**VIETNAM GENERAL CONFEDERATION OF LABOR**

**TON DUC THANG UNIVERSITY**

**FACULTY OF INFORMATION TECHNOLOGY**



**FINAL REPORT**

**THE DESIGN AND ANALYSIS OF ALGORITHMS**

***Instructor***: Nguyen Chi Thien

***Student***: Hoang Dinh Quy Vu - 521H0517

Tran Nhut Anh – 521H0491

Nguyen Hoang Phuc – 521H0509

**HO CHI MINH CITY, 2023**

**VIETNAM GENERAL CONFEDERATION OF LABOR**

**TON DUC THANG UNIVERSITY**

**FACULTY OF INFORMATION TECHNOLOGY**



**FINAL REPORT**

**THE DESIGN AND ANALYSIS OF ALGORITHMS**

***Instructor***: Nguyen Chi Thien

***Student***: Hoang Dinh Quy Vu - 521H0517

Tran Nhut Anh – 521H0491

Nguyen Hoang Phuc – 521H0509

**HO CHI MINH CITY, 2023**

# ACKNOWLEDGEMENT

I would like to thank the teachers of Ton Duc Thang University in Ho Chi Minh City, as well as the teachers of the Department of the design and analysis of algorithms for IT for their enthusiastic guidance during my time studying at the school; Special thanks to Mr. Thien who directly guided me to complete my graduation thesis.

I sincerely thank!

**PROJECT COMPLETED**

**AT TON DUC THONG UNIVERSITY**

I hereby certify that this is my thesis project and was conducted under the guidance of Mr Thien. The research content and results in this topic are honest and have not been previously published in any form. The data in the tables are collected by the author from various sources clearly stated in the reference section for analysis, comments, and evaluation.

Furthermore, this thesis includes comments, evaluations, and data from other authors and organizations, all of which are cited and annotated with their sources.

If any fraud is detected, I will fully take responsibility for the content of my thesis. Ton Duc Thang University is not liable for any copyright infringement or violation caused by me during the implementation process (if any).

*Ho Chi Minh city, 9th December, 2023*

*Author*

*(Sign and write your full name)*

# TEACHER'S CONFIRMATION AND ASSESSMENT SECTION

**The confirmation part of the instructor**

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

*Ho Chi Minh city, 11th April, 2023*

**The evaluation part of the teacher marks the test**

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

*Ho Chi Minh city, 11th April, 2023*

# TABLE OF CONTENTS

# LIST OF SYMBOLS AND ABBREVIATIONS

**SYMBOLS**

**ABBREVIATIONS**

# CHAPTER 1: AN IMPORVED CHEMICAL REACTION OPTIMIZATION ALGORITHM FOR SOLVING THE SHORTEST COMMON SUPERSEQUENCE PROBLEM

## 1.1 Introduction

### 1.1.1 Shorest common supersequence – SCS

* The SCS problem is a classical NP-hard problem that is normally solved by heuristic algorithms.
* It was first defined by David Maier in 1976, and it seeks to find the common supersequence with the minimum length.
* SCS has various applications in fields such as DNA sequencing, data compression, AI optimization, and multiple sequence alignment

### Chemical reaction optimization - CRO

* The CRO algorithm, initially introduced by Lam and Li in 2010, draws inspiration from the sequential steps involved in chemical reactions.
* In a chemical reaction, various sub-reactions occur, leading to a series of intermediate states.
* At each state, the energy of the molecule decreases, resulting in increased stability. This concept can be applied to the step-wise search approach used in optimization problems.

### Improved chemical reaction optimization – IMCRO

* A new algorithm called IMCRO is introduced for solving the SCS problem.
* The key contribution of this paper is the enhancement of the existing CRO\_SCS framework by introducing two new operators for decomposition and inter-molecular ineffective collisions in two of the four reactions of CRO.
* The results show that these new operators significantly improve the performance and efficiency of CRO in solving the SCS problem

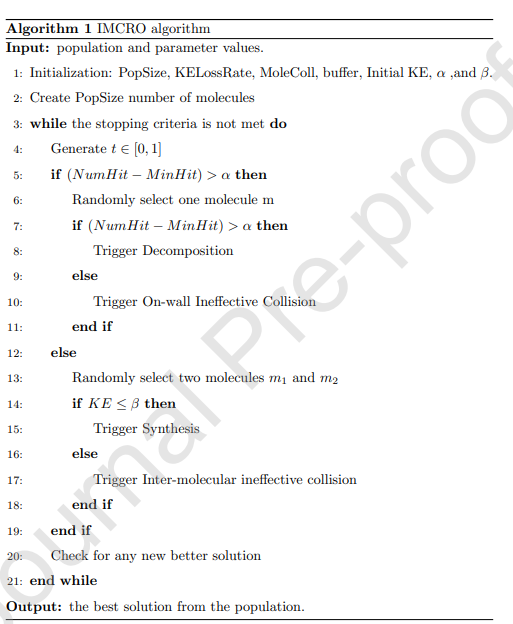
### Different approaches

* Various approaches have been proposed in order to address the SCS problem and find the optimal solution. Some significant proposals found in the literature include greedy methods, ant colony optimization (ACO), and artificial bee colony (ABC),…

## IMCRO

### 1.2.1 Framework

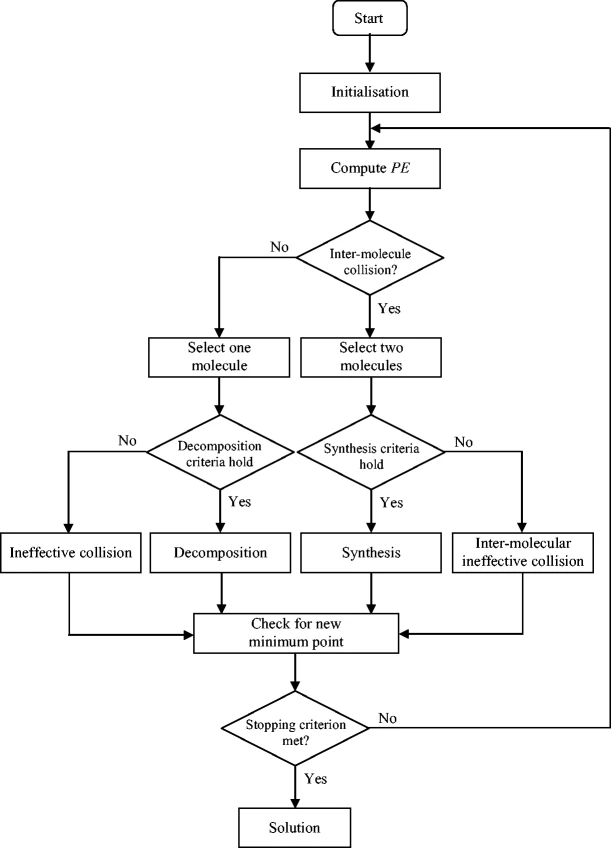
* The proposed framework in our proposal is an enhanced extension of the CRO-based algorithm, known as CRO SCS. It is composed of three stages, namely initialization, iteration, and finalization, which are described in detail in Algorithm 1.



**Algorithm 1**

### 1.2.2 Activity chain

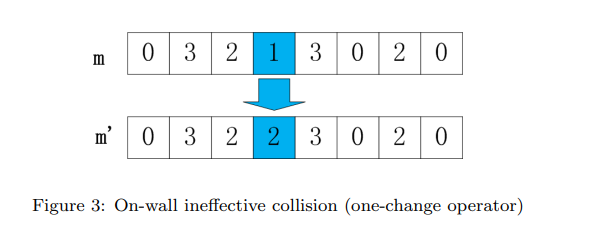
* Initialization (Stage 1):
  + Elements and molecules initialized, including PopSize, KELossRate, MoleColl, buffer, Initial KE, α, and β.
  + Molecule energy includes potential energy (PE) and kinetic energy (KE).
  + PE is related to the objective function (Formula 2), KE reflects tolerance.
  + Population generation and supersequence representation involve random insertion operations.
  + Integer encoding used to represent symbols from the alphabet.
* Iteration (Stage 2):
* Divided into reaction and repair subtasks.
* Reaction step involves on-wall ineffective collision, decomposition, inter-molecular ineffective collision, and synthesis.
* Uni-molecule and inter-molecule reactions determined by parameters α and β.
* Main iteration randomly generates parameter t to decide reaction type.
* Algorithm 1 IMCRO:
* Input: Population and parameter values.
* Initialization creates PopSize number of molecules.
* Main loop continues until stopping criteria are met.
* Reaction type determined by random parameter t.
* Different reactions (decomposition, on-wall ineffective collision, synthesis, inter-molecular ineffective collision) triggered based on conditions.
* Check for any new better solution in each iteration.
* Validation and Finalization (Stages 3 and 4):
* Validity check: If solution does not meet requirements, a repair algorithm is applied.
* Final stage checks stopping criteria (e.g., CPU time, function evaluations, objective function value).
* IMCRO outputs the best solution found with its objective value and terminates the procedure.

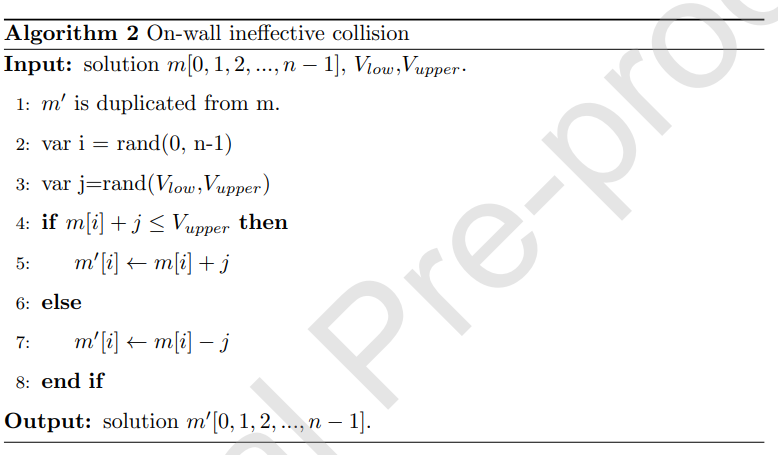


### 1.2.3 Operators

#### 1.2.3.1 On-wall ineffective collision

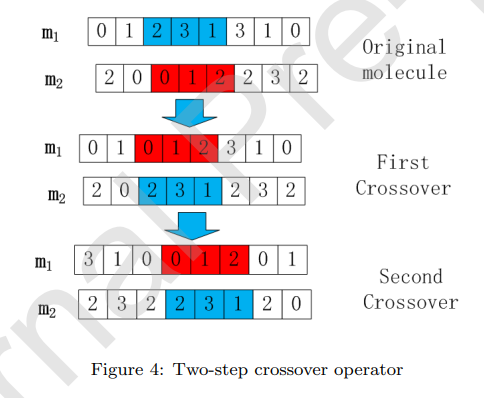
* The one-difference operator is an algorithm that modifies a character in a supersequence, which helps to explore nearby solutions without reducing the overall length. This operator expands the possibilities for finding the Shortest Common Supersequence (SCS).

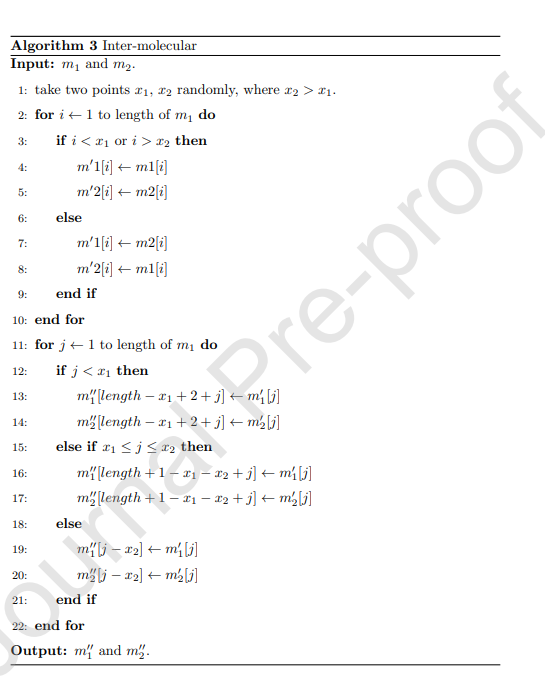




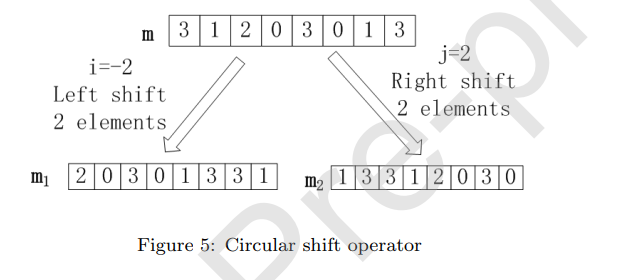
#### 1.2.3.2 Inter-molecular ineffective collision

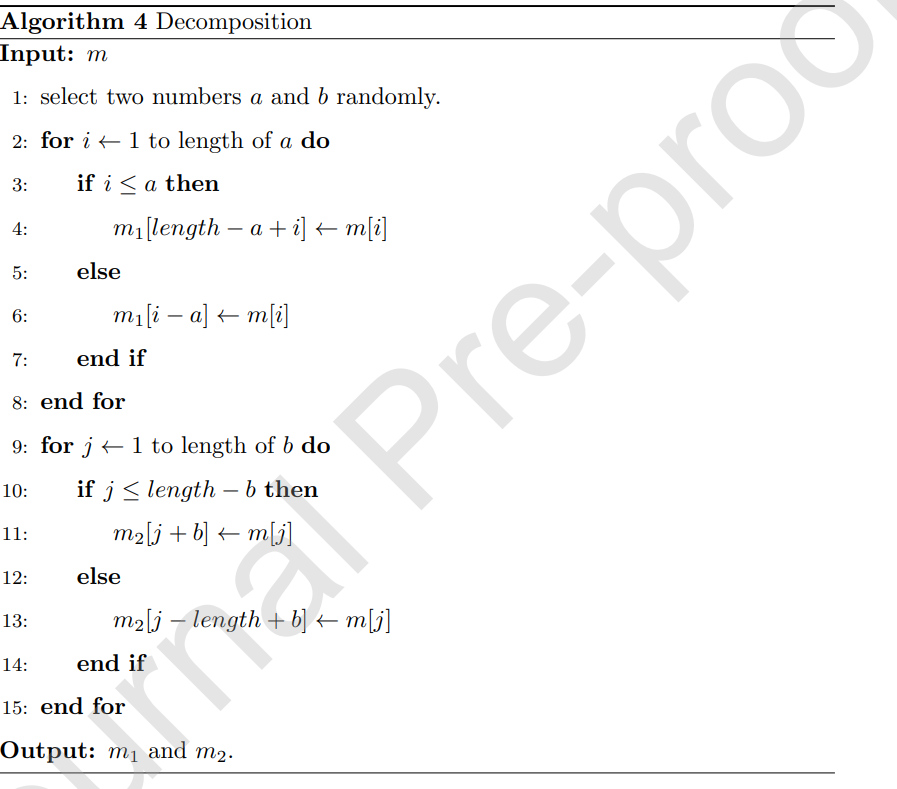
* This algorithm randomly selects two molecules, m1 and m2, and applies two crossover operators: the first between two different molecules, and the second within each individual molecule, resulting in two new solutions, m'1 and m'2.





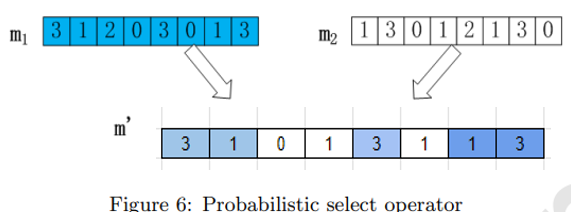
#### 1.2.3.3 Decomposition

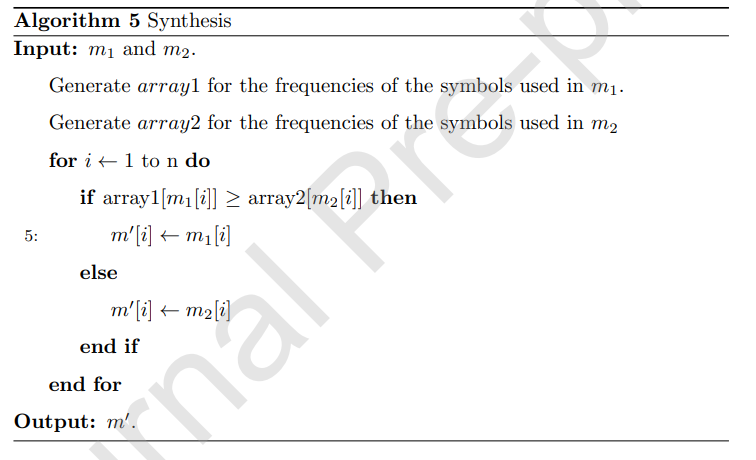
* The decomposition reaction enables exploration of a different search space, causing significant changes in molecular structures.
* In a circular shift operator, generated within the range [-n, n].
* Two randomly chosen integers,
* -i : shift left i times
* j : shift right j times



#### 1.2.3.4 Synthesis

* This algorithm combines two molecules, m1 and m2, to form a new molecule, m', through a process opposite to a decomposition reaction. An enhanced probabilistic selection operator variant expedites convergence by exploring diverse molecular structures and frequencies.





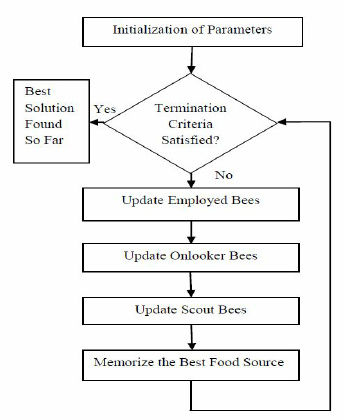
#### 1. 3.2.5 Repair function

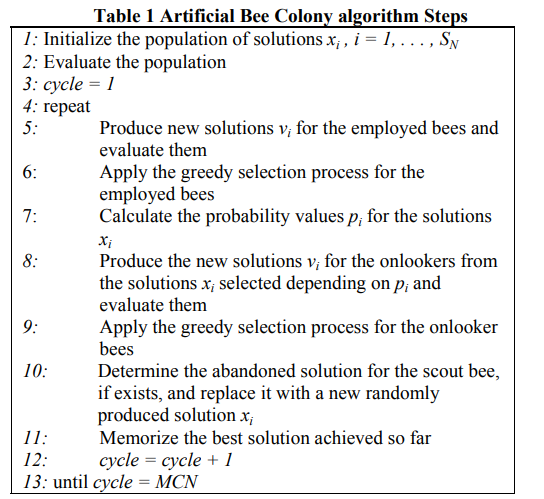
* This algorithm includes a two-phase process to ensure the validity of the new supersequence. The first phase involves validation, where the algorithm checks for any violations.
* If no violations are found during validation, the new molecule is seamlessly inserted into the population. However, if a violation is detected, the algorithm initiates the repair phase.
* During repair, the new molecule undergoes scrutiny against a predefined violation threshold (VT).
* If the extent of violation surpasses the threshold, the algorithm undertakes corrective measures to bring the molecule into compliance, ensuring the integrity of the resulting supersequence.

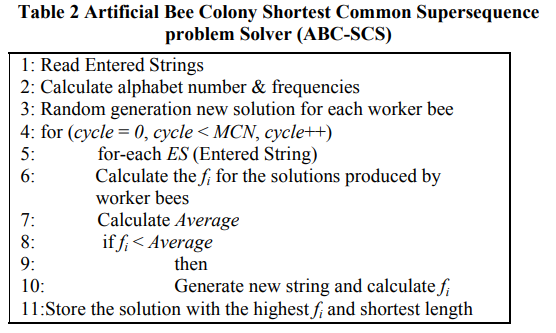
## Algorithms extends

### 1.3.1 Artificial Bee Colony - ABC

* The Artificial Bee Colony (ABC) is an optimization algorithm inspired by the foraging behavior of honeybees. Developed by Dervis Karaboga in 2005, the ABC algorithm is a technique based on swarm intelligence.
* It mimics the collaboration and communication observed in real honeybee colonies to solve complex optimization problems. The algorithm consists of three components: employed bees, onlooker bees, and scout bees, which work together to explore and exploit the solution space.
* The algorithm starts with a population of artificial bees that represent potential solutions.
* The employed bees evaluate the fitness of these solutions and share their findings with the onlooker bees.
* The onlooker bees then choose solutions based on the fitness information provided. Additionally, the scout bees contribute to the diversity of the population by exploring new and unexplored solutions.

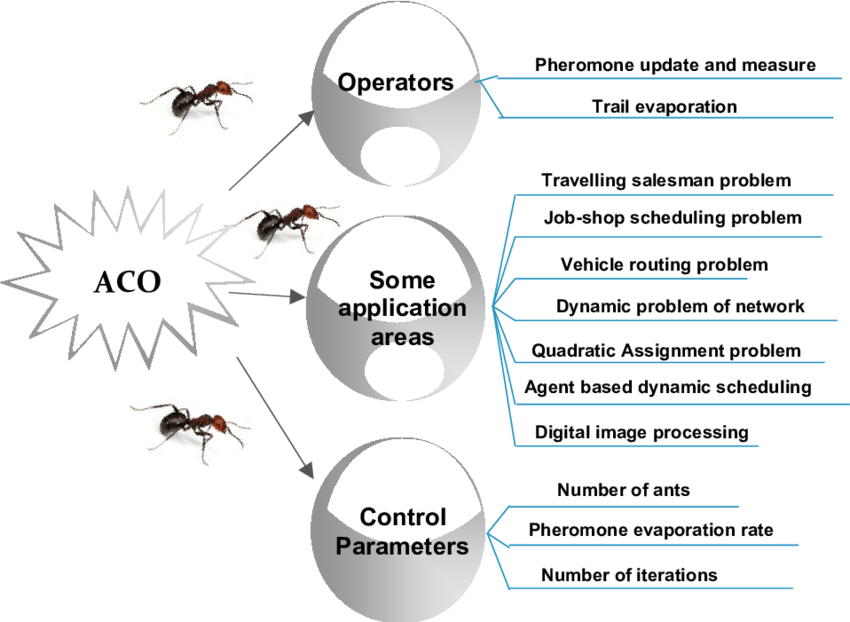




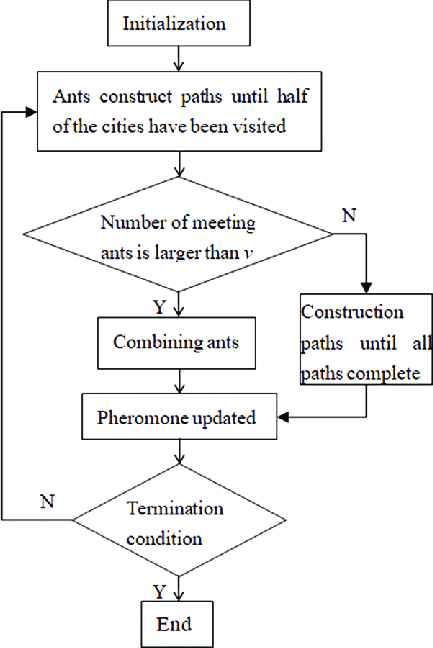


### 1.3.2 Ant colony optimization- ACO

* ACO algorithm mimics ant foraging to find an optimal path. Ants explore paths, leaving pheromones. Paths with more pheromones are favored. For the SCS problem, ants represent paths in a sequence, and pheromones indicate promising subsequences. The algorithm leverages cumulative pheromone information to guide optimal path selection.



* This is activity chain of algorithm:
* 1. Initialize ants:
  + Create ants with random paths.
  + Calculate ant path distances.
* 2. Main loop:
  + 2.1 Select best ant.
  + 2.2 Perform local pheromone updates:
    - For each ant except the best 10%:
      * Swap random cities.
      * Update ant path if improved.
  + 2.3 Perform global pheromone updates:
    - Swap three random cities for the best ant.
    - Update ant path if improved.
  + 2.4 Update pheromone levels.
  + 2.5 Sort ants by distance.
* 3. Output shortest path:
  + Return path of the ant with the shortest distance.



# CHAPTER 2: CODE

## 2.1 Initialization

## 2.2 Molecule

## 2.4 Operator

## 2.5 ACB

## 2.6 ACO

## 2.7 Main

# CHAPTER 3: RESULT AND ANALYSIS

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| popSize | iterations | IMCRO | ABC | ACO |
| 20 | 10 | 0.005 | 0.006 | 0.023 |
| 50 | 100 | 0.002 | 0.003 | 0.042 |
| 100 | 200 | 0.003 | 0.005 | 0.15 |
| 500 | 500 | 0.002 | 0.04 | 0.104 |

# REFERENCES

[1] Fei Luo, Cheng Chen, Joel Fuentes, An improved chemical reaction optimization algorithm for solving the shortest common supersequence problem (2020)   
[2] A. S. Jaradat, M. M. Noaman, Solving shortest common supersequence problem using artificial bee colony algorithm, International Journal of ACM Jordan 2 (1) (2011) 180–185.

[3] S. Rajendran, C. Rajendran, H. Ziegler, An ant-colony algorithm to transform jobshops into flowshops: A case of shortest-common-supersequence stringology problem, in: Bio-Inspired Models of Network, Information, and Computing Systems - 5th International ICST Conference, BIONETICS 2010, Boston, MA, USA, December 1-3, 2010, Revised Selected Papers, 2010, pp. 413–424