ε.δε.μ²
ΕΠΙΣΤΗΜΗ ΔΕΔΟΜΕΝΩΝ & ΜΗΧΑΝΙΚΗ ΜΑΘΗΣΗ

ΣΤΟΧΑΣΤΙΚΕΣ ΔΙΕΡΓΑΣΙΕΣ & ΒΕΛΤΙΣΤΟΠΟΙΗΣΗ ΣΤΗ ΜΗΧΑΝΙΚΗ ΜΑΘΗΣΗ

Ενισχυτική Μάθηση - Δυναμικός Προγραμματισμός:

1. Markov Decision Processes

2. Bellman's Optimality Criterion

3. Αλγόριθμος Policy Iteration

4. Αλγόριθμος Value Iteration

καθ. Βασίλης Μάγκλαρης
maglaris@netmode.ntua.gr
www.netmode.ntua.gr

Αίθουσα 002, Νέα Κτίρια ΣΗΜΜΥ
Τρίτη 8/4/2024

ΣΤΟΧΑΣΤΙΚΕΣ ΔΙΑΔΙΚΑΣΙΕΣ & ΒΕΛΤΙΣΤΟΠΟΙΗΣΗ ΣΤΗ ΜΗΧΑΝΙΚΗ ΜΑΘΗΣΗ

Reinforcement Learning - Markov Decision Processes

Supervised Learning - Teacher

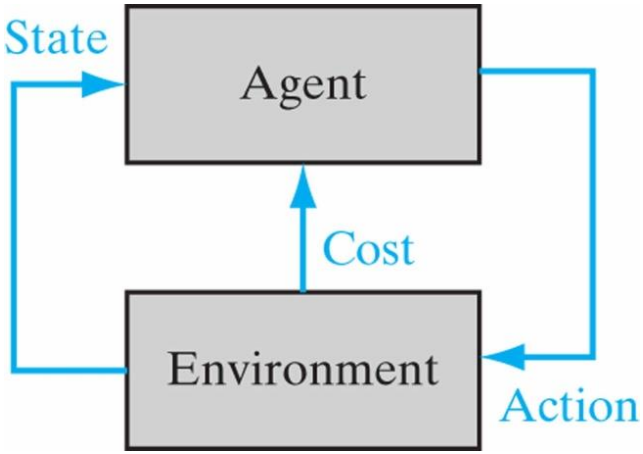
Βέλτιστη ρύθμιση του συστήματος με βοήθεια εξωτερικού δασκάλου, γνώστη επιθυμητών εξόδων (***labels***) δεδομένων ***προϋπάρχοντος*** δείγματος μάθησης (***labeled*** training examples)

Unsupervised Learning

Αυτορυθμιζόμενο σύστημα που κωδικοποιεί στις παραμέτρους του ιδιότητες ***προϋπάρχοντος*** δείγματος μάθησης (***unlabeled*** training examples) ελπίζοντας σε γενίκευση με δεδομένα test

Reinforcement Learning - Agent

- Αποφάσεις (***actions***) από ***agent*** σε ορίζοντα K βημάτων που επηρεάζουν την εξέλιξη καταστάσεων (***states***) περιβάλλοντος (***environment***) με συνεπαγόμενο κόστος/όφελος
- Σχεδιασμός πολιτικής (***policy planning***) καταστάσεων - αποφάσεων (***states*** - ***actions***) του ***agent*** για μέσο - μακροπρόθεσμο στόχο μέσω ***διαδραστικού*** σεναρίου μάθησης
- Εργαλεία: Δυναμικός Προγραμματισμός (***Dynamic Programming***), Στοχαστικές Διαδικασίες Αποφάσεων Markov (***Markov Decision Processes***)
- Δείγμα μάθησης: ***Training dataset*** όχι αναγκαστικά ***προϋπάρχον***, δυνατότητα on-line προσαρμογής μάθησης/λειτουργίας από μεταβολή κατάστασης λόγω αποφάσεων του ***agent***



- Τα Παραδείγματα Μάθησης (Training Examples, Sample Points) σε ***Supervised*** & ***Unsupervised Learning*** *συνήθως* προσεγγίζονται σαν ***ανεξάρτητες*** τυχαίες μεταβλητές σε πολυπληθή δείγματα
- Στο ***Reinforcement Learning*** η μάθηση *συνήθως* βασίζεται σε δυναμικά σενάρια εξέλιξης, με εξαρτήσεις καταστάσεων ***Markov*** του ***Περιβάλλοντος*** από πολιτικές του ***Agent***

ΣΤΟΧΑΣΤΙΚΕΣ ΔΙΑΔΙΚΑΣΙΕΣ & ΒΕΛΤΙΣΤΟΠΟΙΗΣΗ ΣΤΗ ΜΗΧΑΝΙΚΗ ΜΑΘΗΣΗ

Reinforcement Learning - Markov Decision Processes (1/2)

Ορισμοί Markov Decision Processes

- Πεπερασμένος Δειγματικός Χώρος $\mathcal{X}$ διακριτών καταστάσεων (***states***) περιβάλλοντος σε διακριτές χρονικές στιγμές (βήματα) $n = 0,1,2, \dots, K$

  Τυχαία μεταβλητή $X_n \in \mathcal{X}$ που λαμβάνει διακριτές τιμές $X_n = i,\ 1 \le i \le N$

- Πεπερασμένος Δειγματικός Χώρος $\mathcal{A}_i$ διακριτών αποφάσεων (***actions***) που ορίζει ο ***agent*** όταν το περιβάλλον βρίσκεται στη κατάσταση $X_n = i$:

  Τυχαία μεταβλητή $A_n \in \mathcal{A}_i$ απόφασης στο n, με τιμές a_{ik} όταν $X_n = i$

- Μεταβάσεις ***Markov*** $p_{ij}(a)$ από κατάσταση περιβάλλοντος i σε κατάσταση j υπό την επήρεια της απόφασης του ***agent*** a στα διακριτά βήματα $n = 0,1,2, \dots, K$

  $p_{ij}(a) = P(X_{n+1} = j | X_n = i, A_n = a),\ p_{ij}(a) \ge 0,\ \sum_j p_{ij}(a) = 1$

- Άμεσο κόστος (***observed cost***) του ***agent*** στο βήμα n όταν παίρνει απόφαση a_{ik} που οδηγεί σε μετάβαση $(X_n = i) \to (X_{n+1} = j)$:

  $g(i, a_{ik}, j)$ και με ***απόσβεση*** $\gamma^n g(i, a_{ik}, j)$ με συντελεστή $0 \le \gamma < 1$ (***discount factor***)

  ✓ Αν $\gamma = 0$ ο ***agent*** δεν ενδιαφέρεται για μελλοντικές επιπτώσεις αποφάσεών του (***myopic***)

  ✓ Όσο $\gamma \to 1$ οι αποφάσεις του ***agent*** καθορίζονται σημαντικά από μελλοντικές επιπτώσεις

- Πολιτική (***policy***): $\pi = \{\mu_0, \mu_1, \dots, \mu_n, \dots, \mu_{K-1}\}$ όπου μ_n συνάρτηση που στο βήμα n ορίζει αντιστοίχηση μιας κατάστασης $X_n = i$ σε αποφάσεις $A_n = a$ του ***agent***

  $\mu_n(i) \in \mathcal{A}_i$ για όλες τις καταστάσεις $i \in \mathcal{X}$ (π ***admissible policies***)

Αν $\mu_n(i) = \mu(i)$ σε κάθε βήμα n η πολιτική $\pi = \{\mu, \mu, \dots\}$ είναι χρονοσταθερή (***stationary***) και οι μεταβάσεις $p_{ij}(a)$ ορίζουν (χρονοσταθερή) ***αλυσίδα Markov*** $(X_n = i) \to (X_{n+1} = j)$

ΣΤΟΧΑΣΤΙΚΕΣ ΔΙΑΔΙΚΑΣΙΕΣ & ΒΕΛΤΙΣΤΟΠΟΙΗΣΗ ΣΤΗ ΜΗΧΑΝΙΚΗ ΜΑΘΗΣΗ

Reinforcement Learning - Markov Decision Processes (2/2)

Ορισμοί Βελτιστοποίησης Δυναμικού Προγραμματισμού

Το συνολικό κόστος εκτιμάται σε ***πιθανές τροχιές*** (***trajectories***) πεπερασμένων βημάτων K (***Finite-Horizon***) με (επαναλαμβανόμενα) επεισόδια (***episodes***) ή απεριόριστων $K \to \infty$ (***Infinite-Horizon***) αθροίζοντας τα ***άμεσα κόστη μεταβάσεων Markov*** $X_n \to X_{n+1}$ λόγω $\mu_n(X_n)$:

$$g(X_n, \mu_n(X_n), X_{n+1})$$

Το συνολικό αναμενόμενο κόστος (***Total Discounted Expected Cost-to-Go***) σε ορίζοντα K και με πολιτική $\pi = \{\mu_0, \mu_1, \dots, \mu_{K-1}\}$ από αρχική κατάσταση $X_0 = i$ και απόσβεση γ είναι:

$$J^{\pi}(i) = \mathrm{E}\left[\sum_{n=0}^{K} \gamma^n g(X_n, \mu_n(X_n), X_{n+1}) | X_0 = i\right]$$

Ζητείται πολιτική π ελαχιστοποίησης του $J^{\pi}(i)$: $J^*(i) \triangleq \min_{\pi} J^{\pi}(i)$

Η ανωτέρω πολιτική π είναι άπληστη (***greedy***) με την έννοια του ότι ο ***agent*** επιλέγει αποφάσεις που ελαχιστοποιούν το ***Expected Cost-to-Go*** $J^{\pi}(i)$ από αρχική κατάσταση $X_0 = i$ αδιαφορώντας για πιθανά καλύτερες εναλλακτικές τροχιές

> Αν η πολιτική περιορίζεται σε χρονοσταθερές αποφάσεις $\pi = \{\mu, \mu, \dots\}$ τότε $J^{\pi}(i) \triangleq J^{\mu}(i)$ και το τελικό ζητούμενο είναι η βέλτιστη συνάρτηση $\mu(X_n)$ που ελαχιστοποιεί τα $J^{\mu}(i) = J^*(i)$ για όλες τις αρχικές καταστάσεις $X_0 = i$

<u>Σημείωση</u>: Εναλλακτικά με το κριτήριο ***Total Discounted Expected Cost-to-Go*** μπορεί να οριστεί κριτήριο χωρίς απόσβεση π.χ. ***Expected Average Cost*** ανά βήμα σε ***Infinite Horizon*** (Sheldon Ross, "**Applied Probability Models with Optimization**", Dover, 1992)

ΣΤΟΧΑΣΤΙΚΕΣ ΔΙΑΔΙΚΑΣΙΕΣ & ΒΕΛΤΙΣΤΟΠΟΙΗΣΗ ΣΤΗ ΜΗΧΑΝΙΚΗ ΜΑΘΗΣΗ

Principle of Optimality (*Bellman 1957*) – Finite Horizon Problem

Έστω διαδικασία αποφάσεων Markov σε ορίζοντα πεπερασμένων βημάτων $n \leq K$ με κόστη $g_n(X_n, \mu_n(X_n), X_{n+1}) \triangleq \gamma^n g(X_n, \mu_n(X_n), X_{n+1}), n < K$ και κόστος τερματικής κατάστασης $g_K(X_K)$. Το ***Expected Cost-to-Go*** σε K βήματα και αναμενόμενες τροχιές $\{X_0, X_1, \dots, X_K\}$ είναι:

$$J_0(X_0) = \mathrm{E}\left[\left\{g_K(X_K) + \sum_{n=0}^{K-1} g_n(X_n, \mu_k(X_n), X_{n+1})\right\} \mid X_0\right]$$

Μια βέλτιστη πολιτική $\pi^* = \{\mu_0^*, \mu_1^*, \mu_2^*, \dots, \mu_{K-1}^*\}$ οδηγεί το περιβάλλον μετά από n βήματα, $n < K$ σε τροχιά καταστάσεων $\{X_0, X_1, \dots, X_n\}$. Το υπολειπόμενο ***Expected Cost-to-Go*** για $\{X_{n+1}, X_{n+2}, \dots, X_K\}$ είναι:

$$J_n(X_n) = \mathrm{E}\left[\left\{g_K(X_K) + \sum_{k=n}^{K-1} g_k(X_k, \mu_k(X_k), X_{k+1})\right\} \mid X_n\right]$$

Τότε η περικομμένη (***truncated***) πολιτική $\{\mu_n^*, \mu_{n+1}^*, \dots, \mu_{K-1}^*\}$ είναι βέλτιστη για την υπολειπόμενη διαδικασία $\{X_{n+1}, X_{n+2}, \dots, X_K\}$ με κατάσταση εκκίνησης την X_n (***Principle of Optimality***)

Ερμηνεία: Αν η περικομμένη πολιτική δεν ήταν βέλτιστη, τότε μόλις η συνολικά βέλτιστη πολιτική π^* οδηγούσε το περιβάλλον στη κατάσταση X_n ο agent θα μπορούσε να αλλάξει πολιτική για τα υπολειπόμενα βήματα $\{n+1, n+2, \dots, K\}$ και θα πετύχαινε μικρότερο συνολικό κόστος από την π^*

ΣΤΟΧΑΣΤΙΚΕΣ ΔΙΑΔΙΚΑΣΙΕΣ & ΒΕΛΤΙΣΤΟΠΟΙΗΣΗ ΣΤΗ ΜΗΧΑΝΙΚΗ ΜΑΘΗΣΗ

Δυναμικός Προγραμματισμός (*Bellman 1957*) – Finite Horizon Problem

Το ***Bellman Principle of Optimality*** οδηγεί σε αλγόριθμο **Δυναμικού Προγραμματισμού** (***Dynamic Programming***) προσδιορισμού **Βέλτιστης Πολιτικής** $\pi^* = \{\mu_0^*, \mu_1^*, \mu_2^*, \dots, \mu_{K-1}^*\}$ σε τρία στάδια με ανάστροφη χρονική σειρά $K \to (K-1) \to (K-2) \to \cdots \to 1 \to 0$

1. Εύρεση βέλτιστης πολιτικής μ_{K-1}^* για το τελικό βήμα $X_{K-1} \to X_K$
2. Για τα δύο τελικά βήματα $X_{K-2} \to X_{K-1} \to X_K$ εύρεση της μ_{K-2}^* με αναλλοίωτη την μ_{K-1}^*
3. Επανάληψη μέχρι το βήμα $n = 0$ και προσδιορισμός της μ_0^* που συμπληρώνει την π^*

Αλγόριθμος Δυναμικού Προγραμματισμού

1. Εκκίνηση με $J_K(X_K) = g_K(X_K)$ για όλες τις τελικές καταστάσεις X_K
2. Για $n = \{K-1, K-2, \dots, 1, 0\}$ υπολογίζουμε αναδρομικά τα υπολειπόμενα ***Expected Cost-to-Go*** $J_n(X_n)$ για όλες τις καταστάσεις X_n και τις αποφάσεις $\mu_n(X_n)$ με τον ***Αναδρομικό Τύπο*** άπληστων (***greedy***) αποφάσεων:

$$J_n(X_n) = \min_{\mu_n(X_n)} \mathrm{E}[g_n(X_n, \mu_n(X_n), X_{n+1}) + J_{n+1}(X_{n+1})]$$

   Η μέση τιμή στον αναδρομικό τύπο αφορά σε όλες τις πιθανές καταστάσεις X_{n+1}
3. Τελικός προσδιορισμός των $J_0(X_0)$ για όλες τις αρχικές καταστάσεις X_0 και της βέλτιστης $\pi^* = \{\mu_0^*, \mu_1^*, \dots, \mu_{K-1}^*\}$ των αποφάσεων μ_n^* που ελαχιστοποιούν τον αναδρομικό τύπο
4. Για χρονοσταθερές πολιτικές $\pi = \{\mu, \mu, \dots\}$ ο αναδρομικός τύπος απλοποιείται με $\mu_n = \mu$

ΣΤΟΧΑΣΤΙΚΕΣ ΔΙΑΔΙΚΑΣΙΕΣ & ΒΕΛΤΙΣΤΟΠΟΙΗΣΗ ΣΤΗ ΜΗΧΑΝΙΚΗ ΜΑΘΗΣΗ

Optimality Equations – Infinite Horizon Problem, Stationary Policy

- Έστω διαδικασία αποφάσεων ***Markov*** πεπερασμένων καταστάσεων $X_n \in \{1,2,\dots,N\}$ σε άπειρο ορίζοντα βημάτων $n = 0,1,2,\dots$ με **χρονοσταθερές** πολιτικές $\pi = \{\mu,\mu,\dots\}$, απόσβεση $0 < \gamma < 1$, κόστη $g_n(X_n,\mu(X_n),X_{n+1}) \triangleq \gamma^n g(X_n,\mu(X_n),X_{n+1})$ και αρχική κατάσταση $X_0 \to X_1$. Ζητείται η π για ελάχιστο ***Expected Cost over Infinite Horizon***
- Με επαναδιατύπωση του ***Αναδρομικού Τύπου Δυναμικού Προγραμματισμού*** και ***αναστροφή της χρονικής εξέλιξης*** έχουμε για εκκίνηση X_0 και πεπερασμένο ορίζοντα $n \le K$
$J_{n+1}(X_0) = \min_{\mu} \mathrm{E}[(g(X_0,\mu(X_0),X_1) + \gamma J_n(X_1))|X_0]$ με αρχική συνθήκη $J_0(X) = 0, \forall X$
- Για άπειρο ορίζοντας και $X_0 = i$ η βέλτιστη πολιτική π δίνει κόστη $J^*(i) = \lim_{K\to\infty} J_K(i)\,, \forall i \Rightarrow$
$$J^*(i) = \min_{\pi} \mathrm{E}[(g(i,\mu(i),X_1) + \gamma J^*(X_1))|X_0 = i]$$
- Ορίζουμε $c\big(i,\mu(i)\big)$ το ***άμεσο αναμενόμενο κόστος*** (***Immediate Expected Cost***) κατάστασης $X_0 = i$ όταν λαμβάνεται η απόφαση $\mu(i)$ και η μέση τιμή αφορά στις καταστάσεις $X_1 = j$:
$$c\big(i,\mu(i)\big) \triangleq \mathrm{E}[g(i,\mu(i),X_1 = j)|X_0 = i] = \sum_{j=1}^{N} p_{ij}(\mu(i))g(i,\mu(i),j)$$
- Η βέλτιστη μ έχει αναμενόμενο κόστος σε 1^{o} βήμα $\mathrm{E}[J^*(X_1)|X_0 = i] = \sum_{j=1}^{N} p_{ij}(\mu)J^*(j)$

Προκύπτουν οι N εξισώσεις βελτιστοποίησης του ***Bellman*** (***Bellman's Optimality Equations***):
$$J^*(i) = \min_{\mu}\left(c\big(i,\mu(i)\big) + \gamma\sum_{j=1}^{N} p_{ij}(\mu(i))J^*(j)\right), i = 1,2,\dots,N$$
Οι N εξισώσεις προσδιορίζουν τα βέλτιστα $J^*(i)$ και τη βέλτιστη πολιτική μ μέσω αλγορίθμων ***Policy Iteration*** ή ***Value Iteration*** (με ***γνώση*** των $p_{ij}(a)$ για ***Model-based learning***)

ΣΤΟΧΑΣΤΙΚΕΣ ΔΙΑΔΙΚΑΣΙΕΣ & ΒΕΛΤΙΣΤΟΠΟΙΗΣΗ ΣΤΗ ΜΗΧΑΝΙΚΗ ΜΑΘΗΣΗ

Model-based Learning: Αλγόριθμος Policy Iteration (1/2)

Ορισμός Q-factor

- Έστω χρονοσταθερή πολιτική $\pi = \{\mu, \mu, \dots\}$ που οδηγεί σε αναμενόμενα ***costs-to-go*** $J^{\mu}(i), \forall i \in \mathcal{X}$ (καταστάσεις του ***περιβάλλοντος***) με αποφάσεις του ***agent*** $a = \mu(i) \in \mathcal{A}_i$
- Για κάθε ζεύγος (i, a) στο υπό εξέταση βήμα και πολιτική για τα υπολειπόμενα βήματα $\pi = \{\mu, \mu, \dots\}$ ορίζω τους ***Q-factors*** $Q^{\mu}(i, a)$ σαν μέτρο κατάταξης εναλλακτικών άμεσων αποφάσεων $a \in \mathcal{A}_i$ του ***agent*** που θα οδηγούσαν το ***περιβάλλον*** από παρούσα κατάσταση i σε κατάσταση j με αναμενόμενα υπολειπόμενα ***costs-to-go*** $J^{\mu}(j), \forall j \in \mathcal{X}$

$$Q^{\mu}(i, a) \triangleq c(i, a) + \gamma \sum_{j=1}^{N} p_{ij}(a)\, J^{\mu}(j)$$

- Μια πολιτική $\pi = \{\mu, \mu, \dots\}$ ικανοποιεί τις συνθήκες απληστίας (***greedy conditions***) σε σχέση με τα αναμενόμενα ***costs-to-go*** $J^{\mu}(j)$ στα υπολειπόμενα βήματα όταν σε κάθε βήμα και $\forall i \in \mathcal{X}$ ο ***agent*** επιλέγει $a = \mu(i)$ ώστε

$$Q^{\mu}(i, \mu(i)) = \min_{a \in \mathcal{A}_i} Q^{\mu}(i, a), \forall i \in \mathcal{X}$$

- Μια πολιτική $\pi^* = \{\mu^*, \mu^*, \dots\}$ είναι βέλτιστη για όλα τα βήματα αν ικανοποιεί τις συνθήκες απληστίας (***greedy conditions***) του δυναμικού προγραμματισμού:

$$Q^{\mu^*}(i, \mu^*(i)) = \min_{a \in \mathcal{A}_i} Q^{\mu^*}(i, a)$$

Σημείωση: Όταν τα άμεσα αναμενόμενα κόστη $c(i, a)$ αντικαθίστανται από ***rewards*** $r(i, a)$, τα ***costs-to-go*** $J^{\mu}(i)$ αποκαλούνται ***Value Functions*** $V^{\mu}(i)$ και έχουμε κατ' αντιστοιχία:

$$Q^{\mu}(i, a) \triangleq r(i, a) + \gamma \sum_{j=1}^{N} p_{ij}(a)\, V^{\mu}(j) \text{ και } Q^{\mu^*}(i, \mu^*(i)) = \max_{a \in \mathcal{A}_i} Q^{\mu^*}(i, a)$$

ΣΤΟΧΑΣΤΙΚΕΣ ΔΙΑΔΙΚΑΣΙΕΣ & ΒΕΛΤΙΣΤΟΠΟΙΗΣΗ ΣΤΗ ΜΗΧΑΝΙΚΗ ΜΑΘΗΣΗ

Model-based Learning: Αλγόριθμος Policy Iteration (2/2)

Αρχιτεκτονική *Actor – Critic*

(***A.G. Barto***, ***R.S. Sutton*** & ***C.W. Anderson***, "Neuronlike adaptive elements that can solve difficult learning control problems," IEEE Transactions on Systems, Man, and Cybernetics, vol. SMC-13, Sept. – Oct. 1983)

Επαναλήψεις $n = 0,1,2, \ldots$ από δύο βήματα μέχρι $\mu_{n+1}(i) = \mu_n(i),\ J^{\mu_{n+1}}(i) = J^{\mu_n}(i)\ , \forall i$

Βήμα 1. **Policy Evaluation** (ο ***critic*** αναλύει τις αποφάσεις του ***agent***):

Με βάση την παρούσα πολιτική $\pi_n = \{\mu_n, \mu_n, \ldots\}$ υπολογίζονται τα ***costs-to-go***

$J^{\mu_n}(i) = c\big(i, \mu_n(i)\big) + \gamma \sum_{j=1}^{N} p_{ij}(\mu_n(i))\, J^{\mu_n}(j)$ για $\forall i$

Για $\forall i$ και $\forall a \in \mathcal{A}_i$ υπολογίζονται τα ***Q-factors***: $Q^{\mu_n}(i,a) = c(i,a) + \gamma \sum_{j=1}^{N} p_{ij}(a)\, J^{\mu_n}(j)$

Βήμα 2. **Policy Improvement** (ο ***actor*** καθοδηγεί τις αποφάσεις του ***agent***):

Η πολιτική π_n βελτιώνεται σε π_{n+1} μέσω της $\mu_{n+1}(i) = \arg\min_{a \in \mathcal{A}_i} Q^{\mu_n}(i,a)$ για $i = 1,2, \ldots, N$

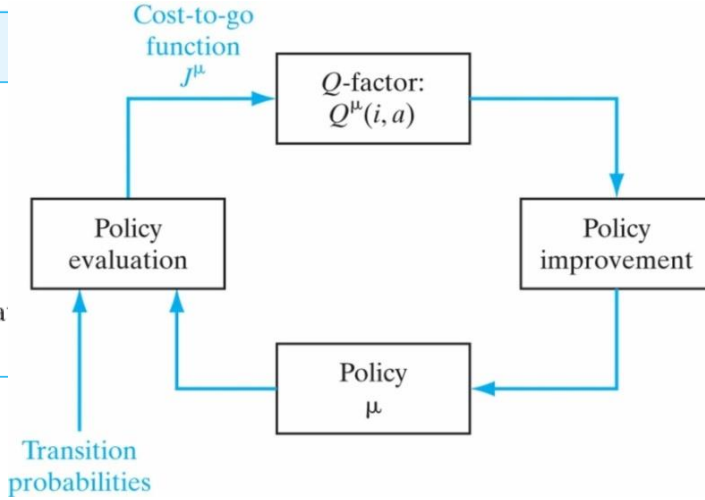
TABLE 12.1 Summary of the Policy Iteration Algorithm

1. Start with an arbitrary initial policy μ_0.
2. For $n = 0, 1, 2, \ldots$, compute $J^{\mu_n}(i)$ and $Q^{\mu_n}(i, a)$ for all states $i \in \mathcal{X}$ and actions $a \in \mathcal{A}_i$.
3. For each state i, compute

$$\mu_{n+1}(i) = \arg\min_{a \in \mathcal{A}_i} Q^{\mu_n}(i, a)$$

4. Repeat steps 2 and 3 until μ_{n+1} is not an improvement on μ_n, at which point the algorithm termina with μ_n as the desired policy.

$\arg\min_x f(x)$: Η τιμή της x που οδηγεί την $f(x)$ σε ελάχιστο



Ο αλγόριθμος συγκλίνει σε βέλτιστη πολιτική σε πεπερασμένα βήματα n λόγω πεπερασμένου πλήθους καταστάσεων N και πεπερασμένων επιλογών αποφάσεων

ΣΤΟΧΑΣΤΙΚΕΣ ΔΙΑΔΙΚΑΣΙΕΣ & ΒΕΛΤΙΣΤΟΠΟΙΗΣΗ ΣΤΗ ΜΗΧΑΝΙΚΗ ΜΑΘΗΣΗ

Model-based Learning: Value Iteration Algorithm

Εκτίμηση των Συναρτήσεων Cost-to-Go μέσω Διαδοχικών Προσεγγίσεων $J_n(i) \to J_{n+1}(i)$

- Εκκίνηση με αυθαίρετες τιμές $J_0(i) \ \ \forall i$
- Επαναλήψεις $n \to n+1$ μέχρι ***ανεκτή σύγκλιση*** (θεωρητικά $n \to \infty$) μέσω σχέσεων ***backup***:
  $J_{n+1}(i) = \min_{a \in \mathcal{A}_i} \{ c(i,a) + \gamma \sum_{j=1}^{N} p_{ij}(a) J_n(j) \}$ για $i = 1,2,\dots,N$ (από εξισώσεις ***Bellman***)
- Τελικός υπολογισμός των βέλτιστων ***Costs-to-Go***
  $$J^*(i) = \lim_{n \to \infty} J_n(i), \ Q^*(i,a) = c(i,a) + \gamma \sum_{j=1}^{N} p_{ij}(a) J^*(j)$$
  και προσδιορισμός της ***βέλτιστης πολιτικής*** $\mu^*(i) = \arg\min_{a \in \mathcal{A}_i} Q^*(i,a)$ για $i = 1,2,\dots,N$

TABLE 12.2 Summary of the Value Iteration Algorithm

1. Start with arbitrary initial value $J_0(i)$ for state $i = 1, 2, \dots, N$.
2. For $n = 0, 1, 2, \dots$, compute

$$J_{n+1}(i) = \min_{a \in \mathcal{A}_i} \left\{ c(i,a) + \gamma \sum_{j=1}^{N} p_{ij}(a) J_n(j), \right\}, \qquad \begin{matrix} a \in \mathcal{A}_i \\ i = 1, 2, \dots, N \end{matrix}$$

Continue this computation until

$$|J_{n+1}(i) - J_n(i)| < \epsilon \qquad \text{for each state } i$$

where ϵ is a prescribed tolerance parameter. It is presumed that ϵ is sufficiently small for $J_n(i)$ to be close enough to the optimal cost-to-go function $J^*(i)$. We may then set

$$J_n(i) = J^*(i) \qquad \text{for all states } i$$

3. Compute the Q-factor

$$Q^*(i,a) = c(i,a) + \gamma \sum_{j=1}^{N} p_{ij}(a) J^*(j) \qquad \begin{matrix} \text{for } a \in \mathcal{A}_i \text{ and} \\ i = 1, 2, \dots, N \end{matrix}$$

Hence, determine the optimal policy as a greedy policy for $J^*(i)$:

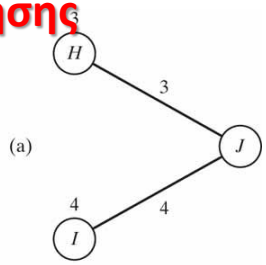
$$\mu^*(i) = \arg\min_{a \in \mathcal{A}_i} Q^*(i,a)$$

- Ο αλγόριθμος ***Value Iteration*** αν συγκλίνει σε ικανοποιητικό χρόνο, αποφεύγει υπολογισμούς ***Q-factors*** και ενδιάμεση ανανέωση πολιτικής σε κάθε βήμα όπως ο ***Policy Iteration***
- Απαιτεί, όπως και ο ***Policy Iteration***, γνώση των $p_{ij}(a)$ (***Model-based Learning***)
- Εναλλακτικές μέθοδοι (***Model-free Learning***) προσεγγίζουν την εύρεση βέλτιστων πολιτικών χωρίς γνώση των $p_{ij}(a)$ με προσομοιώσεις ***Monte Carlo***, αλγορίθμους ***Q-Learning***…

ΣΤΟΧΑΣΤΙΚΕΣ ΔΙΑΔΙΚΑΣΙΕΣ & ΒΕΛΤΙΣΤΟΠΟΙΗΣΗ ΣΤΗ ΜΗΧΑΝΙΚΗ ΜΑΘΗΣΗ

Παράδειγμα Δυναμικού Προγραμματισμού: Βελτιστοποίηση Δρομολόγησης

Εύρεση Δρόμων Ελάχιστου Κόστους από Κόμβο A *σε* Κόμβο J μέσω του μονοκατευθυντικού γράφου όπως στο σχήμα με κατεύθυνση γραμμών ***A → Δ***



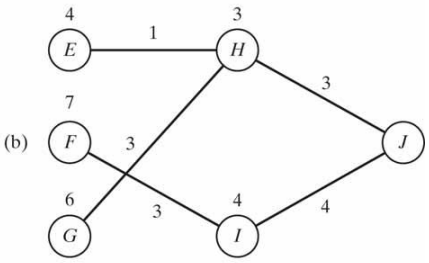
Ενδεικτικό κόστος γραμμών: $A \to B: 2,\ B \to A: \infty$

$B \to F: 4,\ F \to B: \infty$

Ενδεικτικό κόστος δρόμου: Δρόμος $\{A, B, F, I, J\}$: $2 + 4 + 3 + 4 = 13$

Κατάσταση Περιβάλλοντος: Κόμβος σε παρούσα διερεύνηση $\{A, B, \dots, J\}$

Αποφάσεις Agent: Επόμενος κόμβος για διερεύνηση $\{up, down, staight\}$



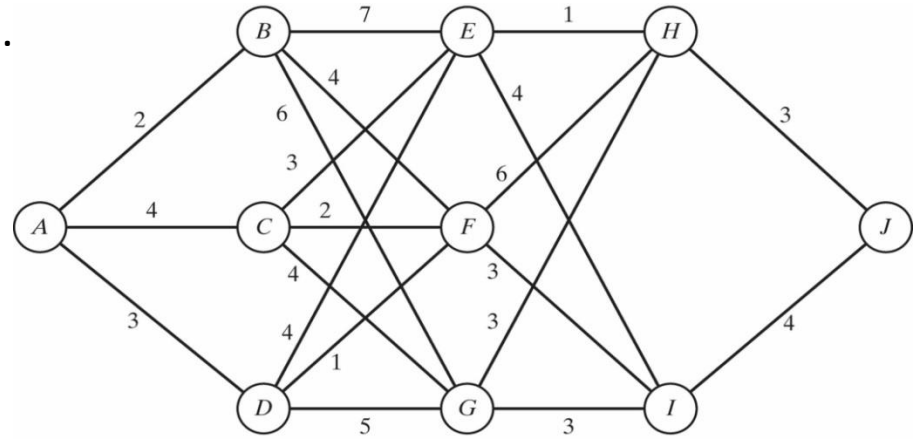
Αναδρομικός Υπολογισμός *Q-Factors* (οι βέλτιστες επιλογές με **bold**):

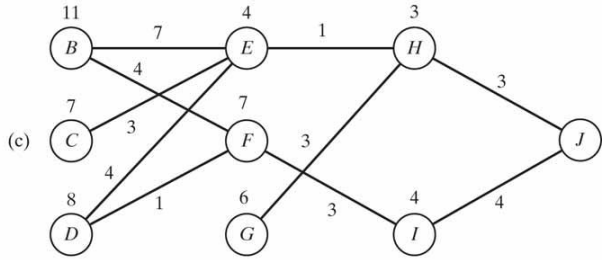
$\boldsymbol{Q(H, down) = 3}$, $\boldsymbol{Q(I, up) = 4}$

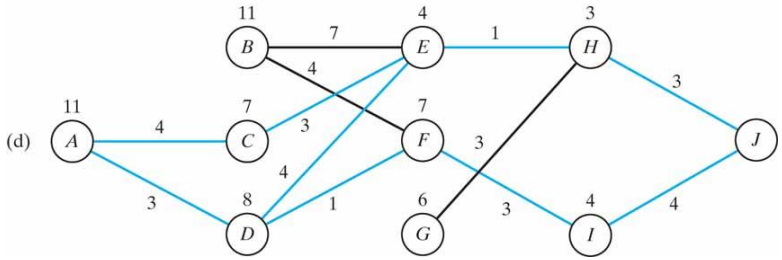
$\boldsymbol{Q(E, staight) = 1 + 3 = 4}$, $Q(E, down) = 4 + 4 = 8$

$Q(F, up) = 6 + 3 = 9$, $\boldsymbol{Q(F, down) = 3 + 4 = 7}$

.....







Κατεύθυνση Γραμμών

A (Αριστερά) ⟶ ***Δ*** (Δεξιά)

Βέλτιστοι Δρόμοι Κόστους 11:

$\{A, C, E, H, J\}, \{A, D, E, H, J\}, \{A, D, F, I, J\}$

Αλγόριθμοι Δυναμικού Προγραμματισμού ***Bellman-Ford*** στηρίζουν την δρομολόγηση ***Border Gateway Protocols*** (***BGP***) ανάμεσα στα ~78,000 Αυτόνομα Συστήματα (***Autonomous Systems***, ***AS***) στο ***Internet*** (~900,000 γνωστά δίκτυα)