εφαρμογή της ενέργειας a η οποία πραγματοποιήθηκε στο προηγούμενο βήμα. Έτσι, ο ΔΠΠΧ καθώς και ο αλγόριθμος LAO* ενημερώνουν καταστάσεις τις οποίες μπορεί κανείς να προσεγγίσει από μια δεδομένη αρχική κατάσταση, όταν οι ενέργειες επιλέγονται άπληστα βασισμένοι στην τρέχουσα συνάρτηση αποτίμησης. Αυτό έχει σαν αποτέλεσμα τόσο ο ΔΠΠΧ όσο και ο αλγόριθμος LAO* να μπορούν να αγνοήσουν μεγάλες περιοχές του χώρου καταστάσεων γιατί πολύ απλά ο πράκτορας δεν θα βρεθεί ποτέ σε αυτές τις περιοχές ακολουθώντας μια βέλτιστη πολιτική. Έτσι, αν και η λύση την οποία βρίσκουν οι δύο αυτές μέθοδοι καθορίζει μια ενέργεια για κάθε κατάσταση στην οποία μπορεί να βρεθεί ο πράκτορας από μια δεδομένη αρχική θέση ακολουθώντας μια βέλτιστη πολιτική, μπορεί να μην καθορίζει μια ενέργεια για πολλές άλλες καταστάσεις. Για το λόγο αυτό λέμε ότι οι αλγόριθμοι αυτοί υπολογίζουν μια *μερική πολιτική (partial policy)*.

Οι δύο μέθοδοι στηρίζονται στη χρήση παραδεκτών (admissible) ευρετικών συναρτήσεων, προκειμένου να υπολογίζονται κάποιες αρχικές εκτιμήσεις της συνάρτησης αποτίμησης για διάφορες καταστάσεις. Μάλιστα, αποδεικνύεται πως όσο πιο κοντινές τιμές στις πραγματικές αποτιμήσεις δίνουν οι ευρετικές συναρτήσεις οι οποίες χρησιμοποιούνται, τόσο μικρότερος θα είναι κι ο χώρος αναζήτησης ο οποίος θα εξερευνηθεί προκειμένου ο πράκτορας να καταλήξει σε μια βέλτιστη μερική πολιτική πάνω στο χώρο καταστάσεων - ενεργειών. Από την άλλη, ο LAO* δεν αναπαριστά τη λύση ενός προβλήματος σαν μια απλή απεικόνιση από καταστάσεις σε ενέργειες, όπως για παράδειγμα κάνει ο ΔΠΠΧ και άλλοι αλγόριθμοι Δυναμικού Προγραμματισμού. Αντίθετα, αναπαριστά τη

λύση του εκάστοτε προβλήματος με ένα γράφο ο οποίος ενδέχεται να περιέχει κύκλους με μια προκαθορισμένη αρχική κατάσταση. Αυτή η αναπαράσταση γενικεύει τις γραφικές αναπαραστάσεις λύσεων οι οποίες χρησιμοποιούνται σε διαδικασίες αναζήτησης όπως είναι η A^* (ένα απλό μονοπάτι) και η AO^* (ένας γράφος χωρίς κύκλους). Το πλεονέκτημα της αναπαράστασης μιας λύσης με τη μορφή ενός γράφου είναι το γεγονός πως παρουσιάζεται ρητά η συνδεσιμότητα των διαφόρων καταστάσεων. Ο ενδιαφερόμενος αναγνώστης μπορεί να βρει περισσότερες πληροφορίες για τον Δυναμικό Προγραμματισμό Πραγματικού Χρόνου στο [3] ενώ για τον αλγόριθμο LAO^* στο [16]. Στα άρθρα αυτά μπορεί κανείς να βρει αποδείξεις για τις ιδιότητες των μεθόδων αυτών καθώς επίσης συσχετισμούς των μεθόδων αυτών με παραδεκτές συναρτήσεις και τέλος διάφορες επεκτάσεις των μεθόδων.

5.2 Βελτίωση συνάρτησης αποτίμησης.

Η «κατάρα της διαστασιμότητας» του Bellman μας αναγκάζει πολλές φορές να βρούμε μια χοντρική προσέγγιση της πραγματικής συνάρτησης αποτίμησης με τις παραδοσιακές μεθόδους Ενισχυτικής Μάθησης που παρουσιάστηκαν νωρίτερα. Παρ'όλ'αυτά, η συνάρτηση αποτίμησης αυτή, προσδιορίζει χοντρικά μια βέλτιστη πολιτική για οποιαδήποτε θέση του χώρου αναζήτησης. Ένα κρίσιμο επομένως ζήτημα είναι τι μπορούμε να κάνουμε προκειμένου να βελτιώσουμε την πολιτική που θα ακολουθήσει ο πράκτορας.

Όπως και στην προηγούμενη περίπτωση, οι προτάσεις που έχουν γίνει μέχρι σήμερα ασχολούνται με την ειδική περίπτωση εύρεσης μιας βέλτιστης πολιτικής από την εκάστοτε θέση στην οποία βρίσκεται ο πράκτορας. Η επιτυχία αυτών των μεθόδων έγκειται στο συνδυασμό μιας συνάρτησης αποτίμησης, η οποία παρέχει μια χοντρική λύση στο δύσκολο πρόβλημα εύρεσης καλών μονοπατιών από οποιαδήποτε κατάσταση, με διαδικασίες αναζήτησης σε πραγματικό χρόνο, ώστε τελικά να προκύπτει μια ακριβής λύση στο ευκολότερο πρόβλημα εύρεσης ενός καλού μονοπατιού από μια συγκεκριμένη κατάσταση.

5.2.1 "Τοπική" Αναζήτηση

Μια πρώτη ιδέα για βελτίωση της πολιτικής που ακολουθεί ο πράκτορας είναι ένα είδος "τοπικής" αναζήτησης ("local" search) για την εύρεση ενός καλού μονοπατιού πεπερασμένου (και συνήθως μικρού) πλήθους βημάτων. Η αναζήτηση είναι "τοπική" υπό την έννοια ότι πραγματοποιείται σε μια περιοχή κοντινή στην τρέχουσα κατάσταση στην οποία βρίσκεται ο πράκτορας.

Η συνήθης τακτική για τους πράκτορες που ακολουθούν μια άπληστη πολιτική πάνω σε μια συνάρτηση αποτίμησης είναι στην ουσία μια αναζήτηση ενός βήματος ή αναζήτηση ενός μονοπατιού μήκους 1 με μέγιστη προσδοκώμενη επιστροφή. Η ιδέα της "τοπικής" αναζήτησης είναι να πραγματοποιείται μια πιο εξαντλητική αναζήτηση σε πραγματικό χρόνο προκειμένου ο πράκτορας να λαμβάνει την απόφαση για το ποια ενέργεια θα εφαρμόσει σε μια δεδομένη

κατάσταση. Επομένως, σύμφωνα με αυτή την ιδέα, πραγματοποιείται μια αναζήτηση για βέλτιστο μονοπάτι μήκους $d > 1$, το οποίο φαίνεται να έχει τη μέγιστη επιστροφή σύμφωνα με το μοντέλο του περιβάλλοντος. Στη συνέχεια, αφού βρεθεί το "τοπικά" βέλτιστο μονοπάτι πραγματοποιείται η πρώτη ενέργεια προς την κατεύθυνση αυτή. Στην κατάσταση η οποία προκύπτει, ο πράκτορας επαναλαμβάνει τη διαδικασία "τοπικής" αναζήτησης και συνεχίζει με αυτόν τον τρόπο έως ότου καταλήξει σε μια τερματική κατάσταση.

5.2.2 Ολική Αναζήτηση

Μια λογική επέκταση της προηγούμενης μεθόδου είναι να μην κάνουμε αναζήτηση "τοπικά", αλλά να συνεχίσουμε να επεκτείνουμε το δέντρο αναζήτησης έως ότου βρεθεί μια τερματική κατάσταση. Το πρόβλημα όμως στην περίπτωση αυτή θα ήταν το μεγάλο πλήθος καταστάσεων το οποίο θα προέκυπτε με αποτέλεσμα το υπολογιστικό κόστος για τη διαδικασία αυτή να είναι απαγορευτικό.

Μια πρόταση είναι η δημιουργία συνόλων καταστάσεων με μια αντιπροσωπευτική κατάσταση για κάθε σύνολο. Έτσι, μπορούμε να πραγματοποιήσουμε την προηγούμενη διαδικασία "τοπικής" αναζήτησης, προκειμένου να βρίσκουμε "τοπικά" βέλτιστα μονοπάτια από ένα σύνολο καταστάσεων σε ένα άλλο. Τελικά, επαναλαμβάνοντας αυτή τη διαδικασία, προσπαθούμε να κατασκευάσουμε ένα μονοπάτι προς μια τερματική κατάσταση χρησιμοποιώντας το μοντέλο που έχουμε για το περιβάλλον. Όταν τελικά ένα τέτοιο μονοπάτι βρεθεί, ο πράκτορας το ακολουθεί μέχρι τέλους. Φυσικά, υπάρχει το ενδεχόμενο ο πράκτορας να έχει ένα λανθασμένο μοντέλο για το περιβάλλον, με αποτέλεσμα να μην είναι δυνατόν να ολοκληρωθεί το προσχεδιασμένο μονοπάτι του πράκτορα μέχρι τέλους. Στην περίπτωση αυτή όμως, ο πράκτορας θα φτάσει μέχρι κάποιο σημείο στο μονοπάτι όπου η πραγματική διαδρομή θα αποκλίνει από την προσχεδιασμένη, με αποτέλεσμα να μπορεί να βελτιώσει το μοντέλο που έχει για τη συγκεκριμένη περιοχή. Τέλος, η διαδικασία αυτή ολικής αναζήτησης (global search) μέχρι κάποια τερματική κατάσταση μπορεί να επεκταθεί με εκτεταμένη χρήση μιας προσεγγιστικής συνάρτησης αποτίμησης η οποία να καθοδηγεί την αναζήτηση στη μορφή της A^* αναζήτησης.

Περισσότερες πληροφορίες για αυτά τα είδη "τοπικής" και ολικής αναζήτησης μπορεί κανείς να βρει στα [12] και [13].

5.2.3 Μη-Ντετερμινιστικοί Χώροι Αναζήτησης.

Οι προηγούμενες μέθοδοι "τοπικής" και ολικής αναζήτησης έχουν παρουσιαστεί κι εξετασθεί επιτυχώς σε συνεχείς-ντετερμινιστικούς χώρους. Ένα κρίσιμο επομένως ζήτημα είναι τι γίνεται σε μη-ντετερμινιστικούς χώρους αναζήτησης, οι οποίοι δεν είναι και λίγοι σε πραγματικές εφαρμογές.

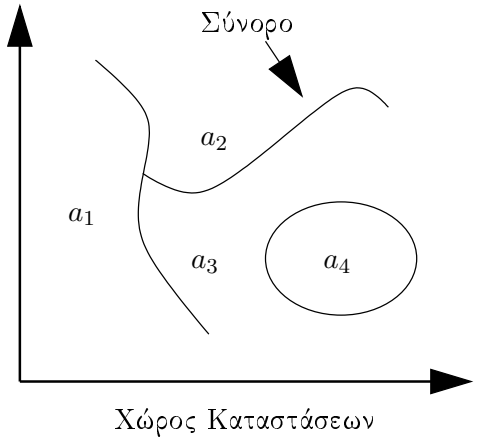
Η σημαντικότερη προσπάθεια η οποία έχει γίνει μέχρι σήμερα ίσως είναι ο αλγόριθμος Min-Max $LRTA^*$ ο οποίος έχει σχεδιαστεί για μη-ντετερμινιστικούς

χώρους αναζήτησης. Η λογική η οποία ακολουθείται στον συγκεκριμένο αλγόριθμο είναι όμοια με αυτή της "τοπικής" αναζήτησης η οποία παρουσιάστηκε νωρίτερα. Η διαφορά με τον προηγούμενο αλγόριθμο είναι πως στη συγκεκριμένη περίπτωση κατά τη διαδικασία αναζήτησης ο πράκτορας λαμβάνει υπ' όψιν του τις χειρότερες πιθανές μεταβάσεις οι οποίες μπορούν να προκύψουν μέσα από την αλληλεπίδρασή του με το περιβάλλον. Προφανώς, η χειρότερη πιθανή μετάβαση για έναν πράκτορα ο οποίος βρίσκεται σε μια κατάσταση s και εφαρμόζει μια ενέργεια a_i θα είναι να οδηγηθεί σε μια νόμιμη κατάσταση s' - επόμενη της s - για την οποία όμως μετάβαση ο πράκτορας θα λάβει την ελάχιστη ανταμοιβή μεταξύ όλων των νόμιμων μεταβάσεων που υπάρχουν στην κατάσταση s εφαρμόζοντας την ενέργεια a_i. Τελικά, ο πράκτορας ακολουθεί την πρώτη ενέργεια από το μονοπάτι εκείνο το οποίο κατά τη διαδικασία αναζήτησης φάνηκε να έχει την μέγιστη ελάχιστη επιστροφή. Δηλαδή, αν η διαδικασία αναζήτησης είναι ενός μόνο βήματος, τότε ο πράκτορας θα εφαρμόσει την ενέργεια a_j στην κατάσταση s για την οποία η ελάχιστη δυνατή ανταμοιβή $\tilde{r}^s_{a_j}$ την οποία μπορεί να λάβει είναι μεγαλύτερη από κάθε άλλη ελάχιστη ανταμοιβή $\tilde{r}^s_{a_i}$ εφαρμόζοντας κάποια ενέργεια $a_i \neq a_j$ στην κατάσταση s. Για το λόγο αυτό η μέθοδος θα έπρεπε να λέγεται Max-Min LRTA*, αφού επιλέγεται η ενέργεια εκείνη κάθε φορά η οποία μεγιστοποιεί (Max) την ελάχιστη (Min) δυνατή επιστροφή από το περιβάλλον. Ο λόγος για τον οποίο οι τελεστές αυτοί βρίσκονται με διαφορετική σειρά στην ονομασία του αλγορίθμου είναι γιατί ο αλγόριθμος δημιουργήθηκε για προβλήματα ελαχιστοποίησης κι όχι μεγιστοποίησης όπως παρουσιάζονται στη συγκεκριμένη πτυχιακή με αποτέλεσμα οι τελεστές αυτοί να εφαρμόζονται με αντίστροφη σειρά.

Ένα ενδιαφέρον χαρακτηριστικό της μεθόδου είναι η ικανότητα του πράκτορα να βελτιώνει τον απαιτούμενο χρόνο εκτέλεσης σχεδίων καθώς επιλύει παρόμοιες εργασίες με κατάστρωση σχεδίων. Περισσότερες όμως πληροφορίες για τη συγκεκριμένη μέθοδο, τη σχέση της με ευρετικές (παραδεκτές) συναρτήσεις καθώς επίσης και μια πλήρη μαθηματική της θεμελίωση μπορεί κανείς να βρει στο [19] ή στη σελίδα του Sven Koenig:
"http://www.cc.gatech.edu/fac/Sven.Koenig/rts.html".

5.3 Εξαιρετικά μεγάλοι χώροι αναζήτησης.

Οι ιδέες οι οποίες έχουν παρουσιαστεί μέχρι στιγμής αναφέρονται σε προβλήματα με όχι ιδιαίτερα μεγάλους χώρους καταστάσεων. Τη μοναδική προσπάθεια γενίκευσης κάποιων καταστάσεων μπορούμε να τη βρούμε στη διαδικασία ολικής αναζήτησης, όπου διάφορες καταστάσεις ομαδοποιούνται και για κάθε ομάδα καταστάσεων έχουμε μια αντιπροσωπευτική κατάσταση. Από την άλλη, προβλήματα με πολύ μεγάλους χώρους αναζήτησης είναι τα πιο συνηθισμένα σε καθημερινή βάση και αυτά αποτελούν τον απώτερο στόχο των μεθόδων της Ενισχυτικής Μάθησης. Η χρήση τεχνικών προσέγγισης συναρτήσεων προκειμένου ο πράκτορας να μπορεί να γενικεύει συμπεράσματα για διαφορετικές



Σχήμα 5.1: Παράδειγμα διαμέρισης του χώρου καταστάσεων

καταστάσεις και η μετέπειτα χρήση αυτών των γενικεύσεων προκειμένου να κατευθύνεται η αναζήτηση καλύτερων πολιτικών είναι μια πιθανή λύση στο πρόβλημα της «κατάρας της διαστασιμότητας». Παρ' όλ' αυτά, ακόμα κι όταν οι τεχνικές προσέγγισης συναρτήσεων επιτυγχάνουν ορθές γενικεύσεις, ο χώρος αναζήτησης καλύτερων πολιτικών παραμένει αμετάβλητος, με αποτέλεσμα πολλές φορές το υπολογιστικό κόστος να εξακολουθεί να είναι απαγορευτικό.

Προς την κατεύθυνση αντιμετώπισης του προβλήματος αυτού έχει παρουσιαστεί μια νέα μέθοδος. Σκοπός της μεθόδου είναι η μείωση του υπολογιστικού κόστους της αναζήτησης σε χώρους μεγάλων διαστάσεων εξερευνώντας μόνο ορισμένες περιοχές του χώρου καταστάσεων. Προκειμένου να επιτευχθεί κάτι τέτοιο, η συγκεκριμένη μέθοδος αναφέρεται σε μια κλάση προβλημάτων όπου η ενέργεια η οποία πραγματοποιείται από τον πράκτορα καθορίζεται ρητά από τη θέση στην οποία βρίσκεται ο πράκτορας στο χώρο καταστάσεων. Ένα απλοϊκό παράδειγμα ενός τέτοιου προβλήματος είναι αυτό του σχήματος 5.1 όπου ο χώρος καταστάσεων έχει διαμεριστεί σε 4 χωρία και ανάλογα με τη θέση στην οποία βρίσκεται ο πράκτορας πραγματοποιείται κάποια από τις ενέργειες a_1, a_2, a_3, a_4.

Προκειμένου η μέθοδος να βρίσκει καλύτερες λύσεις στο πρόβλημα το οποίο αντιμετωπίζεται, εφαρμόζει κλασικές μεθόδους αναζήτησης καλύτερων πολιτικών σε στενές περιοχές γύρω από τα σύνορα διαχωρισμού των διαφόρων ενεργειών. Έτσι, η μέθοδος ενδιαφέρεται αποκλειστικά και μόνο στην ακριβή ενημέρωση των θέσεων από τις οποίες θα περνάει το εκάστοτε σύνορο διαχωρισμού ενεργειών προκειμένου η προσδοκώμενη επιστροφή του πράκτορα να μεγιστοποιείται. Εξαιτίας του τρόπου αυτού λειτουργίας της μεθόδου, της δόθηκε το όνομα Ενισχυτική Μάθηση Επικεντρωμένη στα Σύνορα[1] (Boundary Localized Reinforcement Learning - BLRL).

[1] Ελεύθερη Μετάφραση.

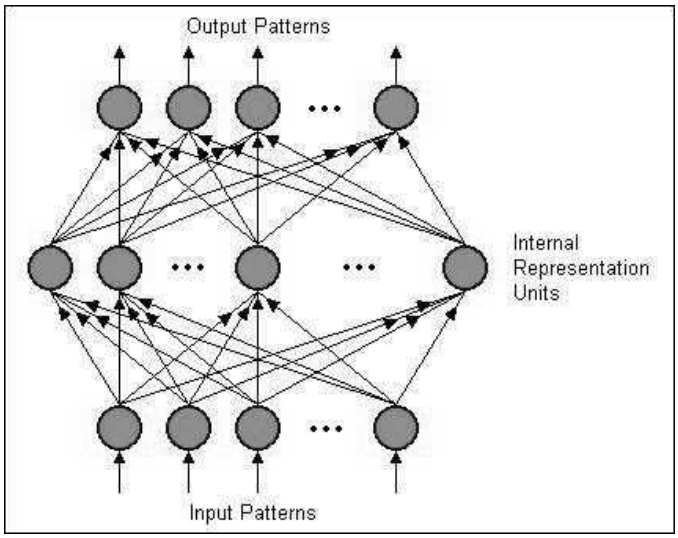
Για τη συγκεκριμένη μέθοδο είναι εξασφαλισμένο το γεγονός πως θα συγκλίνει σε ένα τοπικό βέλτιστο για τη συνάρτηση αποτίμησης του πράκτορα. Το γεγονός πως το βέλτιστο δεν είναι ολικό αλλά τοπικό δεν επηρεάζει την ευρύτητα της εφαρμογής της συγκεκριμένης μεθόδου μιας και η εύρεση τοπικών βέλτιστων είναι σχεδόν πάντα ότι καλύτερο μπορούμε να κάνουμε σε προβλήματα με πολύ μεγάλους χώρους αναζήτησης. Ο ενδιαφερόμενος αναγνώστης μπορεί να βρει περισσότερες πληροφορίες για τη συγκεκριμένη μέθοδο καθώς επίσης και μια περιγραφή για τη μαθηματική της θεμελίωση στο [15].

5.4 TD-Gammon

Κλείνοντας το συγκεκριμένο κεφάλαιο, θα αναφερθούμε στην πιο σημαντική εφαρμογή η οποία έχει γίνει μέχρι σήμερα χρησιμοποιώντας τεχνικές από το χώρο της Ενισχυτικής Μάθησης. Η εφαρμογή αυτή είναι ένα πρόγραμμα το οποίο παίζει το παιχνίδι «πόρτες» στο τάβλι και γράφτηκε από τον *Gerald Tesauro*. Το πρόγραμμα αυτό, το οποίο ονομάζεται *TD-Gammon*, αυτή τη στιγμή συγκαταλέγεται ανάμεσα στους κορυφαίους παίκτες στον κόσμο στο συγκεκριμένο παιχνίδι και αποτελεί τον εγκυρότερο αναλυτή για την αποτίμηση θέσεων και ενεργειών των παικτών σε πραγματικές παρτίδες.

Ιστορικά, η πρώτη απόπειρα του Gerald Tesauro να δημιουργήσει ένα πρόγραμμα το οποίο να παίζει καλά το παιχνίδι «πόρτες» είχε γίνει με το πρόγραμμα *Neurogammon* το οποίο το 1989 είχε καταφέρει να κερδίσει το πρωτάθλημα στη Διεθνή Ολυμπιάδα Υπολογιστών. Παρ' όλ' αυτά, εκείνο το πρόγραμμα είχε προέλθει από το χώρο της Επιβλεπόμενης Μάθησης (Supervised Learning) και το επίπεδό του σύμφωνα με τους ειδικούς κυμαινόταν στα όρια ενός ισχυρού παίκτη μέτριου επιπέδου (strong intermediate player). Φαινόταν λοιπόν, πως ήταν αναγκαίες κάποιες επιπλέον ιδέες προκειμένου το πρόγραμμα να μπορέσει να έχει καλύτερη συμπεριφορά.

Οι ιδέες αυτές προήλθαν από το χώρο της Ενισχυτικής Μάθησης. Συγκεκριμένα, το νέο πρόγραμμα του Gerald Tesauro, TD-Gammon 0.0, δημιουργήθηκε εξ ολοκλήρου από την αρχή βασισμένο αποκλειστικά στις ιδέες Μάθησης Χρονικών Διαφορών (Temporal-Difference Learning), απ' όπου προέκυψε και η ονομασία του. Το αποτέλεσμα ήταν η δημιουργία ενός παίκτη του επιπέδου του Neurogammon. Κάτι τέτοιο όμως ήταν εκπληκτικό για δύο κυρίως λόγους. Ο πρώτος λόγος έχει να κάνει με τον τρόπο μάθησης των δύο προγραμμάτων. Έτσι, ενώ το Neurogammon έφθασε στο συγκεκριμένο επίπεδο «μελετώντας» παρτίδες πολύ ισχυρών παικτών, το TD-Gammon κατάφερε να φθάσει στο ίδιο επίπεδο παίζοντας με αντίπαλο τον εαυτό του. Από την άλλη, ο πραγματικός σκοπός για τον οποίο είχε δημιουργηθεί το TD-Gammon δεν ήταν η δημιουργία ενός προγράμματος καλύτερου από οποιοδήποτε άλλο πρόγραμμα υπήρχε έως τότε για το συγκεκριμένο παιχνίδι. Όπως αποκαλύπτει ο ίδιος ο Gerald Tesauro, ο πραγματικός λόγος δημιουργίας του TD-Gammon ήταν η εξερεύνηση των δυνατοτήτων και των προοπτικών των μεθόδων Ενισχυτικής Μάθησης.



Σχήμα 5.2: Η αρχιτεκτονική του TD-Gammon
Το σχήμα προέρχεται από την ηλεκτρονική διεύθυνση του TD-Gammon.

Σαν επιστέγασμα των προσπαθειών του Tesauro ήρθε η έκδοση TD-Gammon 1.0, η οποία συνδύαζε χαρακτηριστικά τόσο από το TD-Gammon 0.0 όσο και από το Neurogammon. Ο μηχανισμός μάθησης της έκδοσης 1.0 του TD-Gammon παρέμενε ο ίδιος με αυτόν της προηγούμενης έκδοσης ενώ προστέθηκαν στις δυνατότητες του προγράμματος κάποια χαρακτηριστικά και ευρετικές συναρτήσεις δανεισμένες από το Neurogammon. Το αποτέλεσμα ήταν η δημιουργία ενός προγράμματος το οποίο ξεπέρασε κατά πολύ οποιοδήποτε άλλο πρόγραμμα είχε αναπτυχθεί για το συγκεκριμένο παιχνίδι και από την έκδοση αυτή και μετά, το TD-Gammon πρωταγωνίστησε σε παγκόσμιο επίπεδο.

Πριν όμως συνεχίσουμε με την περιγραφή των επιτευγμάτων του TD-Gammon είναι χρήσιμο να δούμε το μηχανισμό λειτουργίας του προγράμματος. Ο βασικός μηχανισμός μάθησης του TD-Gammon είναι μια μορφή μη-γραμμικής TD(λ) μεθόδου. Η αποτίμηση $V_t(s)$, μιας οποιασδήποτε κατάστασης s, απεικονίζει την πιθανότητα νίκης ξεκινώντας από τη συγκεκριμένη κατάσταση. Προκειμένου να επιτευχθεί κάτι τέτοιο, οι ανταμοιβές είναι μηδέν για όλες τις μεταβάσεις του παιχνιδιού εκτός από τις στιγμές εκείνες στις οποίες το παιχνίδι τελειώνει. Προκειμένου τώρα να προσεγγιστεί η συνάρτηση αποτίμησης, το TD-Gammon, όπως και ο πρόγονός του Neurogammon, έχει σαν βασικό μηχανισμό μάθησης ένα νευρωνικό δίκτυο. Ο λόγος για τον οποίο επιλέχθηκε ένα νευρωνικό δίκτυο είναι το γεγονός πως είναι αδύνατο να φυλαχθούν σε έναν πίνακα όλες οι πιθανές διαφορετικές θέσεις του παιχνιδιού οι οποίες υπολογίζεται πως είναι περισσότερες από 10^{20}. Η μορφή του νευρωνικού δικτύου φαίνεται στο σχήμα 5.2. Το δίκτυο αποτελείται από τρία επίπεδα. Στο πρώτο επίπεδο βρίσκεται ένα σύνολο μονάδων εισόδου (Input Patterns), το

Έκδοση	Παρτίδες Εκπαίδευσης	Αντίπαλοι	Αποτέλεσμα
0.0	300,000	Προγράμματα	Ισόπαλο στην κορυφή.
1.0	300,000	Robertie, Magriel, ...	-13 πόντους σε 51 παρτίδες.
2.0	800,000	Sylvester, Russell, ...	-7 πόντους σε 38 παρτίδες.
2.1	1,500,000	Robertie	-1 πόντο σε 40 παρτίδες.
3.0	1,500,000	Kazaros	+6 πόντους σε 20 παρτίδες.

Πίνακας 5.1: Κρισιμότερα ματς των διαφόρων εκδόσεων TD-Gammon.

οποίο δέχεται την αναπαράσταση μιας θέσης του παιχνιδιού. Στο δεύτερο επίπεδο βρίσκονται κάποιες κρυφές μονάδες για επεξεργασία της εκάστοτε θέσης (Internal Represantation Units) και στο τρίτο και τελευταίο επίπεδο συναντάμε ένα σύνολο εξόδων (Output Patterns) το οποίο παρέχει μια εκτίμηση για την αποτίμηση της τρέχουσας θέσης. Τελικά, η μάθηση επέρχεται μέσα από παρτίδες τις οποίες παίζει το πρόγραμμα με αντίπαλο τον εαυτό του.

Ολοκληρώνοντας την σύντομη αυτή παρουσίαση του TD-Gammon πρέπει να αναφέρουμε κάποια αξιοσημείωτα επιτεύγματα διαφόρων εκδόσεων του προγράμματος. Πρώτ' απ' όλα, η έκδοση 1.0 έγινε το πρώτο πρόγραμμα το οποίο έπαιζε αδιαμφισβήτητα καλύτερα από κάθε άλλο που είχε παρουσιαστεί έως τότε. Η παράδοση αυτή, του παγκόσμιου πρωταθλητή μεταξύ των προγραμμάτων, συνεχίστηκε και στις επόμενες βελτιωμένες εκδόσεις του TD-Gammon και φαίνεται ιδιαίτερα δύσκολο να βρεθεί κάποιο πρόγραμμα το οποίο να μπορεί να πάρει τα σκήπτρα. Από την έκδοση μάλιστα αυτή κι έπειτα, το TD-Gammon ήταν το πρώτο πρόγραμμα που μπορούσε να επιδιώξει νίκες σε ματς με αντιπάλους κορυφαίους παίκτες στον κόσμο. Αν και οι εκδόσεις 1.0, 2.0 και 2.1 δεν κατάφεραν να επιτύχουν νίκη σε τέτοια ματς, εντούτοις η έκδοση 3.0 τα κατάφερε. Η παρουσία του TD-Gammon σε αυτά τα ματς είχε καταλυτική επιρροή στο ρεπερτόριο ανοιγμάτων κορυφαίων παικτών. Κινήσεις οι οποίες θεωρούνταν ως οι καλύτερες δυνατές σε συγκεκριμένες θέσεις, τέθηκαν υπό αμφισβήτηση και μάλιστα με το πέρασμα του χρόνου εγκαταλείφθηκαν και αντί των κινήσεων αυτών προτιμούνται πλέον οι κινήσεις του TD-Gammon. Τέλος, το TD-Gammon αποτελεί σήμερα έναν από τους πιο έγκριτους αναλυτές παρτίδων και χρησιμοποιείται εκτεταμένα από τους επαγγελματίες παίκτες του χώρου, μιας και έχει πολλά να τους διδάξει για την πολιτική που πρέπει να ακολουθούν.

Στον πίνακα 5.1 παρουσιάζουμε συνοπτικά το αποτέλεσμα από το κρισιμότερο ματς που έδωσε η κάθε έκδοση του προγράμματος. Ο ενδιαφερόμενος αναγνώστης μπορεί να βρει περισσότερες πληροφορίες στο δικτυακό τόπο της IBM: "http://www.research.ibm.com/massive/tdl.html". Στη διεύθυνση αυτή υπάρχουν επιπλέον σύνδεσμοι για σχετικές πληροφορίες καθώς επίσης και αποτελέσματα από εφαρμογή του συγκεκριμένου μηχανισμού μάθησης σε άλλα παιχνίδια, όπως είναι το σκάκι, το Go, το Othello κ.ά.

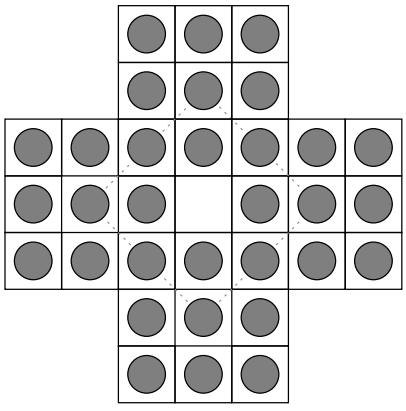
Κεφάλαιο 6

Εφαρμογή

Στο παρόν κεφάλαιο θα παρουσιάσουμε μια εφαρμογή των αλγορίθμων που προτείναμε νωρίτερα μέσα από ένα παιχνίδι ενός παίκτη. Προκειμένου να μπορούμε να συγκρίνουμε με κάποιο τρόπο τα πειραματικά αποτελέσματα, εφαρμόσαμε στο συγκεκριμένο παιχνίδι και τον απλό αλγόριθμο Q-Learning ο οποίος δίνει ένα άνω φράγμα στη συμπεριφορά των αλγορίθμων μάθησης με κατάστρωση σχεδίων.

6.1 Το παιχνίδι SOLO

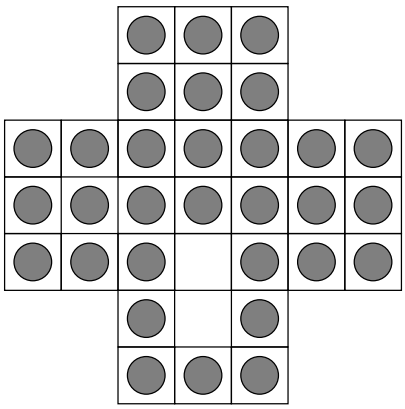
Το παιχνίδι στο οποίο εφαρμόσαμε τους προτεινόμενους αλγορίθμους είναι το παιχνίδι SOLO. Το SOLO παίζεται πάνω σε ένα πίνακα 33 θέσεων με 32 πιόνια. Αρχικά τοποθετείται ένα πιόνι σε κάθε θέση του πίνακα εκτός της κεντρικής θέσης. Στο σχήμα 6.1 φαίνεται η αρχική θέση του παιχνιδιού.



Σχήμα 6.1: Η αρχική θέση στο SOLO

6.1.1 Κανόνες του Παιχνιδιού

Δύο πιόνια ονομάζονται *γειτονικά* εάν οι θέσεις στις οποίες βρίσκονται είναι η μια δίπλα στην άλλη κατά τον οριζόντιο ή κάθετο άξονα. Έτσι κάθε πιόνι έχει το πολύ τέσσερα γειτονικά πιόνια. Η διακεκομμένη γραμμή του σχήματος 6.1 δείχνει τα οκτώ πιόνια τα οποία έχουν τέσσερα γειτονικά όταν το παιχνίδι ξεκινάει. Μια θέση ονομάζεται *κενή* εάν δεν περιέχει κάποιο πιόνι. Έτσι, στην αρχική θέση του παιχνιδιού μόνο η κεντρική θέση είναι *κενή*. Ο παίκτης πραγματοποιεί κινήσεις οριζόντια ή κάθετα, ποτέ διαγώνια, μετακινώντας ένα πιόνι κάθε φορά. Ένα πιόνι μετακινείται οριζόντια ή κάθετα κατά 2 θέσεις εφόσον κατά τη διεύθυνση κίνησης υπάρχει ένα *γειτονικό* πιόνι και η θέση στην οποία καταλήγει είναι *κενή* πριν την πραγματοποίηση της κίνησης. Μετά την ολοκλήρωση της κίνησης το γειτονικό πιόνι απομακρύνεται από τον πίνακα αφήνοντας την θέση στην οποία βρισκόταν *κενή* καθώς επίσης μένει *κενή* η θέση στην οποία βρισκόταν το πιόνι το οποίο μετακινήθηκε πριν την πραγματοποίηση της κίνησης. Έτσι, στην αρχική θέση του παιχνιδιού μόνο τέσσερα πιόνια επιτρέπεται να μετακινηθούν και είναι αυτά τα οποία βρίσκονται στις κορυφές του τετραγώνου που δημιουργεί η διακεκομμένη γραμμή. Το σχήμα 6.2 δείχνει μια πιθανή θέση του παιχνιδιού μετά από μια μόνο κίνηση.



Σχήμα 6.2: Πιθανή θέση μετά από μια κίνηση

Σκοπός του παιχνιδιού είναι να μείνει κανείς με όσο το δυνατόν λιγότερα πιόνια μπορεί, ιδανικά με ένα μονάχα πιόνι. Ανάλογα με το πόσα πιόνια αφήνει κανείς στο τέλος κατατάσσεται σε μια από τις κατηγορίες του πίνακα 6.1 στις οποίες αντιστοιχούν οι ανάλογοι βαθμοί που εκφράζουν το διανοητικό πηλίκο του κάθε ανθρώπου, υποθέτοντας ότι το 100 ανήκει στον φυσιολογικό άνθρωπο.

Εναπομείναντα Πιόνια	Κατάταξη	Διανοητικό Πηλίκο
9	Καθυστερημένος	$0 - 50$
8	Ηλίθιος	$50 - 70$
7	Βλάκας	$70 - 80$
6	Βραδύστροφος	$80 - 95$
5	Φυσιολογικός	$95 - 105$
4	Έξυπνος	$105 - 120$
3	Ιδιοφυΐα	$120 - 150$
2	Μεγαλοφυΐα	$150 - 180$
1	Σπάνια Μεγαλοφυΐα	$180 - 240$

Πίνακας 6.1: Κατάταξη παίκτη ανάλογα με την επίδοση.

6.1.2 Μετα-καταστάσεις

Πριν προχωρήσουμε στη μοντελοποίηση του προβλήματος κρίνουμε αναγκαία την παρουσίαση κάποιας τεχνικής η οποία εφαρμόζεται σε πολλές κατηγορίες προβλημάτων ενισχυτικής μάθησης. Η τεχνική αυτή αποσκοπεί στη μείωση των ζευγαριών καταστάσεων-ενεργειών τα οποία αντιμετωπίζει ένας πράκτορας κατά τη διάρκεια μάθησης.

Η τεχνική αυτή χρησιμοποιείται στις περιπτώσεις εκείνες κατά τις οποίες ένας πράκτορας χρησιμοποιεί ζεύγη καταστάσεων-ενεργειών για τη συνάρτηση αποτίμησης. Ο πιο φυσιολογικός τρόπος κωδικοποίησης των διαφόρων ενεργειών είναι μέσω της κατάστασης s από την οποία είναι διαθέσιμες οι κινήσεις μαζί με κάποια επιπλέον χαρακτηριστικά τα οποία προσδιορίζουν μοναδικά την μια κίνηση από την άλλη. Όμως, η κωδικοποίηση αυτή δεν είναι η πιο αποδοτική τις περισσότερες φορές. Έτσι, η συγκεκριμένη τεχνική κωδικοποιεί τις διάφορες ενέργειες *βάσει της κατάστασης η οποία προκύπτει* με εφαρμογή της εκάστοτε ενέργειας. Για το λόγο αυτό, μιας και οι καταστάσεις οι οποίες προκύπτουν είναι άρρηκτα συνδεδεμένες με τις ενέργειες οι οποίες οδήγησαν στις συγκεκριμένες καταστάσεις ονομάζουμε τις καταστάσεις *μετα-καταστάσεις (afterstates)*. Οι συναρτήσεις αποτίμησης κατ' επέκταση καλούνται *συναρτήσεις αποτίμησης μετα-καταστάσεων*.

Τα πλεονεκτήματα τα οποία μας παρέχει η συγκεκριμένη κωδικοποίηση θα γίνουν αμέσως προφανή. Έστω ότι σε μια κατάσταση s_1 ο πράκτορας εφαρμόζει μια ενέργεια a_1 και οδηγείται σε μια κατάσταση s'. Τότε, η βέλτιστη αποτίμηση του ζευγαριού $\langle s_1, a_1 \rangle$ δίνεται από τον τύπο:

$$Q^*(s_1, a_1) = \tilde{r}_{s_1,a_1} + \max_{a' \in A(s')} Q^*(s', a') = \tilde{r}_{s_1,a_1} + V^*(s'), \qquad (6.1)$$

όπου $\tilde{r}_{s_1,a_1}$ η άμεση ενίσχυση η οποία συνοδεύει τη μετάβαση από την κατάσταση s_1 στην κατάσταση s' μέσω της ενέργειας a_1. Εάν τώρα σε μια κατάσταση s_2 εφαρμοστεί η ενέργεια a_2 και ο πράκτορας οδηγηθεί πάλι στην κατάσταση

s', τότε η βέλτιστη αποτίμηση του ζευγαριού $\langle s_2, a_2 \rangle$ θα δίνεται από τον τύπο:

$$Q^*(s_2, a_2) = \tilde{r}_{s_2,a_2} + \max_{a' \in A(s')} Q^*(s', a') = \tilde{r}_{s_2,a_2} + V^*(s'). \tag{6.2}$$

Με αφαίρεση κατά μέλη των εξισώσεων (6.1), (6.2) προκύπτει:

$$Q^*(s_1, a_1) - Q^*(s_2, a_2) = \tilde{r}_{s_1,a_1} - \tilde{r}_{s_2,a_2}. \tag{6.3}$$

Όμως, στην περίπτωση που ισχύει

$$\tilde{r}_{s_1,a_1} = \tilde{r}_{s_2,a_2}, \tag{6.4}$$

τότε από τη σχέση (6.3) προκύπτει πως ισχύει:

$$Q^*(s_1, a_1) = Q^*(s_2, a_2),$$

δηλαδή η αποτίμηση δύο διαφορετικών ζευγαριών καταστάσεων-ενεργειών είναι η ίδια. Μάλιστα είναι αρκετά συχνό το φαινόμενο της σχέσης (6.4) σε προβλήματα ενισχυτικής μάθησης. Ο λόγος είναι πως οι στόχοι καθορίζονται από τις ανταμοιβές τις οποίες λαμβάνει ο πράκτορας και είναι βέβαιο πως σε ντετερμινιστικούς χώρους το εκάστοτε πρόβλημα μάθησης μπορεί να μοντελοποιηθεί με τέτοιο τρόπο ώστε οι ανταμοιβές τις οποίες λαμβάνει ένας πράκτορας να είναι κοινές στις περισσότερες μεταβάσεις μέχρι την ολοκλήρωση ενός επεισοδίου. Κάτι τέτοιο οφείλεται σε δύο λόγους:

- Στις περιπτώσεις κατά τις οποίες θέλουμε ένας πράκτορας να επιτύχει κάποιο συγκεκριμένο στόχο - ανεξάρτητα από το πλήθος ενεργειών το οποίο απαιτήθηκε για το λόγο αυτό - είναι πολύ φυσιολογικό το σήμα ενίσχυσης $\tilde{r}_{s_i,a_i}$ να είναι μηδέν για όλες τις μεταβάσεις οι οποίες δεν οδηγούν σε τερματική κατάσταση και μόνο για τις μεταβάσεις εκείνες οι οποίες οδηγούν σε τερματική κατάσταση να λαμβάνει μια διαφορετική τιμή η ανταμοιβή, ώστε να αντικατοπτρίζεται ο βαθμός στον οποίο επιτεύχθηκε κάποιος στόχος.

- Σε άλλες περιπτώσεις θέλουμε ένας πράκτορας να επιτύχει κάποιο στόχο με όσο το δυνατόν λιγότερες ή περισσότερες ενέργειες. Πάλι, σε αυτές τις περιπτώσεις όταν προκύπτουν μεταβάσεις από μη-τερματικές καταστάσεις σε μη-τερματικές καταστάσεις είναι βολικό να δίνουμε την ίδια (μη μηδενική) ανταμοιβή $\tilde{r}_{s_i,a_i}$, ενώ όταν ο πράκτορας οδηγείται σε μια τερματική κατάσταση να του δίνεται κάποια άλλη ανταμοιβή.

Όπως κι αν έχουν όμως τα πράγματα, από τα παραπάνω γίνεται φανερό πως σε πάρα πολλές μεταβάσεις πολλών προβλημάτων ενισχυτικής μάθησης ισχύει η σχέση (6.4), με αποτέλεσμα σε όλα αυτά τα περιβάλλοντα να μπορούμε να κωδικοποιήσουμε τις ενέργειες βάσει των *μετα-καταστάσεων* στις οποίες οδηγούν.

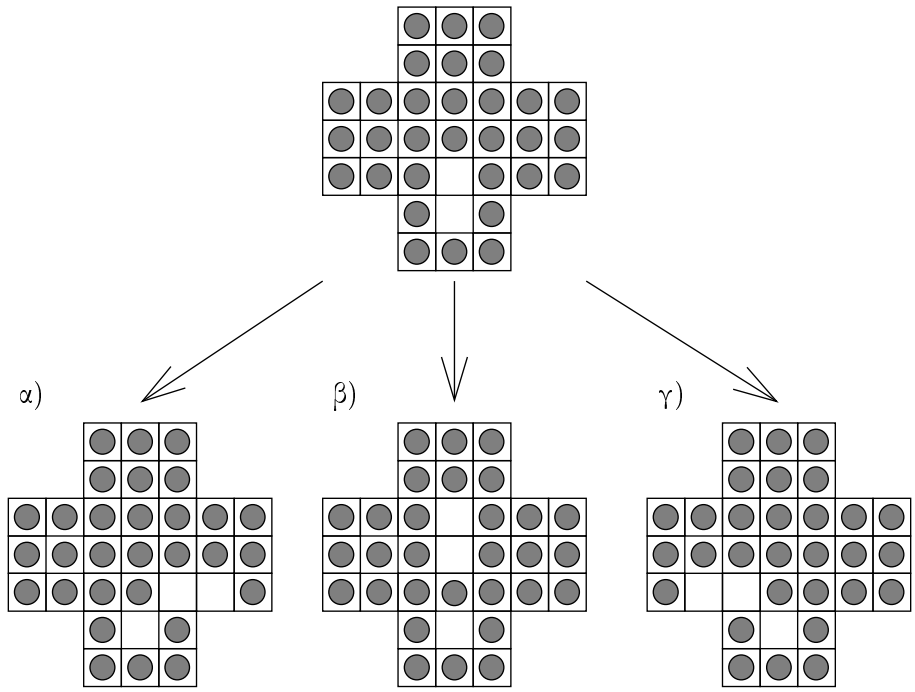
6.1.3 Μοντελοποίηση και Περιβάλλον

Προκειμένου να μοντελοποιήσουμε το παιχνίδι SOLO σαν ένα πρόβλημα ενισχυτικής μάθησης πραγματοποιήσαμε τα ακόλουθα. Οι ανταμοιβές οι οποίες δίνονται στον πράκτορα κατά τις διάφορες μη-τερματικές μεταβάσεις είναι όλες μηδέν, ενώ όταν ο πράκτορας επισκέπτεται μια τερματική κατάσταση λαμβάνει ανταμοιβή της οποίας η απόλυτη τιμή είναι ίση με το πλήθος των πιονιών τα οποία έχουν απομείνει στον πίνακα, όμως η πραγματική τιμή της ανταμοιβής είναι αρνητική, ώστε ο πράκτορας να επιδιώκει το στόχο του προβλήματος μέσα από τη διαδικασία μεγιστοποίησης της προσδοκώμενης επιστροφής. Μια εναλλακτική προσέγγιση θα ήταν να δίναμε στον πράκτορα μια (θετική) μονάδα ανταμοιβής κατά την πραγματοποίηση κάθε κίνησης.

Όποια προσέγγιση κι αν ακολουθήσει κανείς, παρατηρούμε ότι έχουμε να κάνουμε με μια εργασία της οποίας οι ανταμοιβές δεν υπόκεινται σε έκπτωση, δηλαδή ο ρυθμός έκπτωσης είναι $\gamma = 1.0$. Επίσης, σύμφωνα με όσα παρατηρήσαμε στην προηγούμενη παράγραφο, παρατηρούμε ότι με τη συγκεκριμένη μοντελοποίηση μπορούμε να πραγματοποιήσουμε κωδικοποίηση των διαφόρων ενεργειών βάσει των μετα-καταστάσεων του παιχνιδιού και πράγματι το γεγονός αυτό το εκμεταλλευτήκαμε κατά την υλοποίηση. Επιπλέον, στο συγκεκριμένο παιχνίδι μπορούμε να εκμεταλλευτούμε κάτι ακόμα πιο ισχυρό. Είναι γεγονός πως εμφανίζεται ένα είδος οκταπλής συμμετρίας, αφού πολλές θέσεις ταυτίζονται με κάποιες άλλες αν περιστρέψουμε τον πίνακα κυκλικά, ή πάρουμε την κατοπτρικά συμμετρική θέση από την θέση την οποία εξετάζουμε μαζί με τις όποιες κυκλικές περιστροφές μπορούν να προκύψουν από τη συμμετρική θέση.

Έτσι, στην αρχική θέση του παιχνιδιού, ο πράκτορας δεν έχει να αντιμετωπίσει τέσσερις διαφορετικές κινήσεις αλλά μονάχα μια, αφού οποιαδήποτε κίνηση επιλέξει οδηγεί σε μια κυκλικά συμμετρική θέση από αυτή του σχήματος 6.2. Κατά συνέπεια, οι πιθανές θέσεις μετά από μια κίνηση δεν είναι τέσσερις αλλά μόνο μια. Φυσικά, αυτή η διαδικασία επεκτείνεται και σε όλες τις υπόλοιπες πιθανές καταστάσεις μειώνοντας έτσι δραματικά τον χώρο αναζήτησης. Προκειμένου να δώσουμε τώρα κι ένα παράδειγμα κατοπτρικά συμμετρικών θέσεων ας παρατηρήσουμε πως οι διαθέσιμες κινήσεις του πράκτορα από τη θέση του σχήματος 6.2 είναι τρεις και οδηγούν στις καταστάσεις που εικονίζονται στο σχήμα 6.3. Στο σχήμα αυτό, στις εικόνες (α), (β) και (γ) παρατηρούμε τις καταστάσεις οι οποίες προκύπτουν μετά από εφαρμογή κάθε πιθανής κίνησης στη θέση του σχήματος 6.2. Είναι φανερό, πως καμία από αυτές τις καταστάσεις δεν ταυτίζεται με τις υπόλοιπες μέσω κυκλικής συμμετρίας. Όμως μια πιο προσεκτική ματιά μας οδηγεί στο συμπέρασμα πως οι καταστάσεις (α) και (γ) οφείλουν να έχουν την ίδια προσδοκώμενη επιστροφή, μιας και ταυτίζονται μέσω κατοπτρικής κυκλικής συμμετρίας. Έτσι, οι διαθέσιμες κινήσεις στην προηγούμενη κατάσταση δεν ήταν τρεις αλλά δύο με αποτέλεσμα και οι πιθανές επόμενες διαφορετικές καταστάσεις να είναι μόνο δύο.

Σχετικά με τη συνάρτηση αποτίμησης θα μπορούσαμε να χρησιμοποιήσου-



Σχήμα 6.3: Πιθανές θέσεις μετά από δύο κινήσεις

με την τεχνική *αισιόδοξων αρχικών τιμών* και να θέσουμε κάθε ζευγάρι $\langle s, a\rangle$ στην τιμή -1. Έτσι, ο πράκτορας σε αυτή την περίπτωση θα ήταν αισιόδοξος πως κάθε ενέργεια την οποία μπορεί να πραγματοποιήσει είναι δυνατόν να τον οδηγήσει σε τερματική κατάσταση με ένα μόνο πιόνι. Η προσέγγιση που ακολουθήσαμε αρχικοποίησε κάθε ζευγάρι $\langle s, a\rangle$ στην τιμή μηδέν (0). Διαισθητικά, η αρχικοποίηση αυτή σημαίνει πως ο πράκτορας αρχικά πιστεύει πως μπορεί να μένει χωρίς πιόνια στο τέλος κάθε παιχνιδιού για οποιαδήποτε ενέργεια κι αν πραγματοποιήσει. Φυσικά κάτι τέτοιο είναι αδύνατο, αλλά ουσιαστικά πρόκειται για άλλη μια εφαρμογή της χρήσης *αισιόδοξων αρχικών τιμών*. Ουσιαστικά, έχουμε *υπερ-αισιόδοξες αρχικές τιμές* με αποτέλεσμα να αναγκάζουμε τον πράκτορα να διαλέγει ενέργειες οι οποίες τον οδηγούν σε καταστάσεις τις οποίες δεν έχει επισκεφθεί στο παρελθόν· αντί να επισκέπτεται, διαλέγοντας στην τύχη, μεταξύ καταστάσεων οι οποίες έχουν αποτίμηση -1 και τις έχει επισκεφθεί στο παρελθόν και καταστάσεων οι οποίες έχουν αποτίμηση -1 και δεν τις έχει επισκεφθεί στο παρελθόν. Στο θέμα όμως αυτό θα επανέλθουμε όταν εξετάσουμε τα πειραματικά αποτελέσματα των μεθόδων.

Για τον παράγοντα μάθησης α από την άλλη, εκμεταλλευτήκαμε το γεγονός πως ο πράκτορας αντιμετωπίζει ένα ντετερμινιστικό-στατικό περιβάλλον και θέσαμε την τιμή του στον αριθμό $\alpha = 1.0$. Ο λόγος είναι πως ο πράκτορας δεν χρειάζεται να περάσει περισσότερες από μια φορές από ένα ζευγάρι $\langle s, a\rangle$

σε περιβάλλοντα αυτού του τύπου προκειμένου να γνωρίζει τη μέση «άμεση» ανταμοιβή κάθε ενέργειάς του αφού αυτή θα είναι ένας σταθερός αριθμός και πάντοτε ο ίδιος.

Τέλος, για τις προτεινόμενες μεθόδους διαθέσαμε χρόνο για πραγματοποίηση 300 ενημερώσεων μετά το τέλος κάθε επεισοδίου. Σίγουρα η τιμή αυτή είναι εξαιρετικά μεγάλη, μιας και στην καλύτερη περίπτωση ένα επεισόδιο μπορεί να διαρκέσει το πολύ για 31 κινήσεις του πράκτορα. Είναι γνωστό πως οι μέθοδοι ενισχυτικής μάθησης στηρίζονται περισσότερο σε εκτιμήσεις οι οποίες βασίζονται σε μεγάλη πραγματική εμπειρία του πράκτορα με το περιβάλλον (πολλά επεισόδια) κι όχι σε προσομοιωμένη εμπειρία. Όμως, αν φανεί πως οι μέθοδοι κάνουν καλή χρήση του χρόνου που τους διατίθεται για ένα τόσο μεγάλο πλήθος προσομοιωμένων ενημερώσεων, τότε προφανώς η επίδοση αυτή θα είναι εξαιρετική και κατά τη γνώμη μας τα πειραματικά αποτελέσματα αυτό ακριβώς δείχνουν.

6.1.4 Χαρακτηριστικά του παιχνιδιού

Πριν προχωρήσουμε στην παρουσίαση των πειραματικών αποτελεσμάτων κρίνουμε αναγκαία την παρουσίαση διαφόρων χαρακτηριστικών του προβλήματος ώστε να έχει κανείς μια ολοκληρωμένη εικόνα για τις ιδιαιτερότητες και τις δυσκολίες του συγκεκριμένου προβλήματος μάθησης. Επιπλέον, μέσα από αυτή την παράγραφο θα γίνει φανερή η «βοήθεια» της κωδικοποίησης των ενεργειών με χρήση μετα-καταστάσεων.

Αν ξεχάσουμε προς στιγμήν την κωδικοποίηση των ενεργειών με χρήση μετα-καταστάσεων, τότε στον πίνακα 6.2 μπορούμε να παρατηρήσουμε το πλήθος των διαθέσιμων ενεργειών του πράκτορα σε κάθε επίπεδο του χώρου αναζήτησης. Να σημειωθεί πως δύο καταστάσεις βρίσκονται στο ίδιο επίπεδο του χώρου αναζήτησης αν και μόνο αν ο πράκτορας έχει πραγματοποιήσει το ίδιο πλήθος ενεργειών.

Από την άλλη, αν χρησιμοποιήσουμε κωδικοποίηση ενεργειών με χρήση μετα-καταστάσεων, τότε στον πίνακα 6.3 φαίνεται το πλήθος των διαφορετικών μετα-καταστάσεων ανά επίπεδο. Συγκρίνοντας κανείς τις τιμές των πεδίων αυτού του πίνακα με τις αντίστοιχες τιμές στα πεδία του πίνακα 6.2 γίνεται φανερή η σημαντικότατη μείωση του χώρου αναζήτησης τον οποίο θα κληθεί ο πράκτορας να αντιμετωπίσει. Στηριζόμενοι στον πίνακα 6.3 μπορούμε να δείξουμε γραφικά τον μέσο παράγοντα διακλάδωσης για το συγκεκριμένο πρόβλημα στο σχήμα 6.4 καθώς επίσης και την κατανομή των διαφορετικών μετα-καταστάσεων ανά επίπεδο στο σχήμα 6.5, απ' όπου γίνεται φανερό πως ακολουθείται μια κανονική κατανομή.

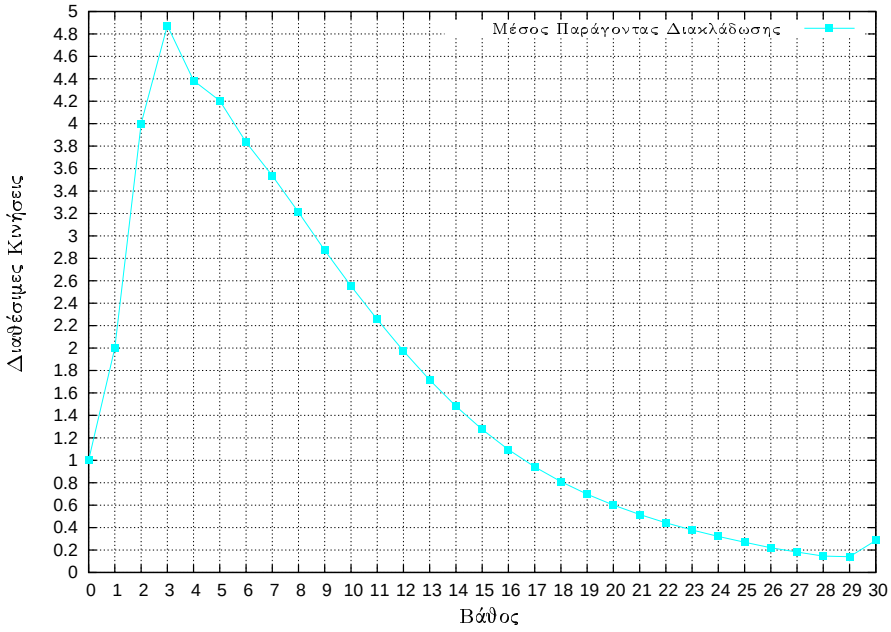
Τέλος, στον πίνακα 6.4 μπορούμε να δούμε το πλήθος των διαφορετικών τερματικών μετα-καταστάσεων του παιχνιδιού καθώς επίσης και στο σχήμα 6.6 την αντίστοιχη γραφική παράσταση της κατανομής αυτής. Παρατηρώντας κανείς αυτά τα δύο στατιστικά στοιχεία του παιχνιδιού μπορεί να δικαιολογήσει την κατάταξη η οποία πραγματοποιείται στον πίνακα 6.1 ανάλογα με το

Επίπεδο	Διαθέσιμες Κινήσεις	Επίπεδο	Διαθέσιμες Κινήσεις
0	4	16	28,939,480
1	3	17	29,624,970
2	10	18	25,738,459
3	51	19	18,985,568
4	294	20	11,882,254
5	1,453	21	6,292,363
6	6,606	22	2,802,072
7	26,912	23	1,042,481
8	99,280	24	322,080
9	325,332	25	81,186
10	940,614	26	16,170
11	2,392,350	27	2,431
12	5,316,996	28	282
13	10,234,767	29	17
14	16,948,714	30	2
15	23,997,734	31	0

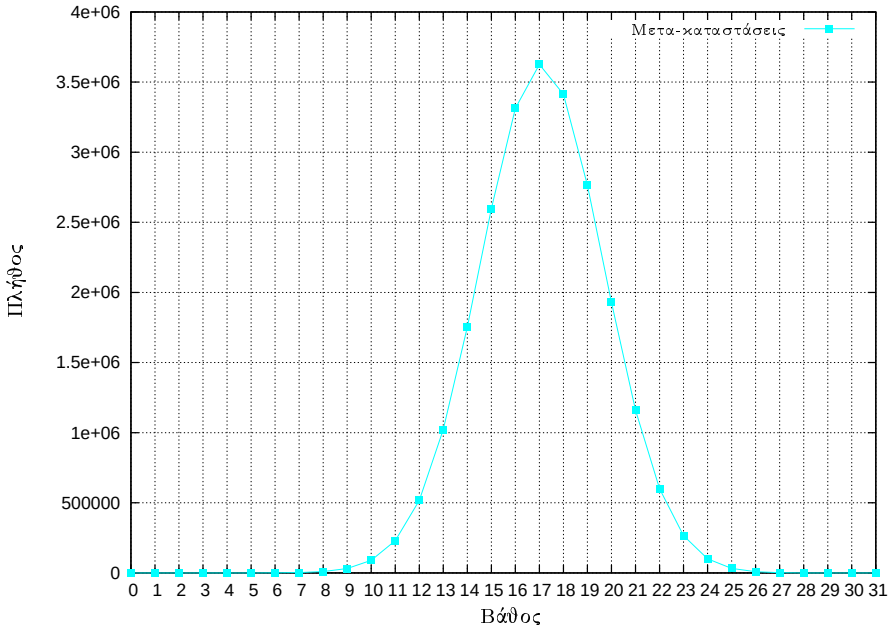
Πίνακας 6.2: Πλήθος διαθέσιμων κινήσεων παίκτη ανά επίπεδο.

Επίπεδο	Μετα-καταστάσεις	Επίπεδο	Μετα-καταστάσεις
0	1	16	3,312,423
1	1	17	3,626,632
2	2	18	3,413,313
3	8	19	2,765,623
4	39	20	1,930,324
5	171	21	1,160,977
6	719	22	600,372
7	2,757	23	265,865
8	9,751	24	100,565
9	31,312	25	32,250
10	89,927	26	8,688
11	229,614	27	1,917
12	517,854	28	348
13	1,022,224	29	50
14	1,753,737	30	7
15	2,598,215	31	2

Πίνακας 6.3: Πλήθος διαφορετικών μετα-καταστάσεων ανά επίπεδο.



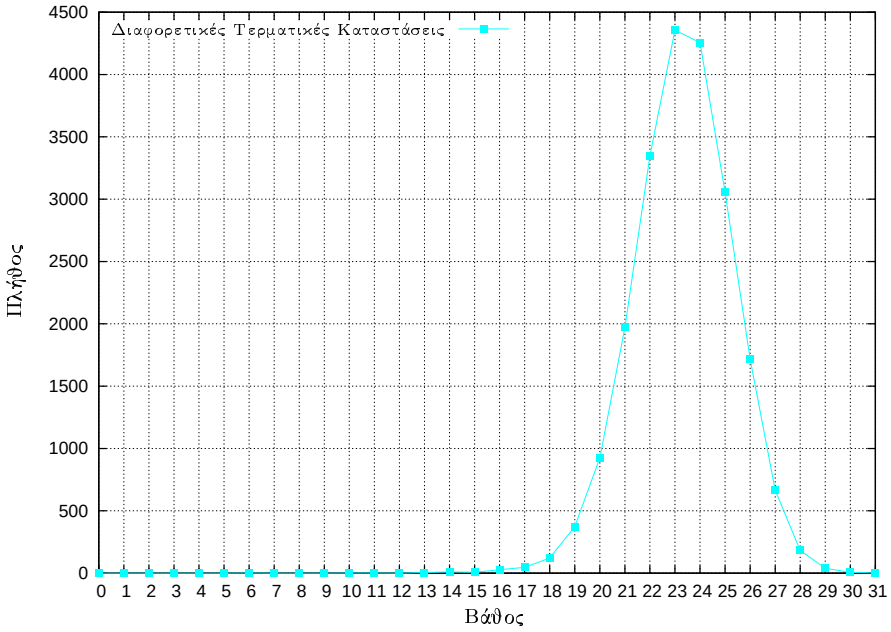
Σχήμα 6.4: Μέσος παράγοντας διακλάδωσης στο παιχνίδι SOLO



Σχήμα 6.5: Κατανομή μετα-καταστάσεων στο παιχνίδι SOLO

Επίπεδο	Μετα-καταστάσεις	Επίπεδο	Μετα-καταστάσεις
0	0	16	27
1	0	17	47
2	0	18	121
3	0	19	373
4	0	20	925
5	0	21	1,972
6	1	22	3,346
7	0	23	4,356
8	0	24	4,256
9	0	25	3,054
10	1	26	1,715
11	1	27	665
12	0	28	182
13	5	29	39
14	10	30	6
15	7	31	2

Πίνακας 6.4: Πλήθος διαφορετικών τερματικών μετα-καταστάσεων ανά επίπεδο.



Σχήμα 6.6: Κατανομή τερματικών μετα-καταστάσεων στο παιχνίδι SOLO

αποτέλεσμα του κάθε παίκτη στο τέλος κάθε παιχνιδιού.

6.2 Πειραματικά αποτελέσματα

Έχοντας πραγματοποιήσει όλη την προηγούμενη περιγραφή χαρακτηριστικών του προβλήματος που καλείται να αντιμετωπίσει ο πράκτορας μάθησης, είμαστε πλέον έτοιμοι να παρουσιάσουμε τα πειραματικά αποτελέσματα τα οποία προέκυψαν από τους προτεινόμενους αλγορίθμους μάθησης του προηγούμενου κεφαλαίου.

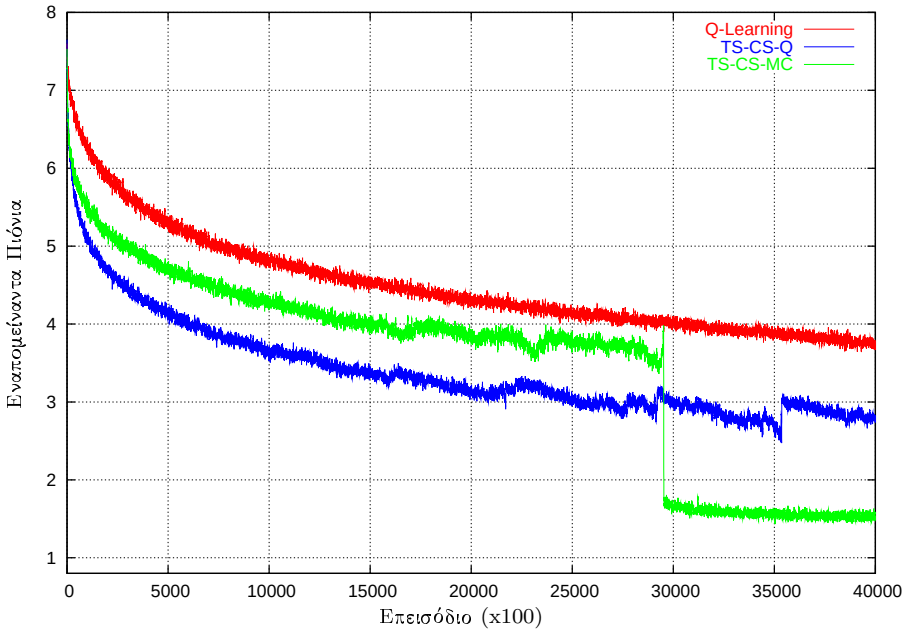
Προκειμένου να μπορούμε να συγκρίνουμε τους διαφόρους αλγορίθμους επιλύσαμε το συγκεκριμένο πρόβλημα και με τη μέθοδο Q-Learning, ώστε να έχουμε και μια άμεση οπτική σύγκριση της επίδοσης των αλγορίθμων σε σχέση με ένα άνω φράγμα συμπεριφοράς. Επιπλέον, χρησιμοποιώντας κωδικοποίηση ενεργειών με χρήση μετα-καταστάσεων, το συνολικό πλήθος ζευγαριών $\langle s, a \rangle$ ανήλθε μόλις στα 23,475,688 καθιστώντας έτσι δυνατή την αντιμετώπιση του προβλήματος με μεθόδους Δυναμικού Προγραμματισμού.

Μάλιστα, η αντιμετώπιση του προβλήματος με μια μέθοδο Δυναμικού Προγραμματισμού μπορεί να πραγματοποιηθεί εξαιρετικά αποδοτικά σε ένα και μόνο πέρασμα μέσα από το χώρο των ζευγαριών $\langle s, a \rangle$. Αυτό οφείλεται στο γεγονός πως οι άμεσες ενισχύσεις τις οποίες λαμβάνει ο πράκτορας κατά τις διάφορες μεταβάσεις είναι κάποιες σταθερές τιμές κι όχι νούμερα τα οποία προκύπτουν μέσα από κάποια κατανομή. Έτσι, πραγματοποιώντας ένα μόνο πέρασμα από καταστάσεις μεγάλου βάθους σε καταστάσεις μικρότερου βάθους είναι δυνατόν να πάρουμε τις βέλτιστες τιμές για τη συνάρτηση αποτίμησης. Διαισθητικά, κάτι τέτοιο ήταν αναμενόμενο εκ των προτέρων, μιας και ο στόχος του πράκτορα στο συγκεκριμένο πρόβλημα είναι να φθάσει όσο το δυνατόν πιο «μακριά» μέσα σε ένα περιβάλλον.

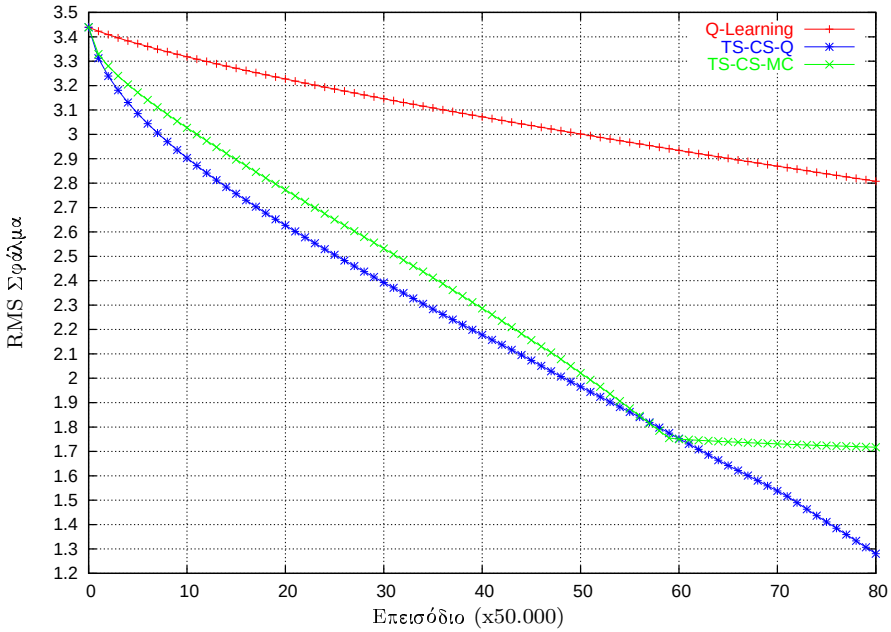
Η επίλυση του προβλήματος με μια μέθοδο Δυναμικού Προγραμματισμού μας επέτρεψε να συγκρίνουμε τους δύο αλγορίθμους βάσει του RMS σφάλματος στη συνάρτηση αποτίμησης όπως αυτή διαμορφωνόταν κατά τη διάρκεια μάθησης, ώστε τελικά να έχουμε ένα επιπλέον κριτήριο σύγκρισης. Για τις διάφορες μεθόδους διαθέσαμε χρόνο μάθησης ίσο με 4,000,000 επεισόδια. Έτσι, στο σχήμα 6.7 παρατηρούμε (εξομαλυσμένα) την επίδοση των μεθόδων, ενώ στο σχήμα 6.8 παρατηρούμε την πορεία ελάττωσης του μέσου τετραγωνικού σφάλματος στη συνάρτηση αποτίμησης για την κάθε μέθοδο. Τέλος στο σχήμα 6.9 φαίνεται η κατανομή τερματικών θέσεων όπως αυτές παρατηρήθηκαν από τους διαφόρους αλγορίθμους κατά τη διάρκεια μάθησης.

6.2.1 Κριτική - Συμπεράσματα

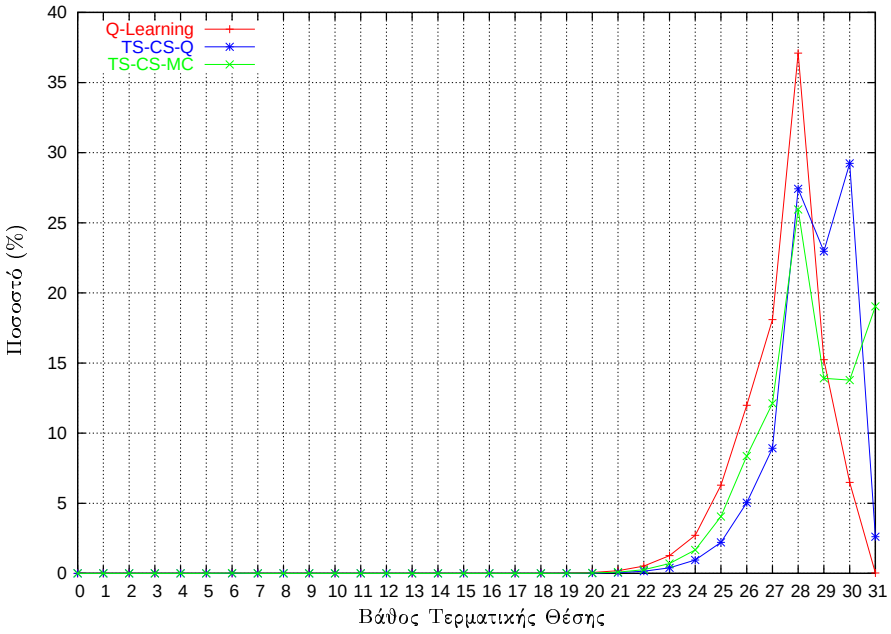
Τα προηγούμενα σχήματα 6.7, 6.8 και 6.9 δικαιολογούν κατά την άποψή μας, το γεγονός πως οι πράκτορες οι οποίοι ακολούθησαν τις προτεινόμενες μεθόδους μάθησης, είχαν αρκετά καλή συμπεριφορά. Ας δούμε όμως ανα-



Σχήμα 6.7: Επιδόσεις Αλγορίθμων
Τα αποτελέσματα της συμπεριφοράς των διαφόρων αλγορίθμων μάθησης.
Προέρχονται από μια εκτέλεση και έχουν εξομαλυνθεί.



Σχήμα 6.8: Ολικό σφάλμα στη συνάρτηση αποτίμησης
Το RMS σφάλμα της συνάρτησης αποτίμησης κάθε μεθόδου όπως αυτή διαμορφώνεται κατά τη διάρκεια εξέλιξης των επεισοδίων.



Σχήμα 6.9: Κατανομή τερματικών θέσεων
Κατανομή των τερματικών θέσεων όπως αυτές παρατηρήθηκαν από τους διαφόρους αλγορίθμους μάθησης.

λυτικότερα κάποια συστατικά στοιχεία τα οποία συνηγορούν προς αυτή την κατεύθυνση.

Πλήθος Επεισοδίων

Από το σχήμα 6.7 παρατηρούμε πως η μέθοδος TS−CS−MC χρειάστηκε κάτι λιγότερο από 3,000,000 επεισόδια προκειμένου να αποκτήσει ϵ-βέλτιστη συμπεριφορά. Αντίθετα, η μέθοδος TS−CS−Q μέσα στο ίδιο πλήθος επεισοδίων μάθησης δεν κατάφερε να αποκτήσει βέλτιστη συμπεριφορά. Είναι γεγονός πάντως πως και οι δύο μέθοδοι κατάφεραν να συμπεριφερθούν πολύ καλύτερα από τη μέθοδο Q-Learning κατά τη διάρκεια μάθησης και μέσα στο ίδιο πλήθος επεισοδίων να έχουν πολύ χαμηλότερο μέσο όρο πιονιών σε τελικές καταστάσεις. Επίσης, να τονίσουμε το γεγονός, πως και η μέθοδος TS−CS−Q τελικά θα παρουσιάσει κάποια στιγμή μια ραγδαία μείωση των πιονιών τα οποία απομένουν στο τέλος κάθε επεισοδίου ενώ αντίθετα η μέθοδος Q-Learning είναι καταδικασμένη να οδηγηθεί σε βέλτιστη συμπεριφορά με πολύ πιο αργούς ρυθμούς. Προς την κατεύθυνση αυτή συνηγορούν και οι γραφικές παραστάσεις του σχήματος 6.8.

Σφάλμα στη συνάρτηση αποτίμησης

Στο σχήμα 6.8 μπορούμε να παρατηρήσουμε το μέσο τετραγωνικό σφάλμα στη συνάρτηση αποτίμησης κάθε μεθόδου. Από το σχήμα αυτό, γίνεται φανερό πως οι μέθοδοι χρησιμοποιούσαν πολύ αποδοτικά το χρόνο μάθησης ο οποίος τους δινόταν μετά το τέλος κάθε επεισοδίου.

Συνήθως, το πλήθος προσομοιωμένων ενημερώσεων το οποίο διατίθεται σε μεθόδους ενισχυτικής μάθησης με κατάστρωση σχεδίων δεν υπερβαίνει κατά πολύ το μέσο πλήθος ενημερώσεων το οποίο θα πραγματοποιείται με βέλτιστη (ή ϵ-βέλτιστη) συμπεριφορά των μεθόδων. Ουσιαστικά, στα περισσότερα προβλήματα μπορούμε να έχουμε ένα βέβαιο άνω φράγμα του πλήθους ενημερώσεων το οποίο θα πραγματοποιείται σε κάθε επεισόδιο και συνήθως αυτή η τιμή (ή κάποια κοντινή της) χρησιμοποιείται σαν το μέγιστο επιτρεπτό πλήθος προσομοιωμένων ενημερώσεων. Ο λόγος για τον οποίο ακολουθείται αυτή η τακτική είναι το γεγονός πως οι μέθοδοι μάθησης δεν πραγματοποιούν πολύ αποτελεσματικές ενημερώσεις κατά τη διάρκεια της προσομοιωμένης εμπειρίας.

Στο συγκεκριμένο πρόβλημα ένα βέβαιο άνω φράγμα για το πλήθος των ενημερώσεων με βέλτιστη συμπεριφορά μιας μεθόδου γνωρίζαμε εκ των προτέρων πως ήταν η τιμή 31. Παρ' όλ' αυτά, η τιμή του πλήθους προσομοιωμένων ενημερώσεων για τις διάφορες μεθόδους που χρησιμοποιήσαμε είχε την τιμή 300, μια τιμή εξαιρετικά μεγάλη σύμφωνα με το συλλογισμό της προηγούμενης παραγράφου. Τα αποτελέσματα όμως επιβραβεύουν την εκλογή που κάναμε για τη συγκεκριμένη παράμετρο και δείχνουν πως περίπου οι 200 ενημερώσεις από αυτές συνέβαλλαν στο τέλος κάθε επεισοδίου σε μείωση του μέσου τετραγωνικού σφάλματος.

Επιπλέον, εκ των προτέρων γνωρίζαμε πως κάποιες ενημερώσεις δεν θα βοηθούσαν προς την κατεύθυνση μείωσης του μέσου τετραγωνικού σφάλματος. Ο λόγος είναι προφανώς η ίδια η μέθοδος χειρισμού αναζήτησης σε συνδυασμό με την εκλογή της παραμέτρου του παράγοντα μάθησης, ο οποίος είχε την τιμή $\alpha = 1.0$ και την ταυτόχρονη χρήση αισιόδοξων αρχικών τιμών. Όπως έχει αναφερθεί, ο χειρισμός αναζήτησης πραγματοποιεί ενημερώσεις σε άσχημες περιοχές της συνάρτησης αποτίμησης, μιας κι από εκείνες τις περιοχές ο πράκτορας θα περάσει σπάνια. Στη συγκεκριμένη όμως εφαρμογή, ο παράγοντας μάθησης έχοντας την τιμή $\alpha = 1.0$, δημιουργεί απότομες διακυμάνσεις πάνω στα διάφορα ζευγάρια $\langle s, a \rangle$ με αποτέλεσμα ο χειρισμός αναζήτησης να αναζητεί χρήσιμες ενημερώσεις σε περιοχές τις οποίες έχει εξερευνήσει στο παρελθόν ο πράκτορας και έχει κάποια εκτίμηση για την προσδοκώμενη επιστροφή. Αυτό, σε συνδυασμό με τη χρήση αισιόδοξων αρχικών τιμών έχει σαν αποτέλεσμα αρκετές ενημερώσεις να πηγαίνουν χαμένες κατά τη διαδικασία προσομοιωμένων ενημερώσεων. Ο λόγος όμως για τον οποίο δεν τροποποιήσαμε το χειρισμό αναζήτησης στη συγκεκριμένη εφαρμογή είναι γιατί προφανώς προσπαθούσαμε να εξετάσουμε τη συμπεριφορά των συγκεκριμένων μεθόδων όσο πιο αντικειμενικά μπορούσαμε σύμφωνα με την πιο γενική τους μορφή. Εξάλλου, μπορεί μεν να είχαμε κατά νου ντετερμινιστικά-στατικά περιβάλλοντα κατά τη δημιουργία των διαφόρων μεθόδων, όμως αυτές επεκτείνονται εύκολα και στα υπόλοιπα περιβάλλοντα και η γενικότητα των προτεινόμενων μεθόδων διαισθητικά αναμένεται να έχει κι εκεί εξίσου καλά (αν όχι καλύτερα) αποτελέσματα.

Τερματικές καταστάσεις

Μέσα από όλη την πορεία μάθησης των 4,000,000 επεισοδίων η μέθοδος Q-Learning παρουσίασε ένα μέσο όρο 4.5502 πιονιών ανά επεισόδιο. Αντίστοιχα η μέθοδος TS−CS−Q είχε έναν μέσο όρο 3.5 πιονιών ανά επεισόδιο ενώ η μέθοδος TS−CS−MC είχε έναν μέσο όρο 3.5552 πιονιών ανά επεισοδιο.

Έτσι, αν και η TS−CS−MC μέθοδος κατάφερε νωρίτερα να αποκτήσει ϵ-βέλτιστη συμπεριφορά όπως φαίνεται στο σχήμα 6.7, εντούτοις η μέθοδος TS−CS−Q είχε καλύτερη μέση επίδοση ανά επεισόδιο. Βέβαια, το αποτέλεσμα αυτό θα μπορούσε να είναι εύκολα διαφορετικό αν για παράδειγμα είχαμε μικρότερη τιμή στην πιθανότητα εξερεύνησης για τη μέθοδο TS−CS−MC. Η τιμή της πιθανότητας εξερεύνησης για τη συγκεκριμένη μέθοδο είχε τεθεί σε 2% ώστε σε ϵ-βέλτιστη συμπεριφορά της μεθόδου να επιτυγχάνεται τουλάχιστον τις μισές φορές βέλτιστη συμπεριφορά σε όλο το επεισόδιο.

Τέλος, στο σχήμα 6.9 μπορούμε να παρατηρήσουμε τον βαθμό (ποσοστό) επίσκεψης των μεθόδων σε τερματικές καταστάσεις συγκεκριμένου βάθους. Από το σχήμα αυτό φαίνεται πως οι μέθοδοι Q-Learning και TS−CS−MC εμφάνισαν τη μεγαλύτερη συγκέντρωση γύρω από τερματικές καταστάσεις με 4 πιόνια, ενώ αντίθετα η μέθοδος TS−CS−Q εμφάνισε μεγαλύτερη συγκέντρωση γύρω από τερματικές καταστάσεις με 2 πιόνια. Όμως, από το σχήμα αυτό παρατηρούμε την εξίσου μεγάλη συγκέντρωση τερματικών θέσεων με 1

πιόνι για τη μέθοδο TS−CS−MC. Έτσι, αν η εξάρτηση μεταξύ των επιθυμητών τερματικών καταστάσεων και της ανταμοιβής δεν ήταν γραμμική, όπως στην περίπτωσή μας, η μέθοδος TS−CS−MC ενδεχομένως θα μπορούσε να έχει πολύ καλύτερο μέσο όρο ανταμοιβών ανά επεισόδιο με αποτέλεσμα να παρουσιάζει πολύ καλύτερη συμπεριφορά από τη μέθοδο TS−CS−Q.

Πλήθος ενημερώσεων

Η επίλυση του προβλήματος με τη μέθοδο του Δυναμικού Προγραμματισμού στηρίχτηκε στην ιδέα η οποία αναφέρθηκε στην αρχή της παραγράφου 6.2. Όμως, προκειμένου να πραγματοποιήσουμε τα περάσματα μέσα από το χώρο των ζευγαριών $\langle s, a \rangle$ και να καταλήξουμε στη βέλτιστη συνάρτηση αποτίμησης ουσιαστικά κάναμε την παραδοχή πως γνωρίζουμε ποιες είναι όλες οι νόμιμες καταστάσεις οι οποίες μπορούν να προκύψουν στο συγκεκριμένο παιχνίδι.

Αν κάτι τέτοιο δεν ήταν γνωστό, τότε θα ήμασταν αναγκασμένοι να πραγματοποιήσουμε τα περάσματα κατά την αντίθετη κατεύθυνση, δηλαδή από καταστάσεις μικρού βάθους προς καταστάσεις μεγαλύτερου βάθους. Σε μια τέτοια περίπτωση όμως, προκειμένου να μεταφερθεί χρήσιμη πληροφορία μέχρι την κορυφή θα απαιτούνταν τουλάχιστον 31 περάσματα μέσα από το χώρο των ζευγαριών $\langle s, a \rangle$ για την ευνοϊκή περίπτωση όπου ο παράγοντας μάθησης έχει την τιμή $\alpha = 1.0$. Τότε όμως το πλήθος των ενημερώσεων το οποίο θα ήταν απαραίτητο προκειμένου να βρεθεί η βέλτιστη συνάρτηση αποτίμησης θα άγγιζε την τιμή των 727,746,000 περίπου. Από την άλλη, η συγκεκριμένη τιμή του παράγοντα μάθησης α είναι ευνοϊκή και για τις προτεινόμενες μεθόδους, όμως μόνο όσον αφορά το πλήθος επεισοδίων το οποίο θα απαιτηθεί προκειμένου να βρεθεί η βέλτιστη συνάρτηση αποτίμησης. Η τιμή αυτή της παραμέτρου δεν είναι καθόλου ευνοϊκή για σύγκριση των προτεινόμενων μεθόδων με τις μεθόδους Δυναμικού Προγραμματισμού. Παρ' όλ' αυτά, οι προτεινόμενες μέθοδοι δεν χρησιμοποίησαν σε καμία περίπτωση περισσότερες από 1,324,000,000 ενημερώσεις, τιμή συγκρίσιμη με τις μεθόδους Δυναμικού Προγραμματισμού.

Αν θέλαμε να κάνουμε μια σύγκριση των μεθόδων που προτείναμε με τις μεθόδους Δυναμικού Προγραμματισμού και να παρατηρήσουμε ποια μέθοδος απαιτεί λιγότερες σε πλήθος ενημερώσεις, τότε θα εκτελούσαμε ένα δυσκολότερο πείραμα σε μη-στατικό περιβάλλον όπου θα ήμασταν αναγκασμένοι να χρησιμοποιήσουμε για την παράμετρο α μια μικρή τιμή (π.χ. 0.1). Όμως, οι μέθοδοι Δυναμικού Προγραμματισμού σε ένα τέτοιο πείραμα θα απαιτούσαν πολλαπλά περάσματα μέσα από το χώρο αναζήτησης μιας και αποσκοπούν στην ελαχιστοποίηση του σφάλματος στη συνάρτηση αποτίμησης σε όλο το χώρο. Αντίθετα, οι μέθοδοι που προτείνουμε έχουν σαν κύριο στόχο την εύρεση μιας βέλτιστης πολιτικής χωρίς να είναι αναγκαία η ελαχιστοποίηση του μέσου τετραγωνικού σφάλματος. Χαρακτηριστικό παράδειγμα είναι η γραφική παράσταση του σχήματος 6.8 όπου ο αλγόριθμος TS−CS−MC έχει ϵ-βέλτιστη συμπεριφορά με αρκετά μεγάλο σφάλμα στη συνάρτηση αποτίμησης. Σε τέτοιες περιπτώσεις επομένως, οι μέθοδοι που προτείνουμε, καθώς επίσης και

σχεδόν όλες οι υπόλοιπες μέθοδοι στην περιοχή της Ενισχυτικής Μάθησης, θα απαιτούσαν πολύ λιγότερες σε πλήθος ενημερώσεις προκειμένου να βρουν μια βέλτιστη πολιτική σε σχέση με τις μεθόδους Δυναμικού Προγραμματισμού.

6.3 Επεκτάσεις και σκέψεις για το μέλλον.

Συμπερασματικά, οι προτεινόμενες μέθοδοι δείχνουν να είναι χρήσιμες εναλλακτικές προτάσεις μεθόδων μάθησης στη συγκεκριμένη περιοχή. Αν και τα αποτελέσματα ήταν αρκετά καλά στη συγκεκριμένη εφαρμογή σίγουρα θα είχε μεγάλο ενδιαφέρον η περαιτέρω εφαρμογή των μεθόδων και σε άλλους τομείς, οπότε και να έχουμε μια πιο ολοκληρωμένη εικόνα των επιδόσεών τους.

Στην παρουσίαση που έχει γίνει έως τώρα έχουμε υποθέσει πως ο πράκτορας γνωρίζει τη δυναμική του περιβάλλοντος ακόμα και για ενέργειες τις οποίες δεν τις έχει εφαρμόσει ποτέ στην πράξη. Κάτι τέτοιο είναι πλήρως αποδεκτό για ορισμένες εφαρμογές και ιδιαίτερα για παιχνίδια, όπου ένας παίκτης ακόμα κι αν δεν έχει επισκεφθεί ορισμένες καταστάσεις, εντούτοις γνωρίζει τα αποτελέσματα διαφόρων ενεργειών που έχει στη διάθεσή του. Από την άλλη, σε αρκετές περιπτώσεις, ιδιαίτερα σε μη-στατικά περιβάλλοντα, ο πράκτορας δεν έχει τέτοιες δυνατότητες και μαθαίνει τη συμπεριφορά του περιβάλλοντος μέσα από την αλληλεπίδραση που έχει με αυτό. Έτσι, θα είχε αρκετά μεγάλο ενδιαφέρον να δούμε τη συμπεριφορά των συγκεκριμένων αλγορίθμων σε αυτά τα περιβάλλοντα. Επιπλέον, μέσα από αυτή τη διαδικασία μάθησης της δυναμικής του περιβάλλοντος, οι προτεινόμενες μέθοδοι θα διέδιδαν εκτιμήσεις της συνάρτησης αποτίμησης σε περιοχές πολύ κοντινές στην αρχική θέση του πράκτορα από τα πρώτα κιόλας επεισόδια, με αποτέλεσμα να κατευθύνουν την αναζήτηση των μεθόδων από πολύ πρώιμα στάδια μάθησης. Βέβαια, εξ'αιτίας των βεβιασμένων ενημερώσεων που θα πραγματοποιούνται σε τέτοιες περιπτώσεις, είναι αναγκαία η μικρή τιμή του παράγοντα μάθησης α καθώς επίσης και μια μικρή τιμή πιθανότητας εξερεύνησης ώστε ο πράκτορας να μην κολλάει σε τοπικά βέλτιστες (ή τοπικά ϵ-βέλτιστες) λύσεις.

Τέλος, σε μη-στατικά περιβάλλοντα μπορούμε εύκολα να κατασκευάσουμε αντιπαραδείγματα στα οποία οι προτεινόμενες μέθοδοι με τη μορφή που παρουσιάστηκαν δεν θα παρουσιάζουν καλή συμπεριφορά. Όπως έχουν δείξει διάφορα πειράματα (π.χ. στο [26]) η δειγματολήπτηση μονοπατιών είναι εξαιρετικά χρήσιμη σε πρώιμα στάδια μάθησης αλλά μακροπρόθεσμα οι τυχαίες προσομοιωμένες ενημερώσεις πάνω στο χώρο αναζήτησης έχουν καλύτερα αποτελέσματα. Από τα πειράματα αυτά γίνεται φανερό πως καλύτερα αποτελέσματα μπορούν να επιτευχθούν για διάφορες μεθόδους εάν οι τελευταίες εμπεριέχουν σε ένα μικρό βαθμό ορισμένες τυχαίες ενημερώσεις πάνω στο χώρο αναζήτησης. Επιπλέον, μπορούμε να εμπλουτίσουμε τις μεθόδους που προτείνουμε με χρήση μιας ακόμη ιδέας η οποία πηγάζει από τους αλγορίθμους περασμάτων προτεραιότητας. Σύμφωνα με αυτή την ιδέα, μπορούμε να διατηρούμε μια επιπλέον λίστα ζευγαριών $\langle s, a \rangle$ για τα οποία οι μεταβολές στη συνάρτηση αποτίμησης

ήταν αρκετά μεγάλες κατά τις πρόσφατες ενημερώσεις του πράκτορα. Έτσι, ένα μικρό ποσοστό του πλήθους των προσομοιωμένων ενημερώσεων μπορούμε να το αφιερώσουμε σε ενημερώσεις πάνω σε αυτά τα ζευγάρια προκειμένου να παρακολουθούμε ταχύτερα τις όποιες μεταβολές στη συνάρτηση αποτίμησης, η οποία είναι απόρροια της μεταβολής της δυναμικής του περιβάλλοντος.

Οι σχεδιαστές συστημάτων ενισχυτικής μάθησης που θα χρησιμοποιήσουν τις συγκεκριμένες μεθόδους θα ήταν ιδιαίτερα κρίσιμο να λάβουν σοβαρά υπ' όψιν τους τα ιδιαίτερα χαρακτηριστικά των μεθόδων και τα σχόλια τα οποία έχουν γίνει γι'αυτές. Θα ήταν ιδιαίτερα ενδιαφέρον να δούμε εφαρμογή των μεθόδων σε δυσκολότερα προβλήματα μάθησης καθώς επίσης και εφαρμογή των μεθόδων αυτών με χρήση νευρωνικών δικτύων. Σίγουρα, σε μελλοντικές εφαρμογές θα πρέπει να δούμε τη συμπεριφορά των συγκεκριμένων μεθόδων συγκριτικά και με άλλες μεθόδους μάθησης που χρησιμοποιούνται αυτή τη στιγμή στη συγκεκριμένη περιοχή. Ότι κι αν γίνει όμως από τα παραπάνω, οι εμπνευστές των συγκεκριμένων μεθόδων είναι αισιόδοξοι πως τα αποτελέσματα τα οποία θα προκύψουν θα είναι αντίστοιχα σε ποιότητα με αυτά τα οποία προέκυψαν από την παρούσα εφαρμογή.

Βιβλιογραφία

[1] Ι. Βλαχάβας και άλλοι. *Τεχνητή Νοημοσύνη*. Εκδόσεις Γαρταγάνη, 1η έκδοση, 2002.

[2] Leemon Baird. *Residual Algorithms: Reinforcement Learning with Function Approximation*. Dept. of Computer Science, U.S. Air Force Academy, CO 80840-6234, 1995. Επικοινωνία: baird@cs.usafa.af.mil ή επισκεφθείτε: http://kirk.usafa.af.mil/~baird.

[3] A.G. Barto, S.J. Bradtke και S.P. Singh. Learning to act using real-time dynamic programming. *Artificial Intelligence*, 72:81–138, 1995.

[4] Dimitri P. Bertsekas. A counterexample to temporal differences learning. *Neural Computation*, 7, 1995.

[5] Dimitri P. Bertsekas. Differential training of rollout policies. Στο *35th Allerton Conference on Communication, Control, and Computing*, Οκτώβριος 1997.

[6] Dimitri P. Bertsekas και Sergey Ioffe. *Temporal Differences-Based Policy Iteration and Applications in Neuro-Dynamic Programming*. Dept. of Electrical Engineering and Computer Science, M.I.T., Cambridge, Mass. 02139, Αύγουστος 1997.

[7] Dimitri P. Bertsekas και John N. Tsitsiklis. *Neuro-Dynamic Programming*. Optimization and Computation series. Athena Scientific, 2η έκδοση, 1996.

[8] Dimitri P. Bertsekas, John N. Tsitsiklis και Cynara Wu. *Rollout Algorithms for Combinatorial Optimazation*. Dept. of Electrical Engineering and Computer Science, M.I.T., Cambridge, Mass. 02139, Απρίλιος (revision: Ιούνιος) 1997.

[9] George Boukeas. *Learning and Adaptation in Multiagent Systems*. Imperial College of Science, Technology and Medicine, Dept. of Computing, 2000. Επικοινωνία: boukeas@di.uoa.gr.

[10] Justin A. Boyan και Andrew W. Moore. *Learning Evaluation Functions for Global Optimization and Boolean Satisfiability.* Computer Science Department - Carnegie Mellon University, Pittsburgh, PA 15213, 1998. Επικοινωνία: jab@cs.cmu.edu και awm@cs.cmu.edu.

[11] Ronen Brafman και Moshe Tennenholtz. R-max - a general polynomial time algorithm for near-optimal reinforcement learning. *Journal of Machine Learning Research*, 3:213–231, 2002. Επικοινωνία: BRAFMAN@CS.BGU.AC.IL και MOSHE@ROBOTICS.STANFORD.EDU.

[12] Scott Davies, Andrew Y. Ng και Andrew Moore. *Applying Online Search Techniques to Continuous-State Reinforcement Learning.* School of Computer Science - Carnegie-Mellon University and Artificial Intelligence Lab - M.I.T., Pittsburgh, PA 15213 and Cambridge, MA 02139, 1998.

[13] Scott Davies, Andrew Y. Ng και Andrew Moore. *Applying Online Search Techniques to Reinforcement Learning.* School of Computer Science, Carnegie-Mellon University, Pittsburgh, PA 15213, 1998. Επικοινωνία: Firstname.Lastname@cs.cmu.edu.

[14] Matt Ginsberg. *Essentials of Artificial Intelligence.* Morgan Kaufmann, 1993.

[15] Greg Grudic και Lyle Ungar. Localizing search in reinforcement learning. Στο *Seventeenth National Conference on Artificial Intelligence, July 30 - August 3, 2000, Austin, Texas*, Institute for Research in Cognitive Science/Computer and Information Science, University of Pennsylvania, Philadelphia, PA USA, 2001. American Association for Artificial Intelligence. Επικοινωνία: grudic@linc.cis.upenn.edu και ungar@cis.upenn.edu.

[16] Eric A. Hansen και Shlomo Zilberstein. *A Heuristic Search Algorithm for Markov Decision Problems.* C.S. Dept. - Mississippi State University and C.S. Dept - University of Massachusetts, 1999. Επικοινωνία: hansen@cs.msstate.edu και shlomo@cs.umass.edu.

[17] Mance H. Harmon και Stephanie S. Harmon. *Reinforcement Learning: A Tutorial.* Wright State University, 156-8 Mallard Glen Drive, Centerville, OH 45458, χ.χ. Επικοινωνία: mharmon@acm.org.

[18] Leslie Pack Kaebling, Michael L. Littman και Andrew W. Moore. Reinforcement learning: A survey. *Journal of Artificial Intelligence Research*, 4:237–285, 1996.

[19] Sven Koenig. *Minimax Real-Time Heuristic Search.* College of Computing, Georgia Institute of Technology, Atlanta, Georgia 30332-0280, 2001.

[20] Michael L. Littman. *Markov games as a framework for multi-agent reinforcement learning*. Dept. of Computer Science - Brown University, Providence, RI 02912-1910, 1994. Επικοινωνία: mlittman@cs.brown.edu.

[21] Pattie Maes και Rodney A. Brooks. *Learning to Coordinate Behaviours*. AI-Laboratory, Massachusetts Institute of Technology, 545 Technology Square, Cambridge, MA 02139, 1990. Επικοινωνία: pattie@ai.mit.edu και brooks@ai.mit.edu.

[22] Jing Peng και Ronald J. Williams. *Efficient Learning and Planning Within the Dyna Framework*. College of Computer Science - Northeastern University, Boston, MA 02115 USA, 1993. Επικοινωνία: jp@ccs.neu.edu και rjw@ccs.neu.edu.

[23] Jing Peng και Ronald J. Williams. *Incremental Multi-Step Q-Learning*. College of Computer Science - Northeastern University, Boston, MA 02115 USA, 1996. Επικοινωνία: jp@ccs.neu.edu και rjw@ccs.neu.edu.

[24] Richard S. Sutton. Learning to predict by the methods of temporal differences. *Machine Learning*, 3:9–44, 1988.

[25] Richard S. Sutton. Generalization in reinforcement learning: Successful examples using sparse coarse coding. *Advances in Neural Information Processing Systems*, 8:1038–1044, 1996. Επικοινωνία: rich@cs.umass.edu.

[26] Richard S. Sutton και Andrew G. Barto. *Reinforcement Learning: An Introduction*. Adaptive Computation and Machine Learning series. MIT Press, 4η έκδοση, 2002.