AIFA
Searching Game Trees

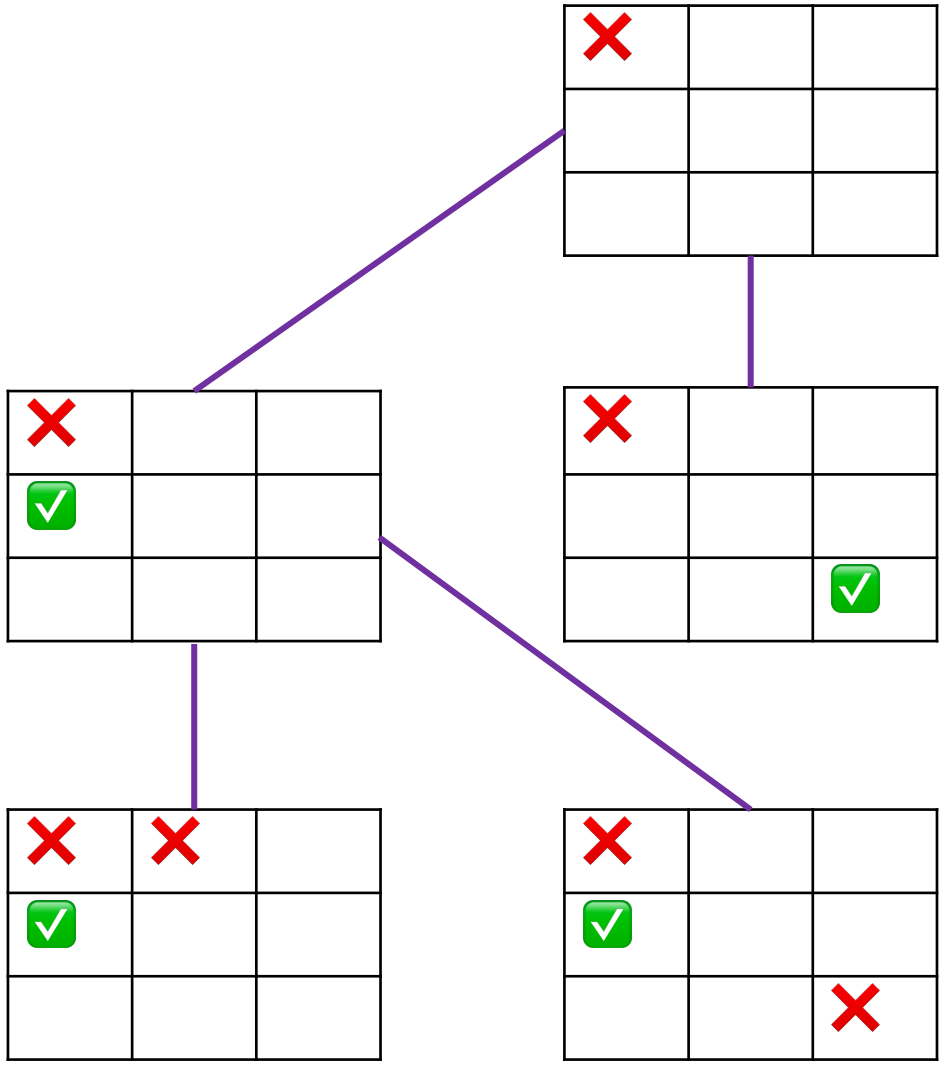
23/01/2024

Koustav Rudra

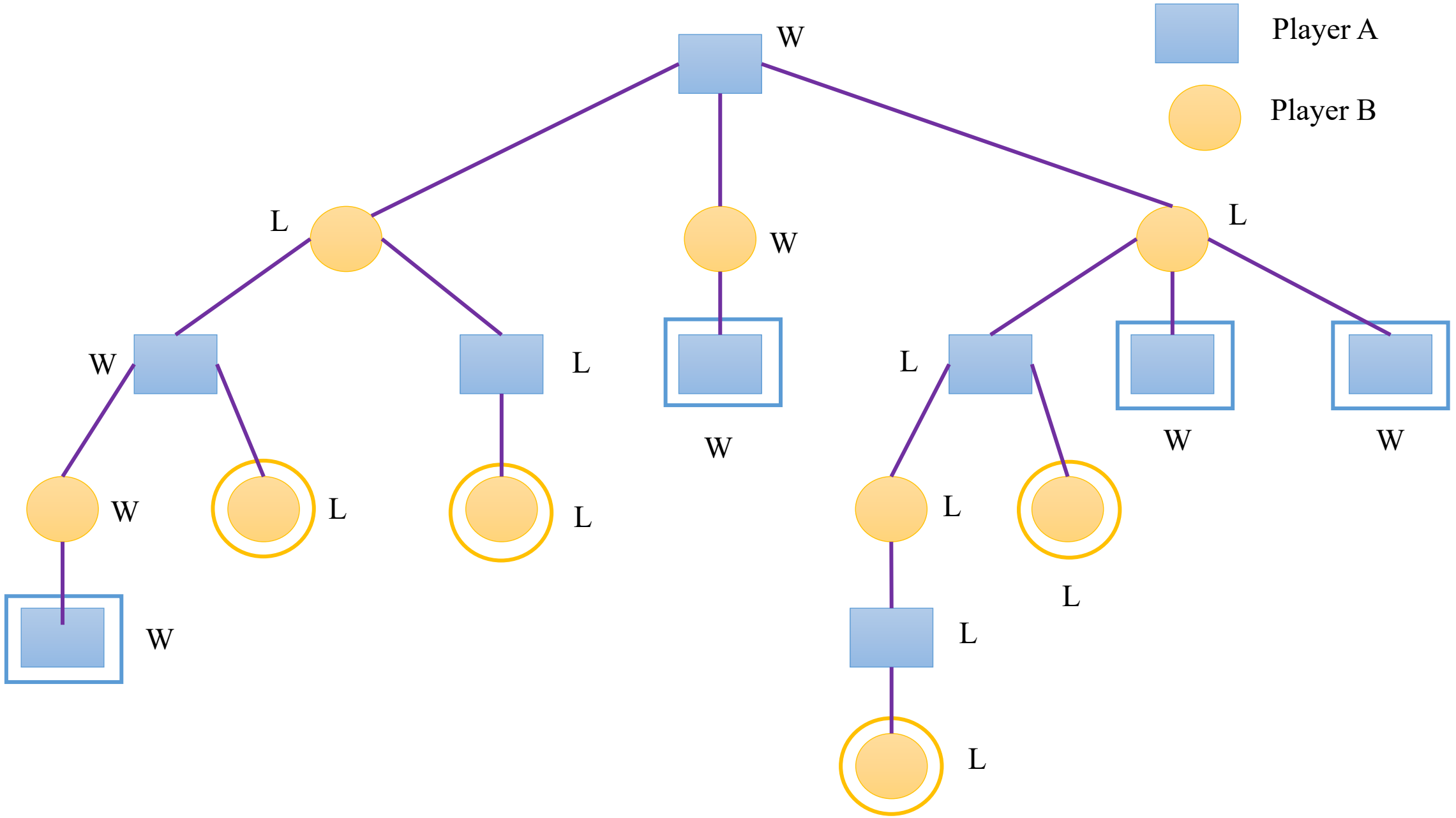
Searching Game Trees

- Consider an OR tree with two types of OR nodes, namely Min nodes and Max nodes
- In Min nodes, select the min cost successor
- In Max nodes, select the max cost successor

- Terminal nodes are winning or losing states
  - It is often infeasible to search up to the terminal nodes
  - We use heuristic costs to compare non-terminal nodes



Player A


Player A
Player B
W
L
W
L
W
L
W
L
W
W
L
L
L
L
L
L
L
W
W
W

Searching Game Trees

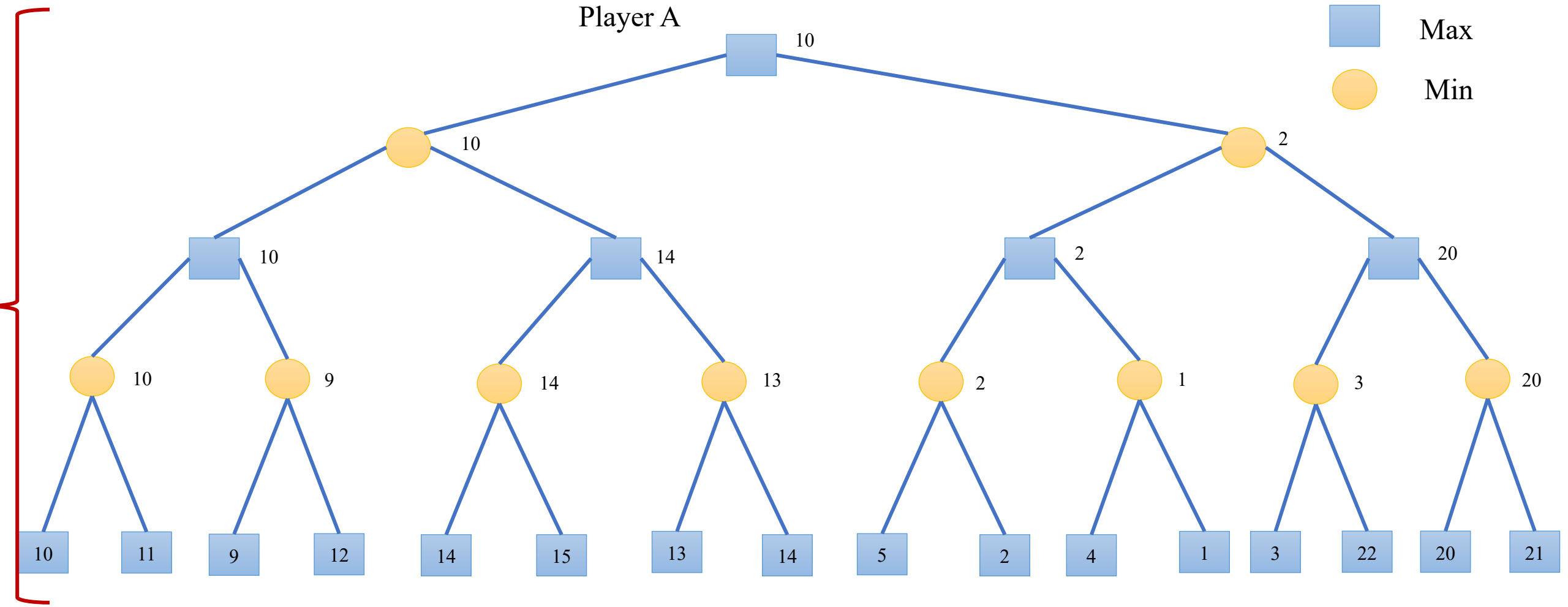
- We will expand these moves up to a certain depth

- We will have some heuristic functions to evaluate the position of the game after that many lookaheads

AIFA
MinMax Trees

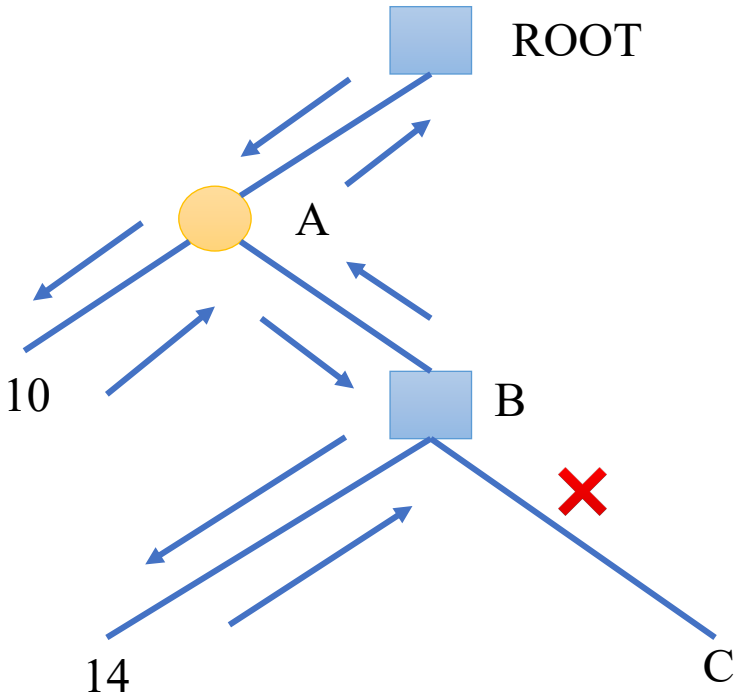
23/01/2024

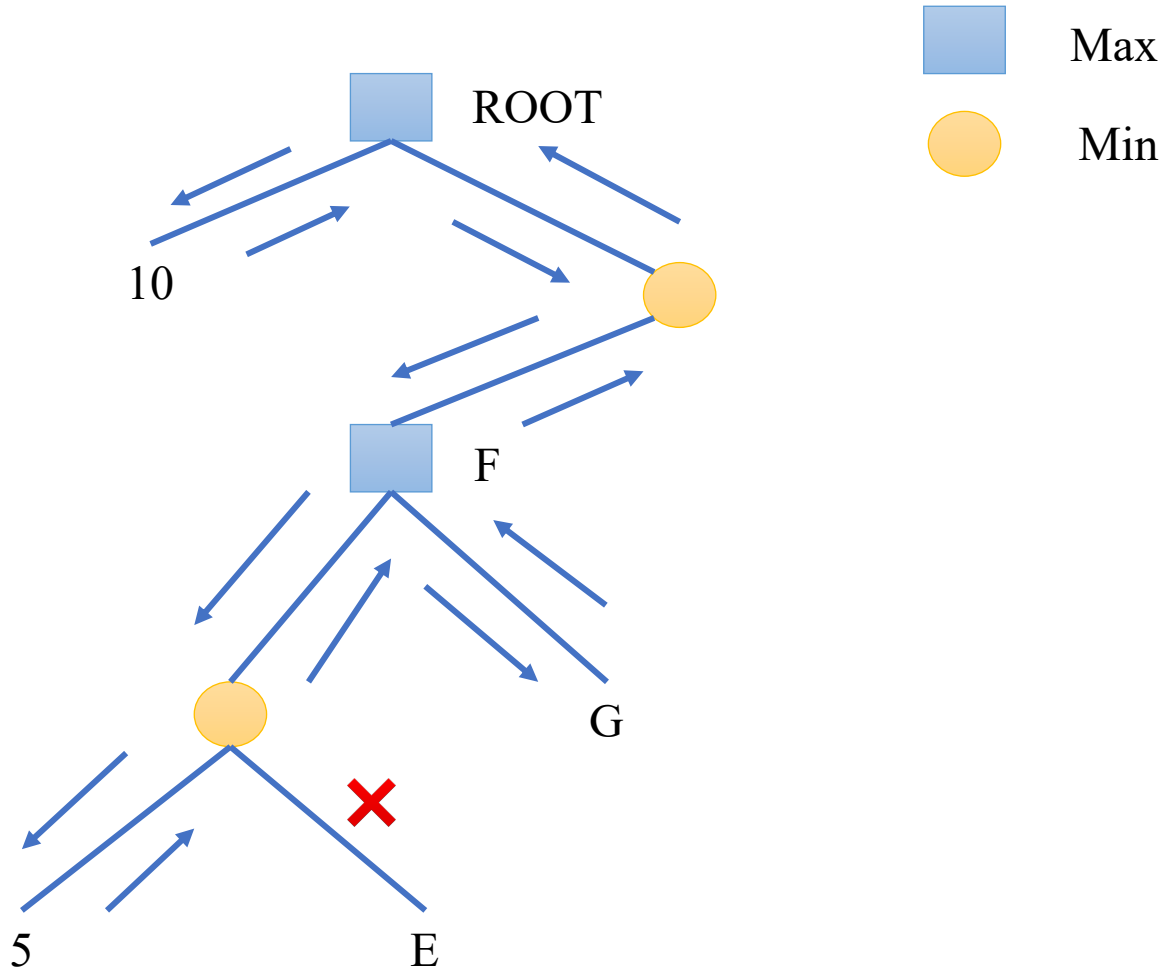
Koustav Rudra



- Looked ahead up to this many number of moves
- Found out the cost value
  - How much cost I have to incur to win the game

Shallow and Deep Pruning





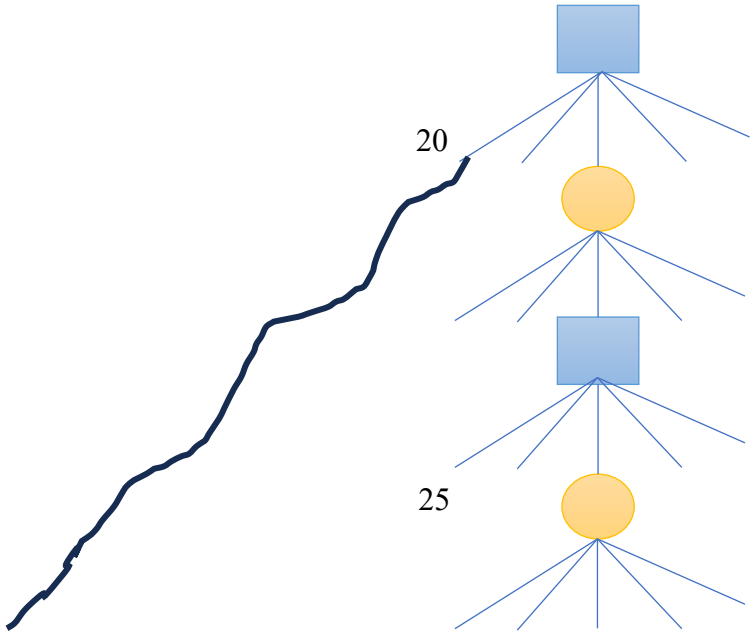
AIFA
AlphaBeta Pruning

23/01/2024

Koustav Rudra

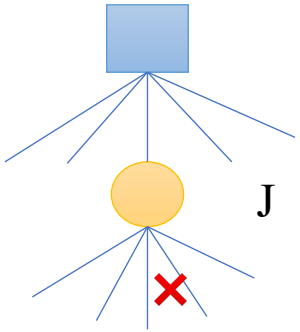
Alpha-Beta Pruning

- Alpha bound of J
  - The max current val of all MAX ancestors of J
  - Exploration of a min node, J, is stopped when its value equals or falls below alpha
  - In a min node, we update beta



What are we looking in MIN node?

Whether its current value fallen below the value backed up in the max ancestor of the node



$\alpha(J) = Current$ max $val\ of\ all\ MAX$ ancestors of J

Alpha-Beta Pruning

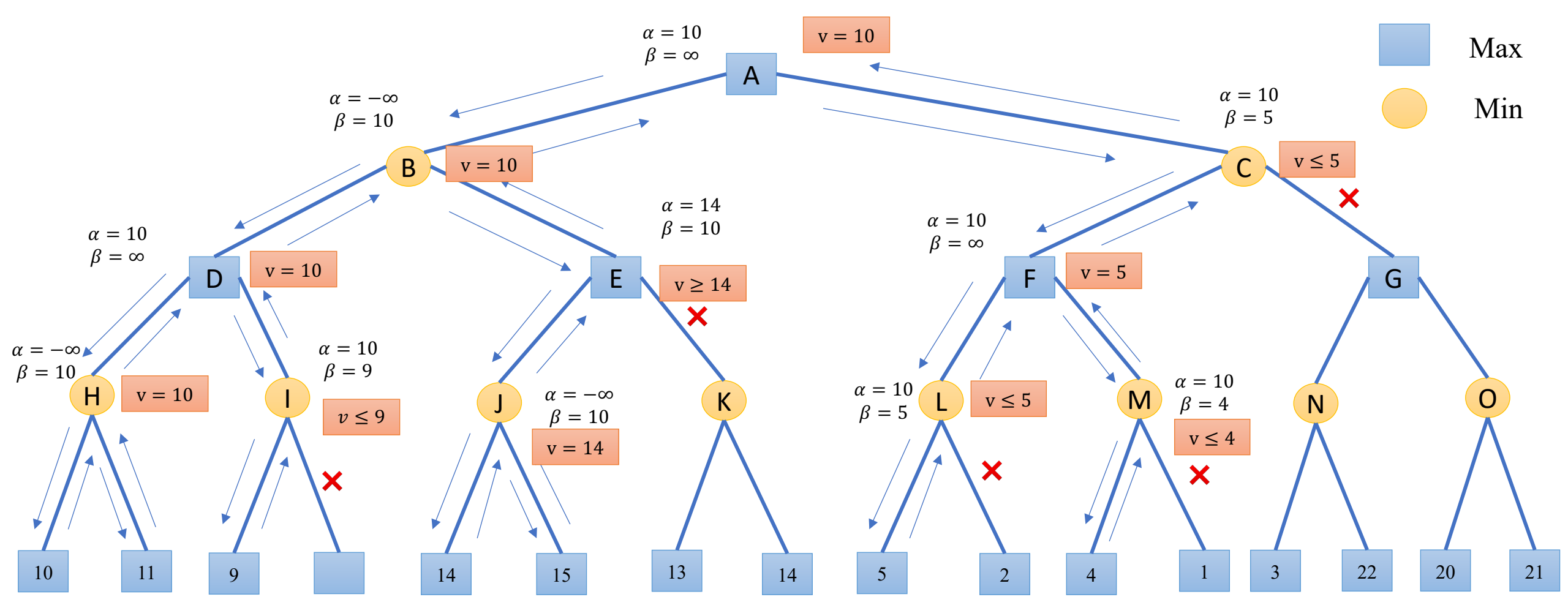
- Alpha bound of J
  - The max current val of all MAX ancestors of J
  - Exploration of a min node, J, is stopped when its value equals or falls below alpha
  - In a min node, we update beta

- Beta bound of J
  - The min current val of all MIN ancestors of J
  - Exploration of a max node, J, is stopped when its value equals or exceeds beta
  - In a max node, we update alpha

- In both min and max nodes, we return when $\alpha \geq \beta$

Alpha-Beta Pruning

- Alpha = best already explored option along path to the root for maximizer
- Beta = best already explored option along path to the root for minimizer



Compute step by step

Alpha-Beta Procedure: V(J; α, β)

1. If J is a terminal, return V(J) = h(J)

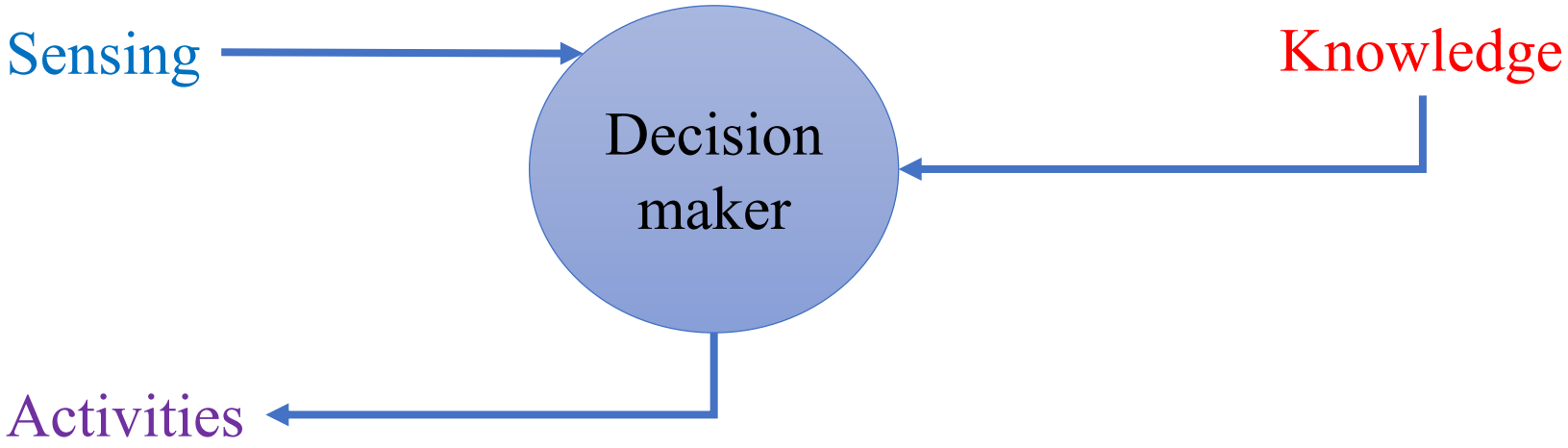
2. If J is a max node:
    1. For each successor J_k of J in succession:
        1. Set $\alpha = max \begin{Bmatrix} \alpha \\ V(J_k; \alpha, \beta) \end{Bmatrix}$
        2. If $\alpha \geq \beta$, then return β, else continue
    2. Return α
3. If J is a min node:
    1. For each successor J_k of J in succession:
        1. Set $\beta = min \begin{Bmatrix} \beta \\ V(J_k; \alpha, \beta) \end{Bmatrix}$
        2. If $\alpha \geq \beta$, then return α, else continue
    2. Return β

Knowledge Based System: Logic and Deduction

23/01/2024

Koustav Rudra

Knowledge and Intelligence



How to act given a particular scenario in the environment?

Machine: It is mandatory to have means of representing knowledge

How to represent knowledge in a way that machine can understand?

Represent knowledge in a machine

- We need a language to **represent** domain knowledge
  - Expect a machine to demonstrate an intelligent behaviour when that machine is left to work in a particular environment in a particular domain, provided we empower the machine with relevant knowledge from that domain

- There must be a method to use the knowledge
  - Understand the knowledge in which it is expressed

- **Inference**
  - Interpret knowledge in response to environmental fact that has been sensed

- **Syntax and semantics of language**
  - Grammar of a language
  - Laughs(Anil) == ?
  - Likes(Ashok, Akash) == ?

Logic is one such formal language

Logic

- A formal system for describing states of affairs, consisting of:
  - Syntax: describes how to make sentences, and
  - Semantics: describes the relation between the sentences and states of affairs

- Propositional Logic
- First Order Logic
- Temporal Logic
- Fuzzy Logic

Logical Deduction Propositional Logic

23/01/2024

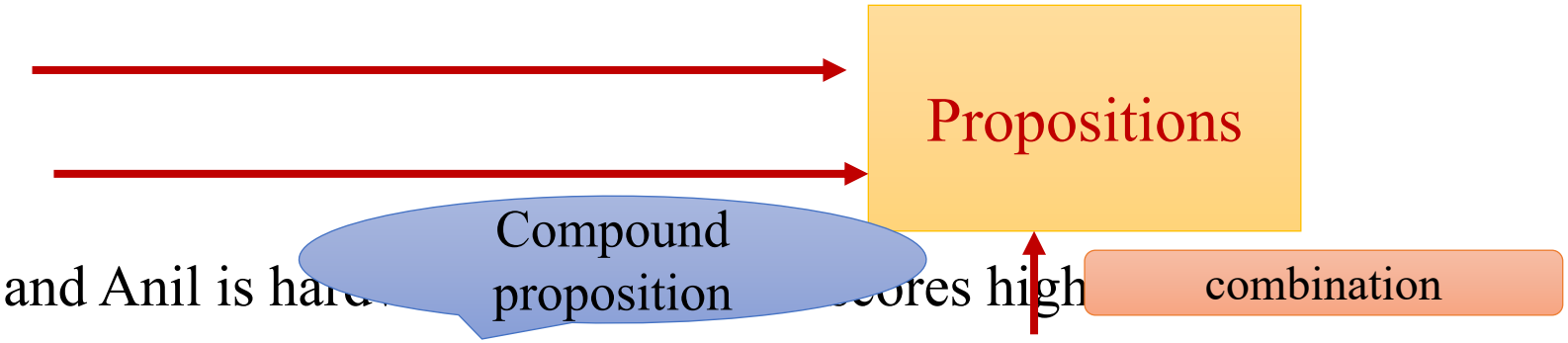
Koustav Rudra

Objective

- How to represent simple facts in the language of propositional logic?
- How can we interpret propositional logic statement?
- How to compute the meaning of compound proposition?

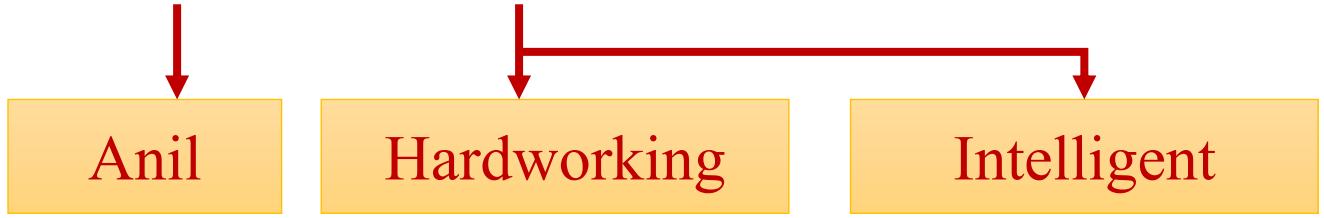
Propositional Logic

- Anil is intelligent
- Anil is hardworking



- If Anil is intelligent and Anil is hard[illegible]scores high

- Objects and Relations



- A Proposition (statement) can either be True or False

- Intelligent_Anil == Anil is intelligent
- Hardworking_Anil == Anil is hardworking

Towards the Syntax

- Let P stands for Intelligent_Anil
- Let Q stands for Hardworking_Anil

- What does P ∧ Q (P and Q) mean?
- What does P ∨ Q (P or Q) mean?

- P ∧ Q and P ∨ Q are compound propositions

Syntactic Elements of Propositional Logic

- Vocabulary
  - A set of propositional symbols (P, Q, R, etc.) each of which can be True or False

  - Set of **logical operators**
    - ∧ (AND), ∨ (OR), ~(NOT), → (implies)
    - Parenthesis () used for grouping

  - There are two special symbols
    - TRUE (T) and FALSE (F)
    - These are **logical constants**

How to form propositional sentences?

- Each symbol (a proposition or a constant) is a sentence

- If P is a sentence and Q is a sentence then
  - (P) is a sentence
  - P∧Q is a sentence
  - P∨Q is a sentence
  - ~P is a sentence
  - P→Q is a sentence

Sentences are called well-formed formulae

Propositional Logic

- Given a set of atomic propositions AP
- Sentence → Atom | ComplexSentence
- Atom → True | False | AP
- ComplexSentence → (Sentence)
  - | Sentence Connective Sentence
  - | ~ Sentence
- Connective →∧ | ∨ | → | ⇔

Implication →

- P→Q

- If P is true then Q is true

- If it rains then the roads are wet

Equivalence (⇔)

- $P \Leftrightarrow Q$

- If P is True then Q is True and If Q is True then P is True

- If two sides of a triangle are equal then two base angles of the triangle are equal

- $(P \rightarrow Q) \wedge (Q \rightarrow P)$

Thank You