Manara University
Faculty of Engineering
Department of IT

Introduction to Artificial Intelligence

Lecture 8
Game Playing

Prof Dr. Eng. Mariam M. Saii

OUTLINE

- Minimax Search with Perfect Decisions
- Minimax Search with Cut-off
- Alpha-Beta Pruning
- Game of chance

What Kinds of Games?

Mainly games of strategy with the following characteristics:

1. Sequence of moves to play
2. Rules that specify possible moves
3. Rules that specify a payment for each move
4. Objective is to maximize your payment

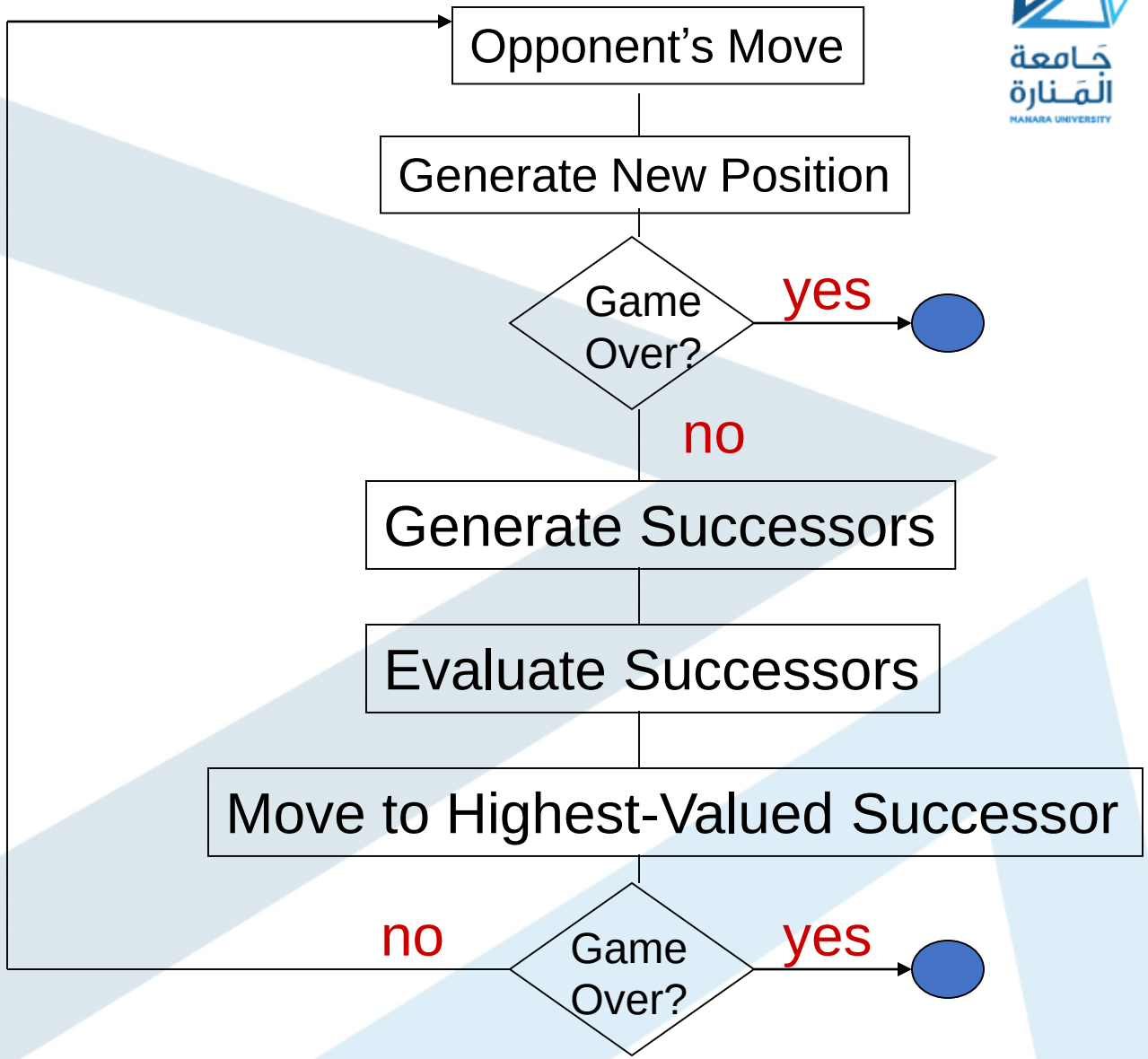
Games vs. Search Problems

- Unpredictable opponent → specifying a move for every possible opponent reply
- Time limits → unlikely to find goal, must approximate

Types of Games

	deterministic	chance
perfect information	chess, checkers, go, othello	backgammon monopoly
imperfect information	battleship Kriegspiel	bridge, poker, scrabble nuclear war

Two-Player Game



Game Tree (2-player, Deterministic, Turns)

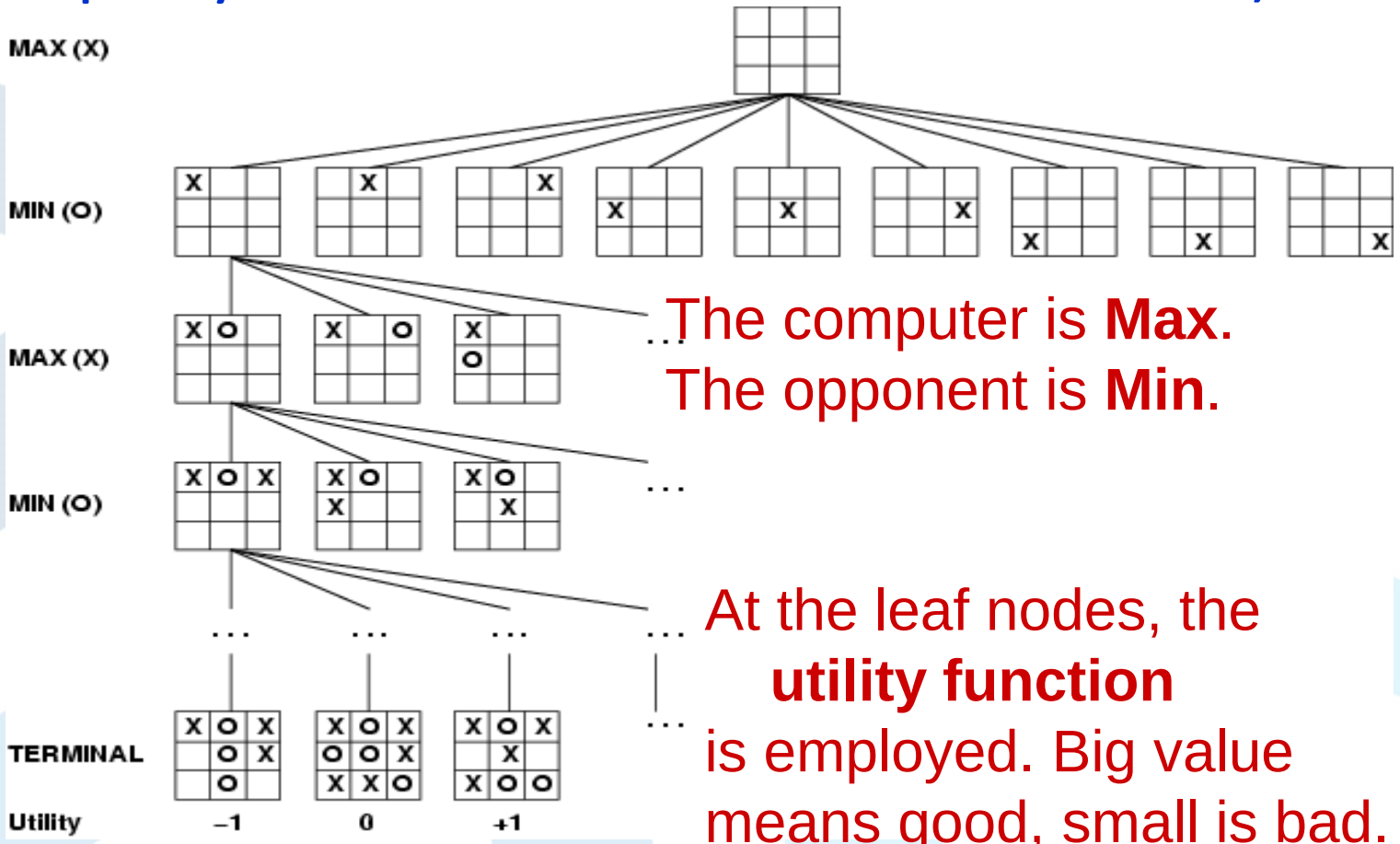
computer's turn

opponent's turn

computer's turn

opponent's turn

leaf nodes are evaluated



The computer is **Max**.
The opponent is **Min**.

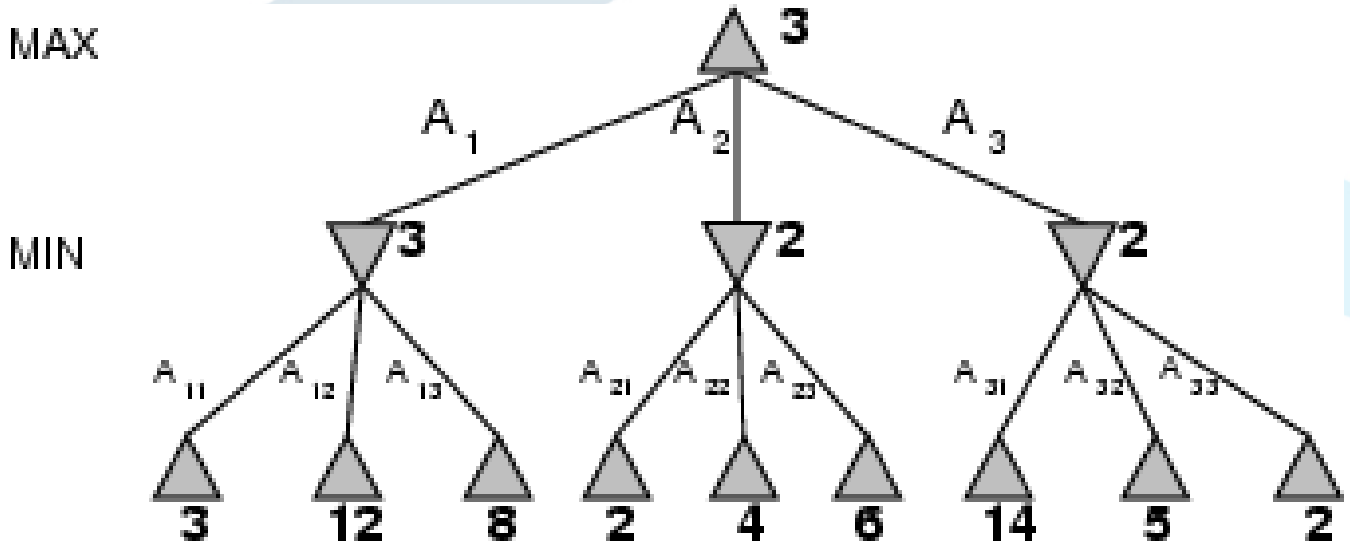
At the leaf nodes, the **utility function** is employed. Big value means good, small is bad.

Mini-Max Terminology

- utility function: the function applied to leaf nodes
- backed-up value
  - of a max-position: the value of its largest successor
  - of a min-position: the value of its smallest successor
- minimax procedure: search down several levels; at the bottom level apply the utility function, back-up values all the way up to the root node, and that node selects the move.

Minimax

- Perfect play for deterministic games
- Idea: choose move to position with highest minimax value
  = best achievable payoff against best play
- E.g., 2-ply game:



Minimax Strategy

- Why do we take the min value every other level of the tree?
- These nodes represent the opponent's choice of move.
- The computer assumes that the human will choose that move that is of least value to the computer.

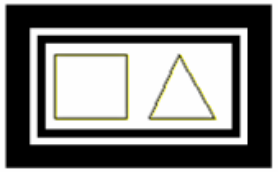
Properties of Minimax

- Complete? Yes (if tree is finite)
- Optimal? Yes (against an optimal opponent)
- Time complexity? $O(b^m)$
- Space complexity? O(bm) (depth-first exploration)

- For chess, b ≈ 35, m ≈100 for "reasonable" games
  → exact solution completely infeasible

Need to speed it up.

Alpha-Beta Procedure



عقدة MAX والهدف جعل الشجرة الفرعية التي جذرها هذه العقدة أكبر مايمكن. وتختار هذه العقدة الابن ذي قيمة تابع التقويم الأعلى وتصبح هذه القيمة هي قيمة عقدة MAX.

عقدة MIN والهدف جعل الشجرة الفرعية التي جذرها هذه العقدة أصغر مايمكن. وتختار هذه العقدة الابن ذي قيمة تابع التقويم الأصغر وتصبح هذه القيمة هي قيمة عقدة MIN.

Beta is the *minimum upper* بيتا هي الحد الأعلى الأصغري للحل المحتمل
bound of possible solutions

Alpha is the *maximum lower* ألفا هي الحد الأدنى الأعظمي للحل المحتمل
bound of possible solutions

- The alpha-beta procedure can speed up a depth-first minimax search.
- Beta: an upper bound on the value that a minimizing node may ultimately be assigned

$$v \leq \beta$$

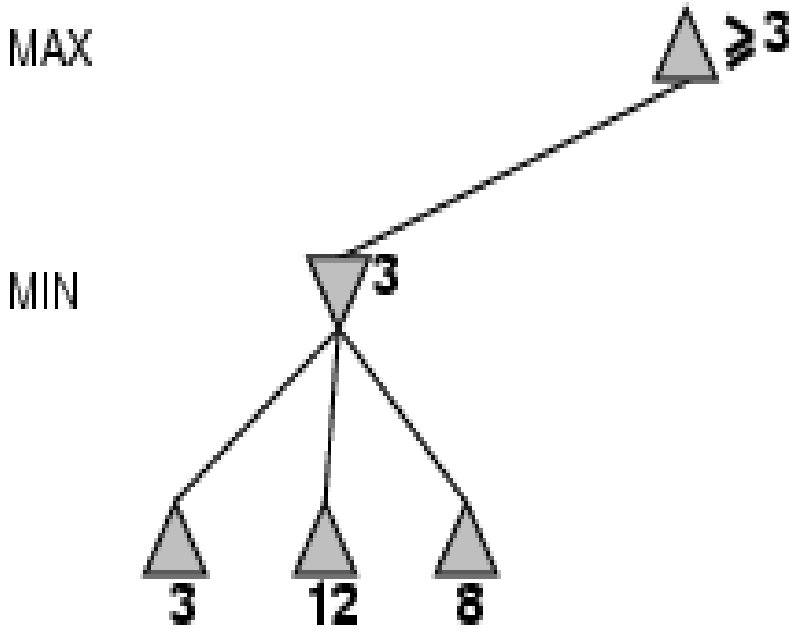
- Alpha: a lower bound on the value that a max node may ultimately be assigned

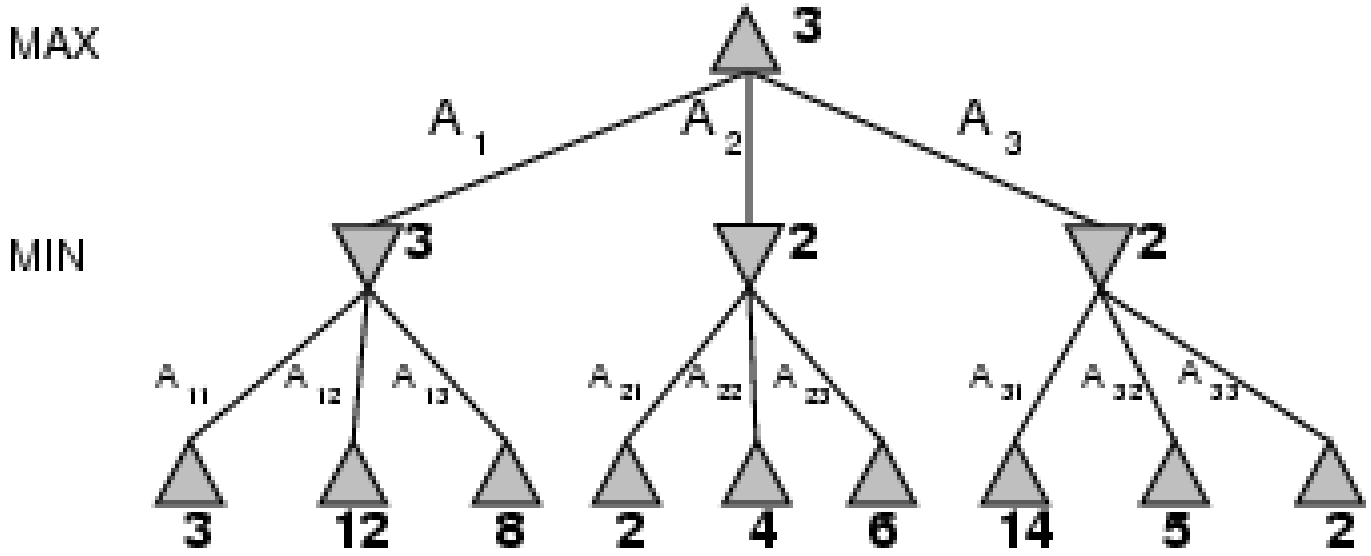
$$v \geq \alpha$$

هو:

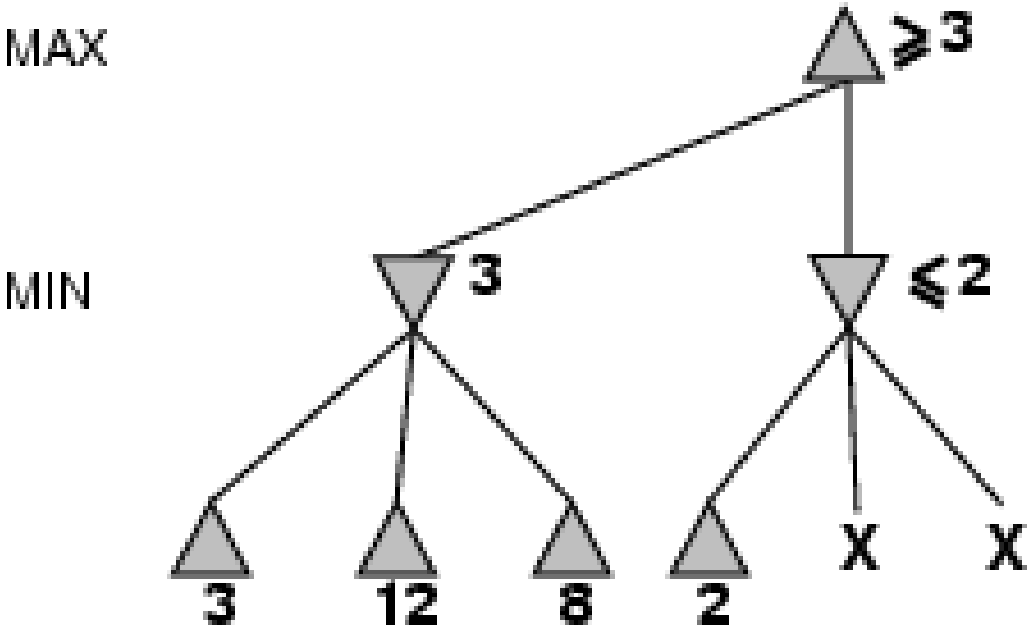
شرط اعتبار أية عقدة جديدة كجزء من ممر الحل المحتمل
حيث N القيمة التقديرية لوزن العقدة قيد البحث

α-β pruning example



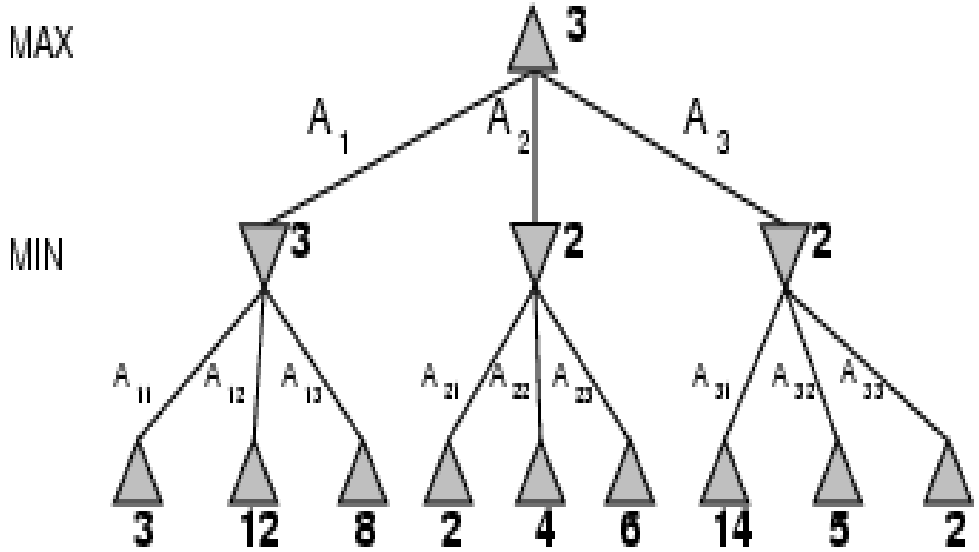


α-β pruning example

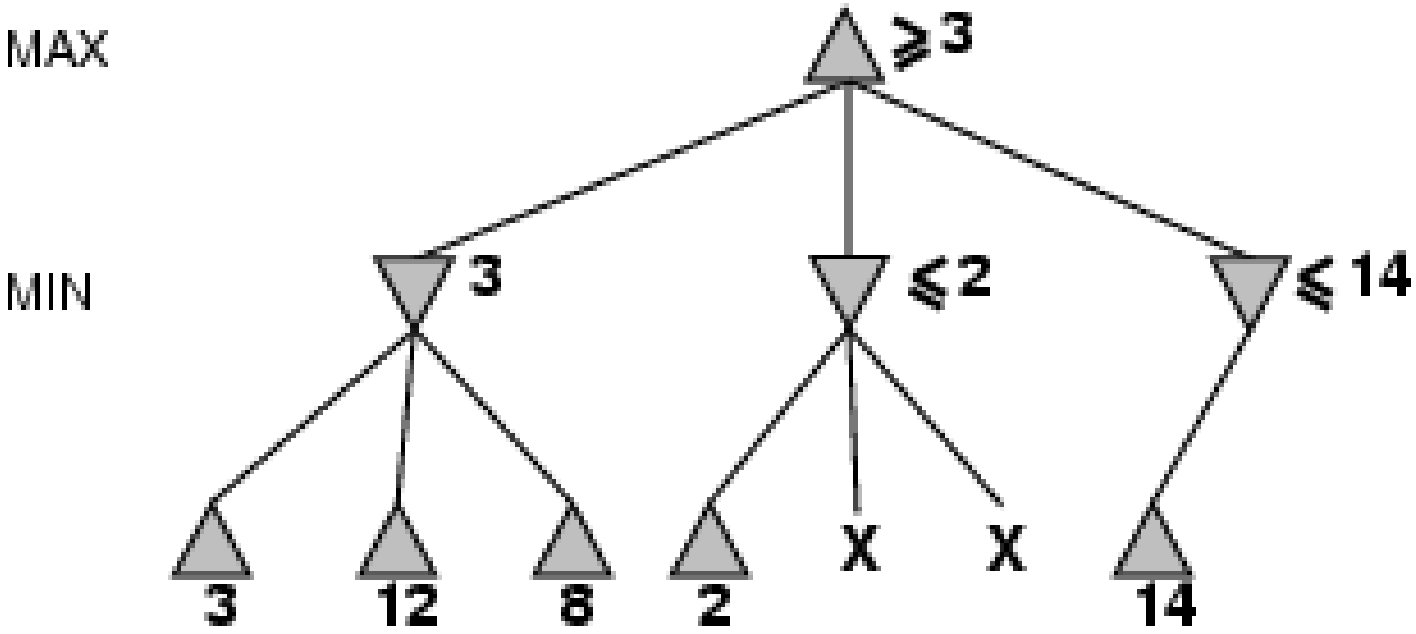


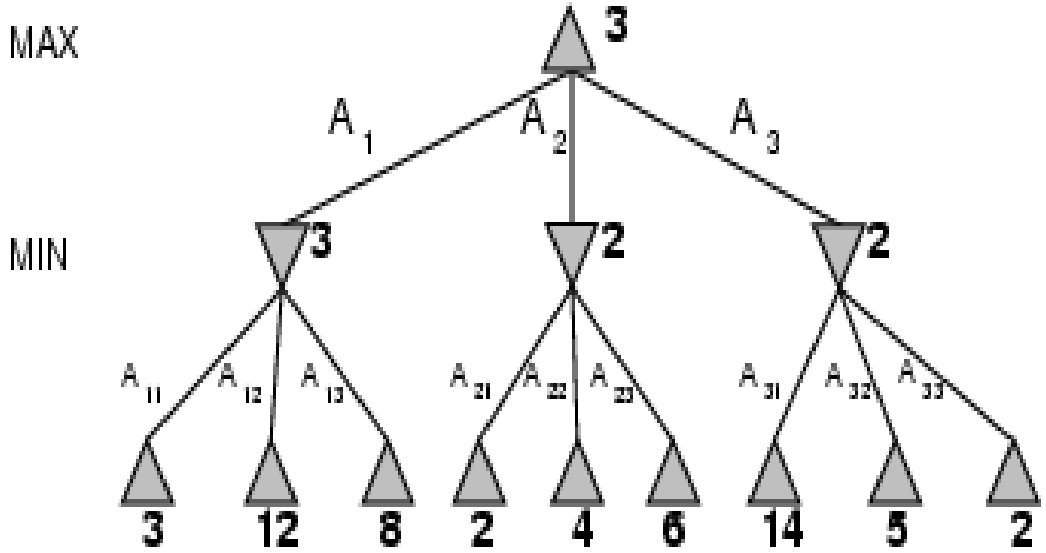
α = 3

alpha cutoff

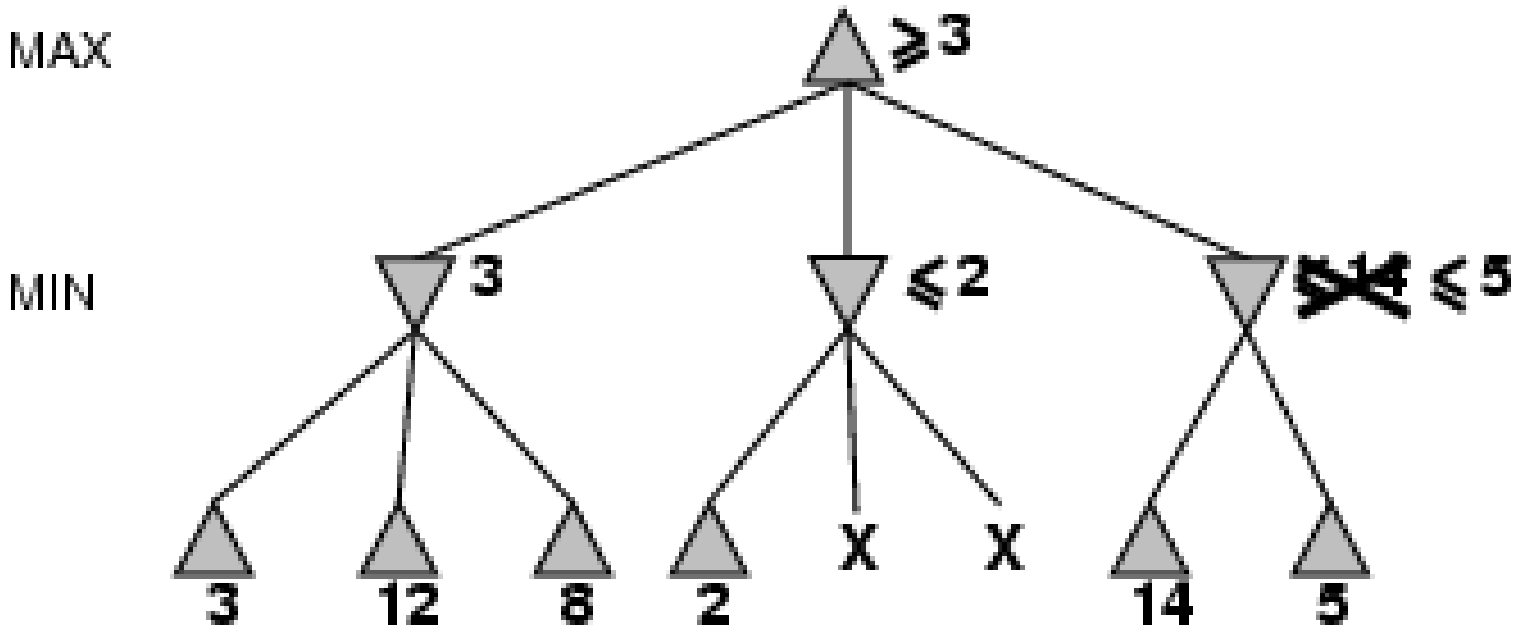


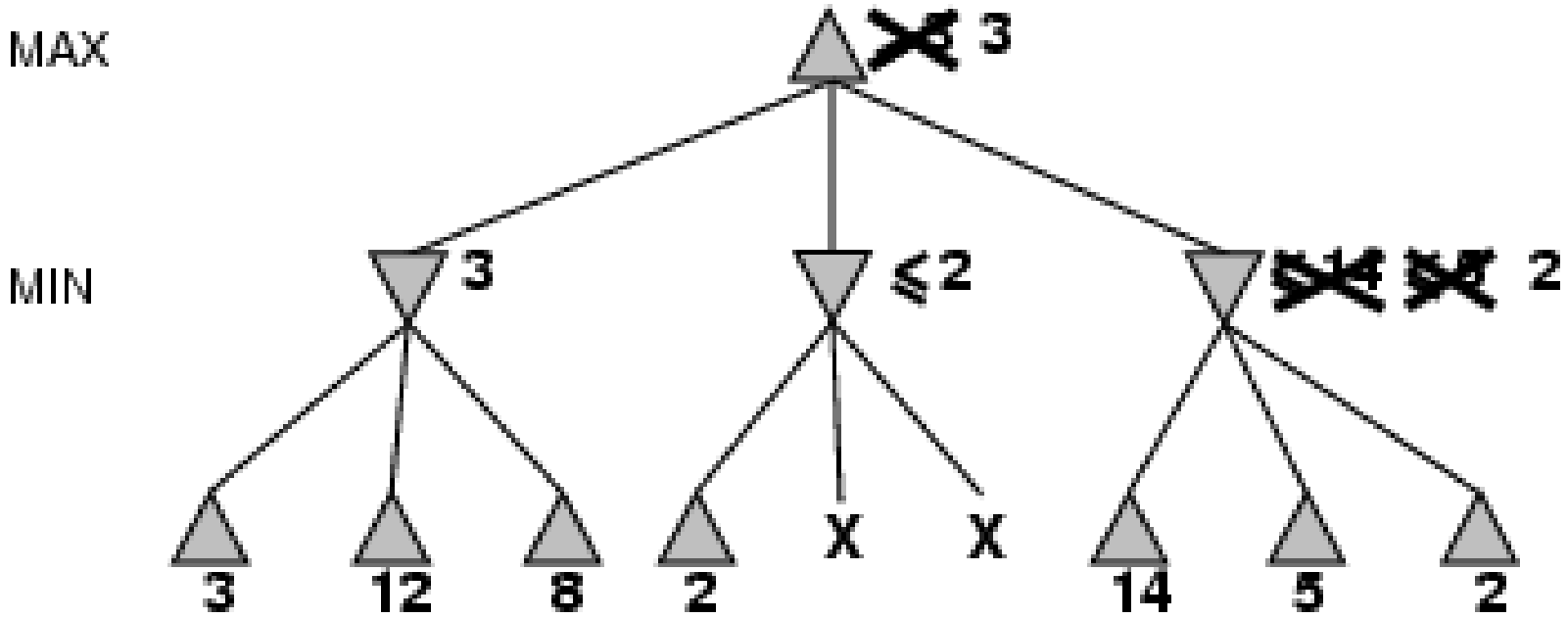
α-β pruning example

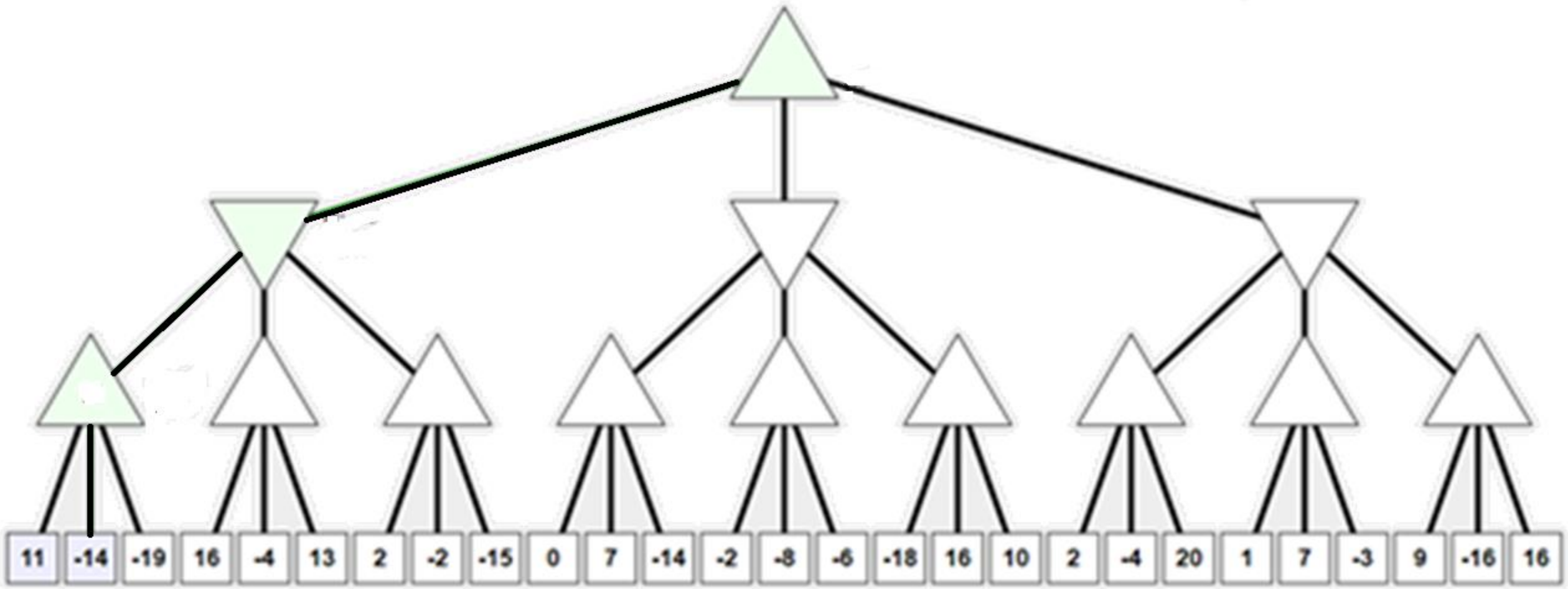


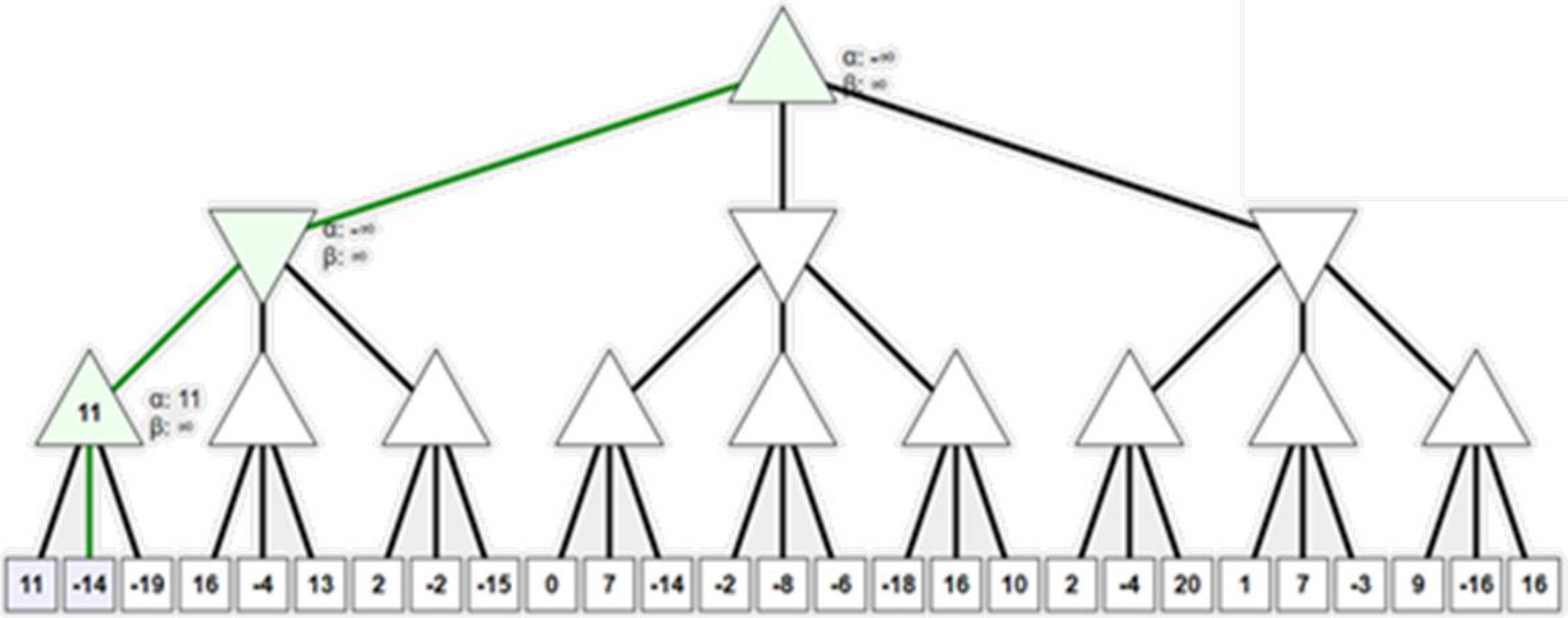


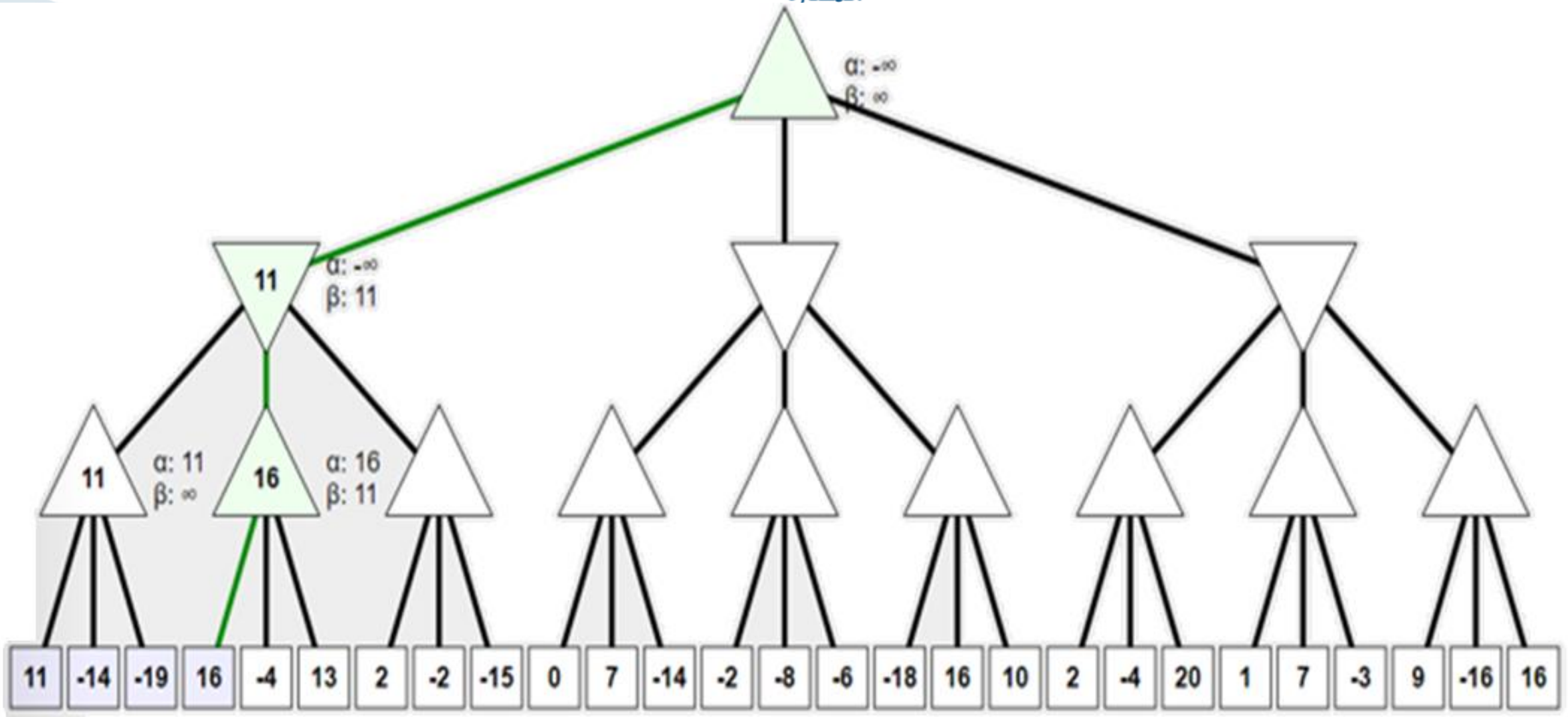
α-β pruning example

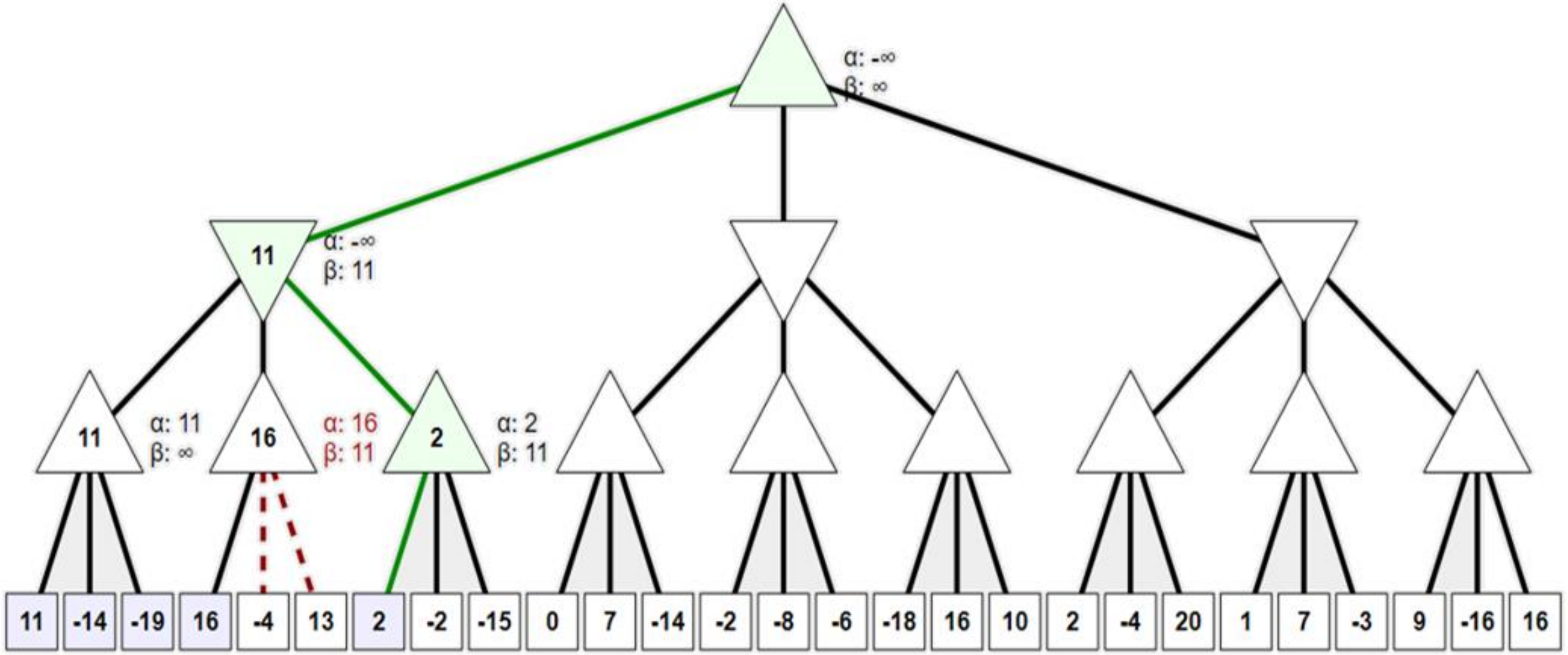


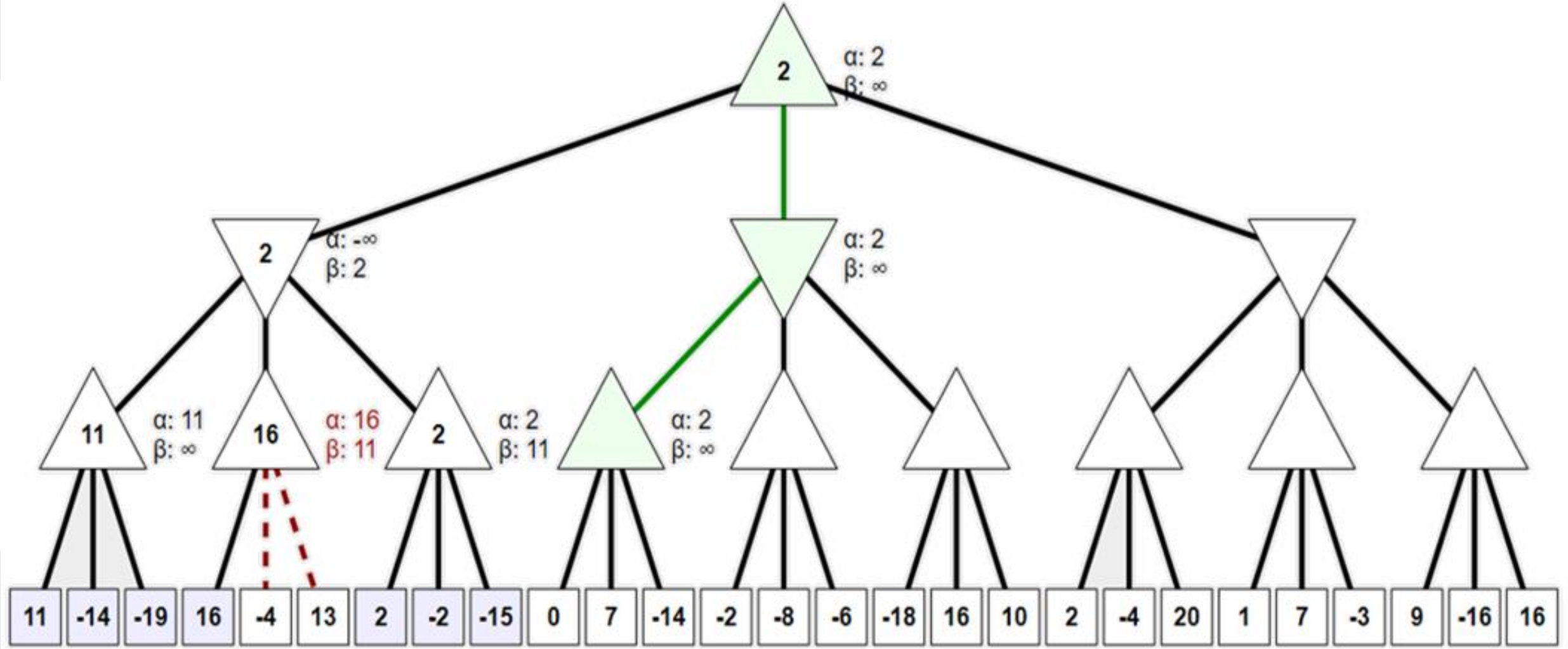

MAX
MIN
3
3
≤2
2
3
12
8
2
X
X
14
5
2

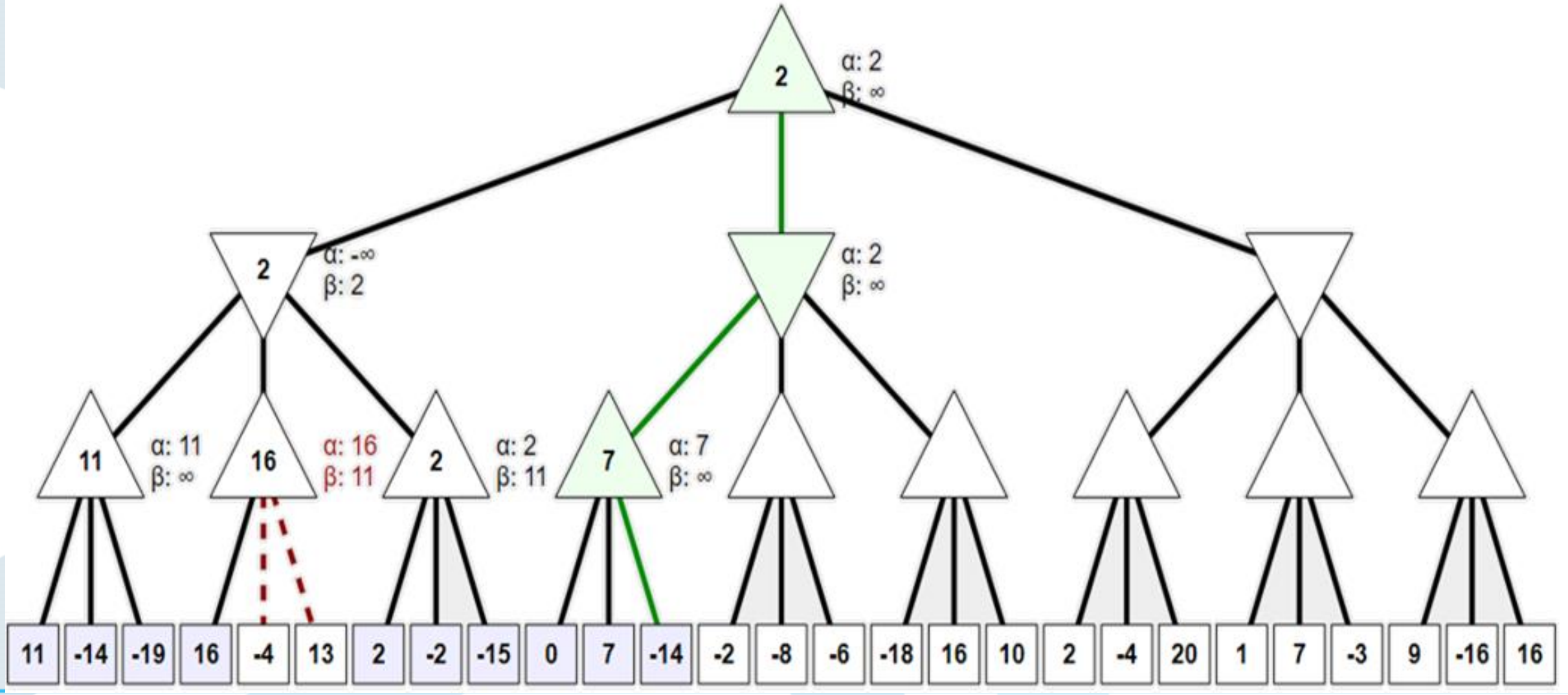

11
-14
-19
16
-4
13
2
-2
-15
0
7
-14
-2
-8
-6
-18
16
10
2
-4
20
1
7
-3
9
-16
16

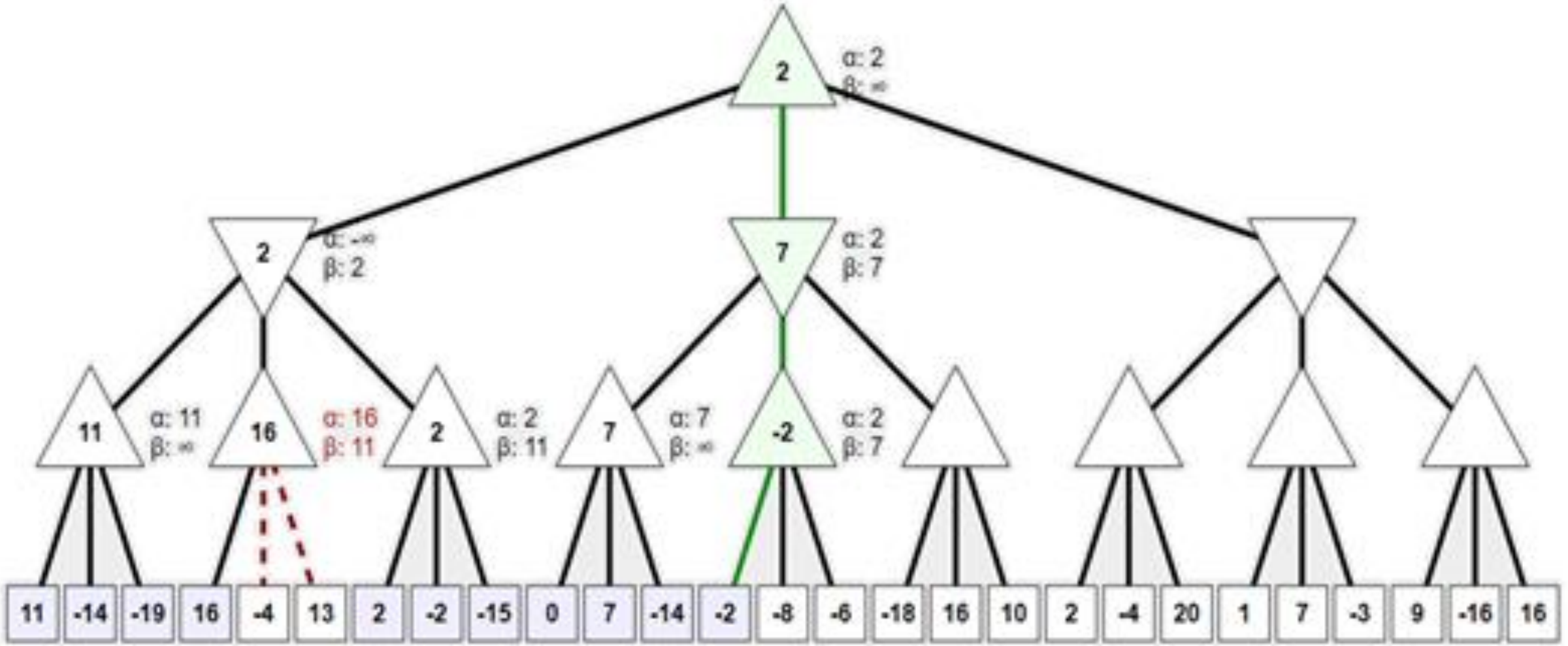

α: -∞
β: ∞
α: -∞
β: ∞
11
α: 11
β: ∞
11 -14 -19 16 -4 13 2 -2 -15 0 7 -14 -2 -8 -6 -18 16 10 2 -4 20 1 7 -3 9 -16 16

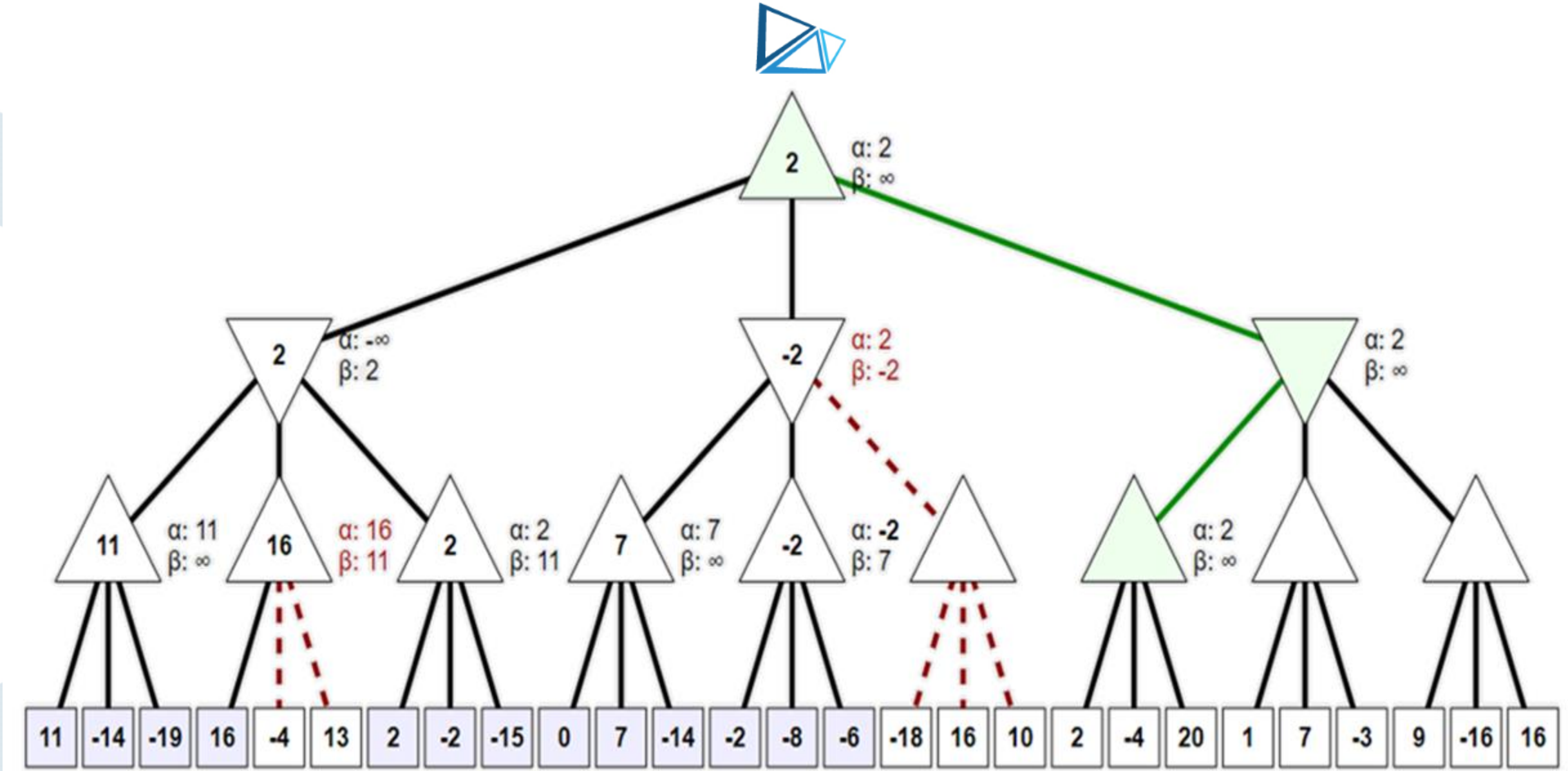

α: -∞
β: ∞
11
α: -∞
β: 11
11
α: 11
β: ∞
16
α: 16
β: 11
11 -14 -19 16 -4 13 2 -2 -15 0 7 -14 -2 -8 -6 -18 16 10 2 -4 20 1 7 -3 9 -16 16

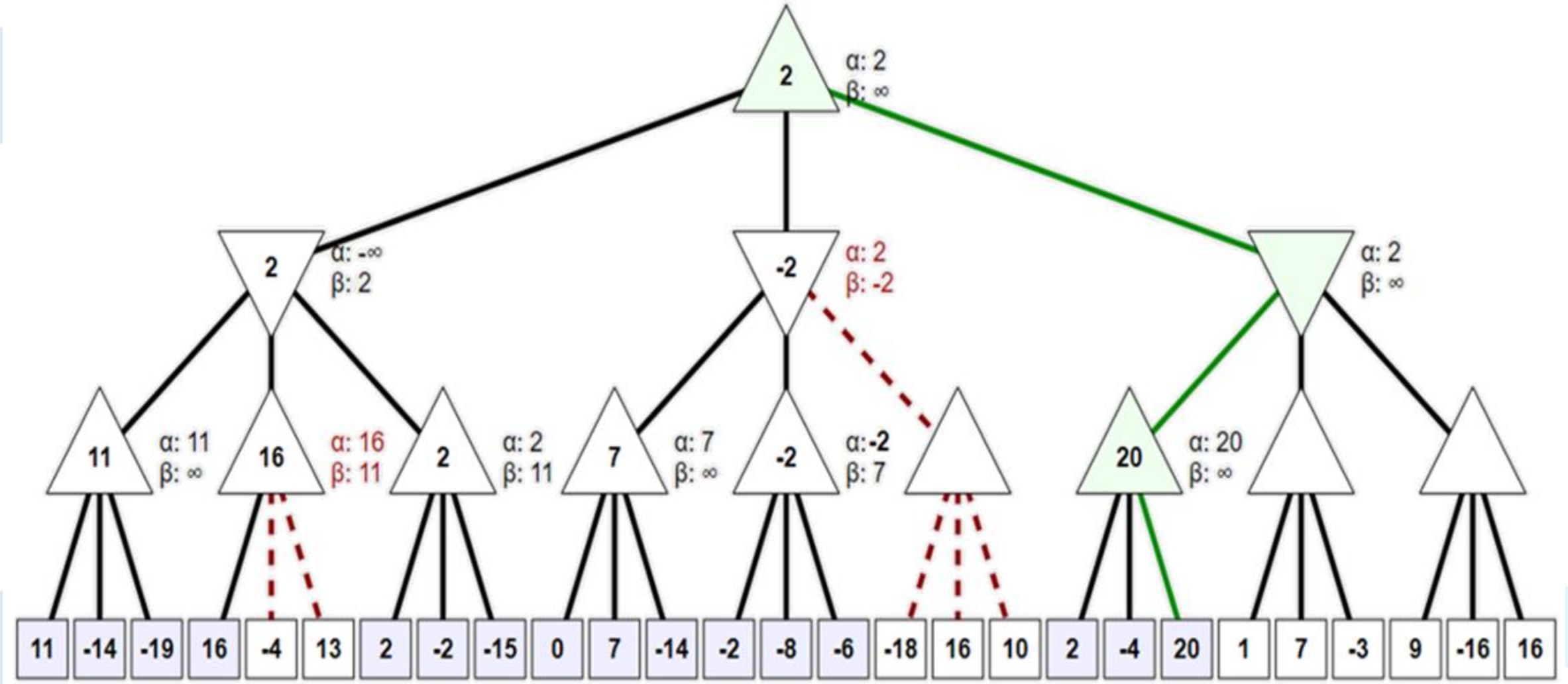

α: -∞
β: ∞
11
α: -∞
β: 11
11
α: 11
β: ∞
16
α: 16
β: 11
2
α: 2
β: 11
11 -14 -19 16 -4 13 2 -2 -15 0 7 -14 -2 -8 -6 -18 16 10 2 -4 20 1 7 -3 9 -16 16

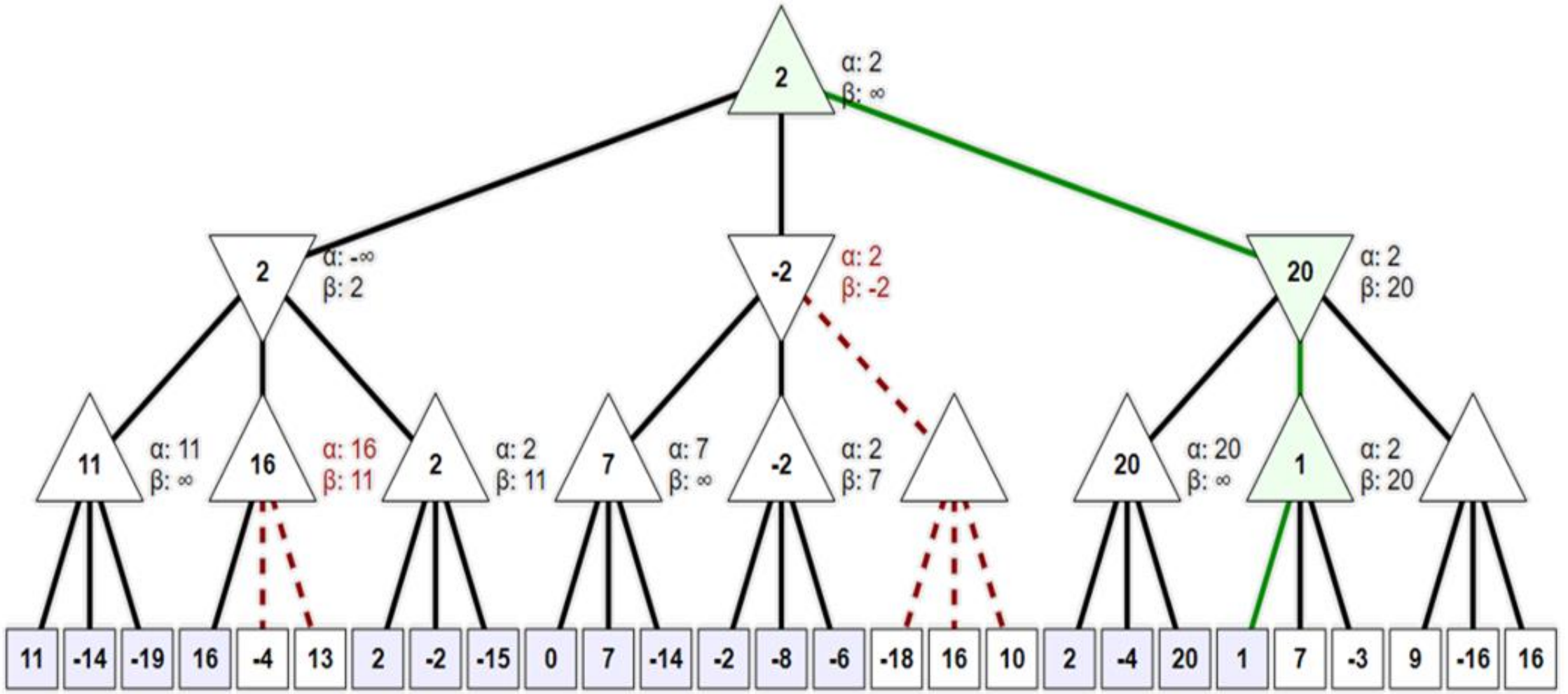

2
α: 2
β: ∞
2
α: -∞
β: 2
α: 2
β: ∞
11
α: 11
β: ∞
16
α: 16
β: 11
2
α: 2
β: 11
α: 2
β: ∞
11 -14 -19 16 -4 13 2 -2 -15 0 7 -14 -2 -8 -6 -18 16 10 2 -4 20 1 7 -3 9 -16 16

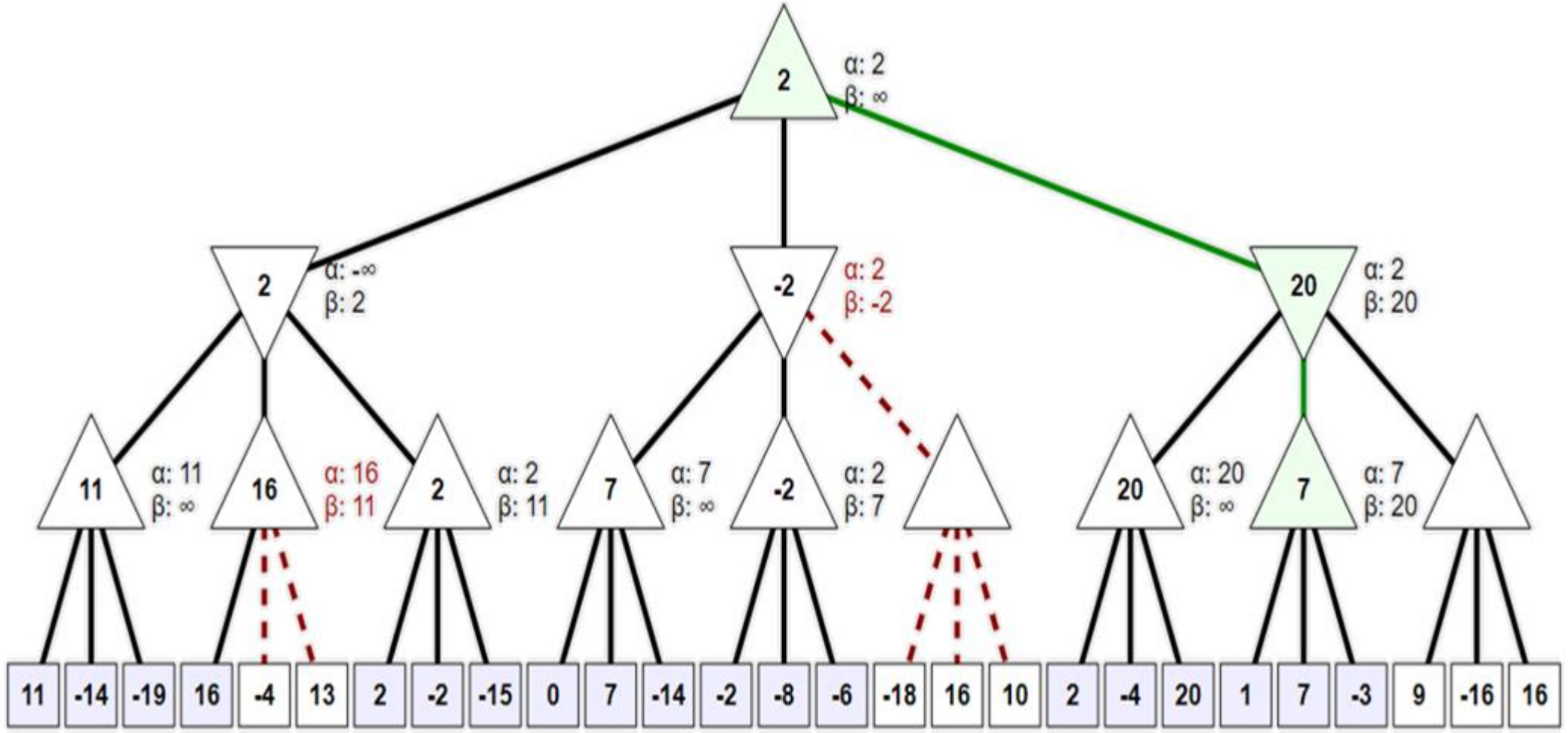

2
α: 2
β: ∞
2
α: -∞
β: 2
α: 2
β: ∞
11
α: 11
β: ∞
16
α: 16
β: 11
2
α: 2
β: 11
7
α: 7
β: ∞
11 -14 -19 16 -4 13 2 -2 -15 0 7 -14 -2 -8 -6 -18 16 10 2 -4 20 1 7 -3 9 -16 16

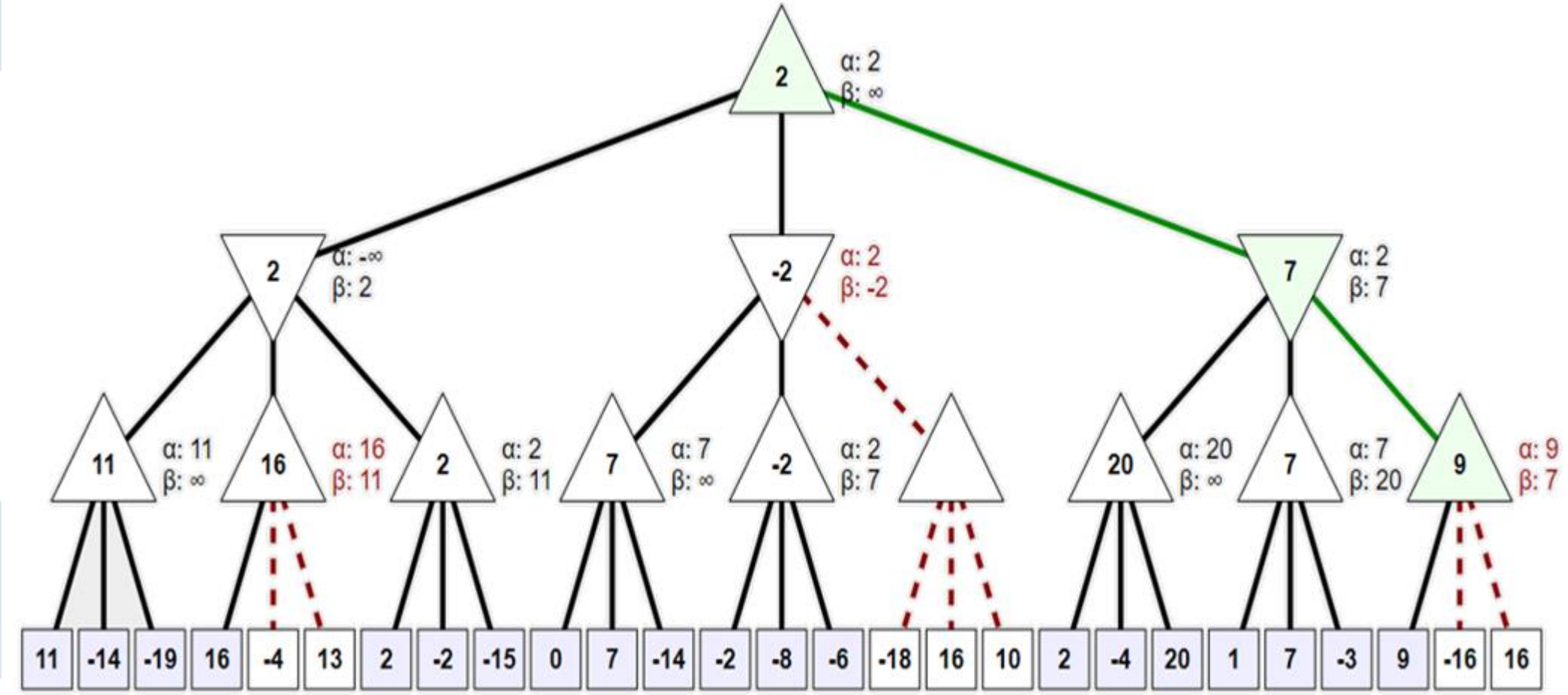

2
α: 2
β: ∞
2
α: -∞
β: 2
7
α: 2
β: 7
11
α: 11
β: ∞
16
α: 16
β: 11
2
α: 2
β: 11
7
α: 7
β: ∞
-2
α: 2
β: 7
11 -14 -19 16 -4 13 2 -2 -15 0 7 -14 -2 -8 -6 -18 16 10 2 -4 20 1 7 -3 9 -16 16

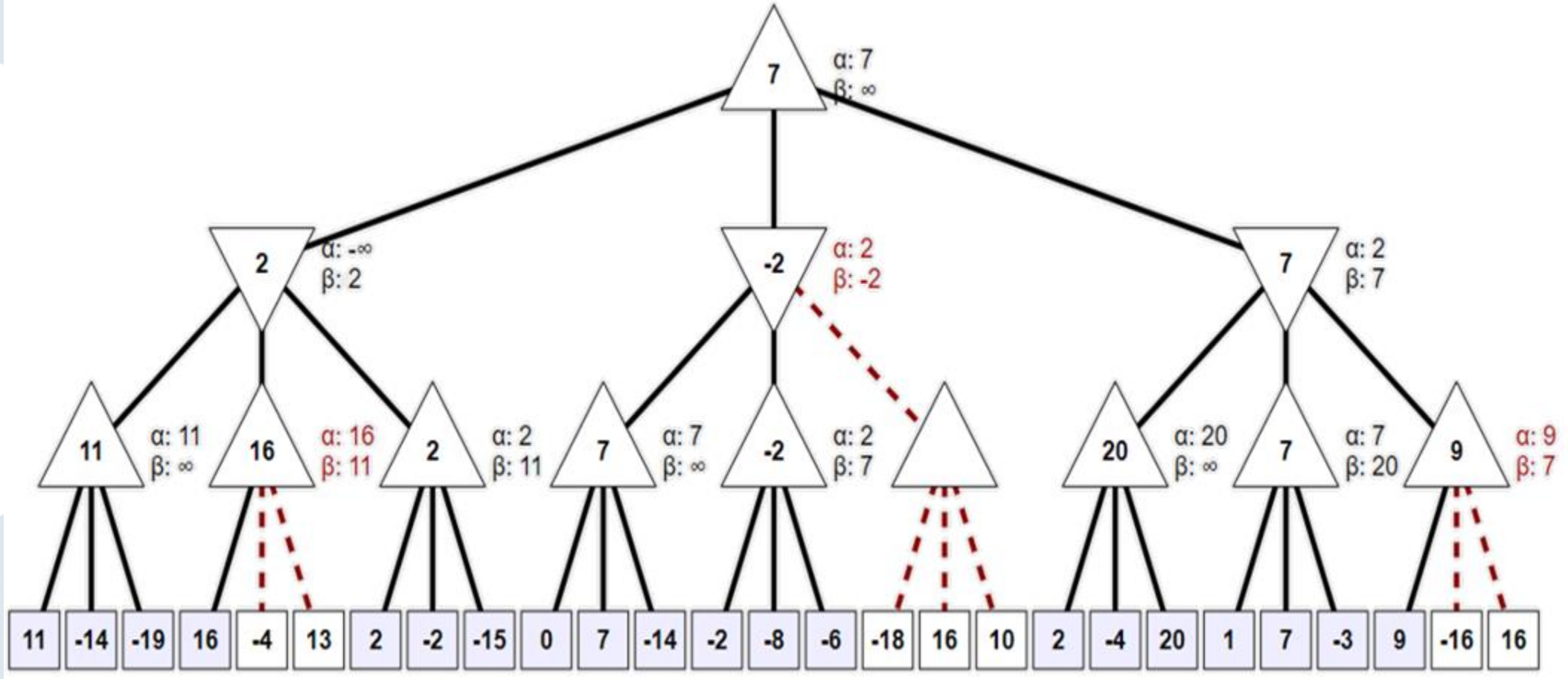

2
α: 2
β: ∞
2
α: -∞
β: 2
-2
α: 2
β: -2
α: 2
β: ∞
11
α: 11
β: ∞
16
α: 16
β: 11
2
α: 2
β: 11
7
α: 7
β: ∞
-2
α: -2
β: 7
α: 2
β: ∞
11 -14 -19 16 -4 13 2 -2 -15 0 7 -14 -2 -8 -6 -18 16 10 2 -4 20 1 7 -3 9 -16 16


2
α: 2
β: ∞
2
α: -∞
β: 2
-2
α: 2
β: -2
α: 2
β: ∞
11
α: 11
β: ∞
16
α: 16
β: 11
2
α: 2
β: 11
7
α: 7
β: ∞
-2
α:-2
β: 7
20
α: 20
β: ∞
11 -14 -19 16 -4 13 2 -2 -15 0 7 -14 -2 -8 -6 -18 16 10 2 -4 20 1 7 -3 9 -16 16


2
α: 2
β: ∞
2
α: -∞
β: 2
-2
α: 2
β: -2
20
α: 2
β: 20
11
α: 11
β: ∞
16
α: 16
β: 11
2
α: 2
β: 11
7
α: 7
β: ∞
-2
α: 2
β: 7
20
α: 20
β: ∞
1
α: 2
β: 20
11 -14 -19 16 -4 13 2 -2 -15 0 7 -14 -2 -8 -6 -18 16 10 2 -4 20 1 7 -3 9 -16 16


2
α: 2
β: ∞
2
α: -∞
β: 2
-2
α: 2
β: -2
20
α: 2
β: 20
11
α: 11
β: ∞
16
α: 16
β: 11
2
α: 2
β: 11
7
α: 7
β: ∞
-2
α: 2
β: 7
20
α: 20
β: ∞
7
α: 7
β: 20
11 -14 -19 16 -4 13 2 -2 -15 0 7 -14 -2 -8 -6 -18 16 10 2 -4 20 1 7 -3 9 -16 16


2
α: 2
β: ∞
2
α: -∞
β: 2
-2
α: 2
β: -2
7
α: 2
β: 7
11
α: 11
β: ∞
16
α: 16
β: 11
2
α: 2
β: 11
7
α: 7
β: ∞
-2
α: 2
β: 7
20
α: 20
β: ∞
7
α: 7
β: 20
9
α: 9
β: 7
11 -14 -19 16 -4 13 2 -2 -15 0 7 -14 -2 -8 -6 -18 16 10 2 -4 20 1 7 -3 9 -16 16


7
α: 7
β: ∞
2
α: -∞
β: 2
-2
α: 2
β: -2
7
α: 2
β: 7
11
α: 11
β: ∞
16
α: 16
β: 11
2
α: 2
β: 11
7
α: 7
β: ∞
-2
α: 2
β: 7
20
α: 20
β: ∞
7
α: 7
β: 20
9
α: 9
β: 7
11 -14 -19 16 -4 13 2 -2 -15 0 7 -14 -2 -8 -6 -18 16 10 2 -4 20 1 7 -3 9 -16 16

Properties of α-β

- Pruning does not affect final result. This means that it gets the exact same result as does full minimax.
- Good move ordering improves effectiveness of pruning
- With "perfect ordering," time complexity = $O(b^{m/2})$
  → doubles depth of search
- A simple example of the value of reasoning about which computations are relevant (a form of metareasoning)

Summary

- Games are fun to work on!
- They illustrate several important points about AI.
- Perfection is unattainable → must approximate.
- Game playing programs have shown the world what AI can do.