

Received December 15, 2021, accepted February 20, 2022, date of publication March 2, 2022, date of current version March 10, 2022.

Digital Object Identifier 10.1109/ACCESS.2022.3156008

Problem Solving in Crowd Management Using Heuristic Approach

ALI M. AL-SHAERY¹, (Member, IEEE), MOHAMED O. KHOZIUM², (Member, IEEE),
NORAH S. FAROOQI³, (Member, IEEE), SHROUG S. ALSHEHRI³, (Member, IEEE),
AND MOHAMMAD ADNAN M.B. AL-KAWA⁴

¹Department of Civil Engineering, Vice Presidency for Innovation and Entrepreneurship (VPiE), Vice Presidency for Development and Community Service (VPDCS), Umm Al-Qura University, Makkah 21955, Saudi Arabia

²Computer Science Research Center, Future University in Egypt, Cairo 11835, Egypt

³College of Computer and Information Systems, Umm Al-Qura University, Makkah 21955, Saudi Arabia

⁴Wentworth College of Computer with Artificial Intelligence, University of York, Heslington, York YO10 5DD, U.K.

Corresponding author: Mohamed O. Khozium (osama@khozium.com)

This work was supported in part by the Deputyship for Research & Innovation, Ministry of Education, Saudi Arabia, under Project RDO-P543.

ABSTRACT There are many problems that procedural algorithms can solve efficiently. However, these algorithms are sometimes too slow to abide by the time available for performing the solution; other times, it is impossible to get a solution using procedural algorithms. A heuristic method is a practical approach that can reach an approximation of an efficient solution where the optimum is not guaranteed. Heuristic techniques are applied in many real-world problems, including crowd management; using heuristic-based models helped to comprehend crowd behavior better and increase simulation reliability. This paper reviews many heuristic-related articles to gather the aspects of the topic in one place and clear the fuzziness to make it easy to comprehend. The paper covers some of the previous works with similar approaches and presents state-of-the-art heuristic solutions for real-world problems. These techniques are discussed under three classifications: simple heuristics, meta-heuristics, and hyper-heuristics. Most importantly, the paper explores the heuristic role in crowd field problems concluding that heuristics are primarily applied in modeling when it comes to the Crowd field. It investigates different heuristics for crowd management. The main intent of this review is to establish a comprehensive understanding of heuristics-related operations in the crowd management field. Moreover, it aims to support other researchers' future work and fill research gaps by highlighting the absence of crowd problems from heuristics literature and the limitations of each heuristics approach.

INDEX TERMS Crowd malmanagement, hyper-heuristics, meta-heuristics, problem-solving, search algorithms.

I. INTRODUCTION

Problem-solving methods aim to provide solutions to specific problems. Typically, algorithms provide procedural steps that guarantee to reach a satisfactory solution in trade of some cost of time or other weights. The optimal solution is the one that sufficiently satisfies the solution with the lowest cost. Where algorithms provide clear steps to reach a solution, heuristics educates the decision-making process to solve the problem instead of offering the solution directly. This distinction inherits more detailed differences in solution optimization,

The associate editor coordinating the review of this manuscript and approving it for publication was Shuihua Wang¹.

linearity, and architecture. Though it cannot guarantee a solution, heuristics can sometimes be the more practical and effective problem-solving approach to a satisfactory solution than procedural algorithms. It is like a recursion algorithm that checks its conditions over time and calls the suitable corresponding functions. These conditions represent the online function that evaluates the parametric inputs and beliefs to decide the following actions over time. This makes heuristics the most suitable solution for artificial intelligence (AI) bots and multi-agent systems (MAS).

First, starting with the difference between heuristics and algorithms in terms of optimization, the optimal solution is the one that sufficiently satisfies the problem conditions with

the minimum cost. Engineers and developers usually adopt the possible solutions and decide which one to use based on performance metric evaluations like the asymptotic analysis. Such evaluations need to consider two things. First, all the compared solutions must sufficiently satisfy the problem. Second, the evaluation test can clarify the significance of the performance, and this means asymptotic analysis, for example, is suitable only when the algorithms belong to different levels of BigO.

When it comes to linearity, problems can be classified as linear or non-linear problems. The linear problem can be interpreted as solving an optimization problem concerning the linear objective function, subject to linear inequality and linear equality constraints [1]. Linear problems are intrinsically simpler and easier to solve, while in general non-linear problems, the solution becomes more complex [1]. If we consider a non-polynomial NP problem, the feasible solution is any solution that satisfies the constraints [2]. There are possible options that are common strategies adopted to develop an algorithm for a given optimization. The first option is to arrive at a good algorithm that always finds the optimal solutions. The simplex algorithm for linear programming is a good example. However, its worst-case complexity is exponential [3]. The second option is to obtain a computationally feasible algorithm that “almost” always finds optimal solutions. The third option is to achieve a computationally feasible algorithm that obtains “reasonably” reasonable feasible solutions. *Heuristic algorithms* have this property. It is an approximation algorithm if the heuristic algorithm is guaranteed to find feasible solutions that have value within a prespecified amount of the optimal value. Two measures are considered here: algorithm complexity which refers to the cost, and solution quality which refers to the efficiency probability.

In terms of architecture, algorithms used in problem-solving sometimes are too slow or fail to find the solution in a reasonable time, while heuristics may find a reasonable guess that is good enough for the problem at hand—this introduced heuristics to problem-solving and decision-making. The heuristic solution may not be the fastest or cheapest of all other solutions or approximate the exact solution. However, it provides a shortcut to reaching a goal in a relatively short time. Moreover, it is the best choice for making quick and informed decisions when problem features are missing and actual data is unavailable or too limited to use in complex problems. This feature made it a well-known approach in operational research, computer science, and artificial intelligence. More problems will be covered in the following sections.

Heuristics is the feasible option for the computational complexity of many problems where the optimal solution cannot be found in a short time. Some problems may be based on inaccurate data, so the solution based on estimated data is not optimal for the actual data. In such scenarios, i.e., computationally complex and ill-defined problems, a good solution that is near-optimum is preferred rather than optimal.

Also, it may require several solutions to make the final decision, mainly in cases where several criteria need to be adjusted using human judgment [4]. These problems involving multiple and conflicting objectives typically yield a set of trade-offs, e.g., the accurate solution versus the fastest one.

While algorithms use procedural steps, heuristics use recursion structure to solve the running problem based on online inputs and offline conditions. Some of the heuristic’s implementations are in transportation, manufacturing, military, and the management of other industrial logistics.

For the Crowd field, the use of heuristic-based models helped comprehend crowd behavior better and increased the reliability of simulation.

In moving pedestrian crowds, the decisions made by individuals over different routes affect the distribution of individuals.

Understanding collective behavior is essential for academic and practical purposes such as safety and management.

It is crucial to study the mechanisms that impact pedestrian route choice. Specifically, it is focused on the effects of time-independent information such as path lengths and time-dependent information as in queue lengths. Some rules can give a model a realistic crowd reaction, such as following the lead and stopping or avoiding a collision.

However, the lack of coverage and the understanding of this topic remains noticeable. This fact encouraged the objectives of this work, which are to present the latest work in heuristics concerning problem-solving in crowd management, discuss the challenges, and suggest additional ways to apply heuristics in the Crowd field.

The paper will essentially focus on heuristics through a defined classification: simple heuristics, meta-heuristics, and hyper-heuristics. Furthermore, it will address the role of heuristics in crowd studies and modeling in a comprehensive survey.

The structure of this paper is composed of five sections. The first section is the introduction. The second section provides an overview of recent reviews and surveys on heuristics, while it also summarizes further work related to this topic and crowd topics. The third section offers a background on the topic. The fourth section discusses the types of heuristics (i.e., their features and shortcomings) and proposes heuristic approaches for classifying and predicting crowd behavior. Finally, the last section concludes the paper.

II. RELATED WORK

Many of the earlier work has been focused on developing the formulations of the problem and their implementations using experimented heuristic and metaheuristic search techniques and their variations. It extends to automating problem-solving using hyper-heuristics. This section provides a comprehensive literature review regarding heuristics in computer science, operational research, and artificial intelligence. It explores perceptions about other review and survey researches of the related fields. Furthermore, it provides a

look over some of the related works which have influenced this research.

It becomes apparent how the field requires more coverage of the subject, especially in crowd studies, in which heuristics are one of the most applied approaches in crowd modeling, simulation, and decision support systems. Hence, this paper contributes to the field by summarizing and categorizing the state-of-art heuristics and problem-solving in crowd management using heuristic techniques. It extends to involve other fields adequately besides crowd study.

The work of Müller-Merbach [5] presented design-oriented heuristics in a relatively traditional and thorough survey. It presented the typical problems mentioned, which most of them are optimization problems, i.e., an optimal solution out of the feasible solutions is sought.

Another paper that discusses heuristics in a design theory is the paper of Gregory and Muntermann [6]. They defined heuristic theorizing as the process of proactively generating design theory for prescriptive purposes from problem-solving experiences.

A comprehensive article reviews the classical and heuristics algorithms for coverage path planning (CPP) in robotics to explore the solution space and find the optimal trajectory [7].

Gao *et al.* [8] reviewed the state-of-the-art of this meta-heuristics to identify the current status of meta-heuristics solving disassembly line balancing, planning, and scheduling problems (DLBPSP) in remanufacturing.

Previously [9], the authors reviewed the nature-inspired algorithms used in metaheuristic solutions. Such as bee colony optimization (BCO), particle swarm optimization, harmony search optimization, ant colony optimization (ACO), genetic algorithm (GA), firefly optimization, cuckoo search optimization, bat algorithm, memetic algorithm, symbiotic organism search optimization, simulated annealing, shuffled frog leaping algorithm, spider web algorithm, and hybrid algorithms.

The authors in [10] reviewed different heuristics for shorter cellular manufacturing systems (CMSs), where designing and implementing effective CMS involves many problems such as cell formation, machine layout, alternative process routes, and inventory lot sizing. Therefore, the meta-heuristic algorithm with hybridization is essential to address multiple optimization problems.

The work of Mostafaie *et al.* [11] reviewed meta-heuristics approaches for solving the graph coloring problem in a systematic review. Covering the algorithms and highlighting the significance of meta-heuristic optimization algorithms and several advantages they provide to tackle challenges associated with graph coloring.

Lei and Cai [12] presented another work contributing to production scheduling in manufacturing, which also addressed meta-heuristics in a survey. First, a new classification of scheduling problems is given, then three algorithms are described and their scheduling applications in a systematic review.

The work of Burke *et al.* [13] introduced hyper-heuristics in a thorough survey paper, including their origin and intellectual roots, the main types of approaches, and an overview of some related areas. The paper also surveyed the heuristic selection and move acceptance strategies (heuristic to choose heuristics) the heuristics generation (heuristics to generate heuristics). It further covers the application domain and the offline and online approaches.

Later on, in the work of [14], the authors continue to explore the classification of heuristics generation and selection and provide state of the art in hyper-heuristics in a comprehensive literature survey.

In Table 1, the paper investigates further related works by briefly reviewing the problems solved by heuristics algorithms and classifying them according to the problem category and the algorithm used in solving the problem. The heuristic techniques are applied and involved in many different fields such as supply chain, transportation, and logistics industry [15]–[20], network problem design [21], [22], machine computing [23]–[25], cloud computing [9], engineering problem design [26], smart cities and parking [27], military operations [28].

Crowd studies particularly have brought the attention by employing heuristic solutions in some areas of crowd applications. For example, in experimental analysis work [29], the crowd monitoring is mapped as a moving target traveling salesman problem (TSP) instance. Experimental evaluation of different heuristics approaches is done to the analyzed problem, and the solution was in favor of the genetic algorithm. In another work in crowd management [30], the system uses the Viola-Jones face detection algorithm to keep a count of the visitors' numbers at railway stations throughout the day and a heuristic data mining technique to analyze the stored data suggest the required action. Also, understanding the mechanisms of crowd dynamics compels involving the local behavior of pedestrians during interactions and the global behavior of the crowd at high density. For this, the author elaborated with a model using simple heuristics in [31], likewise deriving a hierarchy of kinetic and macroscopic models from a noisy variant of the heuristic behavioral Individual-Based Model [32].

III. BACKGROUND

A. INTELLECTUAL ROOTS, ORIGINS, AND DEFINITIONS

The concept of heuristics started to be more common in the sixties, where the word heuristics is originally derived from the Greek word *heurisken*, the word means investigating or discovering a persuasive solution rather than compelling [48], [49]. According to the Cambridge dictionary, heuristics definition is “a method and a way of learning or solving problems which allow people to discover things themselves and learn from their own experiences.” This definition implies the meaning of heuristics as approximate strategies or ‘rules of thumb’ for decision making and problem-solving that do not guarantee an exact solution but typically yield an acceptable solution or bring up one

TABLE 1. Related work taxonomy.

Type	Author, Year	Ref.	Field	Category	Problem	Algorithm
Heuristics	(Martins et al., 2020)	[15]	Supply chain, Transport, and logistics industry	Optimization	Vehicle routing problem (VRP).	Clarke and Wright Savings (CWS)
	(Ghaffari-Nasab et al., 2013)	[16]	Supply chain, Transport, and logistics industry	Computation	Location-routing problem with fuzzy demands (LRPFD).	A hybrid simulated annealing (SA) based heuristic.
	(Bădică et al., 2017)	[17]	Transport and logistics industry	Scheduling	Freight brokerage (vehicle-routing with pickup and delivery) problem.	Defining vehicle routes on the graph of cities, rather than on the graph of customer orders, as in standard set partitioning formulations.
	(Wu et al., 2007)	[23]	Problem Solving	Scheduling	The learning effect; a single-machine maximum lateness minimization problem.	Brach-and-bound, the earliest due date (EDD), SA.
	(Drezner et al., 2015)	[24]	Computing, problem-solving	Allocation	Planar p-median problem.	Hybridization of distribution-based variable neighborhood search and a new GA.
	(Davoodi et al., 2011)	[25]	Computing, problem-solving	Allocation	The p-center problem.	Heuristic Voronoi diagram algorithm.
	(Naik et al., 2013)	[30]	Crowd management, urban assets	Optimization	Providing proper decisions and actions.	Heuristic data mining with a Viola-Jones face detection algorithm.
	(Renaud et al., 2000)	[33]	Problem design	Computation	Pickup and delivery traveling salesman problem.	Two-phase heuristic; Double Insertion heuristic, Deletion, and Re-Insertion heuristic.
	(Amaliah et al., 2020)	[34]	Problem-solving	Computation	Transportation problem.	Bilqis Chastine Erma method (BCE), based on the initial basic feasible solution (IBFS).
	(Wang et al., 2019)	[35]	Bioinformatics, problem-solving	Computation	Capacitated vehicle routing problem.	DNA Algorithm based on Adelman–Lipton model
	(Zografos and Androutsopoulos, 2004)	[36]	Transportation, Risk assessment	Scheduling	Bi-objective vehicle routing problems.	Insertion algorithm; cost and risk minimization.
	(Sahling et al., 2009)	[37]	Problem Solving	Optimization	Multi-level capacitated lot-sizing problem with setup carry-overs (MLCLSP-L).	Fix-and-optimize algorithm
Meta-heuristics	(Milan et al., 2019)	[9]	Cloud computing	Scheduling	Load-balancing in cloud computing.	Nature-inspired algorithms
	(Sadrzadeh, 2012)	[18]	Manufacturing, Problem design	Production	The facility multi-line layout problem (FLP).	GA
	(Li et al., 2020)	[19]	Transport and logistics industry	Computation, Synchronization	Vehicle routing problem with time windows (VRPTW).	Improved artificial bee colony (IABC) algorithm.
	(Pinson et al., 2016)	[20]	Transportation	Scheduling	Dial-a-Ride Problem Applied to Car-pooling.	Greedy randomized adaptive search procedure (GRASP)
	(Poorzahedy and Rouhani, 2007)	[21]	Network design, transportation	Optimization	Transportation network design problem.	Hybridizing an ant system with search processes like GA, SA, and tabu search.
	(Gallo et al., 2010)	[22]	Network design, transportation	Optimization	Urban network design problem.	Scatter search algorithm
	(Hayyolalam and Pourhaji Kazem, 2020)	[26]	Engineering problem design	Optimization	Engineering high-dimension problems	Black Widow Optimization Algorithm (BWO)
	(Abidi et al., 2015)	[27]	Smart cities, parking	Assignment	Parking Assignment Problem.	Hybrid genetic algorithm
	(Chang et al., 2018)	[28]	Military operations	Optimization	Dynamic weapon target assignment (DWT A).	Artificial bee colony (ABC) algorithm with heuristic factor initialization
	(de Moraes and de Freitas, 2019)	[29]	Crowd monitoring	Optimization	The moving target traveling salesman problem	GA.

TABLE 1. (Continued.) Related work taxonomy.

	(Ropke and Pisinger, 2006)	[38]	Problem design	Optimization	Pickup-and-delivery with time windows.	Adaptive large neighborhood (ALNS) search.
	(Ahmadian et al., 2021)	[39]	Problem-solving	Scheduling	Just-in-time job-shop scheduling.	Variable neighborhood search (VNS)
	(Son et al., 2021)	[40]	E-Learning	Optimization	Multi-objective mathematical programming problem	GA, ACO
Hyper-heuristics	(Alinia Ahandani et al., 2012)	[41]	Problem-solving	Scheduling	Examination timetabling.	Hybridizing discrete particle swarm optimization (DPSO) with evolutionary algorithms (EA).
	(Sánchez et al., 2021)	[42]	Problem-solving	Optimization	Balanced partition problem	Simplified MAP-Elites algorithm
	(Zhao et al., 2021)	[43]	Problem-solving	Optimization	Boolean satisfiability (SAT) problems, one-dimensional bin packing problems, multi-objective Permutation flow-shop scheduling problem (PFSP), Personnel scheduling problem (PS), TSP, VRP	Cooperative Multi-Stage Hyper-Heuristic (CMS-HH) algorithm, GA
	(Olivas et al., 2020)	[44]	Fuzzy logic	Optimization	The 0-1 knapsack problem.	Genetic algorithm
	(Sabar and Ayob, 2009)	[45]	Education	Scheduling	Examination timetabling.	Scatter search-based hyper-heuristic (SS-HH).
	(Zheng et al., 2015)	[46]	Transportation	Optimization	Emergency railway transportation problem.	EA.
	(Tsai et al., 2014)	[47]	Cloud computing	Scheduling	Scheduling problem on a cloud computing system.	EA

faster [50]. Heuristics are famous for construction heuristics, using greedy methods that build solutions via iterations and descending heuristics that seek a local optimum from a given solution. It also depends on the problem to find a solution that will likely be the local optimum, mainly to select the nearest neighbor. In other words, when we think of iterative algorithms, heuristics comes up first to the mind, i.e., the converging algorithms and the algorithms with no proved convergence [5]. Heuristics can be used in intelligent systems while searching within the solution space. They are problem-specific approximation solutions, which means the problem is known in advance, and some features of the given problem are available to find solutions. There are numerous implementations of heuristics applications for transportation, planning, and other optimization problems.

Soon after, the complexity of problems has increased, and the conventional methods became more time-consuming to solve these problems. Later, a higher level of heuristics was introduced in the late seventies, called meta-heuristics [8]. The Greek prefix “meta” presented in the name indicates that these algorithms are “higher level” heuristics. A metaheuristic method is designed to solve a wide range of challenging problems without deeply adapting to each problem. In contrast to problem-specific heuristics, meta-heuristics are problem independent. These methods are used

when (I) a piece of incomplete or imperfect information about problem features, (II) a limited computational capacity, and there is a need for a fast, sufficient solution rather than an optimal one for an optimization problem. Some metaheuristics implementations are in generic block-chain and general machine learning problems.

Despite the widespread of heuristic and meta-heuristics search methods in solving real-world computational search problems, there are still some difficulties facing their application to newly encountered problems or even new instances of known problems due to the rapid increase of resources, capabilities, and problems complexity.

Hyper heuristics was introduced at the beginning of the 2000s to automate heuristic methods’ design and tuning to solve complex computational search problems [13], [14]. The main point here is that hyper-heuristics techniques do not search the space of solutions directly; instead, they explore heuristics and the heuristics combination. There are two types of hyper-heuristics: iterative hyper-heuristics and constructive hyper-heuristics. The first receives an initial “base” solution, and, at every iteration, a low-level heuristic is applied to produce a new solution s_0 . It becomes the new base for the following iterations if it is better than the prior one. It may either be discarded or still become the new base with a certain chance if it becomes worse. The way to automate and apply

the selection of low-level heuristics (LLH) is approached differently. The second type, which is more challenging, is where the high-level heuristics (HLH) select a sequence of lower ones to build a solution for the problem from scratch.

The last few years have seen an increase in the number of successful implementations of hyper-heuristics for problems such as production scheduling, timetabling, mapping, routing, and other related ones.

B. METHODOLOGIES AND USES

Naturally, the intention of the previous types of heuristics techniques and their applications in real-life problems differentiate them. This section presents some heuristics methodologies and applications. Table 1 refers to further works in this matter.

1) HEURISTICS

Most of the simple heuristic algorithms are applied for industrial logistic problems such as transportation and manufacturing fields.

In paper [34], to obtain the initial basic feasible solution (IBFS) of transportation problems, a heuristic method called Bilqis Chastine Erma method (BCE) was proposed, which obtained the lower total minimum cost. Another application example of real-life transportation planning using heuristics is carpooling [51]. The authors transformed the original Multiple Criteria Carpooling Optimization (MCCO) problem and assigned greedy heuristic algorithms for passenger assignment and route planning. Additional work that solves the transportation problem using greedy algorithms [52].

The study [53] proposes an adaptive extensive neighborhood search (ALNS) heuristic for robust gate assignment problems (GAP). The purpose of GAP is to assign a gate to the flight activities, including arrival, parking, and departure for a set of aircraft. A typical hub airport has over one hundred gates and more than one thousand flight activities on operation days. GAP's typical objectives include maximizing the airport's one-day throughput, maximizing airline preference, and minimizing towing time. Additional work incorporating heuristics in this field solved aircraft landing problems with a single runway [54]. The authors provide a decomposition-based adaptive large neighborhood search heuristic in which they split the aircraft landing problem into two sub-problems of scheduling and feasibility.

Other work tackles the problem of manufacturing system reconfiguration, known as the scalability problem [55]. The problem is defined as the ability to adjust a manufacturing system's production capacity through reconfiguration with minimal impact on time and cost. A set of practical principles is introduced to seek feasible configurations; a genetic algorithm is designed to search in the global solution space. Finally, an objective balancing function is defined and used to rank the proposed configurations.

Heuristics techniques are also applied to solve routing problems. The VRPSPTL, which is the abbreviation for

Vehicle Routing Problem with Simultaneous Pickup and Delivery with Time Limit, is a variant of the fundamental Vehicle Routing Problem where the vehicles serve delivery and pick up operations of the clients under time and movement restrictions limit. The VRPSPTL assigns a set of vehicle routes originating and terminating at a central depot with the total travel distance minimized. This problem is also a neighborhood search algorithm combined with the classic savings heuristic, variable neighborhood search, and a proposed perturbation mechanism [56]. Further work solves the truck and trailer routing problem (TTRP), a variant of the vehicle routing problem using simulated annealing heuristics [57].

Designing routes for a transit network is a multi-criteria decision-making problem requiring a large solution space to find an optimal solution. Therefore, the authors [58] used a stochastic beam search algorithm with a heuristic-inspired successor operator.

2) META-HEURISTICS

For task Allocation, the paper [59] presented a model to solve the component allocation problem instances and implemented two metaheuristics. The first is a genetic algorithm, and the second is an Ant Colony Optimization algorithm. Although genetic algorithms are widely used to solve component allocation, the paper is the first to implement the ant colony optimization algorithm.

Another critical problem category is scheduling. The authors in [39] introduced a meta-heuristic approach using a variable neighborhood search algorithm for the just-in-time job-shop scheduling (JIT-JSS) problem. Similar to the classical job-shop scheduling problem, the jobs in JIT-JSS have operations that must be performed on the machines in a given order. Each operation has a specific due date, earliness, and tardiness penalty coefficients, and any deviation of the operation completion time from its due date incurs an earliness or tardiness penalty.

A meta-heuristic model is developed for resource-constrained project scheduling problems (RCPSP) [60]. This problem's primary purpose is to find an optimal schedule of activities to satisfy realistic constraints like resource limitation and predecessor relations. They consider two-hybrid multi-objective teaching-learning-based optimization (TLBO) algorithms with a new solution structure based on essential and main concepts of multi-objective particle swarm optimization and multi-objective invasive weeds optimization algorithms to find near-optimal solutions of the model's complicated instances in a reasonable computational time.

Finding effective solutions for the transportation problem is one of the leading industrial and academic concerns.

The work [22] proposes an optimization model and a meta-heuristic algorithm for solving the urban network design problem. The problem involves optimizing an urban road network's layout by designing directions of existing roads and signal settings at intersections. A proposal for a Scatter

Search algorithm based on a random descent method was made and overcame the local optimum.

Furthermore, in work [61], the fuzzy fixed charge solid transportation problem (FFCSTP) considers the variable cost for transporting a batch of products. The fuzzy values are used per the parameter's value in the real world, and six meta-heuristics are utilized to solve the model. Most of these algorithms are used for a mathematical model in transportation problem literature. Tuning the parameters is very important in solving the given problem; therefore, the Taguchi method is used.

Many works have investigated the vehicle routing problem with time windows. One of them is construction [19], dispatching the required prefabricated components to construction sites in a prefabricated system using an improved artificial bee colony (IABC) algorithm.

In another study [62], the authors considered applying an ant colony optimization meta-heuristic to hybrid flow shop scheduling problems with minimizing the earliness-tardiness penalties.

3) HYPER-HEURISTICS

Scheduling is essential in the success of enhancing the performance of information systems. Most job and task scheduling problems performed on cloud computing systems are still either NP-hard or NP-complete. A hyper-heuristic algorithm is proposed to schedule cloud computing systems in [47] to reduce the makespan. Two detection operators -one for diversity detection and one for improvement detection- are proposed for the proposed algorithm to determine the timing to employ the low-level heuristic algorithm. The proposed algorithm can be applied to both sequence-dependent and independent scheduling problems.

Another article [42] proposed using an illumination algorithm called MAP-Elites to train sequence-based hyper-heuristics for the balanced partition problem. It is found that training with ME yields better results even in their lousy performance levels than randomly selected heuristics.

A hyper-heuristic approach is used to solve the emergency railway transportation planning problem [46]. The authors struggled to select an evolutionary algorithm and avoid selecting the appropriate EA for every problem instance. They proposed integrating the following three evolutionary operators into a hyper-heuristic algorithm. The algorithm includes differential evolution DE, fast evolutionary programming (FEP), comprehensive learning PSO (CLPSO) operators.

The work of [45] proposes a hyper-heuristic based on scatter search (SS-HH) to solve examination timetabling problems. The scatter search operates at a high level of abstraction, which intelligently evolves a sequence of low-level heuristics -each of them represents a single neighborhood structure- to use for a given problem.

Another exciting application of hyper-heuristics is in fuzzy logic systems. In the model [44], a GA is responsible for finding the fuzzy rule set that maximizes the fuzzy inference system's performance. The GA used in this work is a custom

discrete version of the original GA, where the genes can only have integer values, i.e., 1, 2, 3, or 4, representing the heuristic selected by the fuzzy rule.

Another aspect is the network mapping and routing. A genetic hyper-heuristic algorithm is used in mapping and routing [63] to co-optimize network-on-chip (NoC). The algorithm consists of bottom-level optimization, including various operators, and top-level optimization, selecting suitable operators through a 'reward' mechanism.

In paper [64], the authors propose a multi-objective hyper-heuristic (MOHH) framework for route planning for walking in a smart city. The authors design a set of low-level heuristics in the search framework to generate new routes. Moreover, adopt a reinforcement learning mechanism to select good low-level heuristics to accelerate searching speed. The study further improved the reinforcement learning-based multi-objective hyper-heuristic (RL-MOHH) algorithm and implemented a parallel version (RL-PMOHH) to obtain 80% optimal solutions in a large-scale road network.

A hyper-heuristic algorithm (HH) that consists of two levels, high-level heuristics and low-level heuristics, is proposed to solve the time-dependent green location routing problem with time windows (TDLGRP) [65]. The study objective is to minimize costs, including opened depot costs, enabled vehicle costs, and fuel consumption costs.

C. HOW HEURISTICS IS RELATED TO CROWD MANAGEMENT

Many crowd management works featured the search and decision-making problems that pursued heuristics approaches.

1) CROWDSOURCING

Crowdsourcing is growing in both industrial and academic fields by introducing new ways of conducting work, improving the understanding of utilizing crowds' potential, and improving crowdsourcing platforms and related practices.

The article [66] proposed a heuristic methodology for Crowd-Based Cooperative Task Allocation (CBCTA) that aims for effective management, coordinated use of crowd resources, and maximization of swarm intelligence. The methodology hinges on evolutionary heuristics to find proper resources that optimally balance matching rates and collaborative quality. The mathematical formulation of the optimal design of CBCTA is produced as a multi-objective optimization problem to find the optimal solution. The author resort to the well-known NSGA-II algorithm. The proposed CBCTA strategy is proven feasible and effective by conducting experiments with different scales of CBCTA problems.

In experimental crowdsourcing work [67], the authors presented a set of heuristics-relevant design guidelines for online communities- to support the expert evaluation of crowdsourcing platforms. Then, they used the guidelines to compare 20 crowdsourcing platforms. Heuristics can be utilized as a tool for crowdsourcing platforms, which raises a research point and proves that heuristic is not necessarily the search

algorithms for mere optimization but could also be a modular approach followed to evaluate a design or make a decision.

2) SIMULATION

Heuristics methods were mainly used in simulation and crowd modeling [68]. Human crowds display a wide variety of collective behaviors that support an efficient motion under conditions such as collision avoidance and pushing, bypassing, backward motion, and not touching other pedestrians.

The heuristic force-based model has been widely used to describe the pedestrian's flow problem. In the model, the dominant component is finding the optimal direction for the pedestrian. The force-based models describe the moving behavior of pedestrian i as the self-driven force f_i^D , the contact force f_i^C , and non-contact force f_i^S , all summed up to give the total force exerted on pedestrians. Moreover, an individual's location and velocity v can be updated using Newton's second law [31].

The remote repulsive force on particles being given by the sum of binary interactions and the calculation of self-driven force depending on the desired direction given by a fixed value may lead to undesired effects and unrealistic movement. In [69], the authors propose a cognitive science approach based on two behavioral heuristics, which overcome the previous problems. The first heuristic was that a pedestrian chooses the direction α_{des} that allows the most direct path to destination point O_i , considering obstacles. The second is a pedestrian who maintains a distance from the first obstacle in the chosen walking direction, ensuring a time to collide at least τ . In other words, the speed $v_{des}(t)$ is given by (1):

$$v_{des}(t) = \min(v_i^0, d_h/\tau) \quad (1)$$

where d_h is the distance between pedestrian i and the first obstacle in the desired direction α_{des} at time τ . The same approach goes for [70], with a slight modification in the cognitive heuristics formulation. The model considers the presence of other pedestrians, the step, and the body size and considers the Imitate the Majority Heuristic [71], in which pedestrians walking behavior influence each other.

In the paper [72], the authors investigated the impact of variable body sizes on direction choice and performed simulation using an improved heuristics-based work model [69]. The cost of adjusting body size is added into the function for optimal direction choice; with the different body size costs, the pedestrian will choose different directions.

The first available kinetic and fluid models were proposed in [32], a hierarchy of three macroscopic models, also derived from the heuristic behavioral Individual-Based Model of [69].

In another heuristics-based crowd modeling approach [73] based on the latter [69] and the Voronoi diagram, the authors consider the local density besides the location, velocity, the neighbor, and the safe distance for the recognition stage and the walking direction and the velocity in the reaction stage.

An agent-based pedestrian behavioral simulation model is presented in work [74] using simple heuristics for exploratory navigation. The model includes the visibility graph analysis (VGA) into the section-and-portal graph for quantifying the exploratory navigation performance of pedestrians. The model focuses on individuals for natural movement and realistic choices, the sensitivity of the model indicates that regional target, the memory of visiting records, and individual crowd tendency can indeed influence the pedestrians' navigation decisions and movement patterns.

In a more recent article [75], the authors proposed a modified heuristic force-based model that simultaneously takes contact forces as the information that affects self-driven forces. The authors expressed self-driven forces through static friction, explaining that pedestrians' motion is dependent on the static friction between pedestrians and ground, which was a different take that has not been considered before.

IV. DISCUSSION

In general, heuristic methods are used to eliminate one of the following two challenges. The first one is to abide by the time limits where the optimal algorithm could not. The second one is to add linearity to a non-linear problem. The first challenge is straight and easy to measure. The second challenge is debatable, and its convention belongs to the presence of NP problems and whether they belong to polynomial problems or not [76]. In both cases, shifting the action orders from the overall offline program to online recursive subprograms would automatically add linearity as online applications are always linear. Moreover, adding an expert system to the program will make the heuristic application fully functional and self-directed toward optimization.

Despite the fact that heuristics approaches are utilized in various fields, the topic of heuristics is still fuzzy. Some studies provided a brief review of the topic, while others merely focused on a specific type of heuristics and its advanced applications. This section provides an overview of the heuristic types and their possible applications in the crowd management field.

A. HEURISTIC TYPES COMPARISON

As presented earlier, heuristics have been reviewed separately and from various points of view. Heuristics are problem-dependent when searching the solution space; they adapt and exploit the given features of the problem, have a deterministic nature, and are usually iterative algorithms. Due to their higher demanding or resources, the heuristic function should be as greedy as possible. Sometimes, this makes them stuck in local optimum and fail to obtain the optimal global solution, especially in complex problems.

The need to avoid the optimization problem and to have a more straightforward implementation of heuristic methods led to the emergence of the meta-heuristic. Meta-heuristic is a problem-independent method that does not take advantage of the specific features of the problem. Instead, it has a

TABLE 2. Heuristics types comparison.

Feature	Simple Heuristics	Meta-heuristics	Hyper-heuristics
Solution Quality	Inexact solution, near-optimal (approximation), fast	Inexact, quick, near-optimal (approximation)	Inexact solution, near-optimal (approximation)
Search Space	Search the solution space.	Search the solution space to get a hopefully better solution.	Do not search the space of solutions rather the space of heuristics or meta-heuristics.
Problems	Problem dependent.	Problem independent; some features of the problem are missing.	Problem-independent; It does not require any problem-specific data, therefore, applied to different problems.
Drawbacks	Getting trapped in a local optimum and failing, in general, to obtain the global optimum solution for complex problems.	Focusing on a single optimization problem as multi-optimization can result in poor performance.	The hyper-heuristics methods are expensive; offline approaches are incredibly cheap and fast to apply since it selects from heuristics; however, the online approaches that generate new heuristics require more effort and cost.
Example	Greedy algorithms	Nature-based, genetic algorithms	Evolutionary algorithms

randomization nature. Therefore, it is usually an iterative and nature-inspired technique.

Meta-heuristics used abstracted heuristics to explore the solution domain and get a better solution that aims to reach the global optimum. However, it is necessary to do some parameter tuning to adapt to the problem. Moreover, many researchers deal with meta-heuristics techniques in which they focus on single optimization problems as multi-optimization problems; this results in poor performance. Thus, a robust meta-heuristic algorithm with hybridization is essential to address multiple problems simultaneously [10].

Hyper-heuristics with artificial intelligence, machine learning, and deep learning incorporated nature seek to automate finding a heuristic way to solve a problem. Hence, their search space is not the usual space of solutions but the space of heuristics; they are presented in the literature as “heuristics to find/generate heuristics.” They are also non-problem specific where the decision for a heuristic selection is based on problem-independent measures. Such as the change in the quality of a solution when the selected heuristic is used. Once implemented, they can be directly applied to other problems. Hyper-heuristics are considered a substitute for the meta-heuristics developed for a specific problem and require fine-tuning related parameters [77]. Compared to a methodology that operates directly on the solution space, the evolutionary heuristic generation process is often computationally expensive, especially if it is not used on other problems.

To sum up, Table 2 distinguishes the three types of heuristics to comprehend the distinctions and assist in selecting a suitable heuristic method for a given problem.

B. HEURISTICS FOR CROWD MANAGEMENT

Heuristics have multiple roles in crowd management, as we could find in the previous sections. This section will concentrate on the applications of heuristics in crowd management. Crowd management includes crowd recognition, tracking,

simulation, and prediction [68]. From a heuristic perspective, tracking is a kind of recognition and prediction results from simulation. Consequently, the applications of heuristics in crowd management fields are in recognition and simulation. Notably, the literature is more driven to crowd modeling, especially agent-based models where agents make a decision that represents a realistic behavior.

The first aspect is the recognition which involves detecting, tracking, and analyzing the crowd. This recognition is also used to detect any abnormal behavior of the crowd, which extends crowd recognition to crowd management. It will also help to develop a crowd risk index (CRI) and decision support systems (DSS) for crowd management [68]. The issue of selecting the optimal method and architecture for crowd detection belongs to the polymerization problem, which may refer to the simplest, the fastest, the most accurate, or the most efficient at producing accuracy while minimizing computational complexity. For example, the most efficient convolutional neural network CNN architectures [78] are used as methods to classify abnormal behavior of the crowd [79]. Using a simple 2D CNN structure to train the network with three layers and two classes, normal or abnormal behavior, the primary goal is to improve the accuracy of the original CNN by applying a simulated annealing algorithm. Where SA is a technique of random search for the global optimization problem. It imitates the process of annealing in material processing. The standard procedure in simulated annealing for optimization problems is to generate the solution vector (randomly selected), calculate the fitness function, and initialize the temperature value. Next, randomly select a new solution to evaluate the solution depending on the fitness function, decrease the temperature, then stop or repeat. However, the SA algorithm is used to find the condition of best accuracy and minimize estimated error and indicator of network complexity [78]. The CNN takes the still frames of crowd video scenes as an input where there are convolutional

and subsampling layers. The last layer calculates the loss function using the computation of the values and weights as solution vector (x) optimized by the SA algorithm by adding Δx randomly. The experiment results showed that the new method, named CNN-SA, presents the best accuracy for all epochs compared to other models and the original CNN model. In the crowd detection case, it is possible to validate the performance of this method on different benchmark datasets; AGORASET is one example of the recommended virtual datasets for crowd video analysis [80], [81].

The second aspect is the application of heuristics in crowd simulation and prediction. The searching nature of heuristic and its applications in AI and MAS makes it the best solution for simulating crowd movement, whether on a microscopic or macroscopic level. The simulation agent A uses the heuristic method H to evaluate the recognized input $X_{A,t}$ of sensing the surrounding environment, including the fixed and moving object at time T and its previous belief in memory $M_{A,t}$ to decide its following action $F_{A,t+1}$.

$$F_{A,t+1} = H(X_{A,t}, M_{A,t}) \quad (2)$$

The belief $M_{A,t}$ consists of the start belief $M_{A,0}$, and recursive recognition of the surrounding environment while processing the previous actions. $M_{A,0}$ includes the conditions of the heuristic function and the offline inputs. This means we can find $F_{A,t+1}$ using a derived non-recursive heuristic function H' that can deal with $M_{A,0}$ and $X_{A,T}$. This macroscopic model of simulation paved the way to linear macroscopic model for crowd prediction based on solving the heuristic recursion function in terms of time instance $t_i \in T$.

$$F_{A,t+1} = H'(X_{A,t}, M_{A,0}) \quad (3)$$

In the case of microscopic simulation, we have multiple simulation agents $a_i \in A$ refers to the moving objects. In this case, the actions $F_{a_i,t+1}$ are calculated separately and affect the inputs of other agents. It makes the complexity of the recursion grows exponentially in terms of time, reaching the worst level, $\text{BigO} \in n^n \Leftrightarrow t+1$. However, the online application of the heuristic method is still running on linear bases $\text{BigO} \in n \Leftrightarrow t$, which makes such simulation possible. Solving the recursion, in this case, is harder than in the macroscopic model. However, in an abstracted and isolated environment, the input X is nothing but the offline environment set in addition to the agents' interactions. In other words, the input X may be replaced by an agent that has its own belief and is affected by the interaction of the other agents. Such as in the following equations.

$$F_{a_i,t+1} = H(X_{a_i,t}, M_{a_i,t}) \quad (4)$$

$$X_{A,t+1} = H'(t, M_{x,0}) \rightarrow X_{a_i,t+1} = H'(t, M_{x,0})_{a_i} \quad (5)$$

$$M_{a_i,t} = H'(t, M_{a_i,0}) \quad (6)$$

$$\begin{aligned} F_{a_i,t+1} &= H\left(H'(t-1, M_{x,0})_{a_i}, H'(t, M_{a_i,0})\right) \\ &= H''(t, M_{x,0}, M_{a_i,0}) \end{aligned} \quad (7)$$

proving that it is mathematically possible to develop a linear prediction model using a non-recursive derivation of the heuristic recursion function in an isolated abstract environment. Still, applying this proof to actual crowd management techniques is expensive in terms of time and equipment and not yet applied. Alternatively, meta-heuristic neural networks are trained using crowd detection techniques, and then used to predict crowd movement. This fact reveals how the meta-heuristic approach applied in deep neural networks could efficiently bypass the limitations and where the trained weights $w_i \in W$ would eventually match the parameters of the offline belief M and the suitable interactions over time T .

V. CONCLUSION

This paper reviewed as many heuristics-related articles as possible to clear any fuzziness around the topic, gather the whole story behind its existence, present the latest advances and touch on its relation to crowd studies. It covers previous work similar to this paper, such as review papers and surveys. It also covers the state-of-the-art heuristics approaches to real-world problems. These techniques fall under this classification: simple heuristics, meta-heuristics, and hyper-heuristics. The paper also explores the origins and the reasons for using heuristics approaches, which come from the following: (I) the classic methods used in problem-solving sometimes are too slow to find a solution that calls for practical and less complicated ones. (II) sometimes, problems have to be built on estimated data or based on inaccurate data that require a flexible approach to tackle such an ill-defined problem.

Firstly, the paper browses the academic history of the topic and the methodologies under the same classification. Then, it directs the focus on the current role of heuristics in crowd studies. It distinguishes the difference between heuristics types. While the simple heuristics are problem dependent, the other two types are problem independent, and they do not require any problem-specific data, hence, applied to different problems. Most importantly, the heuristics and meta-heuristics techniques search the solution space while the hyper-heuristics techniques search the heuristics space. The most perused algorithms for real-world problems in heuristics literature are the greedy neighborhood search algorithms, genetic search algorithms, and evolutionary search algorithms.

Finally, the paper previews the heuristic presence in crowd field problems and concludes that heuristics, in general, are helpful in various crowd management applications. For example, it can be utilized to build effective models or automate the process by relying on data to develop crowd risk indicators (CRI) or decision support systems (DSS). While CRI can monitor, detect, and predict hazardous crises, DSS is set to help the decision process in real-life situations and simulations.

In crowd modeling, simple heuristics can be applied by relying on simple rules and functions for an agent to choose the following action in a simulation. On the other hand, metaheuristics are more generic and applicable to adapt any

problem with ambiguous features to generate more realistic and sophisticated crowd dynamics. Furthermore, hyper Heuristics can be involved in crowd simulation to automate and generate more realistic behavior by constructing a high-level heuristic out of a data-driven low-level heuristic.

Heuristics are highly effective and can enhance the performance of crowd applications, which makes them helpful in automating applications and finding data-based solutions instead of designing a new model.

The authors encourage the other researchers to specify the use of this approach in crowd literature clearly and employ heuristics techniques in this field for further purposes, such as enhancing accuracy or minimizing the computation complexity cost.

REFERENCES

- [1] Z. C. Dagdia and M. Mirchev, "Chapter 15—When evolutionary computing meets astro- and geoinformatics," in *Knowledge Discovery in Big Data from Astronomy and Earth Observation*, P. Škoda and F. Adam, Eds. Amsterdam, The Netherlands Elsevier, 2020, pp. 283–306.
- [2] C. Ryan, "Computer algorithms," in *Encyclopedia of Physical Science and Technology*, R. Meyers, Ed. New York, NY, USA: Academic, 2003, pp. 507–523.
- [3] U. Čibej and J. Mihelič, "Chapter three—Adaptation and evaluation of the simplex algorithm for a data-flow architecture," in *Advances in Computers*, vol. 106, A. R. Hurson and V. Milutinović, Eds. Amsterdam, The Netherlands Elsevier, 2017, pp. 63–105.
- [4] U. Aickelin and A. Clark, "Heuristic optimisation," *J. Oper. Res. Soc.*, vol. 62, no. 2, pp. 251–252, Feb. 2011, doi: [10.1057/jors.2010.160](https://doi.org/10.1057/jors.2010.160).
- [5] H. Müller-Merbach, "Heuristics and their design: A survey," *Eur. J. Oper. Res.*, vol. 8, no. 1, pp. 1–23, 1981, doi: [10.1016/0377-2217\(81\)90024-2](https://doi.org/10.1016/0377-2217(81)90024-2).
- [6] R. W. Gregory and J. Muntermann, "Research note—Heuristic theorizing: Proactively generating design theories," *Inf. Syst. Res.*, vol. 25, no. 3, pp. 639–653, Sep. 2014. [Online]. Available: <http://www.jstor.org/stable/24700315>.
- [7] C. S. Tan, R. Mohd-Mokhtar, and M. R. Arshad, "A comprehensive review of coverage path planning in robotics using classical and heuristic algorithms," *IEEE Access*, vol. 9, pp. 119310–119342, 2021, doi: [10.1109/ACCESS.2021.3108177](https://doi.org/10.1109/ACCESS.2021.3108177).
- [8] K. Z. Gao, Z. M. He, Y. Huang, P. Y. Duan, and P. N. Suganthan, "A survey on meta-heuristics for solving disassembly line balancing, planning and scheduling problems in remanufacturing," *Swarm Evol. Comput.*, vol. 57, Sep. 2020, Art. no. 100719, doi: [10.1016/j.swevo.2020.100719](https://doi.org/10.1016/j.swevo.2020.100719).
- [9] S. T. Milan, L. Rajabion, H. Ranjbar, and N. J. Navimipour, "Nature inspired meta-heuristic algorithms for solving the load-balancing problem in cloud environments," *Comput. Oper. Res.*, vol. 110, pp. 159–187, Oct. 2019, doi: [10.1016/j.cor.2019.05.022](https://doi.org/10.1016/j.cor.2019.05.022).
- [10] V. Kesavan, R. Kamalakannan, R. Sudhakarapandian, and P. Sivakumar, "Heuristic and meta-heuristic algorithms for solving medium and large scale sized cellular manufacturing system NP-hard problems: A comprehensive review," *Mater. Today, Proc.*, vol. 21, pp. 66–72, Jan. 2020, doi: [10.1016/j.matpr.2019.05.363](https://doi.org/10.1016/j.matpr.2019.05.363).
- [11] T. Mostafaie, F. M. Khyiabani, and N. J. Navimipour, "A systematic study on meta-heuristic approaches for solving the graph coloring problem," *Comput. Oper. Res.*, vol. 120, Aug. 2020, Art. no. 104850, doi: [10.1016/j.cor.2019.104850](https://doi.org/10.1016/j.cor.2019.104850).
- [12] D. Lei and J. Cai, "Multi-population meta-heuristics for production scheduling: A survey," *Swarm Evol. Comput.*, vol. 58, Nov. 2020, Art. no. 100739, doi: [10.1016/j.swevo.2020.100739](https://doi.org/10.1016/j.swevo.2020.100739).
- [13] E. Burke, M. Hyde, G. Kendall, G. Ochoa, E. Özcan, and R. Qu, "A survey of hyper-heuristics," *Comput. Sci., Sch. Comput. Sci. Inf. Technol.*, Univ. Nottingham, Nottingham, U.K., Tech. Rep. NOTTCSR-SUB09062414182747, 2009.
- [14] E. K. Burke, M. Gendreau, M. Hyde, G. Kendall, G. Ochoa, E. Özcan, and R. Qu, "Hyper-heuristics: A survey of the state of the art," *J. Oper. Res. Soc.*, vol. 64, no. 12, pp. 1695–1724, Jul. 2013, doi: [10.1057/jors.2013.71](https://doi.org/10.1057/jors.2013.71).
- [15] L. D. C. Martins, C. Bayliss, A. A. Juan, J. Panadero, and M. Marmol, "A savings-based heuristic for solving the omnichannel vehicle routing problem with pick-up and delivery," *Transp. Res. Proc.*, vol. 47, pp. 83–90, Jan. 2020, doi: [10.1016/j.trpro.2020.03.082](https://doi.org/10.1016/j.trpro.2020.03.082).
- [16] N. Ghaffari-Nasab, S. G. Ahari, and M. Ghazanfari, "A hybrid simulated annealing based heuristic for solving the location-routing problem with fuzzy demands," *Sci. Iran.*, vol. 20, no. 3, pp. 919–930, 2013.
- [17] A. Bădică, C. Bădică, F. Leon, and L. Luncean, "Declarative representation and solution of vehicle routing with pickup and delivery problem," *Proc. Comput. Sci.*, vol. 108, pp. 958–967, Jan. 2017, doi: [10.1016/j.procs.2017.05.261](https://doi.org/10.1016/j.procs.2017.05.261).
- [18] A. Sadrzadeh, "A genetic algorithm with the heuristic procedure to solve the multi-line layout problem," *Comput. Ind. Eng.*, vol. 62, no. 4, pp. 1055–1064, May 2012, doi: [10.1016/j.cie.2011.12.033](https://doi.org/10.1016/j.cie.2011.12.033).
- [19] J.-Q. Li, Y.-Q. Han, P.-Y. Duan, Y.-Y. Han, B. Niu, C.-D. Li, Z.-X. Zheng, and Y.-P. Liu, "Meta-heuristic algorithm for solving vehicle routing problems with time Windows and synchronized visit constraints in prefabricated systems," *J. Cleaner Prod.*, vol. 250, Mar. 2020, Art. no. 119464, doi: [10.1016/j.jclepro.2019.119464](https://doi.org/10.1016/j.jclepro.2019.119464).
- [20] C. Pinson, H. M. Afsar, and C. Prodhon, "Heuristic approaches to solve a generalized dial-a-ride problem applied to car-pooling," *IFAC-PapersOnLine*, vol. 49, no. 12, pp. 1187–1191, 2016, doi: [10.1016/j.ifacol.2016.07.668](https://doi.org/10.1016/j.ifacol.2016.07.668).
- [21] H. Poorzahedy and O. M. Rouhani, "Hybrid meta-heuristic algorithms for solving network design problem," *Eur. J. Oper. Res.*, vol. 182, no. 2, pp. 578–596, Oct. 2007, doi: [10.1016/j.ejor.2006.07.038](https://doi.org/10.1016/j.ejor.2006.07.038).
- [22] M. Gallo, L. D'Acerno, and B. Montella, "A meta-heuristic approach for solving the urban network design problem," *Eur. J. Oper. Res.*, vol. 201, no. 1, pp. 144–157, Feb. 2010, doi: [10.1016/j.ejor.2009.02.026](https://doi.org/10.1016/j.ejor.2009.02.026).
- [23] C.-C. Wu, W.-C. Lee, and T. Chen, "Heuristic algorithms for solving the maximum lateness scheduling problem with learning considerations," *Comput. Ind. Eng.*, vol. 52, no. 1, pp. 124–132, Feb. 2007, doi: [10.1016/j.cie.2006.10.007](https://doi.org/10.1016/j.cie.2006.10.007).
- [24] Z. Drezner, J. Brimberg, N. Mladenović, and S. Salhi, "New heuristic algorithms for solving the planar p-median problem," *Comput. Oper. Res.*, vol. 62, pp. 296–304, Oct. 2015, doi: [10.1016/j.cor.2014.05.010](https://doi.org/10.1016/j.cor.2014.05.010).
- [25] M. Davoodi, A. Mohades, and J. Rezaei, "Solving the constrained p-center problem using heuristic algorithms," *Appl. Soft Comput.*, vol. 11, no. 4, pp. 3321–3328, Jun. 2011, doi: [10.1016/j.asoc.2011.01.001](https://doi.org/10.1016/j.asoc.2011.01.001).
- [26] V. Hayyolalam and A. A. P. Kazem, "Black widow optimization algorithm: A novel meta-heuristic approach for solving engineering optimization problems," *Eng. Appl. Artif. Intell.*, vol. 87, Jan. 2020, Art. no. 103249, doi: [10.1016/j.engappai.2019.103249](https://doi.org/10.1016/j.engappai.2019.103249).
- [27] S. Abidi, S. Krichen, E. Alba, and J. M. Molina, "A new heuristic for solving the parking assignment problem," *Proc. Comput. Sci.*, vol. 60, pp. 312–321, Jan. 2015, doi: [10.1016/j.procs.2015.08.132](https://doi.org/10.1016/j.procs.2015.08.132).
- [28] T. Chang, D. Kong, N. Hao, K. Xu, and G. Yang, "Solving the dynamic weapon target assignment problem by an improved artificial bee colony algorithm with heuristic factor initialization," *Appl. Soft Comput.*, vol. 70, pp. 845–863, Sep. 2018, doi: [10.1016/j.asoc.2018.06.014](https://doi.org/10.1016/j.asoc.2018.06.014).
- [29] R. S. de Moraes and E. P. de Freitas, "Experimental analysis of heuristic solutions for the moving target traveling salesman problem applied to a moving targets monitoring system," *Expert Syst. Appl.*, vol. 136, pp. 392–409, Dec. 2019, doi: [10.1016/j.eswa.2019.04.023](https://doi.org/10.1016/j.eswa.2019.04.023).
- [30] A. Naik, D. Darde, S. Chitalia, and V. Zope, "Crowd management using Viola Jones algorithm and heuristic data mining," *Int. J. Comput. Appl.*, vol. 84, no. 17, pp. 9–13, Dec. 2013, doi: [10.5120/14676-2949](https://doi.org/10.5120/14676-2949).
- [31] M. Moussaïd and J. D. Nelson, "Simple heuristics and the modelling of crowd behaviours," in *Pedestrian and Evacuation Dynamics*. 2014, pp. 75–90, doi: [10.1007/978-3-319-02447-9_5](https://doi.org/10.1007/978-3-319-02447-9_5).
- [32] P. Degond, C. Appert-Rolland, M. Moussaïd, J. Pettré, and G. Theraulaz, "A hierarchy of heuristic-based models of crowd dynamics," *J. Stat. Phys.*, vol. 152, no. 6, pp. 1033–1068, 2013, doi: [10.1007/s10955-013-0805-x](https://doi.org/10.1007/s10955-013-0805-x).
- [33] J. Renaud, F. F. Boctor, and J. Ouenniche, "A heuristic for the pickup and delivery traveling salesman problem," *Comput. Oper. Res.*, vol. 27, no. 9, pp. 905–916, 2000, doi: [10.1016/S0305-0548\(99\)00066-0](https://doi.org/10.1016/S0305-0548(99)00066-0).
- [34] B. Amaliah, C. Faticah, and E. Suryani, "A new heuristic method of finding the initial basic feasible solution to solve the transportation problem," *J. King Saud Univ. Comput. Inf. Sci.*, vol. 2020, pp. 1–10, Jul. 2020, doi: [10.1016/j.jksuci.2020.07.007](https://doi.org/10.1016/j.jksuci.2020.07.007).

- [35] Z. Wang, X. Ren, Z. Ji, W. Huang, and T. Wu, "A novel bio-heuristic computing algorithm to solve the capacitated vehicle routing problem based on Adleman–Lipton model," *Biosystems*, vol. 184, Oct. 2019, Art. no. 103997, doi: [10.1016/j.biosystems.2019.103997](https://doi.org/10.1016/j.biosystems.2019.103997).
- [36] K. G. Zografos and K. N. Androutsopoulos, "A heuristic algorithm for solving hazardous materials distribution problems," *Eur. J. Oper. Res.*, vol. 152, no. 2, pp. 507–519, 2004, doi: [10.1016/S0377-2217\(03\)00041-9](https://doi.org/10.1016/S0377-2217(03)00041-9).
- [37] F. Sahling, L. Buschkuhl, H. Tempelmeier, and S. Helber, "Solving a multi-level capacitated lot sizing problem with multi-period setup carry-over via a fix-and-optimize heuristic," *Comput. Oper. Res.*, vol. 36, no. 9, pp. 2546–2553, Sep. 2009, doi: [10.1016/j.cor.2008.10.009](https://doi.org/10.1016/j.cor.2008.10.009).
- [38] S. Ropke and D. Pisinger, "An adaptive large neighborhood search heuristic for the pickup and delivery problem with time windows," *Transp. Sci.*, vol. 40, no. 4, pp. 455–472, 2006. [Online]. Available: <http://www.jstor.org/stable/25769321>.
- [39] M. M. Ahmadian, A. Salehipour, and T. C. E. Cheng, "A meta-heuristic to solve the just-in-time job scheduling problem," *Eur. J. Oper. Res.*, vol. 288, no. 1, pp. 14–29, Jan. 2021, doi: [10.1016/j.ejor.2020.04.017](https://doi.org/10.1016/j.ejor.2020.04.017).
- [40] N. T. Son, J. Jaafar, I. A. Aziz, and B. N. Anh, "Meta-heuristic algorithms for learning path recommender at MOOC," *IEEE Access*, vol. 9, pp. 59093–59107, 2021, doi: [10.1109/ACCESS.2021.3072222](https://doi.org/10.1109/ACCESS.2021.3072222).
- [41] M. A. Ahandani, M. T. V. Baghmisheh, M. A. B. Zadeh, and S. Ghaemi, "Hybrid particle swarm optimization transplanted into a hyper-heuristic structure for solving examination timetabling problem," *Swarm Evol. Comput.*, vol. 7, pp. 21–34, Dec. 2012, doi: [10.1016/j.swevo.2012.06.004](https://doi.org/10.1016/j.swevo.2012.06.004).
- [42] M. Sanchez, J. M. Cruz-Duarte, J. C. Ortiz-Bayliss, and I. Amaya, "Sequence-based selection hyper-heuristic model via MAP-elites," *IEEE Access*, vol. 9, pp. 116500–116527, 2021, doi: [10.1109/ACCESS.2021.3106815](https://doi.org/10.1109/ACCESS.2021.3106815).
- [43] F. Zhao, S. Di, J. Cao, J. Tang, and Jonrinaldi, "A novel cooperative multi-stage hyper-heuristic for combination optimization problems," *Complex Syst. Model. Simul.*, vol. 1, no. 2, pp. 91–108, Jun. 2021, doi: [10.23919/CSMS.2021.0010](https://doi.org/10.23919/CSMS.2021.0010).
- [44] F. Olivias, I. Amaya, J. C. Ortiz-Bayliss, S. E. Conant-Pablos, and H. Terashima-Marin, "A fuzzy hyper-heuristic approach for the 0–1 knapsack problem," in *Proc. IEEE Congr. Evol. Comput. (CEC)*, Jul. 2020, pp. 1–8, doi: [10.1109/CEC48606.2020.9185710](https://doi.org/10.1109/CEC48606.2020.9185710).
- [45] N. R. Sabar and M. Ayob, "Examination timetabling using scatter search hyper-heuristic," in *Proc. 2nd Conf. Data Mining Optim.*, Oct. 2009, pp. 127–131, doi: [10.1109/DMO.2009.5341899](https://doi.org/10.1109/DMO.2009.5341899).
- [46] Y.-J. Zheng, M.-X. Zhang, H.-F. Ling, and S.-Y. Chen, "Emergency railway transportation planning using a hyper-heuristic approach," *IEEE Trans. Intell. Transp. Syst.*, vol. 16, no. 1, pp. 321–329, Feb. 2015, doi: [10.1109/TITS.2014.2331239](https://doi.org/10.1109/TITS.2014.2331239).
- [47] C.-W. Tsai, W.-C. Huang, M.-H. Chiang, M.-C. Chiang, and C.-S. Yang, "A hyper-heuristic scheduling algorithm for cloud," *IEEE Trans. Cloud Comput.*, vol. 2, no. 2, pp. 236–250, Apr./Jun. 2014, doi: [10.1109/TCC.2014.2315797](https://doi.org/10.1109/TCC.2014.2315797).
- [48] D. Rozoff, "Heuristic," *Account. Rev.*, vol. 39, no. 3, pp. 768–769, Nov. 1964. [Online]. Available: <http://www.jstor.org/stable/242477>.
- [49] M. Laguna and R. Martí, "Heuristics," in *Encyclopedia of Operations Research and Management Science*, S. I. Gass M. C. Fu, Eds. Boston, MA, USA: Springer, 2013, pp. 695–703.
- [50] P. M. Todd, "Heuristics for decision and choice," in *International Encyclopedia of the Social & Behavioral Sciences*, N. J. Smelser and P. B. Baltes, Eds. Oxford, U.K.: Pergamon, 2001, pp. 6676–6679.
- [51] G. Filcek, M. Hojda, and J. Žak, "A heuristic algorithm for solving a multiple criteria carpooling optimization (MCCO) problem," *Transp. Res. Proc.*, vol. 27, pp. 656–663, Jan. 2017, doi: [10.1016/j.trpro.2017.12.108](https://doi.org/10.1016/j.trpro.2017.12.108).
- [52] B. Jin, W. Zhu, and A. Lim, "Solving the container relocation problem by an improved greedy look-ahead heuristic," *Eur. J. Oper. Res.*, vol. 240, no. 3, pp. 837–847, Feb. 2015, doi: [10.1016/j.ejor.2014.07.038](https://doi.org/10.1016/j.ejor.2014.07.038).
- [53] C. Yu, D. Zhang, and H. Y. K. Lau, "An adaptive large neighborhood search heuristic for solving a robust gate assignment problem," *Expert Syst. Appl.*, vol. 84, pp. 143–154, Oct. 2017, doi: [10.1016/j.eswa.2017.04.050](https://doi.org/10.1016/j.eswa.2017.04.050).
- [54] S. Vaddlamani and S. Hosseini, "A novel heuristic approach for solving aircraft landing problem with single runway," *J. Air Transp. Manage.*, vol. 40, pp. 144–148, Aug. 2014, doi: [10.1016/j.jairtraman.2014.06.009](https://doi.org/10.1016/j.jairtraman.2014.06.009).
- [55] S. Huan, L. Aiping, L. Xuemei, X. Liyun, and M. Giovanni, "A heuristic approach to solve an industrial scalability problem," *Proc. CIRP*, vol. 63, pp. 21–26, Jan. 2017, doi: [10.1016/j.procir.2017.03.310](https://doi.org/10.1016/j.procir.2017.03.310).
- [56] O. Polat, C. B. Kalayci, O. Kulak, and H.-O. Günther, "A perturbation based variable neighborhood search heuristic for solving the vehicle routing problem with simultaneous pickup and delivery with time limit," *Eur. J. Oper. Res.*, vol. 242, no. 2, pp. 369–382, Apr. 2015, doi: [10.1016/j.ejor.2014.10.010](https://doi.org/10.1016/j.ejor.2014.10.010).
- [57] S.-W. Lin, V. F. Yu, and S.-Y. Chou, "Solving the truck and trailer routing problem based on a simulated annealing heuristic," *Comput. Oper. Res.*, vol. 36, no. 5, pp. 1683–1692, May 2009, doi: [10.1016/j.cor.2008.04.005](https://doi.org/10.1016/j.cor.2008.04.005).
- [58] K. A. Islam, I. M. Moosa, J. Mobin, M. A. Nayeem, and M. S. Rahman, "A heuristic aided stochastic beam search algorithm for solving the transit network design problem," *Swarm Evol. Comput.*, vol. 46, pp. 154–170, May 2019, doi: [10.1016/j.swevo.2019.02.007](https://doi.org/10.1016/j.swevo.2019.02.007).
- [59] I. Al-Azzoni and S. Iqbal, "Meta-heuristics for solving the software component allocation problem," *IEEE Access*, vol. 8, pp. 153067–153076, 2020, doi: [10.1109/ACCESS.2020.3015864](https://doi.org/10.1109/ACCESS.2020.3015864).
- [60] S. Zabihi, M. R. Kahag, H. Maghsoudlou, and B. Afshar-Nadjafi, "Multi-objective teaching-learning-based meta-heuristic algorithms to solve multi-skilled project scheduling problem," *Comput. Ind. Eng.*, vol. 136, pp. 195–211, Oct. 2019, doi: [10.1016/j.cie.2019.07.032](https://doi.org/10.1016/j.cie.2019.07.032).
- [61] S. Golmohamadi, R. Tavakkoli-Moghaddam, and M. Hajiaghahi-Keshтели, "Solving a fuzzy fixed charge solid transportation problem using batch transferring by new approaches in meta-heuristic," *Electron. Notes Discrete Math.*, vol. 58, pp. 143–150, Apr. 2017, doi: [10.1016/j.endm.2017.03.019](https://doi.org/10.1016/j.endm.2017.03.019).
- [62] S. Khaloufi, F. Ghedjati, and A. Hamzaoui, "A meta-heuristic approach to solve a JIT scheduling problem in hybrid flow shop," *Eng. Appl. Artif. Intell.*, vol. 23, no. 5, pp. 765–771, Aug. 2010, doi: [10.1016/j.engappai.2010.01.008](https://doi.org/10.1016/j.engappai.2010.01.008).
- [63] C. Xu, Y. Liu, P. Li, and Y. Yang, "Unified multi-objective mapping for network-on-chip using genetic-based hyper-heuristic algorithms," *IET Comput. Digit. Techn.*, vol. 12, no. 4, pp. 158–166, Jul. 2018, doi: [10.1049/iet-cdt.2017.0156](https://doi.org/10.1049/iet-cdt.2017.0156).
- [64] Y. Yao, Z. Peng, and B. Xiao, "Parallel hyper-heuristic algorithm for multi-objective route planning in a smart city," *IEEE Trans. Veh. Technol.*, vol. 67, no. 11, pp. 10307–10318, Nov. 2018, doi: [10.1109/TVT.2018.2868942](https://doi.org/10.1109/TVT.2018.2868942).
- [65] C. Zhang, Y. Zhao, and L. Leng, "A hyper-heuristic algorithm for time-dependent green location routing problem with time Windows," *IEEE Access*, vol. 8, pp. 83092–83104, 2020, doi: [10.1109/ACCESS.2020.2991411](https://doi.org/10.1109/ACCESS.2020.2991411).
- [66] L. Zhao, W. Tan, L. Xu, N. Xie, and L. Huang, "Crowd-based cooperative task allocation via multicriteria optimization and decision-making," *IEEE Syst. J.*, vol. 14, no. 3, pp. 3904–3915, Sep. 2020, doi: [10.1109/JSYST.2020.2966646](https://doi.org/10.1109/JSYST.2020.2966646).
- [67] S. Campo, V.-J. Khan, K. Papangelis, and P. Markopoulos, "Community heuristics for user interface evaluation of crowdsourcing platforms," *Future Gener. Comput. Syst.*, vol. 95, pp. 775–789, Jun. 2019, doi: [10.1016/j.future.2018.02.028](https://doi.org/10.1016/j.future.2018.02.028).
- [68] A. M. Al-Shaery, S. S. Alshehri, N. S. Farooqi, and M. O. Khozium, "In-depth survey to detect, monitor and manage crowd," *IEEE Access*, vol. 8, pp. 209008–209019, 2020, doi: [10.1109/ACCESS.2020.3038334](https://doi.org/10.1109/ACCESS.2020.3038334).
- [69] M. Moussaïd, D. Helbing, and G. Theraulaz, "How simple rules determine pedestrian behavior and crowd disasters," *Proc. Natl. Acad. Sci. USA*, vol. 108, no. 17, pp. 6884–6888, Apr. 2011, doi: [10.1073/pnas.1016507108](https://doi.org/10.1073/pnas.1016507108).
- [70] Q. Xu, B. Mao, X. Liang, and J. Feng, "Simple cognitive heuristics applied to modeling pedestrian behavior dynamics," *Proc. Social Behav. Sci.*, vol. 43, pp. 571–578, Jan. 2012, doi: [10.1016/j.sbspro.2012.04.130](https://doi.org/10.1016/j.sbspro.2012.04.130).
- [71] G. Gigerenzer, "Why heuristics work," *Perspect. Psychol. Sci.*, vol. 3, no. 1, pp. 20–29, 2008, doi: [10.1111/j.1745-6916.2008.00058.x](https://doi.org/10.1111/j.1745-6916.2008.00058.x).
- [72] N. Guo, M.-B. Hu, and R. Jiang, "Impact of variable body size on pedestrian dynamics by heuristics-based model," *Phys. A. Stat. Mech. Appl.*, vol. 465, pp. 109–114, Jan. 2017, doi: [10.1016/j.physa.2016.08.001](https://doi.org/10.1016/j.physa.2016.08.001).
- [73] Y. Xiao, Z. Gao, Y. Qu, and X. Li, "A pedestrian flow model considering the impact of local density: Voronoi diagram based heuristics approach," *Transp. Res. C, Emerg. Technol.*, vol. 68, pp. 566–580, Jul. 2016, doi: [10.1016/j.trc.2016.05.012](https://doi.org/10.1016/j.trc.2016.05.012).
- [74] W. L. Wang, S. M. Lo, and S. B. Liu, "A cognitive pedestrian behavior model for exploratory navigation: Visibility graph based heuristics approach," *Simul. Model. Pract. Theory*, vol. 77, pp. 350–366, Sep. 2017, doi: [10.1016/j.simpat.2017.07.002](https://doi.org/10.1016/j.simpat.2017.07.002).

- [75] L. Wang and S. Shen, "A pedestrian dynamics model based on heuristics considering contact force information and static friction," *Transportmetrica B, Transp. Dyn.*, vol. 7, no. 1, pp. 1117–1129, Dec. 2019, doi: [10.1080/21680566.2019.1568926](https://doi.org/10.1080/21680566.2019.1568926).
- [76] L. Fortnow, "The status of the p versus NP problem," *Commun. ACM*, vol. 52, no. 9, pp. 78–86, Sep. 2009, doi: [10.1145/1562164.1562186](https://doi.org/10.1145/1562164.1562186).
- [77] E. Özcan, B. Bilgin, and E. E. Korkmaz, "A comprehensive analysis of hyper-heuristics," *Intell. Data Anal.*, vol. 12, no. 1, pp. 3–23, 2008, doi: [10.3233/IDA-2008-12102](https://doi.org/10.3233/IDA-2008-12102).
- [78] L. M. R. Rere, M. I. Fanany, and A. M. Arymurthy, "Metaheuristic algorithms for convolution neural network," *Comput. Intell. Neurosci.*, vol. 2016, May 2016, Art. no. 1537325, doi: [10.1155/2016/1537325](https://doi.org/10.1155/2016/1537325).
- [79] A. A. Afiq, M. A. Zakariya, M. N. Saad, A. A. Nurfarzana, M. H. M. Khir, A. F. Fadzil, A. Jale, W. Gunawan, Z. A. A. Izuddin, and M. Faizari, "A review on classifying abnormal behavior in crowd scene," *J. Vis. Commun. Image Represent.*, vol. 58, pp. 285–303, Jan. 2019.
- [80] P. Allain, N. Courty, and T. Corpetti, "AGORASET: A dataset for crowd video analysis," in *Proc. 1st ICPR Int. Workshop Pattern Recognit. Crowd Anal.*, Nov. 2012, pp. 1–6. [Online]. Available: <https://hal.archives-ouvertes.fr/hal-00904216>.
- [81] S. A. M. Saleh, S. A. Suandi, and H. Ibrahim, "Recent survey on crowd density estimation and counting for visual surveillance," *Eng. Appl. Artif. Intell.*, vol. 41, pp. 103–114, May 2015, doi: [10.1016/j.engappai.2015.01.007](https://doi.org/10.1016/j.engappai.2015.01.007).



NORAH S. FAROOQI (Member, IEEE) received the B.Sc. degree first class Honors with GPA 3.99 out of 4) in computer science from Umm Al-Qura University, Saudi Arabia, in 2007, and the M.Sc. degree (Hons.) in information systems and the Ph.D. degree in computer science from the University of Sheffield, U.K., in 2010 and 2013, respectively. She is currently an Associate Professor in computer science with the Information Science Department, College of Computer and Information Systems, Umm Al-Qura University. She is also a General Supervisor of the Data Management Office. She was the Vice Dean of the Institute of Consulting Research and Studies and the Deputy Director of the Engineering Science Research Center. She has many publications related to databases and security. Her research interests include the area of systems development, information systems, crowd management, and the IoT. She is a reviewer in a number of academic journals related to computer topics.



ALI M. AL-SHAERY (Member, IEEE) received the B.Sc. degree in civil engineering from Umm Al-Qura University, Makkah, Saudi Arabia, in 2003, the M.Sc. degree in surveying from University College London, U.K., in 2006, and the Ph.D. degree in geospatial engineering from the University of New South Wales, Australia, in 2013. He is currently an Associate Professor with the Department of Civil Engineering, the Vice-President of Innovation and Entrepreneurship (VPIE), and the Vice-President of Development and Community Service (VPDCS). He was the Dean of the Institute of Consulting Research and Studies and the Deputy Director of the Transportation and Crowd Management Center of Research Excellence (TCMCORE). His research interests include global navigation satellite systems (GNSS) real-time kinematic positioning RTK, geographic information systems, location-based applications, smart and sustainable cities, and crowd management. He is also the Chair of Governing Board of Center of Safety, Risk and Crisis Management.



MOHAMED O. KHOZIUM (Member, IEEE) received the first M.S. degree in the aviation science (laser applications) from the Air War Studies Institute, in 1994, and the High Diploma, M.S., and Ph.D. degrees in computer science and information system from Cairo University. He made several studies in the electronic warfare from USA and France. Since 2015, he has been a Professor. He is currently a Professor with the Faculty of the Computer Science and Information Systems, Sadat Academy, Egypt, a Consultant of the Computer Science Research Center, Future University in Egypt, and a member of the Institute of Consulting (MIC). His research interests include electronic warfare, AI, ES, BI, crowd management, machine learning, and other simulation aspects. He has been awarded "Doing Duties Honestly and Faithfully Award" and "Excellent Duty Medal from the First Level" from the President of Egypt, in 1996 and 2006, respectively.



MOHAMMAD ADNAN M.B. AL-KAWA received the B.Sc. degree in computer science from the University of the People. He is currently pursuing the M.Sc. degree in computer science with a focus on artificial intelligence with the University of York. He is currently an AI Theorist. His research interests include AI theories, factorization, agents and MAS modeling, uncertainty and fuzzy systems, federated model design, spontaneous robotic interaction, and AI applications in e-Governance.

...