



ΠΑΝΕΠΙΣΤΗΜΙΟ
ΠΑΤΡΩΝ
UNIVERSITY OF PATRAS

ΑΝΟΙΚΤΑ ακαδημαϊκά
μαθήματα ΠΠ

Τεχνητή Νοημοσύνη I

Ενότητα 6: Αναζήτηση με Αντιπαλότητα

Μουστάκας Κωνσταντίνος
Πολυτεχνική Σχολή
Τμήμα Ηλεκτρολόγων Μηχανικών και Τεχνολογίας
Υπολογιστών

Σκοποί ενότητας

- Αναζήτηση με Αντιπαλότητα



Περιεχόμενα ενότητας

- Αναζήτηση με Αντιπαλότητα



Αναζήτηση με Αντιπαλότητα (Adversarial Search)



Παιχνίδια

**Σύμφωνα με τη Θεωρία Παιγνίων,
διακρίνονται σε:**

- Πλήρους/μη-πλήρους Πληροφορίας (πληροφόρησης) – Perfect Information Games = PIGs / Imperfect Information Games = IIGs
- Ντετερμινιστικά / μη-ντετερμινιστικά (τυχαία)
- Μηδενικού αθροίσματος = zero-sum (ανταγωνιστικά) / άλλου αθροίσματος (συνεργατικά)
- 1 παίκτη, 2 παικτών, περισσότερων παικτών



Ιδιαίτερα Χαρακτηριστικά Προβλημάτων Παιχνιδιών

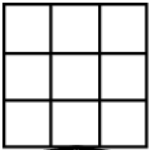
- Απρόβλεπτος αντίπαλος
- Χρονικοί περιορισμοί που απαγορεύουν την άφιξη σε τερματικό κόμβο (προσεγγιστικές τεχνικές)
- Σκοπός είναι η άφιξη σε έναν κόμβο νίκης, όχι απαραίτητα με τη συντομότερη διαδρομή



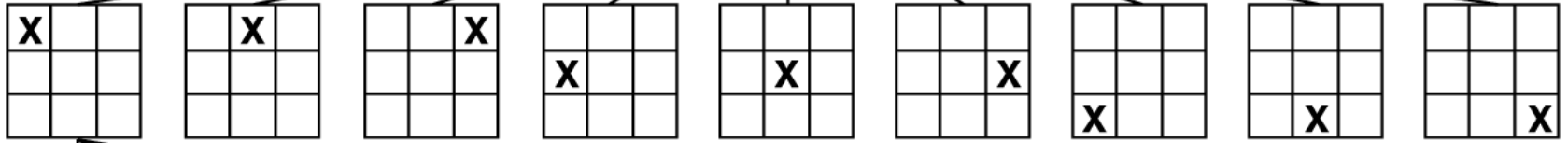
MAX (x)

αρχική κατάσταση

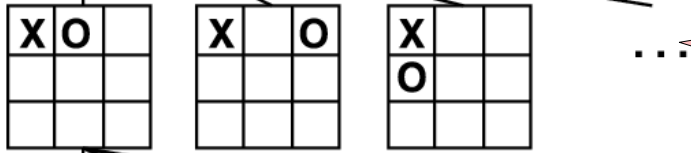
συνάρτηση διαδόχων



MIN (o)

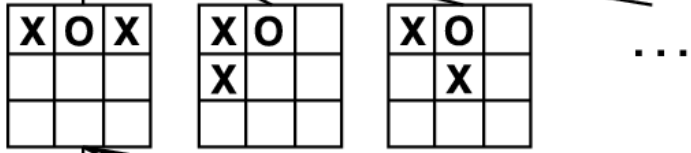


MAX (x)



ply = ημικίνηση (στρώση)
2 plies = 1 κίνηση (move)

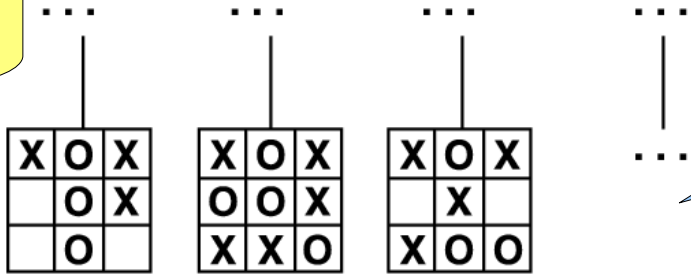
MIN (o)



έλεγχος τερματισμού

ΔΕΝΤΡΟ ΠΑΙΧΝΙΔΙΟΥ
Πολύ μεγάλο δέντρο,
ακόμα και για τα απλούστερα
παιχνίδια
Τρίλιζα : 9! κόμβοι
στο τελευταίο επίπεδο

ΤΕΡΜΑΤΙΚΕΣ



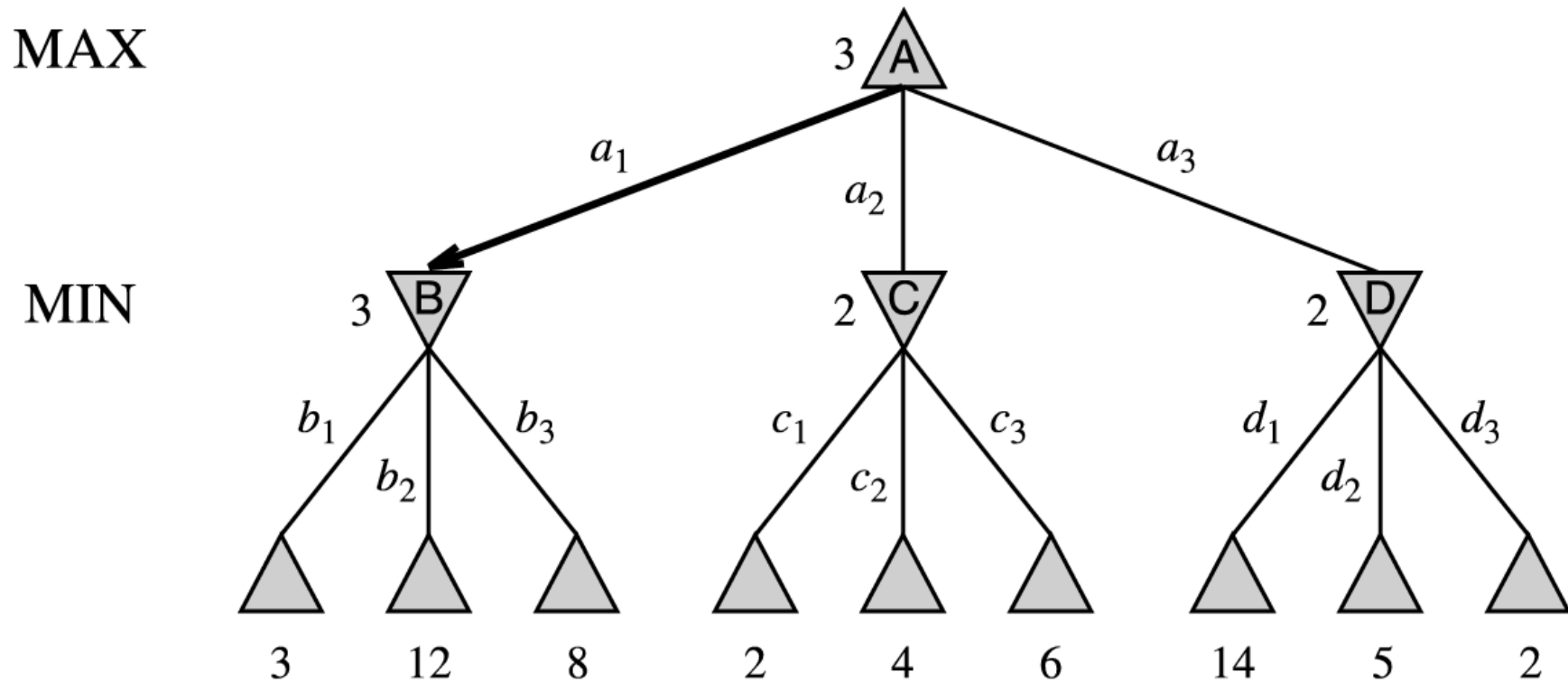
Χρησιμότητα

-1 0 +1

συνάρτηση χρησιμότητας
= utility function



Αλγόριθμος Minimax (ή Min-Max)



function MINIMAX-DECISION(*state*) *returns an action*

$v \leftarrow \text{MAX-VALUE}(state)$

return the *action* in SUCCESSORS(*state*) with value *v*

function MAX-VALUE(*state*) *returns a utility value*

if TERMINAL-TEST(*state*) **then return** UTILITY(*state*)

$v \leftarrow -\infty$

for *a, s* in SUCCESSORS(*state*) **do**

$v \leftarrow \text{MAX}(v, \text{MIN-VALUE}(s))$

return *v*

function MIN-VALUE(*state*) *returns a utility value*

if TERMINAL-TEST(*state*) **then return** UTILITY(*state*)

$v \leftarrow \infty$

for *a, s* in SUCCESSORS(*state*) **do**

$v \leftarrow \text{MIN}(v, \text{MAX-VALUE}(s))$

return *v*

Πηγή: “Artificial Intelligence – A Modern Approach”, 2nd ed., S.Russell & P.Norvig,
Prentice Hall, 2003



```

% A straightforward implementation of the minimax principle. (I.Bratko)
% http://www.pearsoned.co.uk/highereducation/resources/
% bratkoprologprogrammingforartificialintelligence3e

% minimax( Pos, BestSucc, Val):
%   Pos is a position, Val is its minimax value;
%   best move from Pos leads to position BestSucc

minimax( Pos, BestSucc, Val) :-
    moves( Pos, PosList), !,                % Legal moves in Pos produce PosList
    best( PosList, BestSucc, Val)
    ;
    staticval( Pos, Val).                  % Pos has no successors: evaluate statically

best( [ Pos], Pos, Val) :-
    minimax( Pos, _, Val), !.

best( [Pos1 | PosList], BestPos, BestVal) :-
    minimax( Pos1, _, Val1),
    best( PosList, Pos2, Val2),
    betterof( Pos1, Val1, Pos2, Val2, BestPos, BestVal).

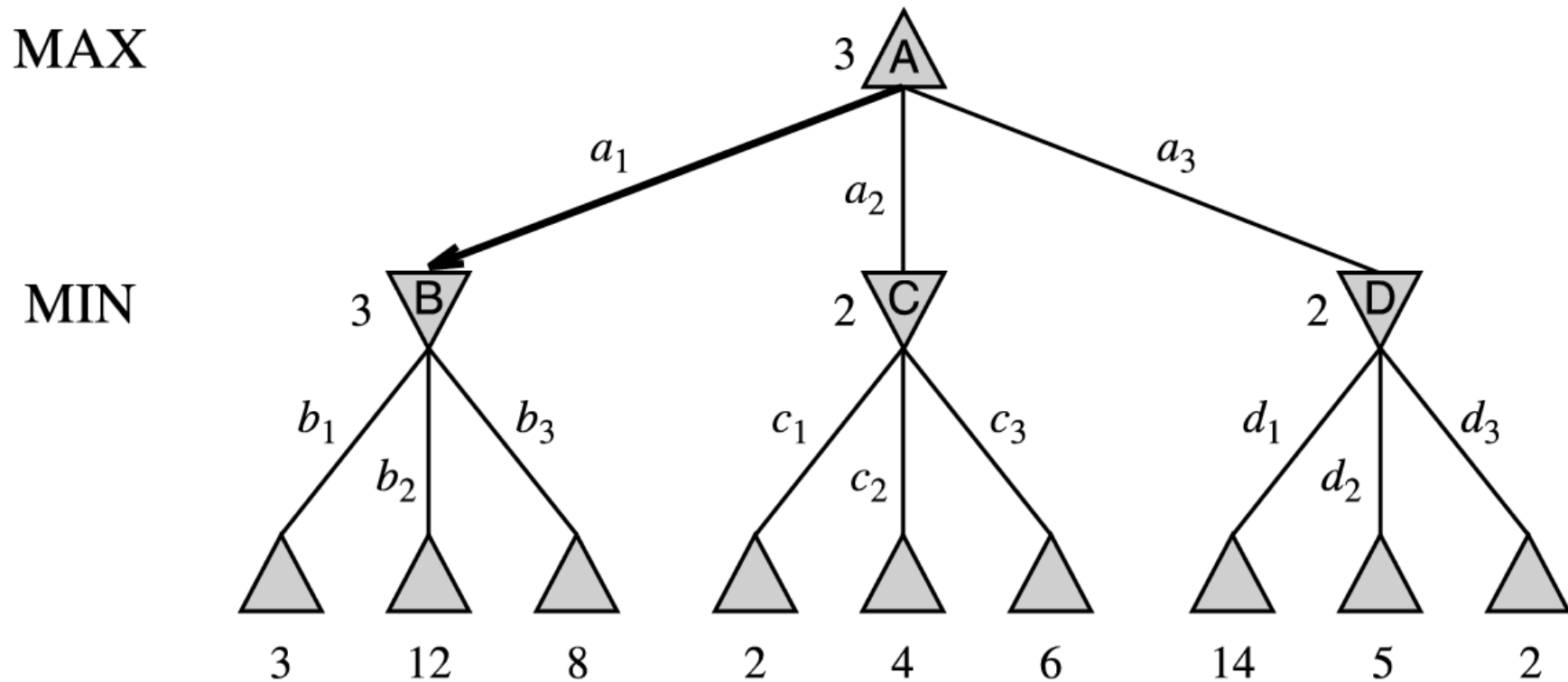
betterof( Pos0, Val0, Pos1, Val1, Pos0, Val0) :-
    min_to_move( Pos0),                    % Pos0 better than Pos1
    Val0 > Val1, !                         % MIN to move in Pos0
    ;                                       % MAX prefers the greater value
    max_to_move( Pos0),                    % MAX to move in Pos0
    Val0 < Val1, !                         % MIN prefers the lesser value

betterof( Pos0, Val0, Pos1, Val1, Pos1, Val1). % Otherwise Pos1 better than Pos0

```



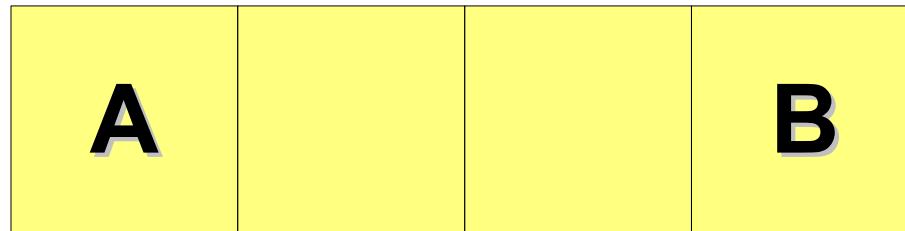
Αλγόριθμος Minimax (ή Min-Max)

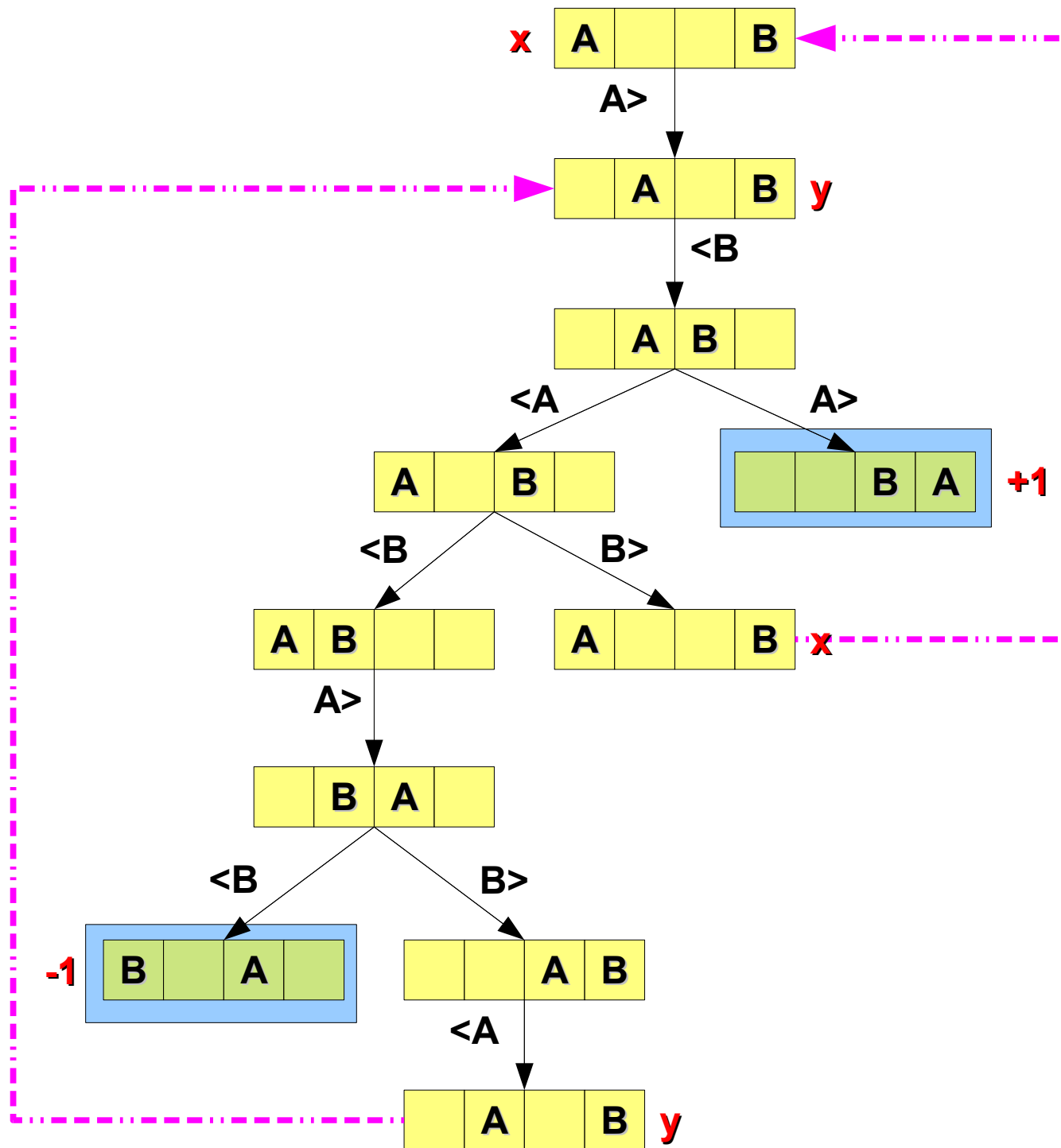


$$A = \max(B, C, D) = \max(\min(3, 12, 8), \min(2, 4, 6), \min(14, 5, 2))$$

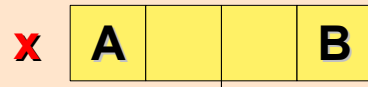


Παράδειγμα



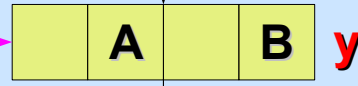


MAX



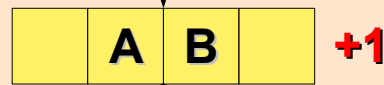
A>

MIN



<B

MAX



<A

A>

MIN



<B

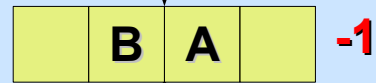
B>

MAX



A>

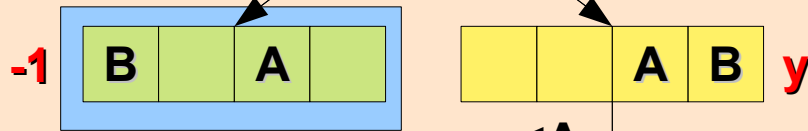
MIN



<B

B>

MAX



<A

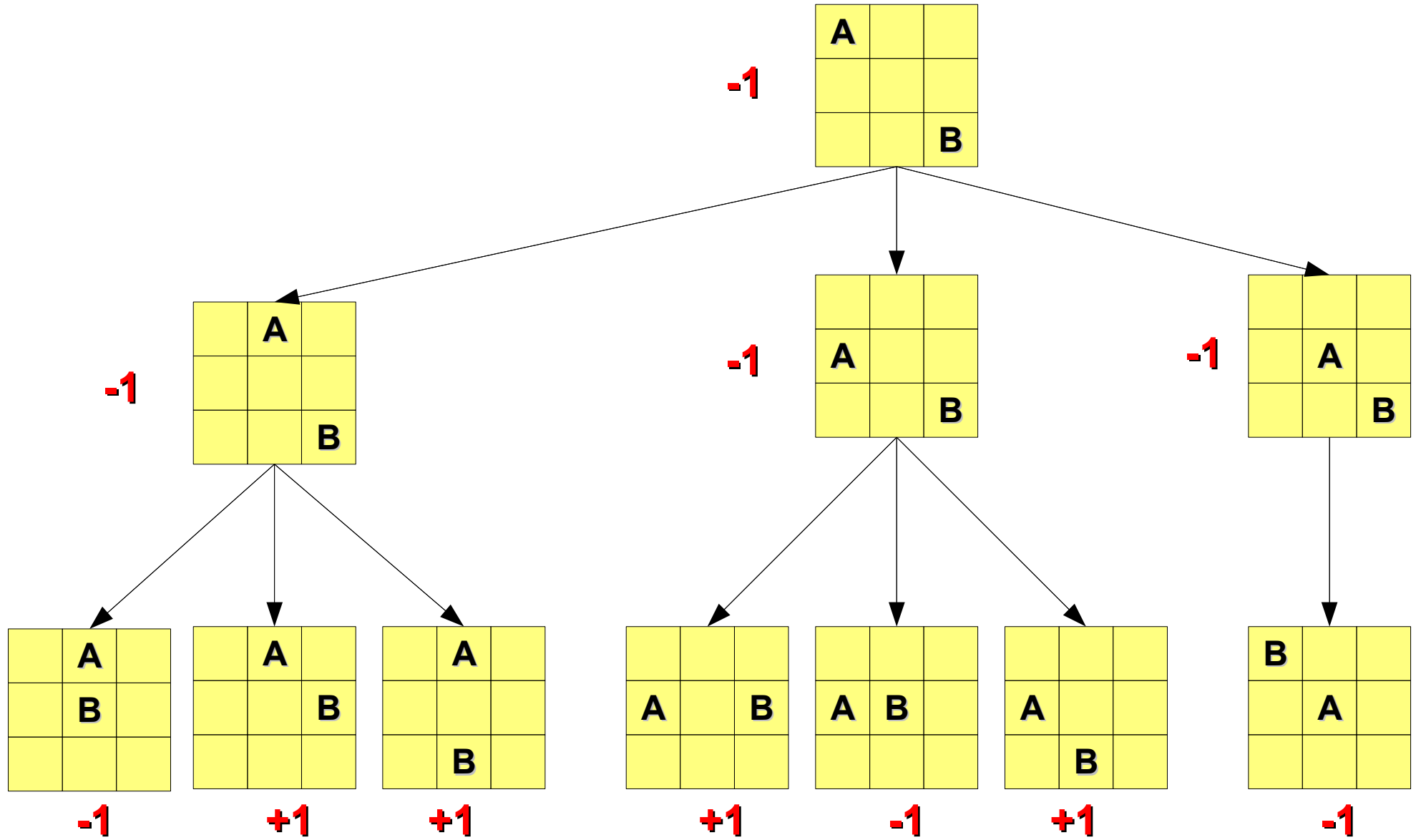
MIN



Άλλο Παράδειγμα

A		
		B





Ανάλυση Παιχνιδιού

NIM

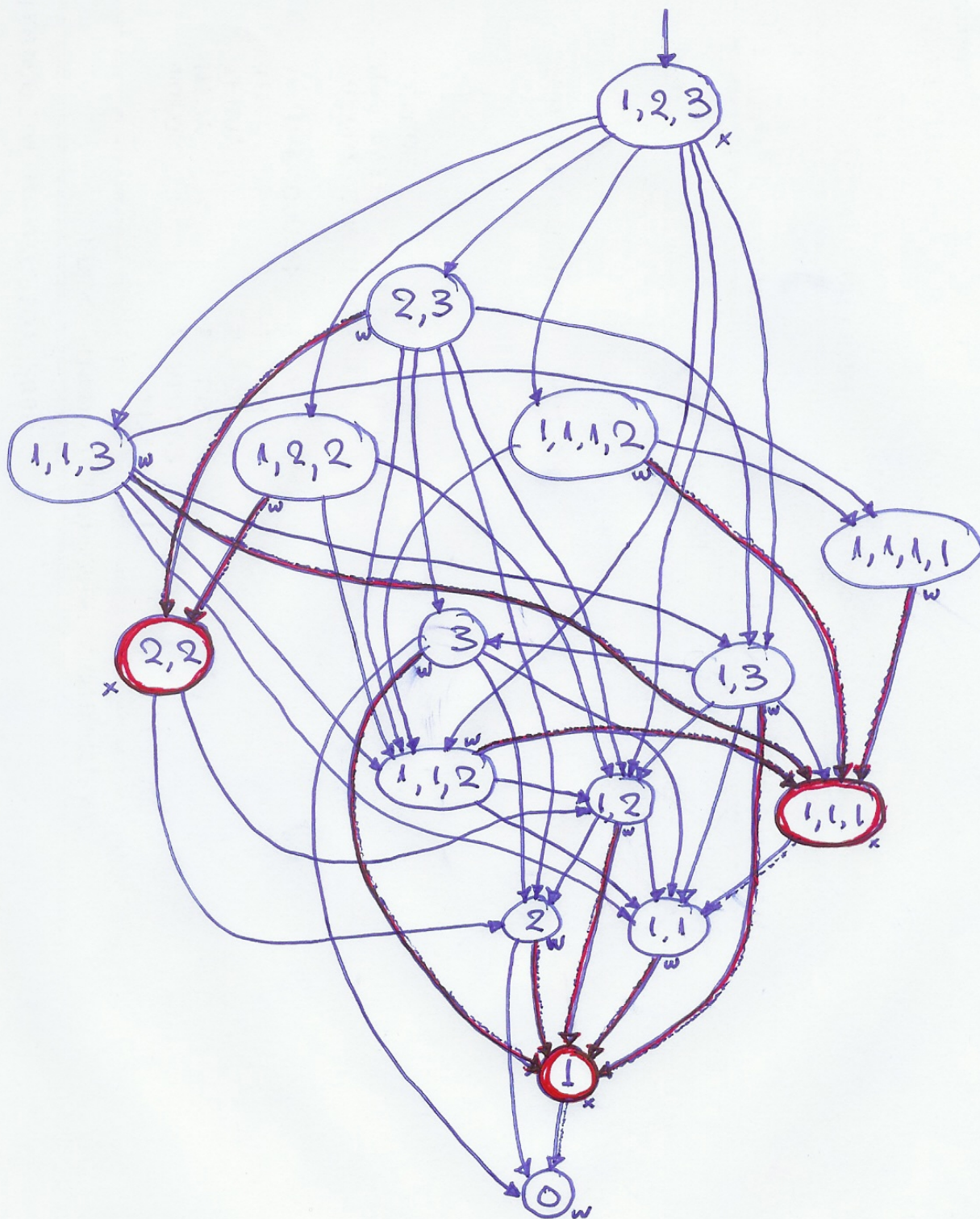
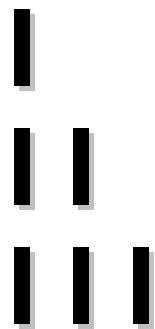
NIM(3,5,7)

ΚΑΝΟΝΕΣ:

- Δυο παίκτες
- Ο καθένας σβήνει 1 ή περισσότερες συνεχόμενες γραμμές στην ίδια σειρά
- Όποιος σβήσει την τελευταία χάνει

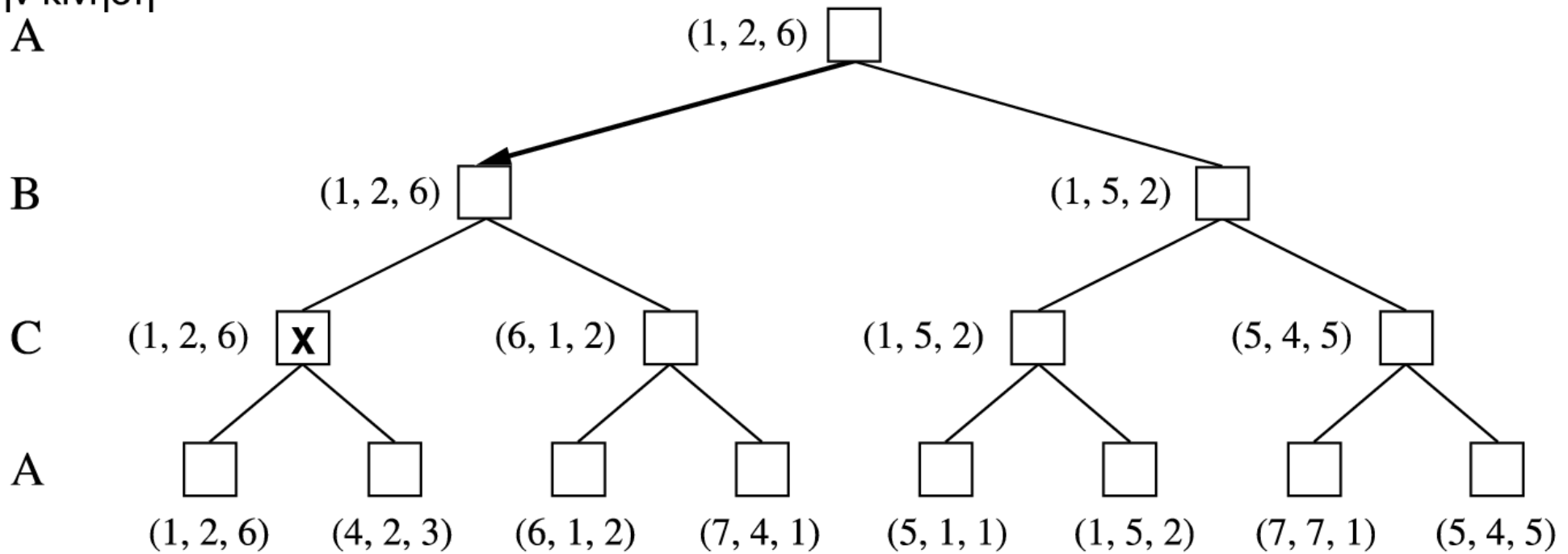


Γράφος NIM(1,2,3)



Επέκταση σε περισσότερους παίκτες

έχει την κίνηση
A

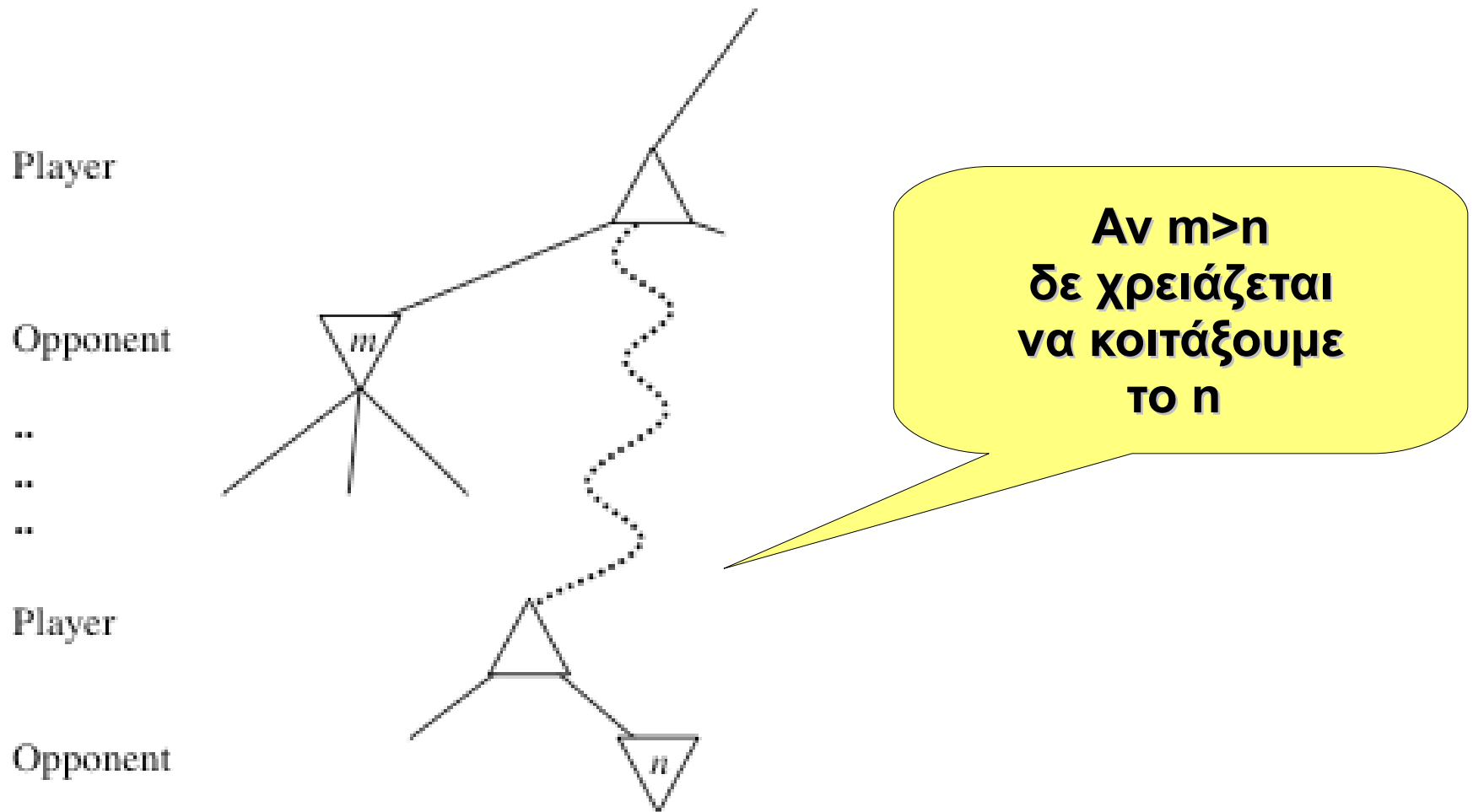


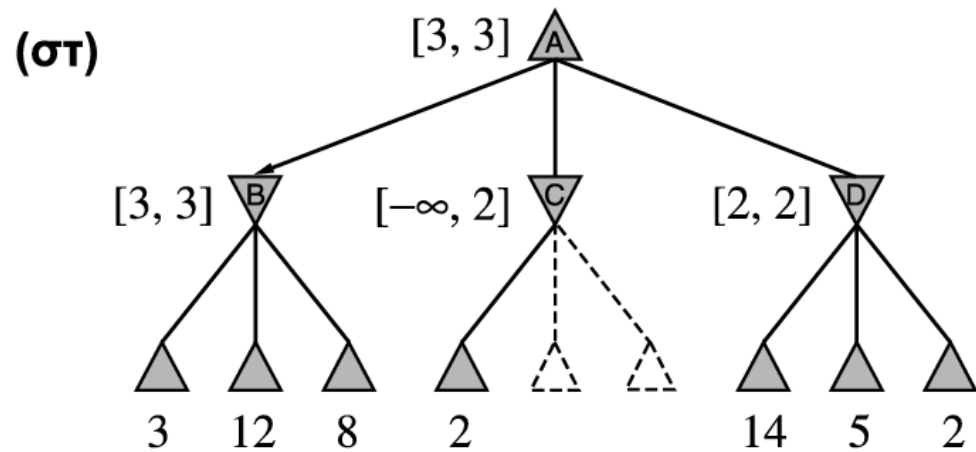
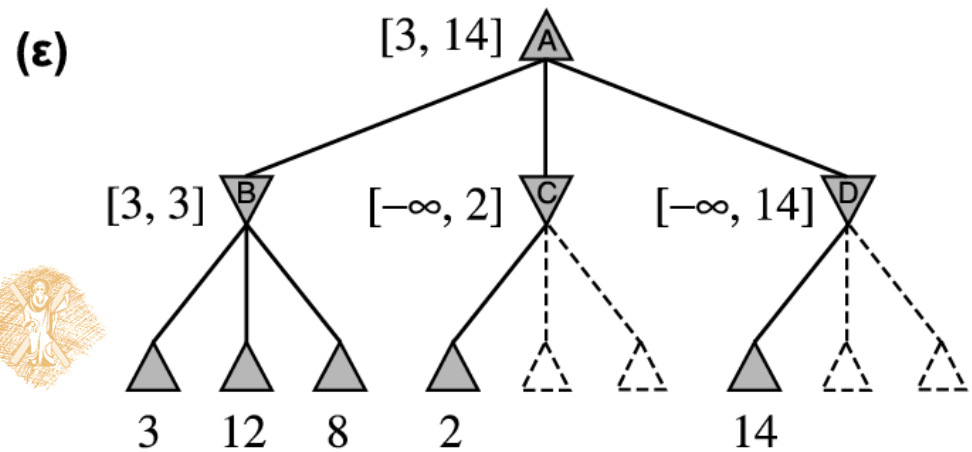
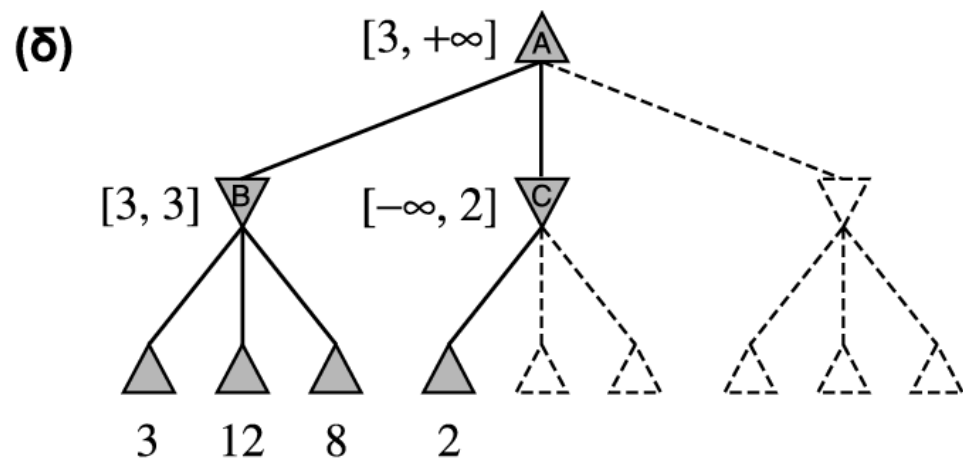
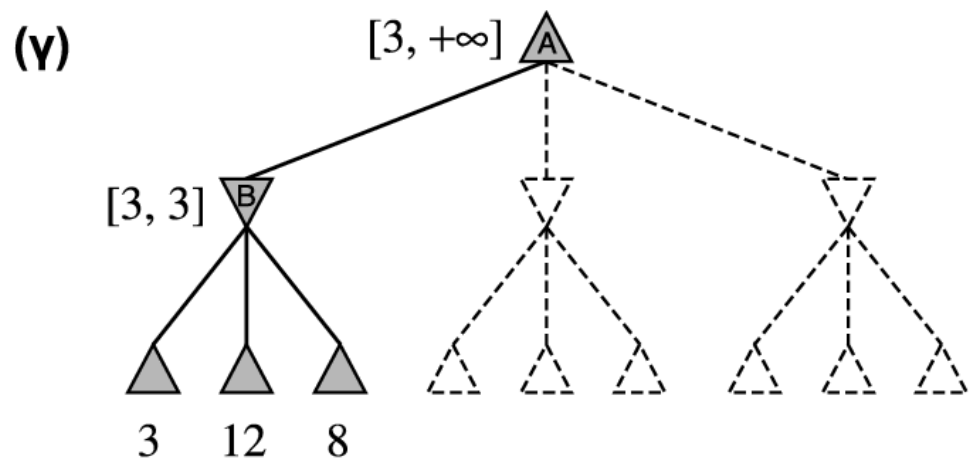
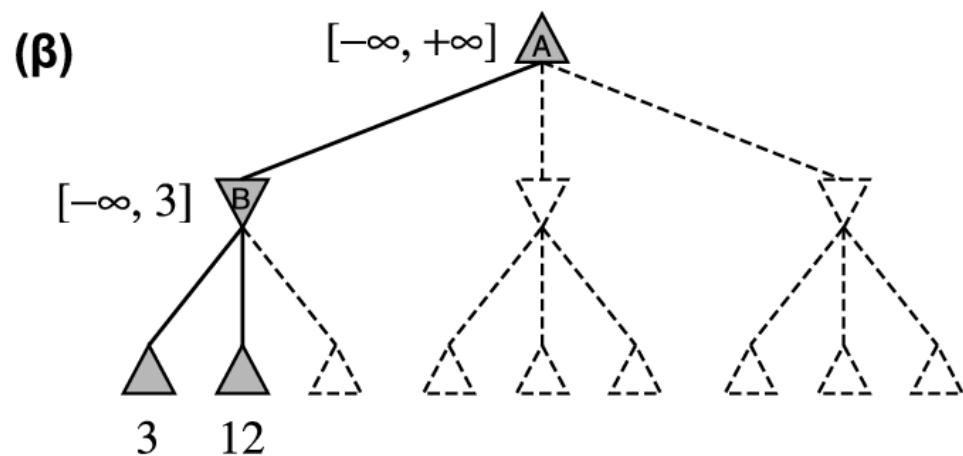
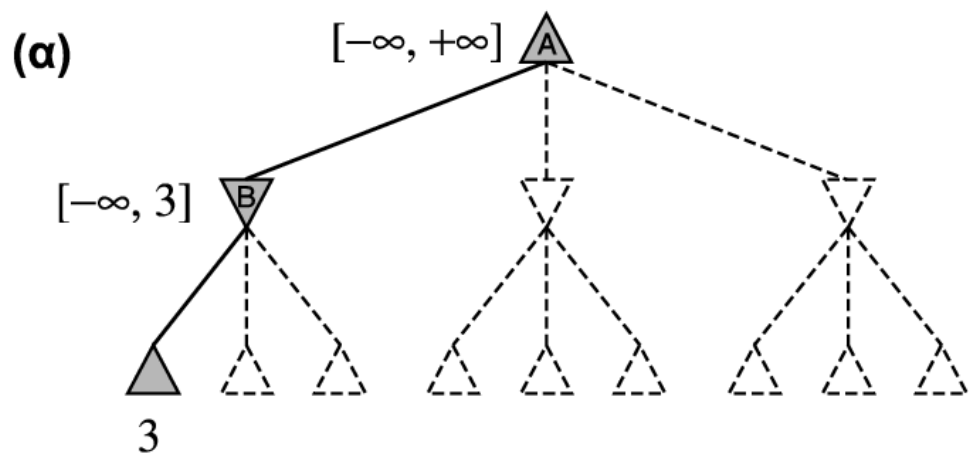
Πολλοί παίκτες ανάλυση

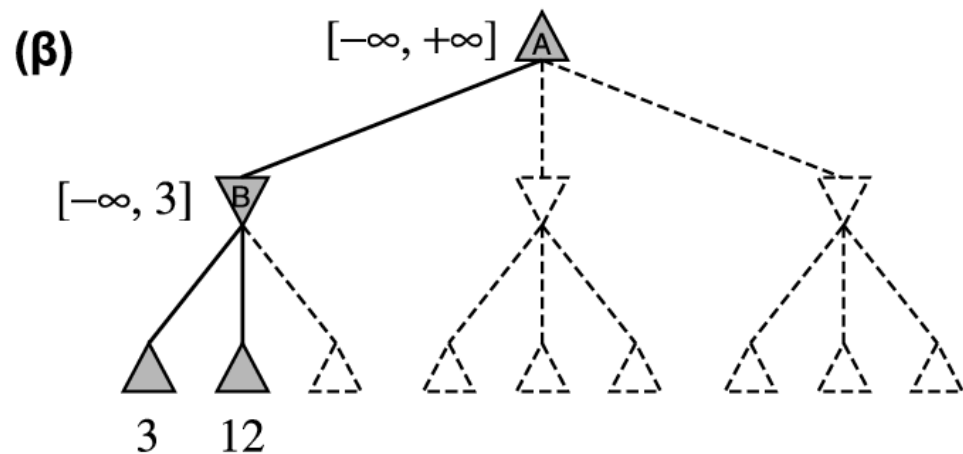
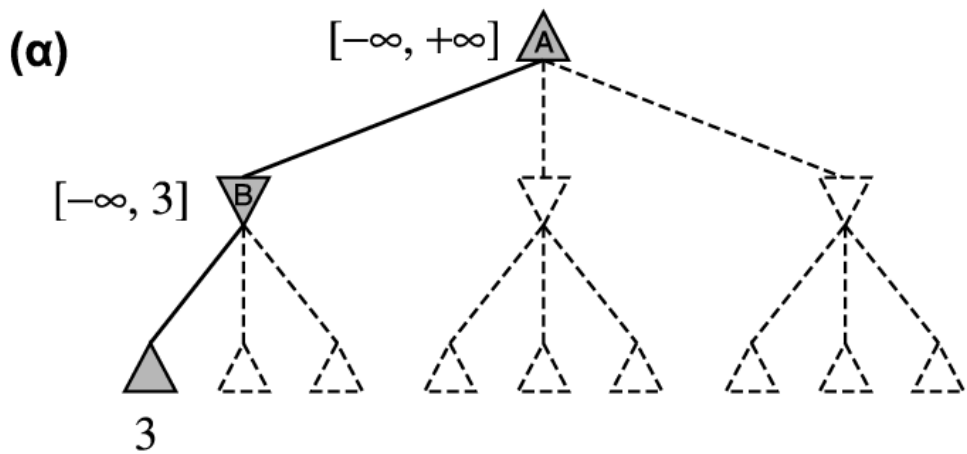
- Όμοια λογική: τιμές \rightarrow διανύσματα
- Συμμαχίες (Risk...)
- Παραβίαση συμμαχίας
- Μη μηδενικού αθροίσματος παιχνίδι έχει ως αποτέλεσμα συμμαχία προς κοινό στόχο



Alpha-Beta Pruning (Κλάδεμα α-β)







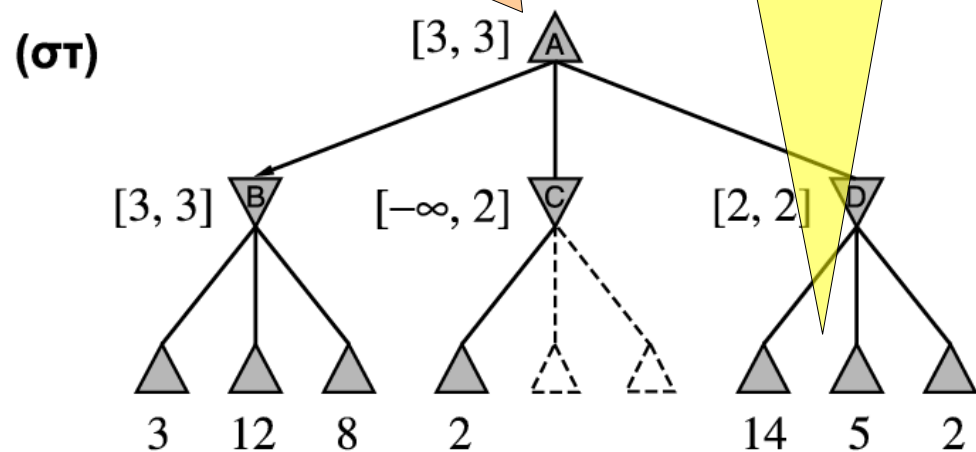
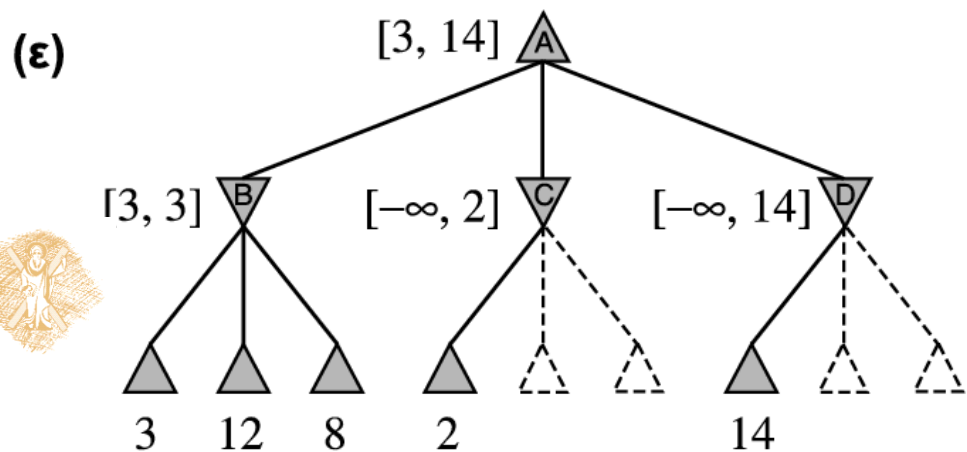
(γ)

Τιμή = $\max(\min(3, 12, 8), \min(2, x, y), \min(14, 5, 2))$
 $= \max(3, \min(2, x, y), 2)$
 $= \max(3, z, 2)$, όπου $z \leq 2$
 $= 3$

$[3, 3]$

3 12 8 3 12

Η σειρά των κόμβων έχει μεγάλη σημασία!



```
% An implementation of the alpha-beta algorithm. (I.Bratko)
```

```
alphabeta( Pos, Alpha, Beta, GoodPos, Val) :-  
    moves( Pos, PosList), !,  
    boundedbest( PosList, Alpha, Beta, GoodPos, Val);  
    staticval( Pos, Val). % Static value of Pos  
  
boundedbest( [Pos | PosList], Alpha, Beta, GoodPos, GoodVal) :-  
    alphabeta( Pos, Alpha, Beta, _, Val),  
    goodenough( PosList, Alpha, Beta, Pos, Val, GoodPos, GoodVal).  
  
goodenough( [], _, _, Pos, Val, Pos, Val) :- !. % No other candidate  
  
goodenough( _, Alpha, Beta, Pos, Val, Pos, Val) :-  
    min_to_move( Pos), Val > Beta, ! % Maximizer attained upper bound  
    ;  
    max_to_move( Pos), Val < Alpha, !. % Minimizer attained lower bound  
  
goodenough( PosList, Alpha, Beta, Pos, Val, GoodPos, GoodVal) :-  
    newbounds( Alpha, Beta, Pos, Val, NewAlpha, NewBeta), % Refine bounds  
    boundedbest( PosList, NewAlpha, NewBeta, Pos1, Val1),  
    betterof( Pos, Val, Pos1, Val1, GoodPos, GoodVal).  
  
newbounds( Alpha, Beta, Pos, Val, Val, Beta) :-  
    min_to_move( Pos), Val > Alpha, !. % Maximizer increased lower bound  
  
newbounds( Alpha, Beta, Pos, Val, Alpha, Val) :-  
    max_to_move( Pos), Val < Beta, !. % Minimizer decreased upper bound  
  
newbounds( Alpha, Beta, _, _, Alpha, Beta). % Otherwise bounds unchanged  
  
betterof( Pos, Val, Pos1, Val1, Pos, Val) :- % Pos better than Pos1  
    min_to_move( Pos), Val > Val1, !  
    ;  
    _to_move( Pos), Val < Val1, !.  
    rof( _, _, Pos1, Val1, Pos1, Val1). % Otherwise Pos1 better
```



α-β ανάλυση

- Εξαρτάται από τη σειρά με την οποία εξετάζονται οι κόμβοι
- Μπορεί να φτάσει ~ 2 φορές πιο μακριά σε σχέση με τον minimax στον ίδιο χρόνο
- Μηχανισμοί καθορισμού προτεραιότητας κινήσεων
- Χειρισμός επαναλαμβανόμενων καταστάσεων



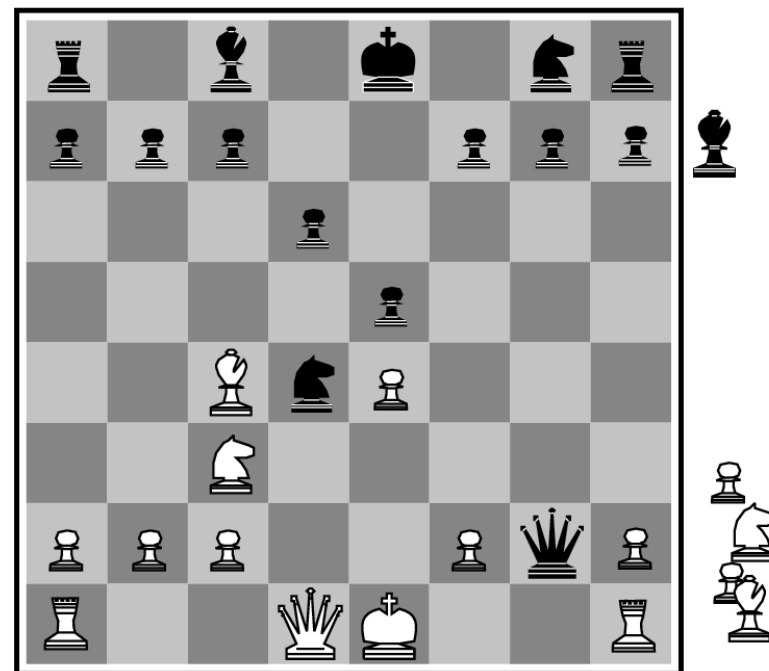
Ατελείς Αποφάσεις σε Πραγματικό Χρόνο

...όταν δεν προλαβαίνουμε να φτάσουμε
στα φύλλα του δέντρου



Συναρτήσεις Αξιολόγησης

- f_1 = αξία υλικού:
 - Πιόνι = 1
 - Αξιωματικός = 3
 - Ίππος = 3
 - Πύργος = 5
 - Βασίλισσα = 9
- f_2 = έλεγχος κέντρου
- f_3 = προστασία βασιλιά
- f_4 = κινητικότητα κομματιών
- ...



Παίζουν τα λευκά

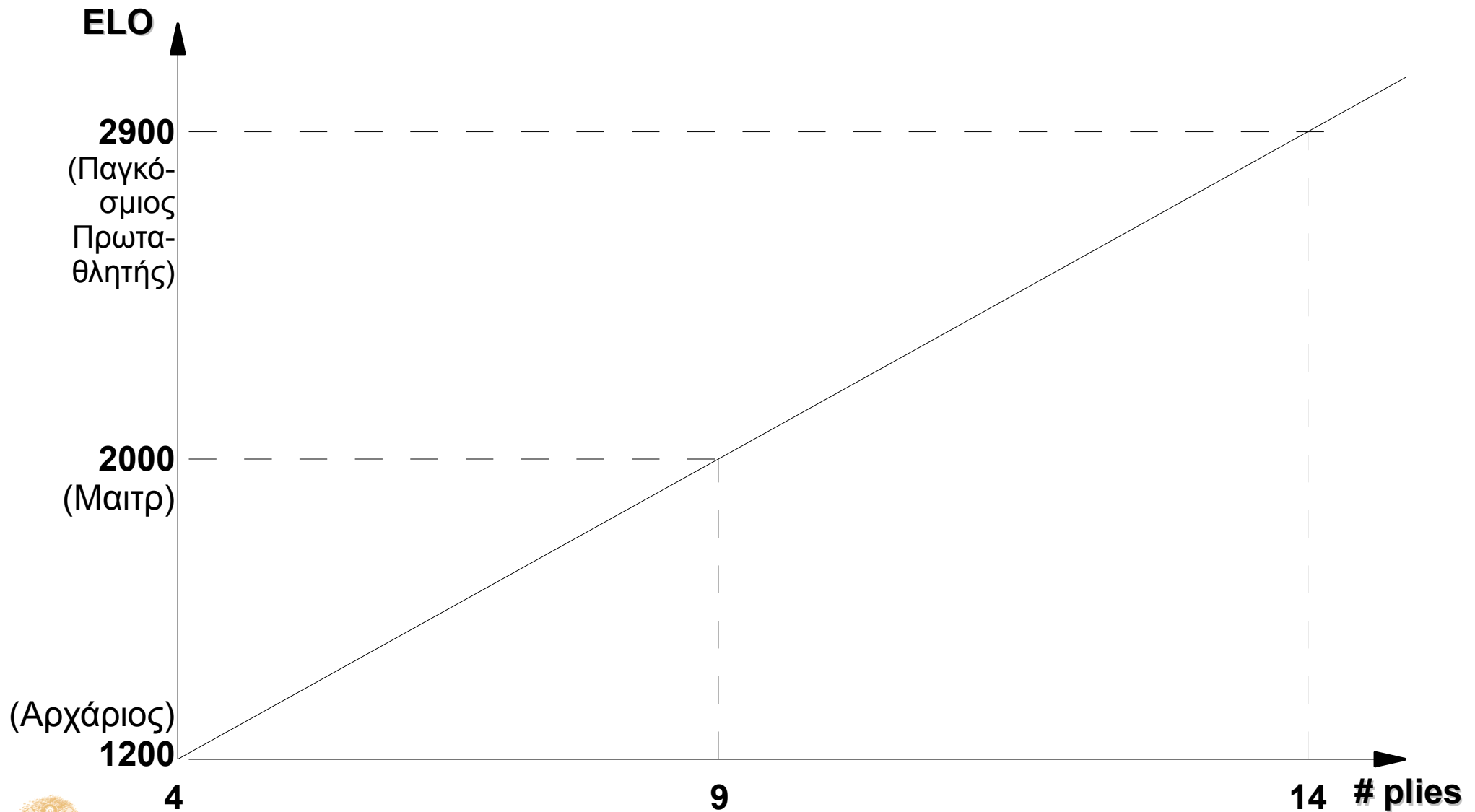
$$Eval(s) = w_1 f_1(s) + w_2 f_2(s) + \dots + w_n f_n(s)$$

όσο χαμηλότερα στο δέντρο την εφαρμόζουμε,
τόσο καλύτερα συμπεριφέρεται

σταθμισμένη γραμμική
συνάρτηση αξιολόγησης

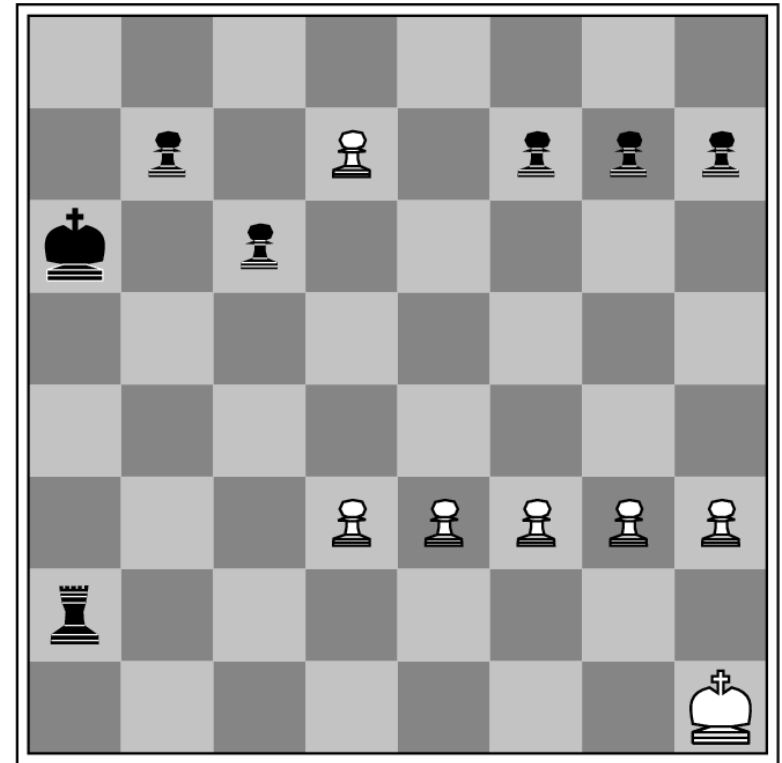


ELO Rating



Επίδραση του Ορίζοντα (Horizon Effect)

- Ο μαύρος μπορεί να καθυστερήσει την προαγωγή του λευκού πιονιού σε βασίλισσα για 14 plies.
 - Το πρόβλημα δεν ανιχνεύεται για μικρούς ορίζοντες.
- Αντιμετώπιση με *μοναδικές επεκτάσεις* (singular extensions) [μπορεί να φτάσει μέχρι και 30 plies]



Παίζουν τα μαύρα



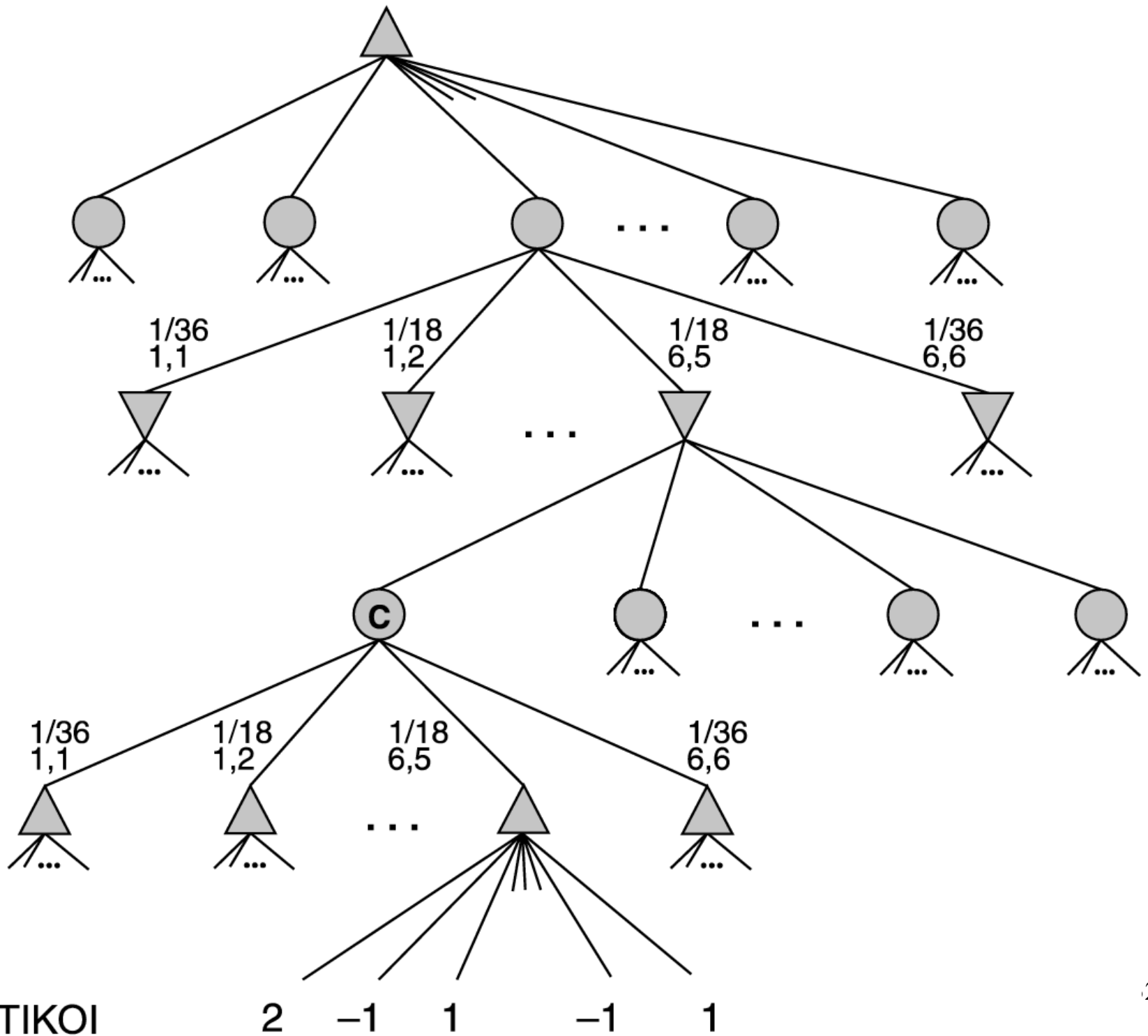
MAX

TYXH

MIN

TYXH

MAX



Παιχνίδια Μη-Πλήρους Πληροφορίας

- Χαρτοπαίγνια
- Δεν γνωρίζουμε την κατάσταση του αντίπαλου.
- Οποιοσδήποτε αλγόριθμος θα πρέπει να εξετάζει *τι πληροφορίες θα έχει* σε κάθε σημείο του παιχνιδιού.
 - Αναζήτηση στο χώρο των καταστάσεων πεπιοίθησης (κρατάμε έναν για κάθε διατεταγμένο ζευγάρι παικτών).
- Συνιστάται να δίνουμε στον αντίπαλο όσο το δυνατόν λιγότερες πληροφορίες ενεργώντας *απρόβλεπτα*.
- Μπορούμε να θεωρήσουμε τυχαία κάθε μεταβλητή που δε γνωρίζουμε την τιμή της και να περιορίσουμε το πεδίο τιμών με κάποια μέθοδο (π.χ. ενίσχυση περιορισμών)



Ανταγωνιστικά & Συνεργατικά Παιχνίδια (Μη-Πλήρους Πληροφορίας) Δημοπρασίες - Ψηφοφορίες

	A	B	Γ	Δ	
α		3	3	3	= 9
β	3		2	2	= 7
γ	2	2		1	= 5
δ	1	1	1		= 3

	A	B	Γ	Δ	
α	4	3	3	3	= 9 + 4
β	3	4	2	2	= 7 + 4
γ	2	2	4	1	= 5 + 4
δ	1	1	1	4	= 3 + 4

	A	B	Γ	Δ	
α		1	1	1	= 7 - 4
β	1		2	2	= 9 - 4
γ	2	2		3	= 11 - 4
δ	3	3	3		= 13 - 4

	A	B	Γ	Δ	
α	4	1	1	1	= 7
β	1	4	2	2	= 9
γ	2	2	4	3	= 11
δ	3	3	3	4	= 13



Θεώρημα John Nash

Τέλος Ενότητας

Χρηματοδότηση

- Το παρόν εκπαιδευτικό υλικό έχει αναπτυχθεί στο πλαίσιο του εκπαιδευτικού έργου του διδάσκοντα
- Το έργο «**Ανοικτά Ακαδημαϊκά Μαθήματα στο Πανεπιστήμιο Αθηνών**» έχει χρηματοδοτήσει μόνο την αναδιαμόρφωση του εκπαιδευτικού υλικού
- Το έργο υλοποιείται στο πλαίσιο του Επιχειρησιακού Προγράμματος «Εκπαίδευση και Δια Βίου Μάθηση» και συγχρηματοδοτείται από την Ευρωπαϊκή Ένωση (Ευρωπαϊκό Κοινωνικό Ταμείο) και από εθνικούς πόρους.



Σημειώματα

Σημείωμα Ιστορικού Εκδόσεων Έργου

Το παρόν έργο αποτελεί την έκδοση **1.0**.

Έχουν προηγηθεί οι κάτωθι εκδόσεις:

- Έκδοση **1.0** διαθέσιμη [εδώ](#).

Σημείωμα Αναφοράς

Copyright Πανεπιστήμιο Πατρών, **Σγάρμπας Κυριάκος**. «**Τεχνητή Νοημοσύνη Ι, Αναζήτηση με Αντιπαλότητα**». Έκδοση: **1.0**. Πάτρα **2014**. Διαθέσιμο από τη δικτυακή διεύθυνση:

https://eclass.upatras.gr/modules/course_metadata/opencourses.php?fc=15

Σημείωμα Αδειοδότησης

Το παρόν υλικό διατίθεται με τους όρους της άδειας χρήσης Creative Commons Αναφορά, Μη Εμπορική Χρήση Παρόμοια Διανομή 4.0 [1] ή μεταγενέστερη, Διεθνής Έκδοση. Εξαιρούνται τα αυτοτελή έργα τρίτων π.χ. φωτογραφίες, διαγράμματα κ.λ.π., τα οποία εμπεριέχονται σε αυτό και τα οποία αναφέρονται μαζί με τους όρους χρήσης τους στο «Σημείωμα Χρήσης Έργων Τρίτων».

[1] <http://creativecommons.org/licenses/by-nc-sa/4.0/>



Ως **Μη Εμπορική** ορίζεται η χρήση:

- που δεν περιλαμβάνει άμεσο ή έμμεσο οικονομικό όφελος από την χρήση του έργου, για το διανομέα του έργου και αδειοδόχο
- που δεν περιλαμβάνει οικονομική συναλλαγή ως προϋπόθεση για τη χρήση ή πρόσβαση στο έργο
- που δεν προσπορίζει στο διανομέα του έργου και αδειοδόχο έμμεσο οικονομικό όφελος (π.χ. διαφημίσεις) από την προβολή του έργου σε διαδικτυακό τόπο

Ο δικαιούχος μπορεί να παρέχει στον αδειοδόχο ξεχωριστή άδεια να χρησιμοποιεί το έργο για εμπορική χρήση, εφόσον αυτό του ζητηθεί.

Διατήρηση Σημειωμάτων

Οποιαδήποτε αναπαραγωγή ή διασκευή του υλικού θα πρέπει να συμπεριλαμβάνει:

- το Σημείωμα Αναφοράς
- το Σημείωμα Αδειοδότησης
- τη δήλωση Διατήρησης Σημειωμάτων
- το Σημείωμα Χρήσης Έργων Τρίτων (εφόσον υπάρχει)

μαζί με τους συνοδευόμενους υπερσυνδέσμους.

Σημείωμα Χρήσης Έργων Τρίτων

Το Έργο αυτό κάνει χρήση των ακόλουθων έργων:

Εικόνες/Σχήματα/Διαγράμματα/Φωτογραφίες