Chapter 6
Machine Leaning

Compiled By: Bal Krishna Nyaupane
balkrishnanyaupane@gmail.com

What is Learning?

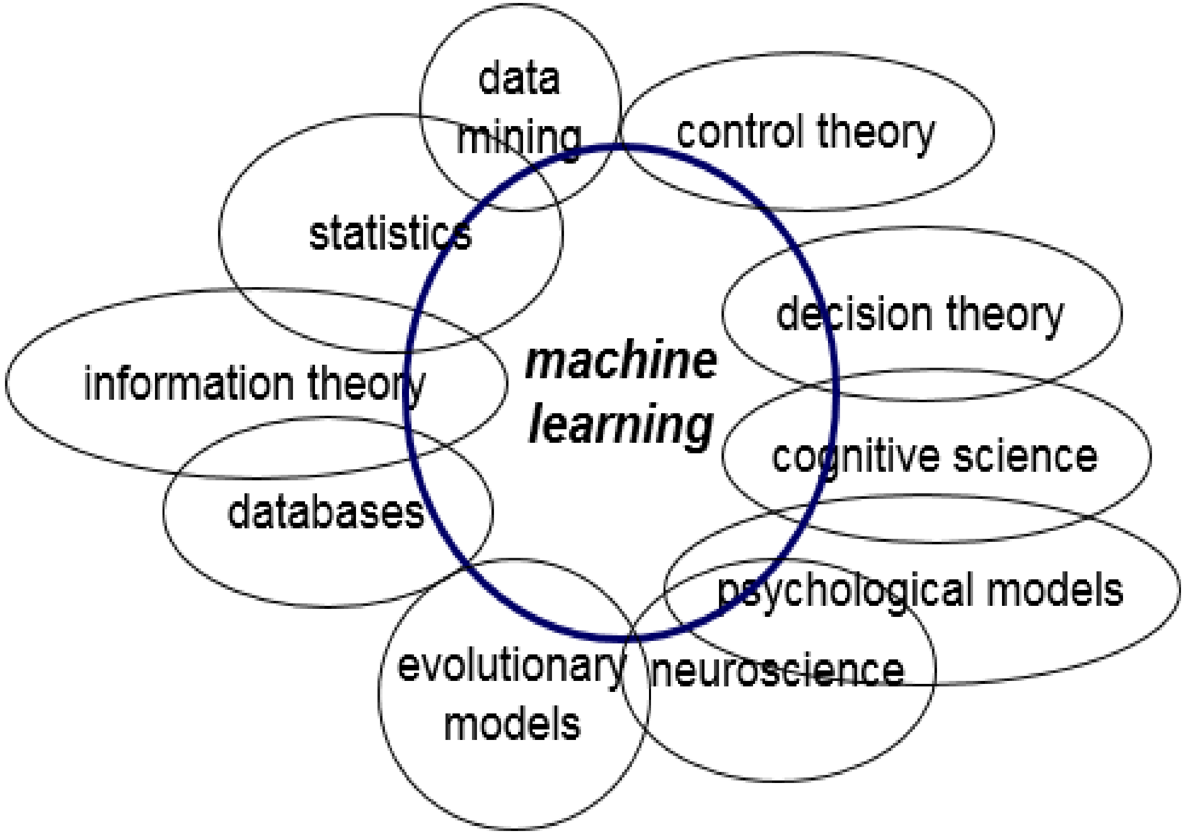
- “Learning is constructing or modifying representations of what is being experienced.” - Ryszard Michalski
- “Learning is making useful changes in our minds.” - Marvin Minsky
- “Learning denotes changes in the system that are adaptive in the sense that they enable the system to do the same task (or tasks drawn from the same population) more effectively the next time.”- Herbert Simon
- "Learning is making useful changes in our minds." --Marvin Minsky
- Learning is acquiring new or modifying existing knowledge, behaviors, skills and may involve synthesizing different types of information
- **Learning involves three factors:**
  - **Changes:** Learning changes the learner. For machine learning the problem is determining the nature of these changes and how to best represent them.
  - **Generalization:** Learning leads to generalization. Performance must improve not only on the same task but on similar tasks.
  - **Improvement:** Learning leads to improvements. Machine learning must address the possibility that changes may degrade performance and find ways to prevent it.

Machine Learning

- Machine learning is a branch of AI that uses algorithm to allow computer to evolve behaviors based on data collected from databases or gathered through sensors.
- A branch of artificial intelligence, concerned with the design and development of algorithms that allow computers to evolve behaviors based on empirical data.
- Machine learning focuses on prediction based on known properties learned from the training data.
- The performance is usually evaluated learned with respect to reproduce known knowledge.
- Machine learning refers to a system capable of the autonomous acquisition and integration of knowledge.
- Machine learning is primarily concerned with the accuracy and effectiveness of the computer system.

Why Machine Learning?

- **Application**
  - Autonomous robot control
  - Data mining and bioinformatics
  - Playing games
  - Fault detection
  - Spam email detection
  - Credit scoring, fraud detection
  - Web mining: search engines
  - Market basket analysis,
  - Recognizing patterns:
    - Facial identities or facial expressions
    - Handwritten or spoken words
    - Medical images
  - Information retrieval:
    - Find documents or images with similar content.
  - Data Visualization:
    - Display a huge database in a revealing way



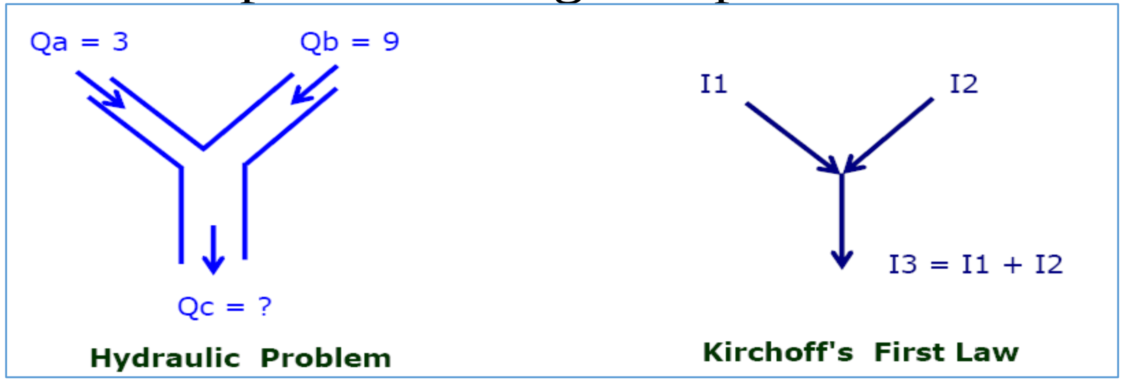
Type of Learning

- **Rote Learning**
  - Rote learning is a simple method used in machine learning, although it does not involve repetition like in the traditional sense of rote learning; instead, it simply makes use of memory to store new knowledge and provides this knowledge to the machine to avoid repeated execution or calculation of the same function on the same input.
  - It is a technique which focuses on memorization. Memorization - saving new knowledge to be retrieved when needed rather than calculated.
  - It avoids understanding the inner complexities and inferences of the subject that is being learned.
  - It works by taking problems that the performance element has solved and memorizing the problem and the solution.
  - Only useful if it takes less time to retrieve the knowledge than it does to recomputed it
  - **Example**: A.L. Samuels Checkers Player (1959-1967). It is a program that knows and follows the rules of checkers. It memorizes and recalls board positions it has encountered in previous games.

Type of Learning

- **Learning by Analogy**

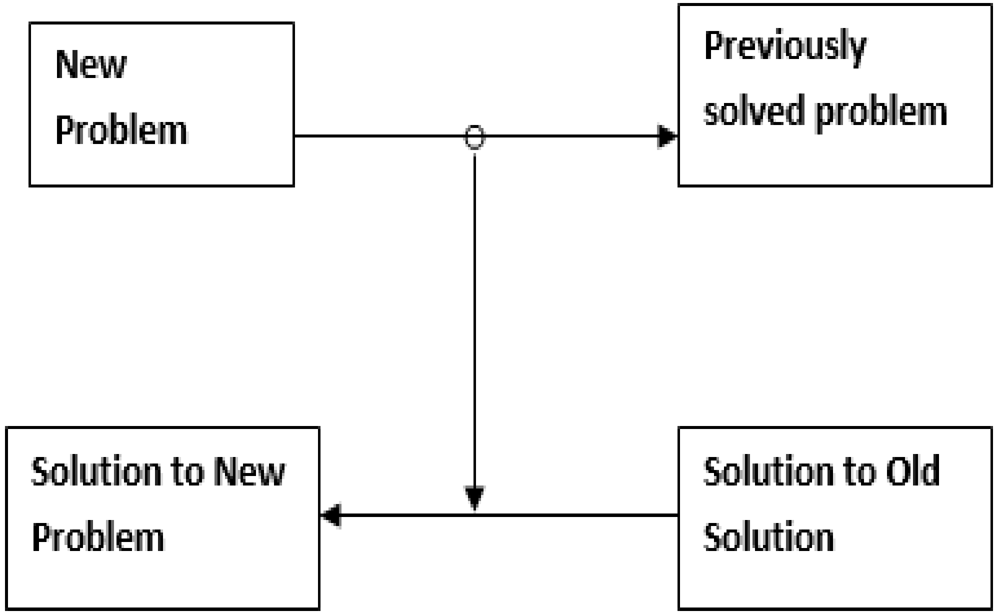
- Learning by analogy means acquiring new knowledge about an input entity by transferring it from a known similar entity.
- This technique transforms the solutions of problems in one domain to the solutions of the problems in another domain by discovering analogous states and operators in the two domains.
- Examples of Analogy Learning: Infer by analogy the hydraulics laws that are similar to Kirchhoff's laws. Pressure drops like voltage drops



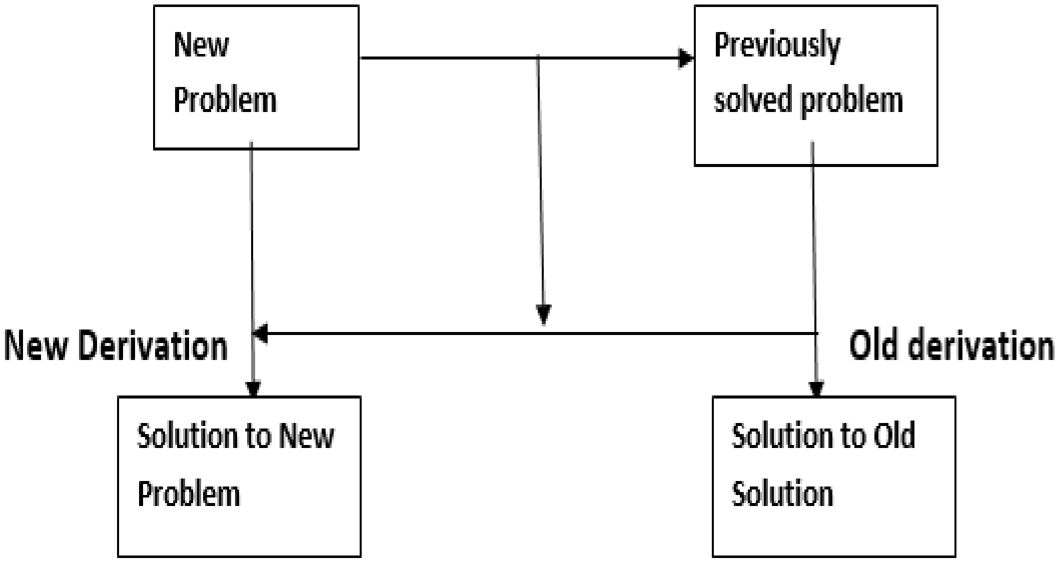
- Hydrogen atom is like our solar system. The Sun has a greater mass than the Earth and attracts it, causing the Earth to revolve around the Sun. The nucleus also has a greater mass then the electron and attracts it. Therefore it is plausible that the electron also revolves around the nucleus

Type of Learning

- **Transformational Analogy:** You might look for a previous theorem that is very similar and copy its proof, making substitutions when necessary. The idea is to transform a solution to a previous problem in to solution for the current problem.



- **Derivational Analogy**: Notice that transformational analogy does not look at how the old problem was solved, it only looks at the final solution. Often the twists and turns involved in solving an old problem are relevant to solving a new problem. The detailed history of problem solving episode is called derivation, Analogical reasoning that takes these histories into account is called derivational analogy.



Type of Learning

- **Explanation Based Learning (EBL)**
  - In the case of differentiation, memorization would remember that the derivative of X^2 with respect to X is 2X. We would like to be able to extract the general rule that for any arithmetic unknown u, the derivative of u^2 with respect to u is 2u.
  - The basic idea behind EBL is first to construct an explanation of the observation using prior knowledge, and then to establish a definition of the class of cases for which the same explanation structure can be used.
  - **The basic EBL process works as follows:**
    1. Given an example, construct a proof that the goal predicate applies to the example using the available background knowledge.
    2. In parallel, construct a generalized proof tree for the variabilized goal using the same inference steps as in the original proof.
    3. Construct a new rule whose left-hand side consists of the leaves of the proof tree, and whose right-hand side is the variabilized goal.
    4. Drop any conditions that are true regardless of the values of the variables in the goal.

Domain Theory

A(x) y B(x) -> C(x)
D(x) y E(y) -> F(x,y)
C(x) y F(x,y) -> G(x,y)

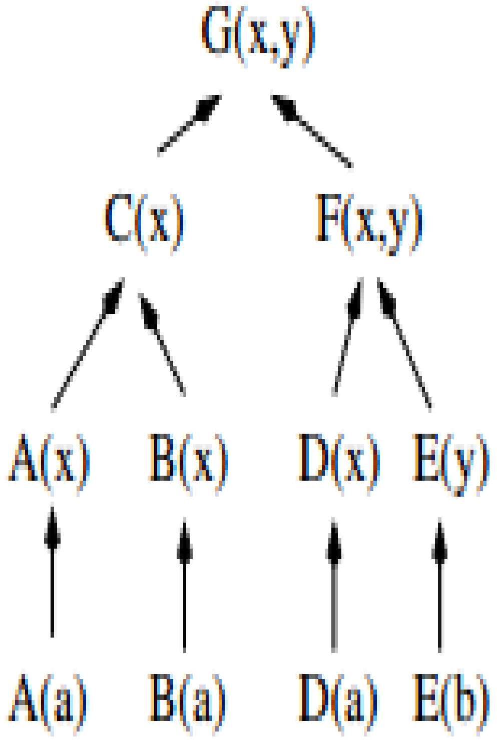
Goal Concept

G

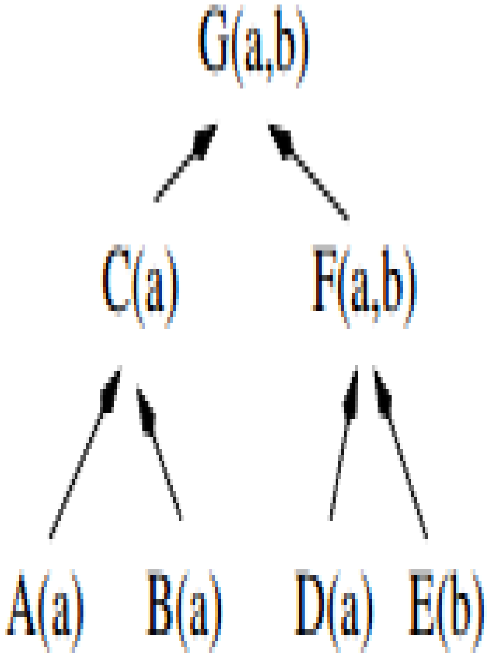
Example

A(a),B(a), D(a), E(b)

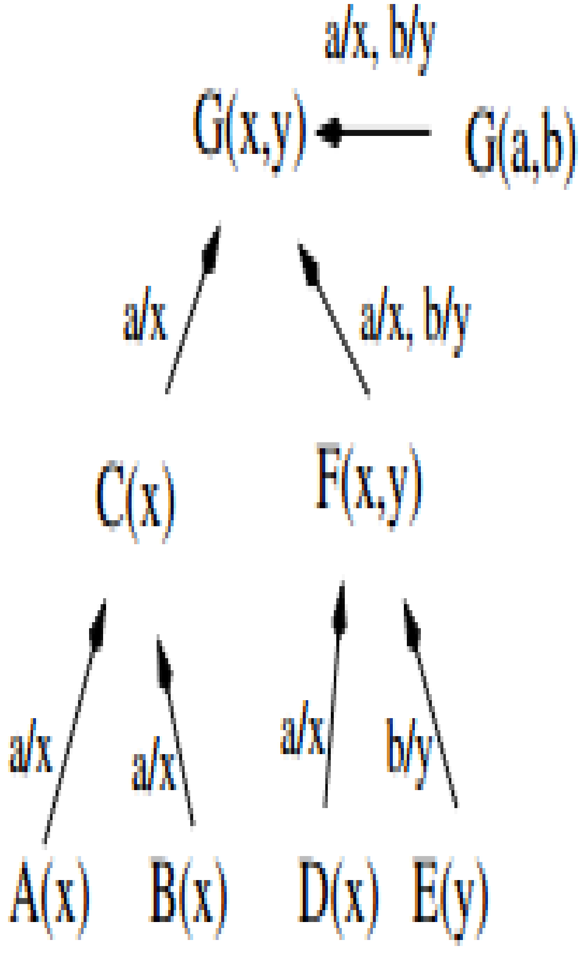
Resolution



Explanation



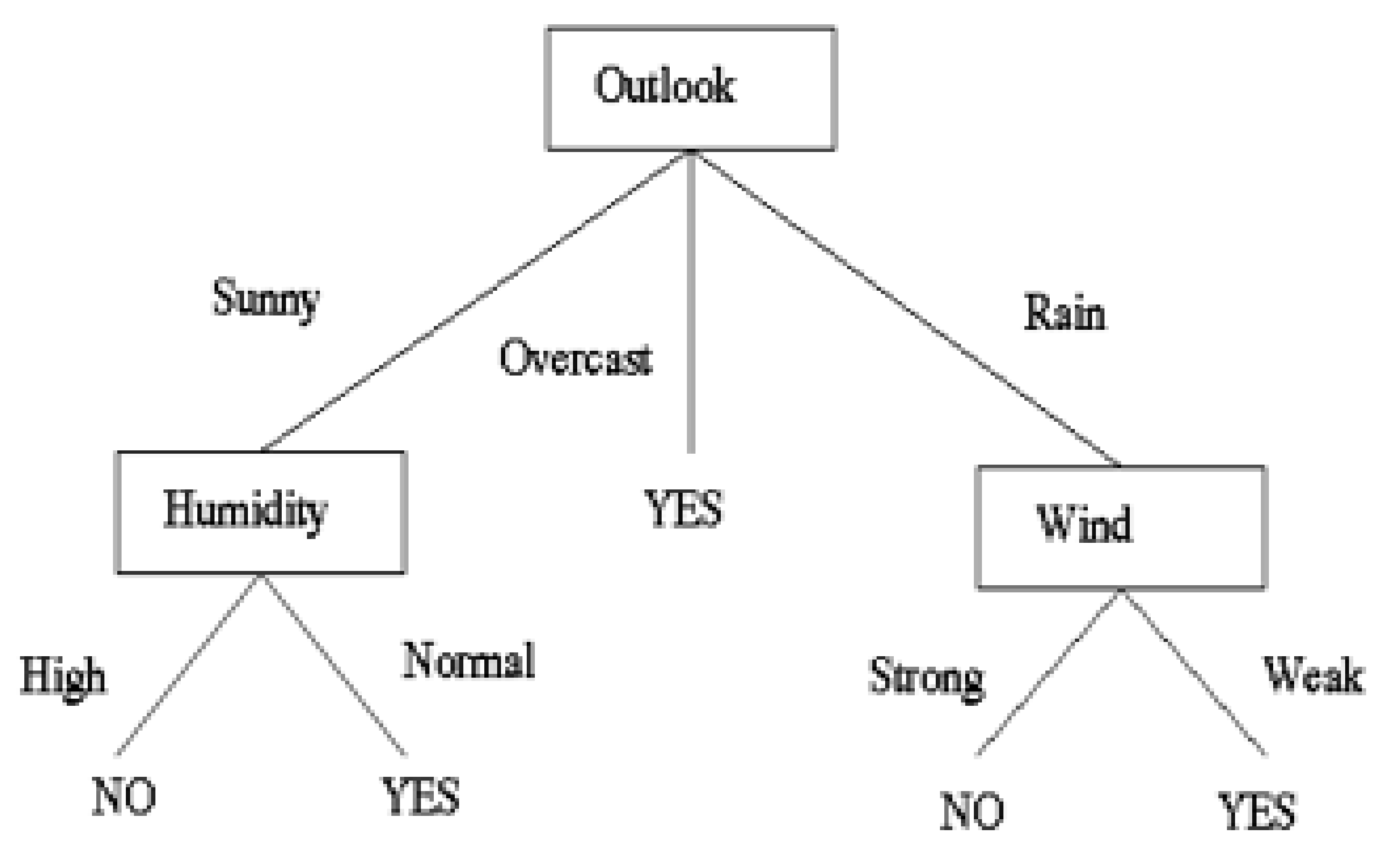
Generalization



Type of Learning

- **Learning by Example (Inductive learning)**
  - **Learning by example** is a general learning strategy where a concept is learned by drawing inductive inferences from a set of fact.
  - AI systems that learn by example can be viewed as searching a concept space by means of a decision tree.
  - The best known approach to constructing a decision tree is called **ID3 (Iterative Dichotomizer 3** developed by J. Ross Quinlan in 1975).
  - **ID3 (Iterative Dichotomizer 3)**
  - Top-down construction of the decision tree by recursively selecting the “best attribute” to use at the current node in the tree
    - Once the attribute is selected for the current node, generate children nodes, one for each possible value of the selected attribute
    - Partition the examples using the possible values of this attribute, and assign these subsets of the examples to the appropriate child node
    - Repeat for each child node until all examples associated with a node are either all positive or all negative

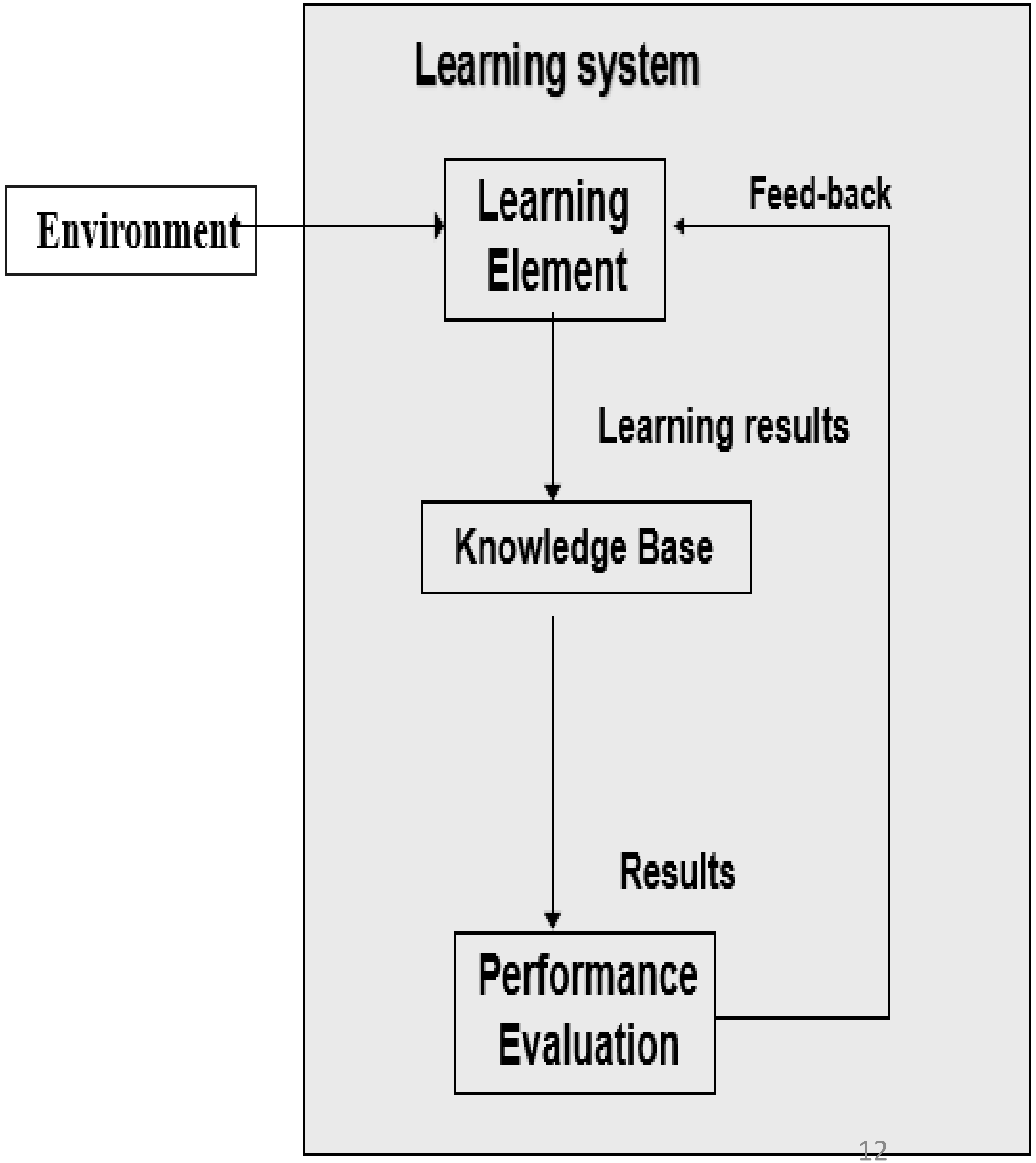
Decision Tree Learning



(Outlook = Sunny ∧ Humidity = Normal) ∨ (Outlook = Overcast) ∨ (Outlook = Rain ∧ Wind = Weak)

Learning Framework

- There are four major components in a learning system:
- **Environment**
  - The environment refers the nature and quality of information given to the learning element.
  - The nature of information depends on its level (the degree of generality with respect to the performance element)
    - ✓High level information is abstract, it deals with a broad class of problems
    - ✓Low level information is detailed; it deals with a single problem.
  - The quality of information involves
    - ✓noise free
    - ✓reliable
    - ✓ordered



Learning Framework

- **Learning Elements**
  - Acquire new knowledge through learning elements. Learning may be of
    - Rote learning ,Learning by examples, Learning by analogy, Explanation based learning…etc
  - The learning elements should have access to all internal actions of the performance element.
- **The Knowledge Base**
  - The knowledge base should be
    1. Expressive: Knowledge should be represented in understandable way.
    2. Modifiable: it must be easy to change the data in the knowledge base
    3. Extendibility”
       - The knowledge base must contain meta-knowledge (knowledge on how the data base is structured) so the system can change its structure

Learning Framework

- **The Performance Evaluation Element**
  - The performance evaluation element analyzes how complex the learning is and how learning is being performed.
  - Complexity depends upon type of task. For learning, the simplest task is classification based on a single rule while the most complex task requires the application of multiple rules in sequence.
  - Should have access to all internal actions of the performance elements.
  - Transparency, the learning element should have access to all the internal actions of the performance element.

Genetic Algorithm

- Developed extensively by **John Holland in mid 70's**
- Uses concepts of "**Natural Selection**" i.e. "**Survival of the fittest"** and "**Genetic Inheritance**"
- A **genetic algorithm** (or **GA**) is a search technique used in computing to find true or approximate solutions to optimization and search problems.
- Genetic algorithms are categorized as global search heuristics.
- Genetic algorithms are a particular class of evolutionary algorithms that use techniques inspired by evolutionary biology such as inheritance, mutation, selection, and crossover (also called recombination).
- **Concepts:**
  - **Population: set** of individuals each representing a possible solution to a given problem.
  - **Gene**: a solution to problem represented as a set of parameters ,these parameters known as genes.
  - **Chromosome**: genes joined together to form a string of values called chromosome.
  - **Fitness score(value)**:every chromosome has fitness score can be inferred from the chromosome itself by using fitness function.

Genetic Algorithm

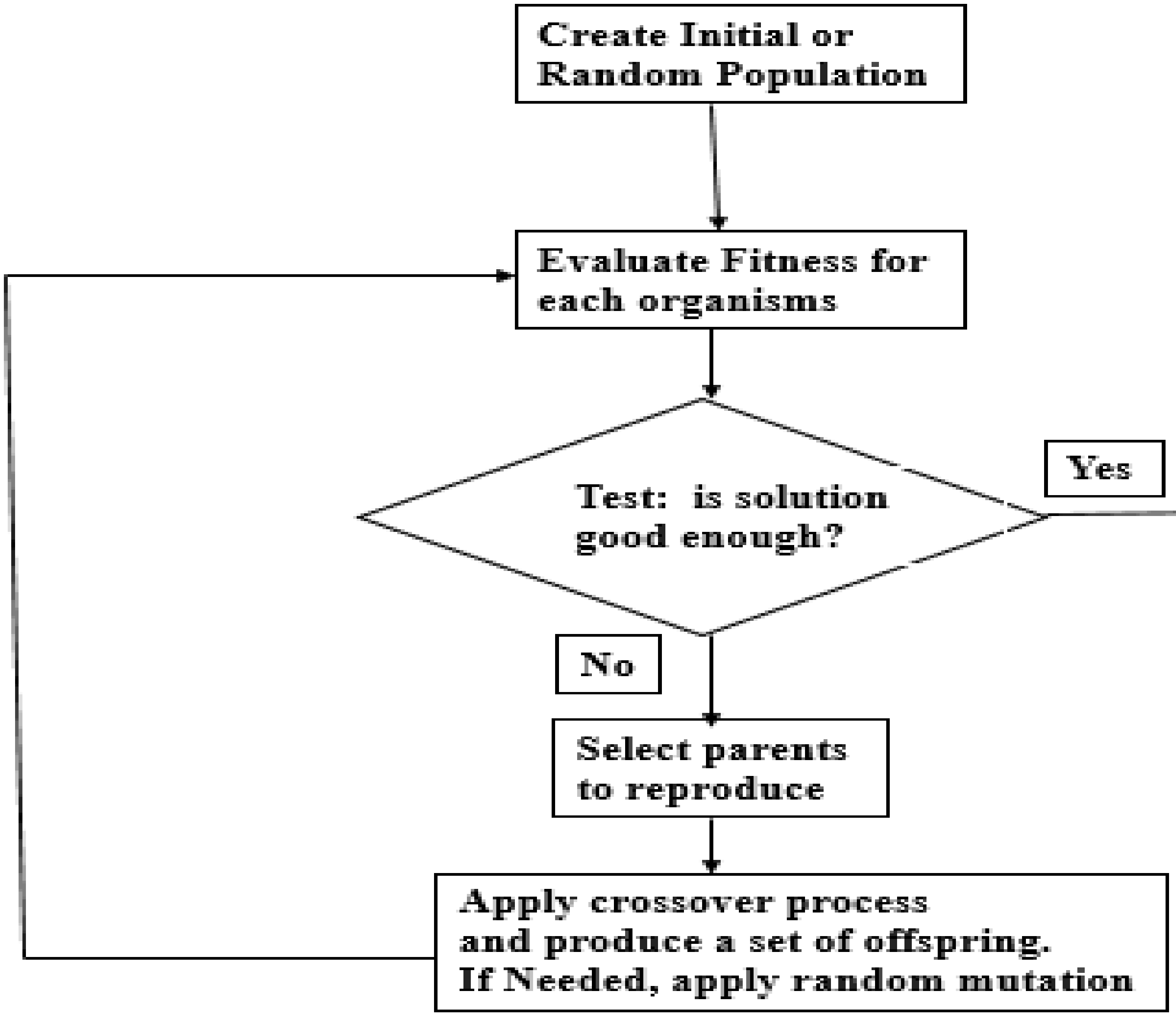
- **Stochastic operators**
  - **Selection** replicates the most successful solutions found in a population at a rate proportional to their relative quality
  - **Recombination** (Crossover) decomposes two distinct solutions and then randomly mixes their parts to form novel solutions
  - **Mutation** randomly perturbs a candidate solution
- **Comparison between Genetic Algorithm and Nature**

Genetic Algorithm	Nature
Optimization problem	Environment
Feasible solutions	Individuals living in that environment
Solution Quality (fitness function)	Individual's degree of adaption to its surrounding environment
A set of feasible solutions	A population of organisms
Stochastic operators	Selection, Crossover and mutation in nature's evolutionary process
Iteratively applying a set of stochastic operators on a set of feasible solutions	Evolution of populations to suit their environment

Basic Genetic Algorithm

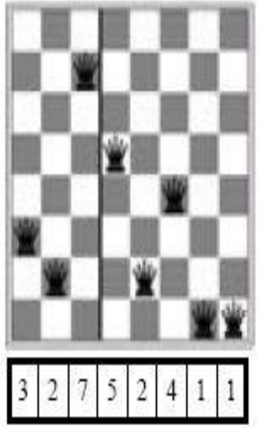
1) **[Start]** Generate random population of n chromosomes (suitable solutions for the problem)
2) **[Fitness]** Evaluate the fitness f(x) of each chromosome x in the population
3) **[New population]** Create a new population by repeating following steps until the new population is complete
4) **[Selection]** Select two parent chromosomes from a population according to their fitness (the better fitness, the bigger chance to be selected) The idea is to choose the better parents.
5) **[Crossover]** With a crossover probability cross over the parents to form a new offspring (children). If no crossover was performed, offspring is an exact copy of parents.
6) **[Mutation]** With a mutation probability mutate new offspring at each locus (position in chromosome).
7) **[Accepting]** Place new offspring in a new population
8) **[Replace]** Use new generated population for a further run of algorithm
9) **[Test]** If the end condition is satisfied, **stop**, and return the best solution in current population
10) **[Loop]** Go to step **2**

Genetic Algorithm Flowchart



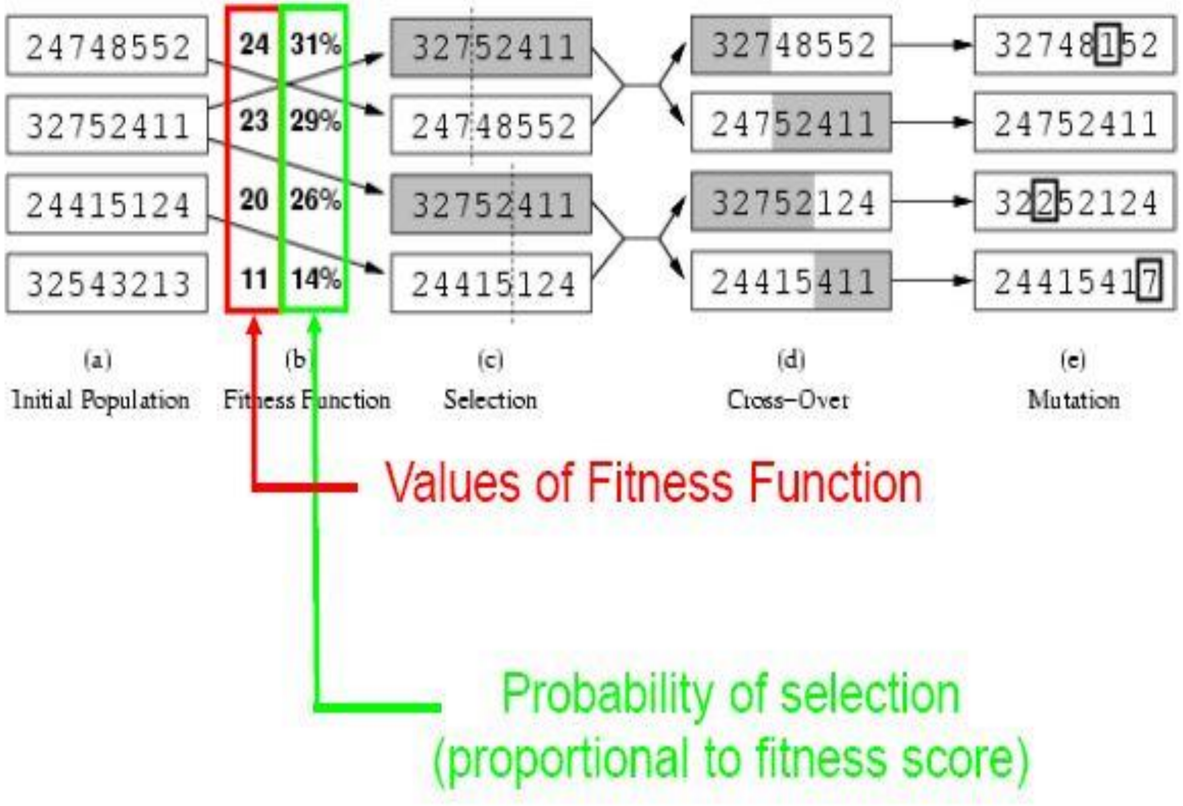
Example: 8-queens

- Fitness Function: number of nonattacking pairs of queens (28 is the value for the solution)
- Represent 8-queens state as an 8 digit string in which each digit represents position of queen



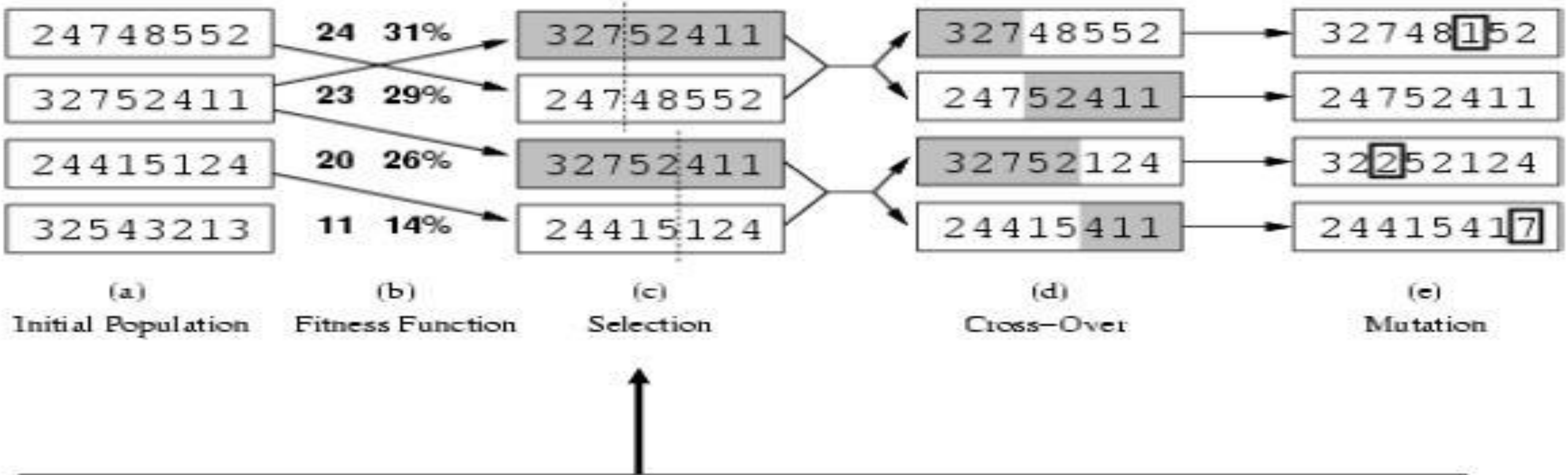
16

Example: 8-queens (Fitness Function)



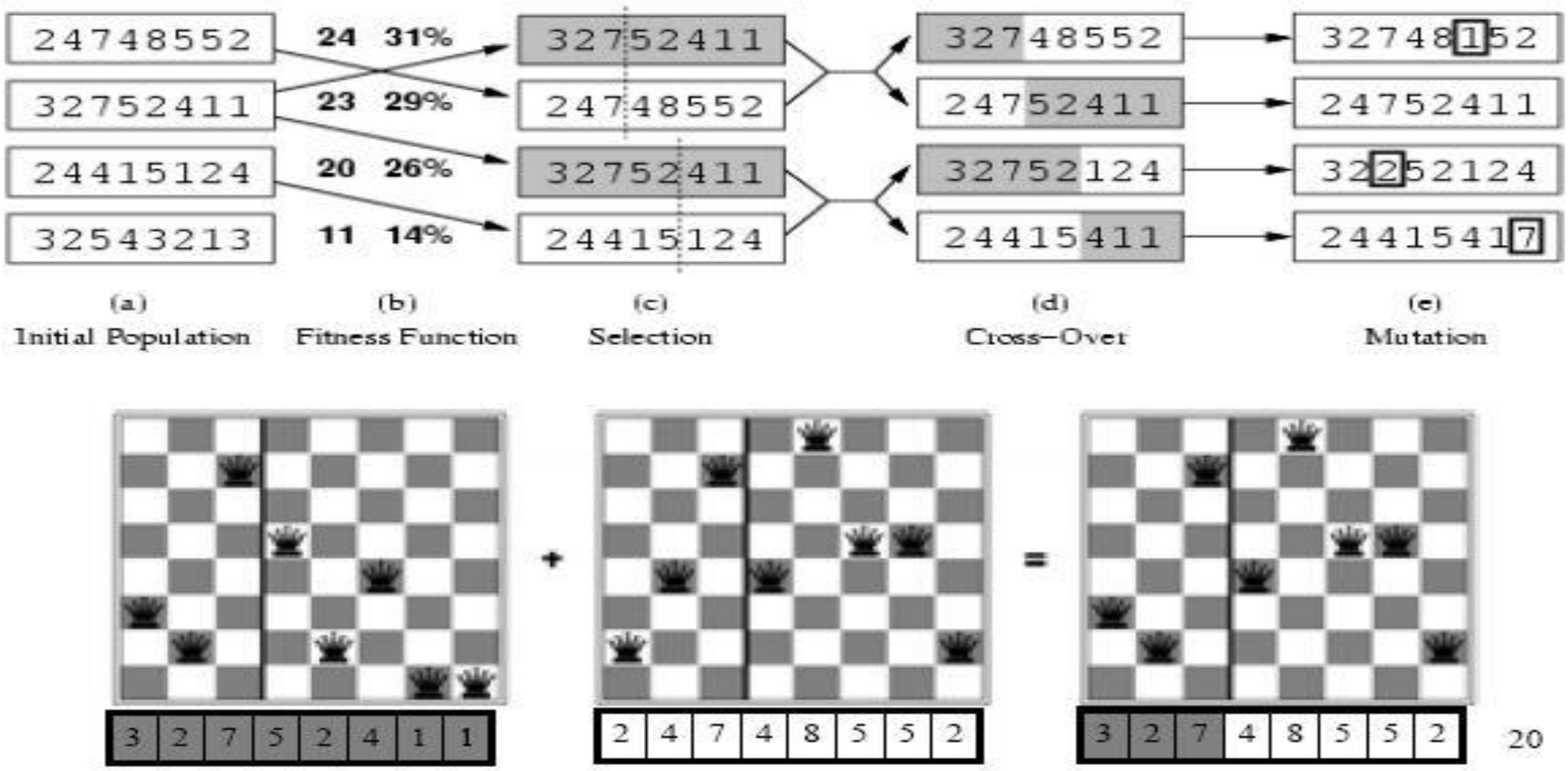
18

Example: 8-queens (Selection)

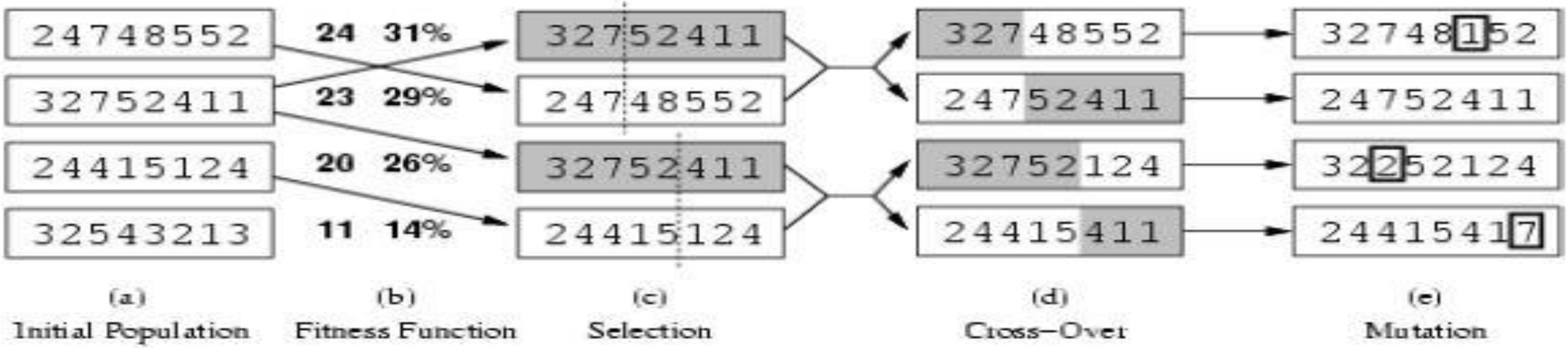


Notice 3 2 7 5 2 4 1 1 is selected twice while
3 2 5 4 3 2 1 3 is not selected at all

Example: 8-queens (Crossover)



Example: 8-queens (Mutation)



Mutation corresponds to randomly selecting a queen and randomly moving it in its column

Application of GA

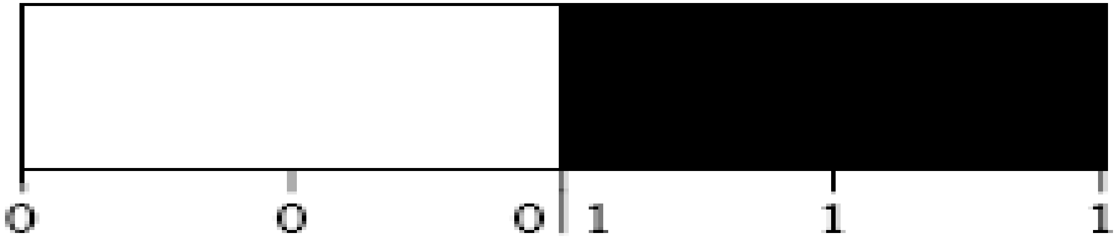
- Dynamic process control
- Induction of rule optimization
- Simulating biological models of behavior and evolution
- Complex design of engineering structures
- Pattern recognition
- Scheduling
- Transportation
- Layout and circuit design
- Telecommunication
- Graph-based problems
- **Disadvantage**
  - GA is better if the problem does not have any mathematical model for the solution.
  - GA is less efficient in terms of speed of convergence.
  - GA has tendency to get stuck in local maxima rather than global maxima.

Fuzzy Learning

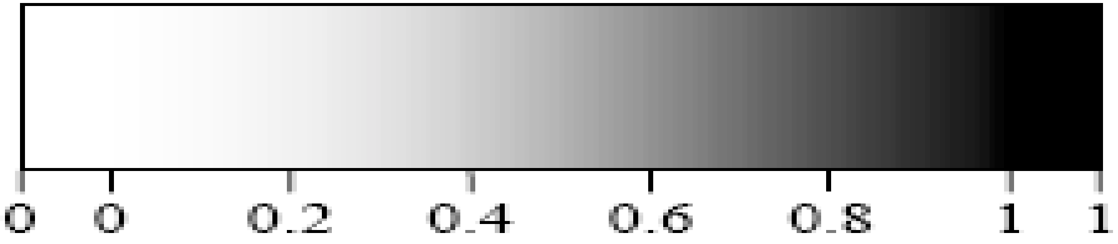
- In 1965 **Lotfi Zadeh**, published his famous paper “fuzzy sets”. Zadeh extended the work on possibility theory into a formal system of mathematical logic and introduced a new concept for applying natural language terms. This new logic for representing and manipulating fuzzy terms was called fuzzy logic.
- **Traditional Logic:** Traditional Boolean logic uses sharp distinctions. The truth values of variables may only be 0 or 1, often called "**crisp**" values. For instance Shyam with height 181cm is tall. If we draw a line at 180 cm, David with height 179cm is small. Is David really small?.
- **Fuzzy logic** is a form of many-valued logic in which the truth values of variables may be any real number between 0 and 1, considered to be "fuzzy".
- Fuzzy logic has been employed to handle the concept of partial truth, where the truth value may range between completely true and completely false.
- Fuzzy logic is a form of multi-valued logic.
- Fuzzy logic reflects how people think. It attempts to model our sense of our decision making and our common sense.
- Example: Temperature, Height, Speed, Distance, Beauty.

Fuzzy Learning

- **Range of logical values in Boolean and fuzzy logic**



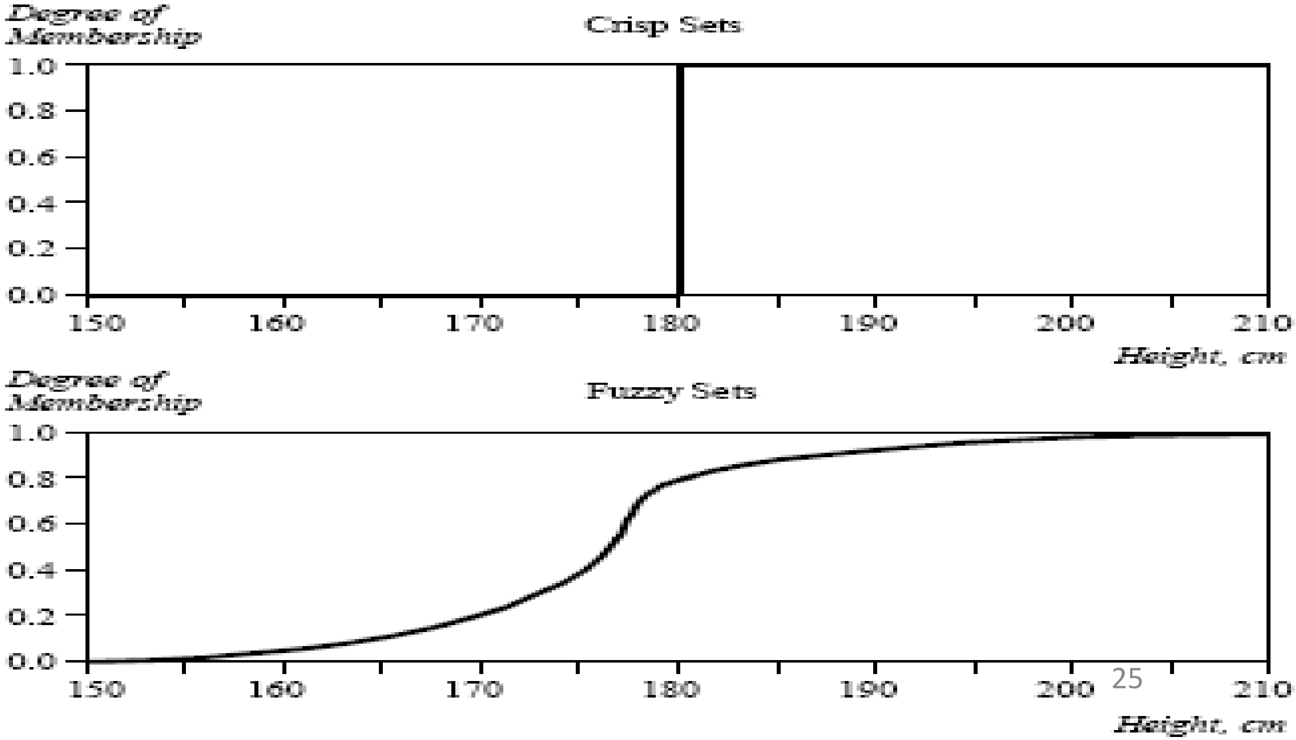
(a) Boolean Logic.



(b) Multi-valued Logic.

- **Crisp and fuzzy sets example**

Name	Height, cm	Degree of Membership	
		Crisp	*Fuzzy*
Chris	208	1	1.00
Mark	205	1	1.00
John	198	1	0.98
Tom	181	1	0.82
David	179	0	0.78
Mike	172	0	0.24
Bob	167	0	0.15
Steven	158	0	0.06
Bill	155	0	0.01
Peter	152	0	0.00

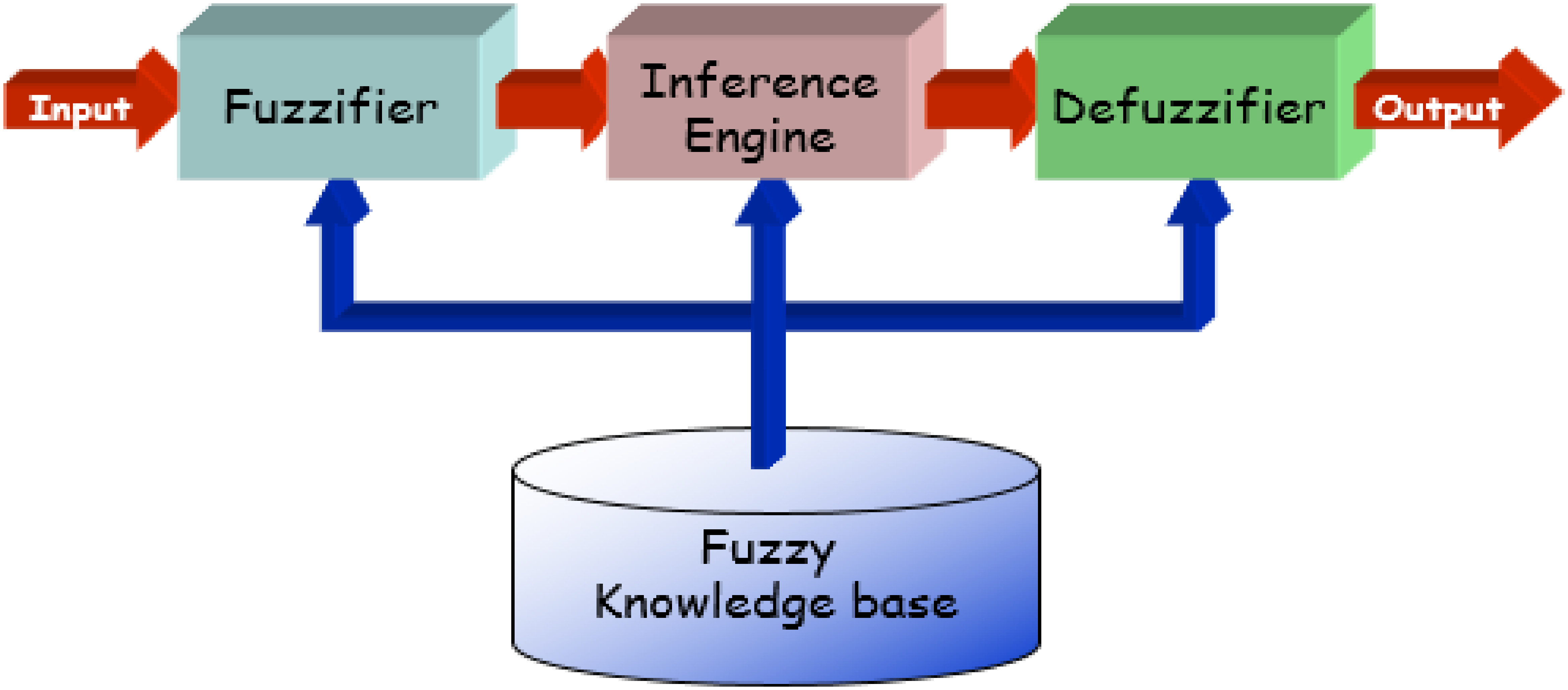


Fuzzy Inferences System (FIS)

- Fuzzy inference is the actual process of mapping from a given input to an output using fuzzy logic.
- A FIS uses a collection of fuzzy membership functions and rules to reason about data.
- FIS have been successfully applied in fields such as automatic control, data classification, decision analysis, expert systems, and computer vision. The FIS is known by a number of names, such as fuzzy-rule-based system, fuzzy expert system, fuzzy model, and fuzzy logic controller etc.
- **The Architecture of Fuzzy Inference Systems**
  - **Fuzzifier:**Converts the crisp input to a linguistic variable using the membership functions stored in the fuzzy knowledge base.
  - **Knowledge Base:** The rule base and the database are jointly referred to as the knowledge base.
    - ✓a rule base containing a number of fuzzy IF–THEN rules;
    - ✓a database which defines the membership functions of the fuzzy sets used in the fuzzy rules.
  - **Inference Engine:** Using If-Then type fuzzy rules converts the fuzzy input to the fuzzy output.
  - **Defuzzifier:** Converts the fuzzy output of the inference engine to crisp using membership functions analogous to the ones used by the fuzzifier.

Fuzzy Inferences System

- **The Architecture of Fuzzy Inference Systems**



Fuzzy Inference Methods

- The most important two types of fuzzy inference method are **Mamdani** and **Sugeno** fuzzy inference methods, **Mamdani fuzzy inference** is the most commonly seen inference method. This method was introduced by Mamdani and Assilian (1975).
- Another well-known inference method is the so- called **Sugeno or Takagi–Sugeno–Kang** method of fuzzy inference process. This method was introduced by Sugeno (1985). This method is also called as TS method. The main difference between the two methods lies in the **consequent of fuzzy rules**.
- **Mamdani fuzzy inference Algorithm**
  - The Mamdani-style fuzzy inference process is performed in four steps:
    1. Fuzzification of the input variables,
    2. Rule evaluation;
    3. Aggregation of the rule outputs, and finally
    4. Defuzzification.

Fuzzy Inference Methods

- We examine a simple two-input one-output problem that includes three rules:

Rule: 1
IF x **is** $A3$
OR y **is** $B1$
THEN z **is** $C1$

Rule: 1
IF ***project_funding*** **is** ***adequate***
OR ***project_staffing*** **is** ***small***
THEN ***risk*** **is** ***low***

Rule: 2
IF x **is** $A2$
AND y **is** $B2$
THEN z **is** $C2$

Rule: 2
IF ***project_funding*** **is** ***marginal***
AND ***project_staffing*** **is** ***large***
THEN ***risk*** **is** ***normal***

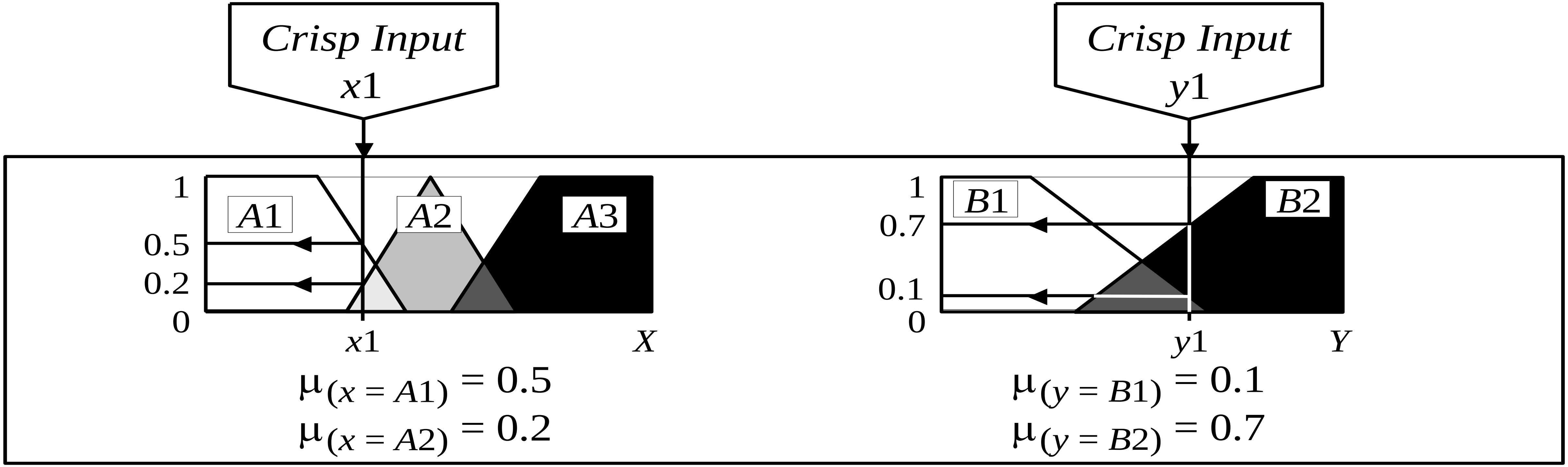
Rule: 3
IF x **is** $A1$
THEN z **is** $C3$

IF ***project_funding*** **is** ***inadequate***
THEN ***risk*** **is** ***high***

29

Fuzzy Inference Methods

- **Step 1-Fuzzification:** The first step is to take the crisp inputs, x1 and y1 (project funding and project staffing), and determine the degree to which these inputs belong to each of the appropriate fuzzy sets.



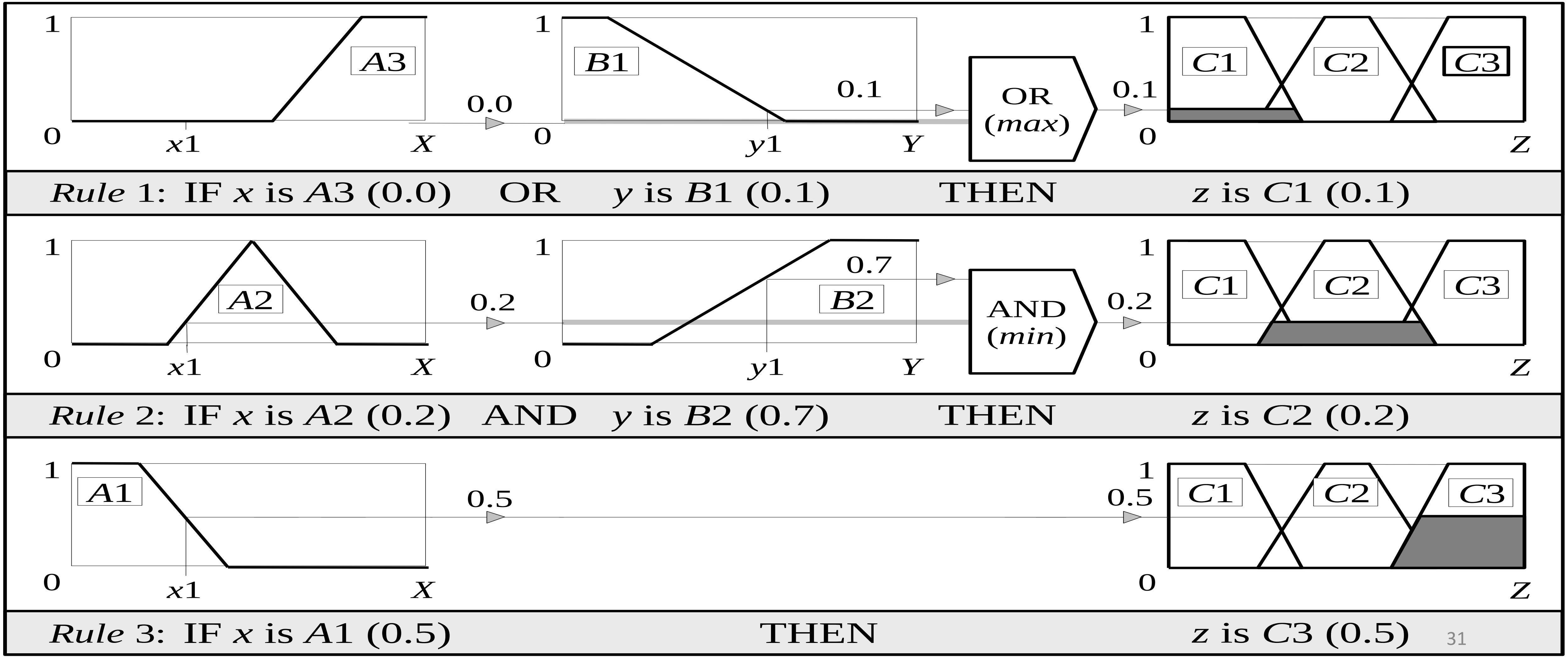
- **Step 2-Rule Evaluation:** The second step is to take the fuzzified inputs, $\mu_{(x=A1)} = 0.5$, $\mu_{(x=A2)} = 0.2$, $\mu_{(y=B1)} = 0.1$ and $\mu_{(y=B2)} = 0.7$, and apply them to the antecedents of the fuzzy rules. If a given fuzzy rule has multiple antecedents, the fuzzy operator (AND or OR) is used to obtain result of the antecedent evaluation.

- To evaluate the disjunction of the rule antecedents, use **OR fuzzy operation**.

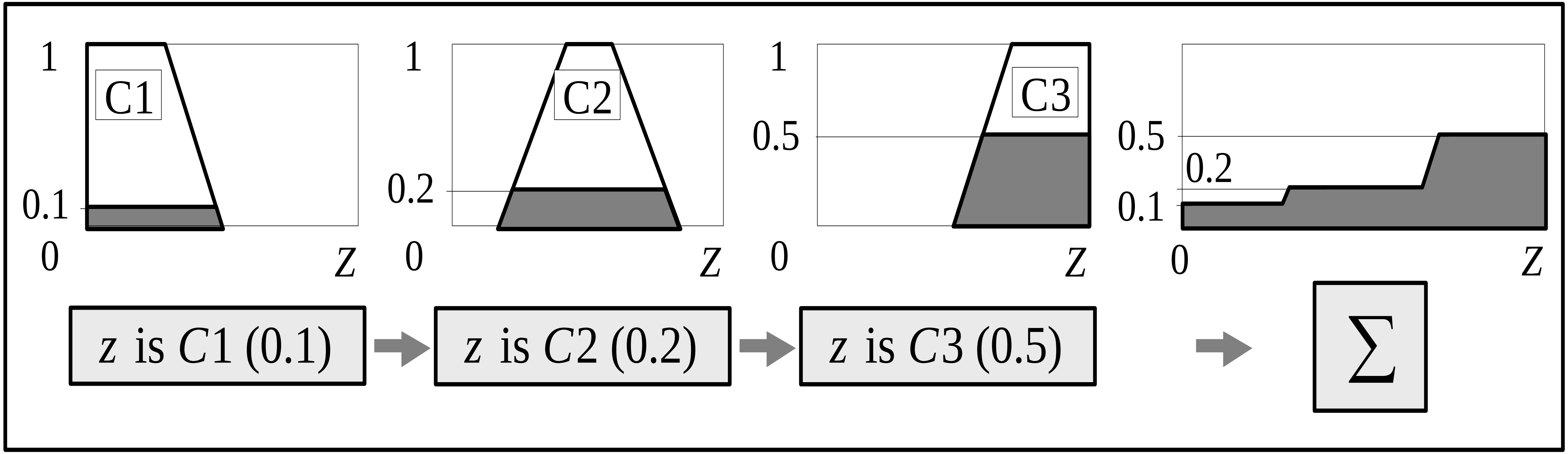
$$\mu_{A \cup B}(x) = max\ [\mu_A(x),\ \mu_B(x)]$$

- To evaluate the conjunction of the rule antecedents, apply the AND fuzzy operation intersection:

$$\mu_{A \cap B}(x) = min\ [\mu_A(x),\ \mu_B(x)]$$

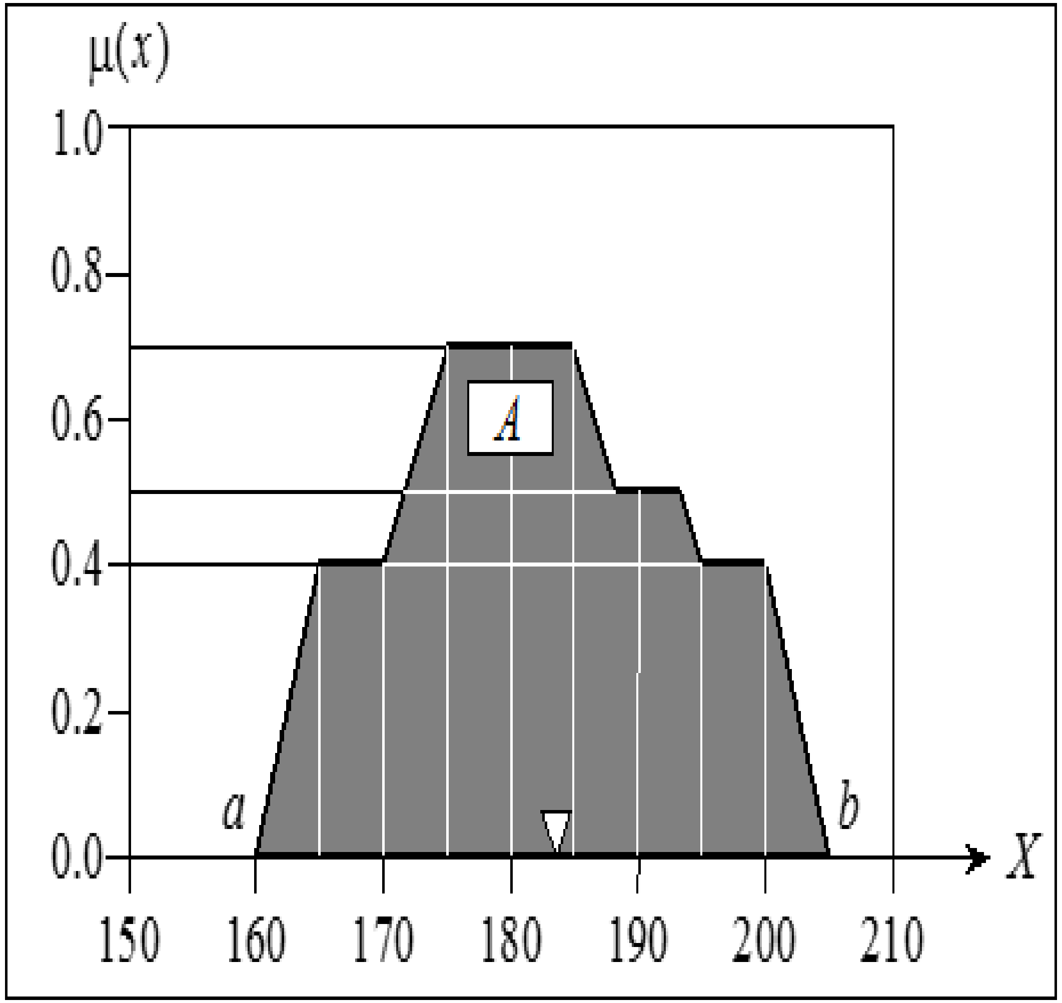


- **Step 3 - Aggregation of the rule outputs:**
  - Aggregation is the process of unification of the outputs of all rules. We take the membership functions of all rule consequents previously clipped or scaled and combine them into a single fuzzy set.
  - The input of the aggregation process is the list of clipped or scaled consequent membership functions, and the output is one fuzzy set for each output variable.



Step 4- Defuzzification : Fuzziness helps us to evaluate the rules, but the final output of a fuzzy system has to be a crisp number. The input for the defuzzification process is the aggregate output fuzzy set and the output is a single number.

- The most popular defuzzification method is the centroid technique. It finds the point where a vertical line would slice the aggregate set into two equal masses.
- Centroid defuzzification method finds a point representing the centre of gravity of the fuzzy set, A, on the interval, ab.



$$COG = \frac{(0+10+20)\times 0.1 + (30+40+50+60)\times 0.2 + (70+80+90+100)\times 0.5}{0.1+0.1+0.1+0.2+0.2+0.2+0.2+0.5+0.5+0.5+0.5} = 67.4$$