The Maximum Flow Problem

Nadia Abd-Alsabour

Cairo University, Egypt.

ORCID: 0000-0003-4080-5615

Abstract

In many fields, there exist various crucial applications that can be handled as maximum flow problems. Examples are: electrical power, airline scheduling, communication networks, computer sciences, etc. This clarifies why various researchers have handled them (which are NP optimization problems and a type of network optimization problems) with the utilization of various approaches. The paper explores this as we tackled it by the utilization of two approaches dealing with different instances from various domains. Furthermore, the approaches based on the network flow are addressed. Additionally, the latest applications of the most well-known as well as the recent approaches tackling this crucial optimization problem are detailed. Further, the approaches for dealing with other optimization problems based on the maximum flow as well as the most recent real-world applications are highlighted.

Keywords: The maximum flow problem; flow-based algorithms; the maximum flow approaches; hybrid and parallel approaches, nature inspired algorithms.

1. INTRODUCTION

Nowadays, the quality of the private and public infrastructure has turned out to be so significant. In numerous nations, some or all the transportation, water, energy, and telecommunications networks do not suit their given tasks. Subsequently, these infrastructure networks have been investigated from different perspectives [1].

In the daily life, numerous various sorts of networks are encountered such as transportation, telecommunications, water, and electricity networks. These are briefly addressed below.

The transportation networks are viewed as one of the most crucial infrastructure systems. Their significant aspects include network design (such as adding extra roads to a highway network), traffic prediction (predicting the traffic levels on the road network), and real-time traffic control (providing real-time information for the vehicles moving through a transportation network) [1]. Inaccurate design for these networks raises noise, pollution, etc. resulting in vehicular traffic existing in the daily life in almost all towns worldwide.

Path direction and traffic control are optimization problems by their nature since the aim is to use the available road network in such a way that the throughput is maximized or the overall network load is minimized [2]. This highlights the significance of studying the optimization in networks through which a large number of real-world networks applications can be tackled.

The telecommunication networks also are considered as one of the most evident infrastructure systems. Their crucial problems are flow control (the management of demands through rejecting or accepting message routing demands) and routing (discovering the ideal routing pattern for currently active message requests) [1].

Electricity networks include crucial infrastructure decision problems. These are estimating the electrical power and emphasizing that the electrical distribution network fulfills the expected requests appropriately.

Water resource networks have numerous related crucial infrastructure networks such as sewer pipelines, water pipelines, and irrigation systems [1]. Water transportation problems refer to where water flows through pipes or their substitutes. Distributing the water raises problems such as its bad distribution, which will make the water move extra distances to reach the required destination [3].

A water distribution network is a group of numerous parts like pumps, reservoirs, pipes, and valves that are linked to supply the water to customers. The interconnecting pipes represent the significant portion of the capital expense. The ideal design of this system is to define the ideal configuration for the pipe sizes, which provide the lowest expense for the particular design of the network with the end goal that the restrictions on water amounts and pressures at the customers' nodes are satisfied [4].

The distribution network operation and planning face huge difficulties in providing a steady, and secure service in the presence of a significant rate of uncertainty in the network conduct. Meanwhile, they additionally advantage from the progressions of futurity and present distribution networks concerning the availability of expanded assets, and diverseness and flexibility of the distribution networks [5]. These challenges include how to adapt to these changes, coordinate an optimization procedure in a problem-solving system to proficiently seek the ideal methodology, and how to enhance the utilization of the resources appropriately so as to accomplish cost-effective operations and offer services' quality as needed [1], [5].

We focus on the flow optimization in this paper. More particularly, the maximum flow problem is addressed. Besides, the algorithms based on the flow in networks are highlighted.

The remainder of this work is as per the following. The maximum flow problem is introduced in the second section. The third section addresses the approaches for dealing with other optimization problems based on the maximum flow. Section 4 demonstrates the flow-based algorithms. The fifth section addresses the most common approaches for handling the maximum flow problem. The sixth section details the carried out experiments. Section 7 discusses the obtained results. The last section addresses the conclusions, and also the future work.

2. THE MAXIMUM FLOW PROBLEM

In the daily life, there are numerous crucial problems that can be modeled as flow networks such as the flow of information through communication networks, current through electrical networks, liquids through pipes, calls in a communication network for particular services, parts through assembly lines [6]-[7], and it also emerges in process scheduling when allotting resources to processes in heterogeneous computing [8]. Weighted directed connected graphs (a weighted directed connected graph is known as a network if the weight of each arc is a non-negative value [9]) are usually utilized to represent these networks that model systems in which a material is created at the starting node (the one and only one that does not have entering edges and it is known also as the source), moves through the system, and at the target node (the one and only one that does not have leaving edges and it is also known as the sink) no longer exists [6]. Every edge has a capacity that is not negative and accepts the one whose quantity must not exceed its capacity. The flow of a flow network is a valued procedure [10] defined on the links of the graph, i.e., assigning real numbers (non-negative) to the arcs fulfilling the following 2 conditions:

1. The flow on every arc must not surpass its capacity and is non-negative [9]. This is known as arc condition or capacity constraint as in Equation 1.

$$0 \leq f_{ij} \leq c_{ij} \tag{1}$$

where f_{ij} is the flow and c_{ij} is the capacity

2. The incoming flow must equal the outgoing one at every node [9]. This is known as the flow constraint.

The maximum flow is the maximum incoming flow value to the sink node. According to the previous constraints, the overall flow to the sink node can be maximized. The following diagram depicts an instance of a flow network.
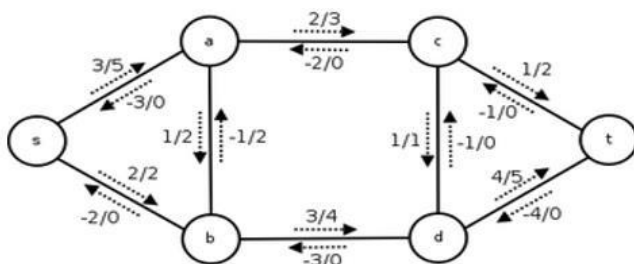

Fig. 1. An instance of flow networks [10].

Getting the maximum flow of such a material in a flow network without breaking the restrictions of the capacity (non-negative integer associated with every edge indicating the maximum allowed amount of such a material which can be sent through this edge and is also known as the weight of this edge) is known as the maximum flow problem [6]. It is the problem of finding how much material can be transported between two locations through limited capacity ways, i.e., getting the uttermost overall flow from the source to the target in a directed weighted graph [10]-[11].

It is one of the classic combinatorial optimization problems, and numerous optimization problems can be reduced to it. It can be considered as a particular instance of network flow issues having increased difficulty like the circulation issue [9], [12].

It has been investigated by numerous researchers from experimental and theoretical perspectives utilizing various techniques since it models various important applications and is encountered in numerous crucial applications [10], [13]-[14] such as designing the transportation paths (deciding on the maximum number of travelers who can be moved between two stations per day, people in a public transportation system, traffic on roads [9]-[10], and other transportation networks as rail, and highway ones), shipping industry (organizations have to keep particular quantities of stocks in various warehouses and from the producer to the customers [9], [15]), distributing water to towns (deciding on how much fluids within pipes) [7], traffic in computer and telephone networks [6], and its applications in other numerous fields such as logistics, computer science, bioinformatics, security, engineering, operations research, biological and logistic networks, and the allotment of hardware resources [8], [11]-[13]. In addition, it has numerous interesting special cases such as the bipartite graphs, which in turn model numerous natural flow problems [14].

To give an illustration, given the one-way roads connecting different towns and the greatest quantity of lorries that can be driven along every road, a factory wants to transport its products to the distribution center in another town. This factory must know the maximum number of these cars, which can be sent to that center [16]. One more illustration: a railway network connecting 2 towns through intermediate towns. In which, every connection holds a digit symbolizing its capacity. The aim is to get the maximum flow from a particular town to another [12].

It is a main problem in graph approaches and optimization [13]. It is classified among quite known fundamental issues for combinatorial optimization in directed weighted graphs [10]. As an optimization problem, it involves discovering the largest possible flow through a flow network having one starting node and one end node [16]. It is an NP-hard problem and also one of the most well-known optimization problems in networks (network analysis is crucial in numerous fields such as geographical information systems (GIS)) [6]–[7], [11], [16]. This highlights the significance of the optimization & their algorithms that can handle such a problem and other optimization applications as most real-life problems are about maximizing or minimizing some measure to optimize some

outcome [18]. Examples are: industry (electric power, airlines, engineering, trucking, mechanics, oil), economics (production, investments, purchasing, distribution), agriculture (distribution, human resources), transportation (goods delivery), and algorithms [16], [19].

3. TACKLING OPTIMIZATION PROBLEMS BASED ON THE MAXIMUM FLOW

Lately, new approaches for dealing with other optimization problems based on the maximum flow have been advanced. For instance, the work of Altiparmak and Tosun [20] proposed the use of maximum flow approaches for retrieving the replicated data from numerous disks in order to ensure the minimum time for retrieving the replicated data in the heterogeneous architectures, which is an essential problem for heterogeneous storage.

One more example is crafted by Mulajkar et al. [21] who executed a maximum flow based approach for mining association rules for heterogeneous parallel disk and distributed architecture in order to get the optimal retrieval time.

Additionally, Peng et al. [22] executed the utilization of the maximum flow theory for assessing the network performance, i.e., they conceived and transformed the performance analysis of 2 sorts of typical architectures of data center networks into the maximum-flow problem. They obtained the critical edges for every architecture with the utilization of the minimal cut sets and the maximum flow [22]. A typical data center is composed of hundreds of servers linked by an immense network and usually administrated by a single operator. It gives cost- efficient and flexible access to scalable storage and computing assets necessitated for the current cloud computing requirements. In order to enhance the performance, and get good quality access to a miscellany of applications and services given by a data center, it is essential to utilize it efficiently [23]. This should be evaluated enough.

In addition, Kumar et al. [24] represented the human brain as a flow network of vertices sending bits of information to one another for the purpose of evaluating the functional brain connectivity, which is utilized for forecasting the individual dissimilarities in conduct.

Besides, by building a directed graph, Ren et al. [25] turned the topology reconfiguration in reconfigurable structures into a maximum flow problem to get the solution with the utilization of a maximum flow approach. Their proposed work greatly enhanced the mend ratio of defective processing components as well as obtained a polynomial reconfiguration time. This is achieved via enhancing the utilization of spare processing components with the least effects on the space, the retard, and the throughput.

4. FLOW-BASED ALGORITHMS

Lately, a variety of flow-based algorithms have been introduced to tackle various problems in numerous fields as addressed below.

Lamghari et al. [26] proposed algorithms on the basis of network-flow for tackling the open-pit mine production scheduling issue. It has gained important attention recently. It is about optimizing the current net value of a mining asset. It is a stochastic problem as metal uncertainty has to be considered. The utilized approaches should consider the mining sequence as well as the flow of the material once mined besides considering that the mined material can have numerous destinations. This is handled in the work of Lamghari and Dimitrakopoulos [26] as the authors developed a heuristic algorithm inspired by the flow of networks for discovering efficiently the enhanced solutions in a huge neighborhood. Besides, they integrated it with a variable neighborhood descent algorithm for getting better results and to deal with their shortcomings when utilizing them individually to tackle this vital problem. Their experimental results show that the performance of the network-flow based algorithms is better than that of other utilized heuristics in terms of robustness and efficiency.

Karpagam et al. [27] proposed a flow-based algorithm for tackling the issue of transporting the maximum capacity with the minimum cost, particularly within the pipeline transportation that can be utilized in the milk and juice industry, etc.

Orecchia and Zhu [28] advanced a flow-based approach to tackle the cut-improvement issue. It is the first local cut-improvement approach (its running time is reliant on the size of the input set in lieu of the whole graph's size). Besides, it accomplishes such a local conduct whilst carefully matching the identical theoretical assurance like the elegant global approach introduced by Andersen et al. [29] (was necessitating dealing with a few sole commodity maximum flow computations through the entire graph). The key objective is developing good local graph segmentation approaches. The previous local approaches were on the basis of random walk.

Gastner et al. [30] proposed a flow-based approach having motion equations, which are numerically simpler to handle in comparison to the previous algorithms. This leads to direct parallelization in such a way the computation only consumes little time when handling detailed and complex data. In spite of the speedup, their approach has the advantages of previous approaches. These advantages are scaling properly all regions, generating a map projection for each point, and properly suiting the areas together.

Svancara and Surynek [31] utilized the search approaches that examine systematically the search space. These algorithms need a well-defined heuristic function to be able to enhance the computational efficiency via altering the order in which the states are extended. More particularly, the authors introduced a novel flow-based heuristic via relaxing the multi-commodity flow decrease in order to enhance the fundamental A* approach.

Through Secure Shell, a hacker is able to enter and dominate remote hosts. Then, he can disrupt the host and even utilize it for assaulting different systems. Therefore, discovering the intrusions is significant for network administrators to avoid the damage to hosts and networks. This is handled in the work

of Hellemons et al. [32] as they introduced a flow-based intrusion discovering system for Secure Shell attacks. This system utilizes an effective approach for real-time discovering the continuing harms and permits the determination of compromised assault preys.

Coverage is so crucial in the wireless sensor networks. The linked target coverage problem considers the sensor location in order to ensure the requirements of connectivity and covering. Shan et al. [33] tackled the type of this problem that is relied upon the probabilistic sensing model that can define the goodness of coverage more properly. The authors defined the minimum e-connected target coverage problem such that it minimizes the numbers of the sensors to meet the needs of connectivity and coverage. Specifically, they mapped it into a flow graph and tackled it by utilizing their novel flow-based approximation algorithm.

5. THE MOST COMMON APPROACHES

In general, there are 2 main classes for dealing with the maximum flow problem; push-relabel (the Goldberg-Tarjan) and augmenting path algorithms (the Ford Fulkerson). In addition, metaheuristics (are approximate approaches and problem independent that direct the search to search efficiently the set of all possible solutions to discover a close optimal solution within acceptable time instead of guaranteeing getting the ideal one) and more particularly nature inspired metaheuristics (metaheuristics that simulate principles and rules from nature) have been utilized for handling it. This is studied in the following subsections [17]-[18], [34]-[35].

5.1. The Ford Fulkerson

It was proposed in 1955 by Ford and Fulkerson [34]-[35]. This is the first, and the most known algorithm utilized for tackling the maximum flow problem in optimizing flow networks as well as a variety of computer applications [9]. It is an easy and iterative approach. It begins with an initial flow for each arc equals zero. Then, it is increased in every iteration until it cannot be incremented any longer [6]. In other words, it begins with the initial flow, builds iteratively a sequence of flow of increasing value (accordingly change the flow), and exits with the maximum flow (being guaranteed to get always the maximum flow if it ends is its main strong merit) [9]. Its key idea is simple; get a flow of an augmenting path of the available paths from the root to the target in the given flow network and then get another one, and so on. It is based on discovering the augmenting path (one at a time) from the root to the target [9] and terminates when there is no augmenting path (a path from the start node to the end node passing through only positive weighted arcs and does not contain any cycles [10], [36]). It is hard to calculate its actual time complexity as it is impossible to exactly determine the number of the augmenting paths [34]-[36]. In the worst case, its time complexity is large [37]. This can be overcome with the use of good computing resources at no cost due to the recent advancements in them.

It should be called a method rather than an algorithm. This is because getting the augmenting paths (its main idea) was not completely determined in its first edition, which is addressed in various implementations having various time complexities [6], [37]. It is not only utilized for tackling various flow problems (as shown below) but also for evidencing different theorems for network flow [38].

Recently, it has been applied for tackling various important real-world flow problems in a variety of domains as follows. Kyi and Naing [9] utilized it in 2018 for discovering the possible maximum flow of the water from the starting node to the target node by utilizing actual data of a particular distribution pipeline network in Pyigyitagon Township in Myanmar.

It means sending the water with the use of various paths (not necessarily to be disjointed) in which everyone consists of pipelines of various capacities (diameters) [36]. Besides the capacity constraints (in any pipe, the flow is equivalent to or less than its capacity), there is no intermediate storage, i.e., at any intermediate node, the flow in equals the flow out [36].

Iqbal et al. [39] in 2018 utilized it for an important real-world application which is airline scheduling (each flight is represented by a node and the arrival time is the capacity), i.e., having an airline schedule such that keeping the airplanes in the flight mode for a maximum amount of time. This is to guarantee the appropriate time maintenance since the overall time of a flight is a crucial reason for passengers based on which tickets may or may not be bought (every flight is operated based on a multi-thousand dollar gamble of whether tickets will be bought or not). Besides, airplanes consume expenses during their flight and also on standby at the airports. Airlines make a great effort to accumulate adequately their flights in order to offer the shortest possible connection time (numerous routes necessitate connections) to make the maximum utilization of their flights as well as maintain them. This is to stay in the airline business with maximum possible profit.

Similarly, it was utilized in Gebreanenya [40] to discover the maximum flow in the Ethiopian airlines. In addition, it was utilized in Zhang [15] for dealing with the maximum flow in a biological network.

Dimri and Ram [36] deal with the maximum flow problem as a linear programming problem with the goal of getting the maximum overall flow from the source to the destination subject to the previous constraints. However, they got the same results of the Ford Fulkerson [36]. Tackling the airline scheduling problem properly can lead to a variety of benefits such as saving time and money, using airplanes resources adequately, providing more information about the flight, utilizing the airline team efficiently, and organizing the airline efficiently [39].

Nowadays, almost all the countries are interested in dealing with urban mobility problems (e.g. traffic flows) since they result in traffic congestions (a significant urban transportation problem worldwide affecting the life quality for numerous people, the environment, and the economic output) which take place when traffic surpasses the capability of a current way

ability. This is because of numerous reasons such as the rapid population increase, the liberty of having private vehicles, drivers' behavior, and bad traffic means [41]. This was the motivation of the work of Abdullah and Hua [41] in a real-world application in Kota Kinabalu, since the location is a part of the central business region in which the traffic is larger than that of the others. They discovered the maximum flow (with the use of the Ford Fulkerson), the shortest path, and the bottleneck path from Bandaraya Mosque (the starting node) to Kampung Air (the end node). This is to decide on the roadway facilities that need to be enhanced. In addition, drivers can select alternate routes based on the results of this study.

Later the same authors in Abdullah and Hua [42], extended this work to discover the bottlenecks of the used real-life network (to determine which roads need to be enhanced) with the use of the maximum flow (discovered by the Ford and Fulkerson) and the min-cut theorem [42].

Another real-life application with the use of it was the work of Moore et al. [43] aiming to enhance the roads design, and the traffic flows to overcome the traffic jam problem in Bangkok. The authors modified it in order to tackle the maximum flow problem having extra characteristics that are multiple sources, multiple sinks, allowing both two-way and one-way flows on arcs, and speed dependent capacities. They utilized their algorithm to estimate the maximum traffic flow in a ways' network within Bangkok. In their algorithm, they modeled a roads network by a graph to study the traffic flow problem (where the arcs' weights are the road capacities, i.e., maximum no. of means of transport per hour) [43].

In addition to its previous applications mentioned above, it has a variety of other applications in computer vision, and image processing such as image segmentation, stereo correspondence, and optical flow estimation. This is achieved by converting the problem being solved into the maximum flow problem so that it can be tackled utilizing the Ford Fulkerson [37].

Further, it is the most classic approach for the time-cost trade-off problem that is fundamentally a maximum flow problem. This is an example of the numerous attempts that have been made in the past to deal with numerous practical problems as maximum flow ones that were tackled by the Ford Fulkerson approach [44].

Recently, numerous extensions to it have been introduced as follows.

Wei and Su [44] enhanced it to make it able to deal with the maximum flow in a network having negative flows or capacities rather than only negative ones. This is to tackle more applications since numerous maximum flow problems necessitate non-negative flows and capacities [44].

Besides, Kinsley and Maheswari [38] enhanced it by introducing the concept of a tree flow network as a deduction of the flow network that is a directed flow network having only a unique route between any pair of vertices [38].

Jiang et al. [45] proposed a parallel Ford Fulkerson, where in every iteration, all the edges in the given flow network are processed concurrently in parallel. This is to handle its main

problem, which is the time complexity.

Han et al. [12] studied the utilization of the Ford Fulkerson in uncertain environments, since practical applications usually have uncertain factors. If the arc capacities are uncertain variables, then this network is called uncertain.

5.2. The Push-Relabel Algorithm

It is also known as the Goldberg-Tarjan. It is based on the preflow, which is similar to a flow with only a single difference; the overall incoming flow into a node can surpass the overall outgoing flow. It drives the local flow surplus in the direction of the target through the estimated shortest paths. It is intuitive, quick, and straightforward [46]. This algorithm is widely utilized due to its simplicity.

There are numerous parallelized push-relabel maximum flow approaches. This is because of their memory locality feature, which makes it simple to parallelize them [47].

Recently, numerous extensions to it have been introduced as follows.

Gallo et al. [48] extended it to tackle the monotone parametric maximum flow problems. Their work was later pursued further by Iwata et al. [49] to tackle the polymatroid intersection.

Another extension to it (by utilizing gap and global relabeling) was performed by Sato [50] for the purpose of dealing with the maximum flow issue on top of Pregel (the vertex-centric model where every node is viewed as a processor, and the entire graph executes distributed concurrent computing on the basis of the bulk synchronous parallel model). In addition, He and Hong [51] applied the push-relabel algorithm to be run in parallel on CUDA machines.

Since the matching in bipartite graphs is considered a special case of the maximum flow problem, Deveci et al. [52] simplified it to tackle the bipartite matching issue. The authors advanced atomic and lock-free GPU execution of this approach for discovering maximum cardinality matchings within these graphs.

In order to enhance its capabilities for tackling real-world graph clustering problems, Nate et al. [53] utilized a variant of it (by using a warm-start procedure and global relabeling heuristic to tackle cut problems rapidly) to calculate the minimum s-t cuts (preflows). Local graph clustering is discovering tightly linked clusters of vertices close to the seed ones within a big graph. It has been used in numerous real-life applications such as image segmentation, community detection, information retrieval, etc. In practice, semi-supervised data related to a hidden target module are represented by seed vertices [54].

It has been utilized recently for proposing algorithms based on it for handling other optimization problems such as the work of Chaudhuri et al. [54]. They discovered the minimum cost spanning tree out of the spanning trees with the maximum degree, which is a generalization of combinatorial optimization problems (like the traveling salesman). Another

instance is the proposed work of Langguth et al. [55]. They utilized parallel push–relabel based approaches for calculating the maximum cardinality matching in bipartite graphs that is an essential optimization problem having various applications like chemical structure analysis, scheduling, and bioinformatics. This can benefit various matching applications. Moreover, Fleischer and Iwata [56] proposed a push-relabel frame to deal with the submodular procedure minimization, and then pursued further their work to handle the parametric minimization for a good map succession of submodular functions.

5.3. Nature Inspired Computing for the Maximum Flow Problem

It is an exceptionally interdisciplinary area as it incorporates knowledge from different areas with computer science like chemistry, physics, life science, and biology. It is expanding speedily. As of the date of its origination, numerous nature-inspired computing areas have been introduced lately. They can be categorized into the following categories; biology-based approaches (BBA), chemistry-based approaches (CBA), and physics-based approaches (PBA) [57]-[58].

Newly, nature inspired techniques have been employed successfully to deal with numerous optimization problems [58]. That is why numerous researchers have utilized them for handling a variety of maximum flow problems, as demonstrated below. Barham et al. [59] tackled different examples of the maximum flow issue with by employing a chemical reaction optimization algorithm (a branch of chemistry-based algorithms mimicking particular chemical rules). It basically simulates the molecular interactions in a chemical reaction [58].

As an instance of physics-based algorithms that has been utilized for tackling the maximum flow problem is the flow regime algorithm [60] which is inspired by the classical fluid mechanics, and flow regimes. Saha et al. [60] executed another physics-based global non-gradient approach for dealing with the optimal power flow issue. It mimics vaporizing a slight volume of the water molecules on the hard surface having various wet abilities.

Electricity networks aim at providing high quality electricity to the customer efficiently and economically. The applications expansion prompts recurrent alterations in the network variables and hence constituting a defiance to the current systems to deliver the required power.

Recently, biology based algorithms have been utilized for dealing with this optimization problem such as genetic algorithms (an instance of evolutionary approaches that imitates the natural evolution, aiming at making PCs do what nature can) in Sallam et al. [7]. The fitness procedure reflects the saturation percentage of the flow and balancing vertices. Every solution was defined by a flow matrix.

Surco et al. [61] executed a particle swarm optimization based (PSO) algorithm for enhancing the water distribution networks, which can be utilized for verifying the node pressures, fluid flow rate and velocity, and head losses in every pipe. PSO is a population-based nature-inspired metaheuristic for dealing with continuous optimization problems. It mimics a certain behavior of the particles' groups, fish schools, and bird flocks.

There are several options for water distribution systems those necessitate mandatory restrictions in addition to the large cost of water distribution systems. This makes determining the ideal scheme having the lowest cost for a water distribution system as an optimization problem. Yilmaz et al. [62] utilized particle swarm optimization, genetic algorithm, and artificial bee colony algorithm for tackling this optimization problem.

A water distribution network is consisted of a large number of nodes connected by means of a large number of interconnecting links. The main issue related to optimizing the expense in designing these networks is a direct result of the nonlinear connection between the flow and the head waste, and the discrete nature of the pipe sizes. The significance and large capital expense of the system prompts extensive consideration on looking for the ideal cost plan [4]. Uma [4] handled this with the use of the differential evolution. It can be considered a member of the best effective population-based evolutionary algorithms for dealing with continuous optimization applications. Its use of the progressive iterations which are equivalent to generations in nature, and its utilized operations imitate elements from biology [17]. In addition, Ghosh et al. [63] executed a particle swarm approach to tackle the load flow issue (more particularly, to minimize the power loss).

Raviprabakaran and Subramanian [64] proposed an ant colony optimization (ACO) technique for tackling the power flow issue having ecological pollution restrictions. Their objective was decreasing the overall fuel cost of ecological pollution and fossil thermal power generators. This is achieved by maintaining a sufficient level of bus voltage, the actual & reactive power, as well as the power flow within the transference media. This lessens the processing time with meeting each of the constraints of the power flow. ACO algorithms are nature inspired population-based metaheuristics that mimic the foraging conduct of several ant types. Close optimal solutions can only arise because of the consequence of the aggregate cooperation between the artificial ants, which is achieved through indirect communication [65].

This problem (which is extremely restricted) was also handled with the use of another biology based algorithm (flower pollination algorithm imitating the intriguing process in nature) in Shilaja and Ravi [66]. The authors tackled the problem having multiple objectives; enhancing the voltage stability, minimizing the transmission loss, the generating cost, and the power plant's emission.

Moreover, it was tackled by the use of another biology based algorithm, which is an artificial bee colony algorithm (a general-purpose biology and population-based optimization algorithm mimicking the intelligent foraging behavior of honey bees demonstrating the highly organized way in which they communicate with each other in order to search for their food) in Al-Anbarri and Naief [67] to minimize the reactive and active power mismatch in every bus.

Besides, Dogra and Gupta [68] handled it with the utilization of glowworm swarm optimization algorithm (imitates the motion of the glowworms in a group on the basis of the amount of a luciferin and the space separating the glowworms). It utilizes a group of agents to be divided into subsets moving to each other concurrently, and rendezvous at numerous optima of the function being handled. In their algorithm, the agents represent the generation values of the generators.

Another biology based and population-based metaheuristic (a bat approach motivated by the echolocation of micro bats) was developed by Kumar and Sivanagaraju in Kumar and Sivanagaraju [69] to solve it with the aim of getting an appropriate fuel cost. This is because when minimizing the generation fuel cost, the net saving of the power system will be increased.

For years, effective traffic flow and path optimization represent a great challenge since more than half of the population worldwide live in the urban environment where traffic jams are common at peak hours. Much of the recent research is about getting the best route to optimize the path to the target within reasonable time. Nevertheless, directing vehicles in the direction of the shorter routes will lead to a great traffic jam. Thus, with guaranteeing the usual level of the traffic flow on every path, the optimal path can be found since distributing the traffic on all the paths helps accomplish the maximum flow as well as the ideal denseness of the paths. This was the main objective of the work of Abdul Rehman et al. [70] that used an ant algorithm.

Ant Colony Optimization has been applied to tackling real-life flow problems like the proposed system of Tuaimah et al. [71]. They handled the Iraqi super high grid having 11 generations and 13 load buses linked together via 400-kV power transmission lines with the use of an ant colony optimization algorithm. Also, Bouktir and Slimani [72] proposed an ant colony algorithm for tackling the optimal power flow issue in an Algerian electrical network consisting of 59 buses, 10 generators, and 83 branches (transformers and lines).

5.4 Hybrid Approaches

Lately, hybridizing optimization algorithms in order to handle a particular problem has become very common and usually yields better results than utilizing these algorithms individually. This research direction is followed by various researchers when tackling the maximum flow problem as follows.

For instance, Delir et al. [73] integrated the firefly algorithm, and the charged system search technique for enhancing the water distribution networks. The firefly algorithms are inspired by the variations in the light strength. A brighter firefly is more appealing and guides its colleagues to the best answer in the solution space. The firefly attraction is because of its brightness of firefly indicating to the goodness of its place (brighter fireflies attract less bright fireflies towards their positions in addition to some random variables to help these moves).

Another instance, Khunkitti et al. [74] integrated the dragonfly approach (DA) with a particle swarm approach in order to balance the exploration, and exploitation features of the dragonfly approach & the particle swarm approach when tackling the multi-objective optimal power flow problem. This is to deal with the optimization problem of PSO and DA (imitates the grouping conduct of dragonflies; partitioning, coherence and alignment, distraction from foes, and the desirability to nourishment) which is occasionally being stopped in the local optima while discovering the ideal outcome. Their proposed algorithm obtained the merits of these two algorithms; the exploitation of PSO and the exploration of DA. At the beginning, the dragonflies explore the search space in order to discover the portion containing the global solution, which is then substituted as the global ideal location to the PSO. This enables the PSO to get the expected optimal outcome.

Another instance, Gonal and Sheshadri [75] introduced a hybrid bat–dragonfly technique to tackle the optimal power flow problem within the wind–solar system. The accuracy of the controller tuning can be enhanced with small processing time through combining the procedures of the dragonfly, and the bat approaches. This is because the bat approach is characterized by less accuracy with small computing time, and the dragonfly approach is characterized by large accuracy with more computing time. Their hybrid system enhances the performance of the entire system as well as offers good balance in energy supply through adjusting the controller parameters.

The best option for satisfying the energy demand worldwide is the use of renewable energy sources as they are non-polluting and can be obtained easily. The optimal power flow problem is highly complicated and hard to be tackled as a result of uncertainties involved in integrating the renewable energy sources having large size and power [75].

5.5. Parallel Approaches

As a kind of high-performance computing, parallel computing emerged due to the huge reduction in the wealth and cost of software and hardware since a few decades. In such a methodology, a group of computations is performed concurrently on the grounds of a large problem can be split into smaller ones to be handled concurrently [76]. For example, Jincheng and Lixin [17] introduced a parallel algorithm with Message Passing Interface for the maximum flow problem. The flow inside every part of the given graph is pushed into its boundary via the hybrid approach iteratively, and a particular algorithm discharges the flow out of this part, decreasing the message passing via several enhancements. Another example, Surakhi et al. [10] utilized a parallel genetic approach for tackling the maximum flow issue. In each iteration, they got the value of the fitness function, known also as the cost function and the evaluation function, for every augmenting path from the destination to the target concurrently in parallel.

6. COMPUTATIONAL EXPERIMENTS

Since this work studies the maximum flow problem, the experiments performed examine a variety of it with various benchmark instances having various network sizes as well as various domains with the utilization of the most common approaches for tackling it. The carried out experiments are as follows.

6.1. Data

Various benchmark networks of various capacities as well as no. of vertices were utilized in the experiments. They are from different sources such as [77]. Their details are in the following table. The 2nd column contains the instance name. The no. of vertices is represented in the 3rd column. The source node and the sink node are contained in the fourth and fifth column respectively. The arcs of these instances are provided in a supplementary file in text format (comma separated).

Table 1. The utilized instances

	Instance	No. of vertices	Source	Sink
1	A02	23	12	6
2	A97	84	1	84
3	C01	33	1	32
4	CPM2	7	1	7
5	MCCABE1	15	14	13
6	MCCABE2	20	1	19
7	MCCABE2A	27	1	27
8	PETRI1	8	1	7
9	PETRI3	11	1	10
10	RAGUSA16	24	1	24

6.2. Method

The following systems were implemented for tackling the used benchmark data utilizing R language [78]:

- The Ford Fulkerson, and
- The push-relabel algorithm.

6.3. Results

The experiments' outcomes are in table 2. In which, the instance ID is contained in the second column. The maximum flow (using the Ford Fulkerson) of each utilized instance is given in the third column. The number of arcs in each of the utilized instance is in the fourth column. The fifth column provides the maximum flow (utilizing the Goldberg-Tarjan) of each utilized instance.

Table 2. The experiments' outcomes

	Instance	Max. flow	No. of arcs	Max. flow
1	A02	3	87	2
2	A97	5	332	3
3	C01	1	146	1
4	CPM2	5	10	4
5	MCCABE1	1	18	1
6	MCCABE2	1	26	1
7	MCCABE2A	1	33	1
8	PETRI1	1	12	1
9	PETRI3	1	18	1
10	Ragusa16	3	81	3

7. DISCUSSION

Many vital applications in a variety of domains can be handled as a maximum flow problem. That is why it has been handled with the use of a variety of approaches. This was researched in this work; the Ford Fulkerson, the push relabel, and many recent approaches.

The Ford Fulkerson can be applied easily for tackling a variety of real-life applications, as it is a simple method having straightforward steps. In fact, it is considered a game changer in the graph approaches [36]. This is obvious from its recent crucial real-life applications aforementioned. Also, the push relabel has its recent applications as well. The difference between the push relabel and augmenting algorithms is rather small.

As nature inspired metaheuristics have been utilized successfully in dealing with a variety of crucial applications, they have been utilized for handling the maximum flow problem in various domains as explained previously. Further, hybrid and parallel approaches are increasingly used for tackling a variety of optimization problems and hence, they are used for handling the maximum flow problem as demonstrated above.

8. CONCLUSION AND FUTURE WORK

This article researched the maximum flow problem from different aspects, such as tackling various benchmark instances with the use of the most widely adopted approaches followed by its recent real-life applications, and recent utilized approaches for tackling it. Besides, the algorithms based on the network flow were highlighted.

For the future work, an interesting point to be considered is to utilize more algorithms. Also, it is crucial to utilize larger networks. Also, since parallel computing has been used recently for enhancing the computation speed of getting a near optimal solution for a variety of real-world optimization problems, parallel maximum flow algorithms and in particular, the coarse-grained approaches can be considered. This is because the coarse-grained approaches have been proven that they outperform other parallel approaches. In addition, most of the current parallel maximum flow approaches are fine-grained that suffer from low speedup.

REFERENCES

[1] T. L. Friesz, and D. Bernstein, Foundations of Network Optimization and Games, Springer, New York, 2016.

[2] J. A. Ruiz-Vanoye, R. Barrera-Cámara, O. Díaz-Parra, A. Fuentes-Penna, J. P. Ortega, B. B. Loranca, A. C. Saenz, and J. D. C. Santiago-Pérez, Surveying the Optimization Problems of Water Distribution Networks, Polish Journal of Environmental Studies, 27(4): 1425-1432, 2018.

[3] H. Liao, Review on distribution network optimization under uncertainty, Energies, 2019, 12, 3369, 2019.

[4] R. Uma, Optimal Design of Water Distribution Network Using Differential Evolution, International Journal of Science and Research, 5(11): 1515-1520, November 2016.

[5] E. Kohler, R. H. Mohring, and M. Skutella, Traffic Networks and Flows over Time, In: J. Lerner, D. Wagner, and K.A. Zweig (Eds.): Algorithmics, LNCS 5515, 166–196, Springer-Verlag Berlin Heidelberg, 2009.

[6] Chapter 10: Iterative Programming- Maximum Flow Problem. Available at: https://www.yumpu.com/en/document/read/49778331/maximum-flow-problem. Last visited on 1-1-2020.

[7] G. Sallam, G. R. Gupta, B. Li, and B. Ji, Shortest Path and Maximum Flow Problems under Service Function Chaining Constraints. In IEEE INFOCOM 2018-IEEE Conference on Computer Communications, p. 2132-2140. IEEE, 2018.

[8] S. Soner, and C. Ozturan, Experiences with Parallel Multi-threaded Network Maximum Flow Algorithm, Partnership for Advanced Computing in Europe, 2013(1): 1-10, 2013.

[9] M. Kyi, and L. Naing, Application of Ford-Fulkerson Algorithm to Maximum Flow in Water Distribution Pipeline Network, International Journal of Scientific and Research Publications, 8(12): 306-310, 2018.

[10] O. Surakhi, M. Qatawneh, and H. Alofeishat, A Parallel Genetic Algorithm for Maximum Flow Problem, International Journal of Advanced Computer Science and Applications, 8(6): 159-164, 2017.

[11] R. Masadeh, A. Alzaqebah, and A. Sharieh, Whale optimization algorithm for solving the maximum flow problem, Journal of Theoretical and Applied Information Technology, 96(8): 2208-220, 2018.

[12] S. Han, Z. Peng, and S. Wang, The maximum flow problem of uncertain network. Information Sciences, 265: 167-175, 2014.

[13] R. K. Ahuja, T. L. Magnanti, and J. B. Orlin, Network Flows: Theory, Algorithms, and Applications. Prentice Hall, Englewood Cliffs, New Jersey, USA, 1993.

[14] C. S. Negruseri, M. B. Pasoi, B. Stanley, C. Stein, and C. G. Strat, Solving Maximum Flow Problems on Real World Bipartite Graphs. Journal of Experimental Algorithmics, 16: 3-5, 2011.

[15] W. Zhang, Finding maximum flow in the network: A Matlab program and application, Computational Ecology and Software, 8(2): 57-61, 2018.

[16] L. Monis, B. Kunjumon, and N. Guruprasad, Implementation of Maximum Flow Algorithm in an Undirected Network, In V. Sridhar et al. (eds.), Emerging Research in Electronics, Computer Science, and Technology, Lecture Notes in Electrical Engineering, 545: 195-202, Springer, Singapore, 2019.

[17] J. Jiang, and L. Wu, A MPI Parallel Algorithm for the Maximum Flow Problem. Available at: http://www.geocomputation.org/2013/papers/65.pdf. Last visited on 1-1-2020.

[18] N. Abd-Alsabour, On Tackling Real-Life Optimization Problems, International journal on advanced science, engineering, and information technology, 9l(2): 640-647, 2019.

[19] E. S. Nicoară, Applying Genetic Algorithms to Optimization Problems in Economics, Economic Insights – Trends and Challenges, IV(3/2015): 125–132, 2015.

[20] N. Altiparmak, and A. Ş. Tosun, Multithreaded Maximum Flow Based Optimal Replica Selection Algorithm for Heterogeneous Storage Architectures, IEEE Transactions on Computers, 65(5): 1543-1557, May 2016.

[21] P. Mulajkar, I. Pai, M. Sakhare, T. Zende, and V. M. Barkade, Multithreaded maximum flow based frequent item selection algorithm for heterogeneous storage architecture, International Journal of Advanced Computational Engineering and Networking, 5(7): 66-70, Jul. 2017.

[22] K. Peng, R. Lin, B. Huang, H. Zou, and F. Yang, Assessment of Performance in Data Center Network Based on Maximum Flow. In Y. M. Huang et al. (eds.), Advanced Technologies, Embedded and Multimedia for Human-centric Computing (p. 429-439). Springer, Dordrecht, 2014.

[23] M. Noormohammadpour, and C.S. Raghavendra, Datacenter traffic control: Understanding techniques and tradeoffs, IEEE Communications Surveys and Tutorials, 20(2): 1492-1525, 2018.

[24] S. Kumar, K. Yoo, M. D. Rosenberg, D. Scheinost, R. T. Constable, S. Zhang, C. R. Li, and M. M. Chun, An information network flow approach for measuring functional connectivity and predicting behavior, Brain and Behavior, 2019; 9:e01346, 2019.

[25] Y. Ren, L. Liu, S. Yin, J. Han, and S. Wei, Efficient fault-tolerant topology reconfiguration using a maximum flow algorithm, ACM Transactions on Reconfigurable Technology and Systems (TRETS), 8(3): 1-26, 2015.

[26] A. Lamghari, and R. Dimitrakopoulos, Network-flow based algorithms for scheduling production in multi-processor open-pit mines accounting for metal uncertainty, European Journal of Operational Research, 250(1): 273–290, 2016.

[27] T. Karpagam, V.J. Sudhakar, N. Arunsankar, and M. Dhanasekaran, Flow Based Algorithm, American Journal of Applied Sciences, 9(2): 238-239, 2012.

[28] L. Orecchia, and Z. A. Zhu, Flow-Based Algorithms for Local Graph Clustering, In Proceedings of the twenty-fifth annual ACM-SIAM symposium on discrete algorithms, (SODA 2014), p. 1267-1286, 2014.

[29] R. Andersen, and K. J. Lang, An algorithm for improving graph partitions, In Proceedings of the nineteenth annual ACM-SIAM symposium on discrete algorithms, (SODA 2008), p. 651-660. 2008.

[30] M. T. Gastner, V. Seguy, and P. More, Fast flow-based algorithm for creating density-equalizing map projections, In Proceedings of the National Academy of Sciences (PNAS), 115(10): E2156–E2164, 2018.

[31] J. Svancara, and P. Surynek, New Flow-based Heuristic for Search Algorithms Solving Multi-agent Path Finding, In Proceedings of the 9th International Conference on Agents and Artificial Intelligence (ICAART 2017), p. 451-458, 2017.

[32] L. Hellemons, L. Hendriks, R. Hofstede, A. Sperotto, R. Sadre, and A. Pras, SSHCure: A Flow-Based SSH Intrusion Detection System, IFIP International Conference on Autonomous Infrastructure, Management and Security. In R. Sadre et al. (Eds.): AIMS 2012, LNCS 7279, p. 86–97, Springer, Berlin, Heidelberg, 2012.

[33] X. Shan, Z. Xu, Cheng, and W. Wang, A Max-Flow Based Algorithm for Connected Target Coverage with Probabilistic Sensors, Sensors, 17(6), article number 1208, 2017.

[34] L. R. Ford, and D. R. Fulkerson, Maximal flow through a network, Canadian Journal of Mathematics, 8: 399-404, 1956.

[35] L.R. Ford, and D.R. Fulkerson, Flows in Networks, Princeton University Press, Princeton, New Jersey, 1962.

[36] S. C. Dimri, and M. Ram, Flow Maximization Problem as Linear Programming Problem with Capacity Constraints, Applications and Applied Mathematics, 13(1): 508 – 515, 2018.

[37] S. Biswas, S. S. Laga, and B. Paul, A Review on Ford Fulkerson Graph Algorithm for Maximum Flow, International Journal of Scientific and Engineering Research, 8(3): 109-112, 2017.

[38] A. A. Kinsley, and B. U. Maheswari, Design of algorithms to Maximum flow Problems in tree flow Networks, International Journal of Science, Engineering and Technology Research, 5(4): 1015-1018, April 2016.

[39] A. Iqbal, Md. S. Hossain, and H. A. Ebna, Airline scheduling with max flow algorithm, In Proceedings of the 5th International Conference on Natural Sciences and Technology (ICNST'18), p. 129-131, Chittagong, Bangladesh, March 30 -31, 2018.

[40] H. Gebreanenya, Maximum Flow Problem in Ethiopian Airlines, Journal of Progressive Research in Mathematics, 9(2): 1371-1380, 2016.

[41] N. Abdullah, and T. K. Hua, The Application of the Shortest Path and Maximum Flow with Bottleneck in Traffic Flow of Kota Kinabalu, Journal of Computer Science and Computational Mathematics, 7(2): 37-43, June 2017.

[42] N. Abdullah, and T. K. Hua, Using Ford-Fulkerson Algorithm and Max Flow-Min Cut Theorem to Minimize Traffic Congestion in Kota Kinabalu, Sabah, Journal of Information System and Technology Management, 2(4): 18-34, 2017.

[43] E. J. Moore, W. Kichainukon, and U. Phalavonk, Maximum flow in road networks with speed-dependent capacities–application to Bangkok traffic, Songklanakarin Journal of Science and Technology, 35(4): 489-499, 2013.

[44] H. Wei, and Z. Su, The Maximum Flow Problem with Negative Capacities or Flows, 2016 Joint International Conference on Service Science, Management and Engineering (SSME 2016) and International Conference on Information Science and Technology (IST 2016), 2016.

[45] Z. Jiang, X. Hu, and S. Gao, A Parallel Ford-Fulkerson Algorithm For Maximum Flow Problem, In Proceedings of the International Conference on Parallel and Distributed Processing Techniques and

Applications (PDPTA), p. 70-73, 2013.

[46] A. V. Goldberg, and R. E. Tarjan, A New Approach to the Maximum-Flow Problem, Journal of the Association for Computing Machinery (ACM), 35(4): 921-940, October 1988.

[47] M. Yu, S. Shen, and Z. Hu, Dynamic Parallel and Distributed Graph Cuts, IEEE Transactions on Image Processing, 25(12): 5511-5525, 2016.

[48] G. Gallo, M. D. Grigoriadis, and R. E. Tarjan, A fast parametric maximum Low algorithm and applications, SIAM Journal on Computing, 18(1): 30–55, 1989.

[49] S. Iwata, K. Murota, and M. Shigeno, A fast parametric submodular intersection algorithm for strong map sequences, Mathematics of operations research, 22(4): 803–813, 1997.

[50] S. Sato, On Implementing the Push-Relabel Algorithm on Top of Pregel, New Generation Computing, 36(4): 419–449, 2018.

[51] Z. He, and B. Hong, Dynamically Tuned Push-Relabel Algorithm for the Maximum Flow Problem on CPU-GPU-Hybrid Platforms, In 2010 IEEE International Symposium on Parallel & Distributed Processing (IPDPS) (p. 1-10), 2010.

[52] M. Deveci, K. Kaya, B Uçar, and Ü. V. Çatalyürek, A Push-Relabel-Based Maximum Cardinality Bipartite Matching Algorithm on GPUs, The 42nd International Conference on Parallel Processing (p. 21-29), IEEE, 2013.

[53] N. Veldt, C. Klymko, and D. F. Gleich, Flow-Based Local Graph Clustering with Better Seed Set Inclusion, In Proceedings of the 2019 SIAM International Conference on Data Mining (p. 378-386), Society for Industrial and Applied Mathematics, 2019.

[54] K. Chaudhuri, S. Rao, S. Riesenfeld, and K. Talwar, Push-Relabel Algorithm for Approximating Degree Bounded MSTs, International Colloquium on Automata, Languages, and Programming, In: M. Bugliesi et al. (Eds.), ICALP 2006, Part I, LNCS 4051, Springer, Berlin, Heidelberg, (p. 191–201), 2006.

[55] J. Langguth, A. Azad, M. Halappanavar, and F. Manne, On parallel push–relabel based algorithms for bipartite maximum matching, Parallel Computing, 40(7): 289–308, July 2014.

[56] L. Fleischer, and S. Iwata, A push-relabel framework for submodular function minimization and applications to parametric optimization, Discrete Applied Mathematics, 131(2): 311–322, September 2003.

[57] N. Siddique, and H. Adeli, Nature inspired computing: an overview and some future directions, Cognitive computation, 7(6): 706-714, December 2015.

[58] N Abd-Alsabour, Nature as a Source for Inspiring New Optimization Algorithms, In Proceedings of the 9th International Conference on Signal Processing Systems 2017 Nov 27, p. 51-56, ACM, 2017.

[59] R. Barham, A. Sharieh, and A. Sliet, Chemical Reaction Optimization for Max Flow Problem, International Journal of Advanced Computer Science and Applications, 7(8): 189-196, 2016.

[60] A. Saha, P. Das, and A. K. Chakraborty, Water evaporation algorithm: A new metaheuristic algorithm towards the solution of optimal power flow, Engineering Science and Technology, an International Journal, 20(6): 1540–1552, 2017.

[61] D. F. Surco, T. P. B. Vecchi, and M. A. S. S. Ravagnani, Optimization of water distribution networks using a modified particle swarm optimization algorithm, Water Science and Technology: Water Supply, 18(2): 660-678, 2018.

[62] V. Yilmaz, M. Büyükyildiz, and Ö. K. Baykan, Cost Optimization of Hanoi Water Distribution Network with Meta-Heuristic Optimization Algorithms, Turkish journal of water science and management, 3(2): 2-15, 2019.

[63] P. Ghosh, A. G. Mukherjee, S. K. Singh, and H. P. Dubey, Application of Particle Swarm Optimization For Load-Flow Computation, International Journal of Advanced Computational Engineering and Networking, 4(7): 70-77, July 2016.

[64] V. Raviprabakaran, and R. C. Subramanian, Enhanced ant colony optimization to solve the optimal power flow with ecological emission, International Journal of System Assurance Engineering and Management, 9(1): 58-65, February 2018.

[65] N. Abd-Alsabour, A Review on Evolutionary Feature Selection, the 8th European Modeling Symposium, 2014 UKSim-AMSS 8th European Modeling Symposium, Italy, (p. 20-26), 2014.

[66] C. Shilaja, and K. Ravi, Multi-Objective Optimal Power Flow Problem Using Enhanced Flower Pollination Algorithm, Gazi University Journal of Science, 30(1): 79-91, 2017.

[67] K. Al-Anbarri, and H. M. Naief, Application of Artificial Bee Colony Algorithm in Power Flow Studies, UHD Journal of Science and Technology, 1(1): 11-16, 2017.

[68] R. Dogra, and N. Gupta, Glowworm Swarm Optimization Technique for Optimal Power Flow, Advance in Electronic and Electric Engineering, 4(2): 155-160, 2014.

[69] Y. N. V. Kumar, and S. Sivanagaraju, Optimal Power Flow With Interline Power Flow Controller Using Modified Bat Algorithm, International Journal of Electrical, Electronics and Data Communication, 3(1): 6-12, January 2015.

[70] A. Rehman, M. M. Rathore, A. Paul, F. Saeed, and R. W. Ahmad, Vehicular traffic optimisation and even distribution using ant colony in smart city environment, IET Intelligent Transport Systems, 12(7):594-601, 2018.

[71] F. M. Tuaimah, Y. N. Abd, and F. A. Hameed, Ant

Colony Optimization based Optimal Power Flow Analysis for the Iraqi Super High Voltage Grid, International Journal of Computer Applications, 67(11): 13-18, April 2013.

[72] T. Bouktir, and L. Slimani, Optimal Power Flow of the Algerian Electrical Network using an Ant Colony Optimization Method, Leonardo Journal of Sciences, July-December 2005(7): 43-57, 2005.

[73] S. Delir, A. Foroughi-Asl, and S. Talatahari, A hybrid charged system search – firefly algorithm for optimization of water distribution networks, International journal of optimization in civil engineering, 9(2): 273-290, 2019.

[74] S. Khunkitti, A. Siritaratiwat, S. Premrudeepreechacharn, R. Chatthaworn, and N. R. Watson, A Hybrid DA-PSO Optimization Algorithm for Multiobjective Optimal Power Flow Problems, Energies 2018, 11, 2270, 2018.

[75] V. Gonal, and G. S. Sheshadri, A hybrid bat–dragonfly algorithm for optimizing power flow control in a grid-connected wind–solar system, Wind Engineering, 2019:0309524X19882429, 1–14, 2019. Available at: https://doi.org/10.1177/0309524X19882429. Last visited on 1-1-2020.

[76] S. Das, S. S. Mullick, and P. N. Suganthan, Recent advances in differential evolution– An updated survey, Swarm and Evolutionary Computation, 27: 1–30, 2016.

[77] http://vlado.fmf.uni-lj.si/pub/networks/pajek/data/. Last visited on 1-1-2020.

[78] R: A Language and Environment for Statistical Computing [http://www.R-project.org]. R Foundation for Statistical Computing, Vienna, Austria.