International Journal of Engineering Research and Technology. ISSN 0974-3154, Volume 12, Number 2 (2019), pp. 180-195

TERMHIGEN – A Hybrid Metaheuristic Technique for Solving Large-Scale Vehicle Routing Problem with Time Windows

[1,2]Temitayo Matthew Fagbola, [1]Surendra Colin Thakur

[1]Department of Information Technology, Durban University of Technology, Durban 4000, South Africa.
[2]Computing & Intelligent Systems Research Group, Department of Computer Science, Federal University, Oye-Ekiti, Nigeria.

ORCID: 0000-0001-6631-1002 (Temitayo Fagbola)

Abstract

Vehicle Routing Problem with Time Windows (VRPTW) involves traversing a coordinated set of vehicular paths such that a set of customers is visited once within a given time-stamped boundary. VRPTW poses a great challenge to logistics distribution and supply chain management systems, due to its characterized stochastic and NP-hard combinatorial properties, which requires that its corresponding optimal path planning and vehicle scheduling solutions be both highly efficient and cost effective even as customers' demands change dynamically. In this paper, a new hybrid metaheuristic scheme, tagged TERMHIGEN, based on the characteristics of the Termite-Hill algorithm and a modified Genetic Algorithm, with its associated adaptive self-learning and tuning schemes, based on is developed and applied to solving a prototype VRPTW specifically with the objective of minimizing overall logistic distribution cost. TERMHIGEN was tested using Solomon's 56 VRPTW instances containing 100 customers. The performance evaluation results of the algorithms reveal that TERMHIGEN produced more optimal and efficient outputs for some problem instances than those produced by some baseline metaheuristic techniques in terms of computational time efficiency and distance travelled.

Keywords: Genetic Algorithm, Solomon Benchmark Problems, TERMHIGEN, Termite-Hill, Time-Windows, Vehicle Routing

I. INTRODUCTION

Polynomial-time (NP)-Hard combinatorial optimization problem with the goal of finding optimal set of paths for delivery trucks to reach the customers requesting for goods or services [1],[2]. This problem has closely drawn strong attentions from researchers in logistics distribution and management, transportation science, intelligent traffic congestion management and task sequencing among others [3]-[9]. Very strong contexts of application of vehicle routing to real-life logistic and distribution-related issues include aircraft and bus scheduling, team rostering and scheduling, postal delivery, grocery delivery, product distribution, pickup logistics, inventory management and game playing [4],[10],[11]. Generally, VRP is often characterized by largely varying objectives and constraints requiring that solutions be tailored and directed to specific pre-defined problems. It is important to note that there is no general methodological approach to solving it [4],[12],[13]. This in turn limits the applicability of generalized solutions to VRP in practice. Hence, the main requirement of VRP is to find the least minimum cost required to service a set of customers with known demands in a set of closed path, starting and terminating at one of the end nodes (depots), while vehicle and depot capacity constraints are satisfied given that one or more depots and a fleet of vehicles exist [14],[15]. Genetic algorithm (GA), Ant Colony Optimization (ACO), fuzzy system, Nearest Neighbor Search (NNS), Simulated Annealing (SA) and Tabu search among other salient metaheuristic techniques have been successfully applied to many theoretical optimization problems especially VRP [16]-[19]. The behavioral patterns of these metaheuristic techniques are strongly inspired by nature which makes them suitable for solving a variety of real life optimization problems.

GA has a good ability to conduct global search and has been successfully applied to other real-life problem domains ranging from fraud detection, traffic congestion management, pattern recognition to classification tasks [20]-[22]. It is also often used to improve the performance of some other nature-inspired algorithms and local search techniques [23]. Although, GA suffers from a number of limitations including its inability to produce global optimal solutions and its premature convergence to local minima due to its total reliance on crossover such that the population becomes homogeneous [24],[25]; consequently, modified and hybrid variants are being developed in more recent works to address these drawbacks. Examples of such variants include GA based on random immigrants and triggered hypermutation [12],[26],[27]. Suffice it to say that most of these solutions are computationally highly expensive, impractical and with limited flexibility and scalability [28]. In contrast, the complex constraints inherent in VRPs require that appreciable and near-optimal solutions be adaptive, holistically robust, computationally efficient, flexible and scalable while possessing some associated self-healing and organization features [29]. Interestingly, most of these requirements are only offered by metaheuristic approximations and recently, hybrid versions of some of these nature-inspired techniques have been developed to solve VRP. A number of VRP solutions based on metaheuristic algorithms is presented by Bhuvaneswari, Sumathy and Rajagopalan [30].

Pratiwi, Pratama, Sa'diyah and Suprajitno [31], Leonid, Sergey and Nadezhda [32] and Doerner, Hartl and Reimann [33] among others justifiably argued the need to combine distinct solutions of different evolutionary algorithms to generate optimal solutions for VRP. In this paper, a hybrid metaheuristic algorithm tagged "*TERMHIGEN*" is developed. TERMHIGEN combines the sophisticated cooperative and self-organization abilities of Termite Hill algorithm, which makes it capable of handling complex tasks in a well-coordinated, efficient and adaptive manner, with the rich global convergence and computational efficiency offerings of a modified Genetic algorithm based on ranking selection to solving a prototype VRP with Time Windows.

The major contributions of this paper include the following:

(1) To the best of our knowledge, this is the first paper to adapt the principle and social behaviour of Termites to solving a prototype VRPTW. Most other applications have been directed homogeneously towards mobile Ad-Hoc and wireless sensor networks;

(2) An extensive review of works on VRP variants and existing solutions as well as Termite-inspired metaheuristic algorithms and their application environments was conducted;

(3) A hybrid metaheuristic scheme, tagged TERMHIGEN, based on the Termite-hill algorithm and a modified Genetic algorithm with ranking selection, for solving VRPTW was proposed and tested using Solomon's 53 benchmark problems containing 100 customers.

The rest of this paper is presented as follows: in section 2, vehicular routing problems and variants as well as several existing approaches for solving them are discussed. In section 3, several related works on VRPTW are reviewed. The Termite-Hill algorithm, the Genetic Algorithm, the proposed TERMHIGEN algorithm, the Solomon's Benchmark problems and performance evaluation metrics for VRPTW solutions are presented under the materials and method in section 4. The experimental results obtained for the VRPTW algorithms considered were comparatively evaluated with some best-known solutions and presented in section 5 while conclusion and future works followed in section 6.

II. VEHICLE ROUTING PROBLEM

VRP is a generalization of the travelling salesman problem. John and Patrick [34] described VRP as a weighted graph $G = (V, A, d)$ where $V = (V_1, V_2, V_3, \dots, V_m)$ depicts the set of vertices for m customers, $A = \{(V_1, V_2, V_3, \dots, V_k): i \neq j\}$ represents the set of arcs and d_{ij} represents the Euclidean distances associated with the arcs and V_0 depicts the take-off depot for all vehicles to reach m customers as depicted in Figure 1. An in-depth fundamental discussion on VRPs is provided in the work of Suresh and Ramasamy [5]. Basically, the goal of VRP is most often associated with reduction in the cost (number of vehicles, time, distance, expenses and so on) incurred on goods to be delivered to a number of customers with prior non-negative effective demand(s) via vehicular routes starting and terminating at one or more central depots [36].

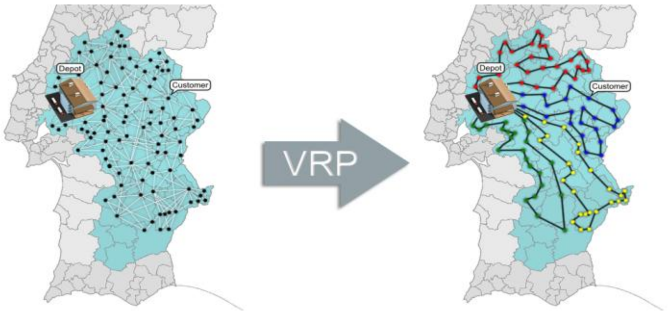

Fig. 1. Description of a Multi-Node Vehicle Routing Problem [35]

For example, this could involve finding the least cost route to be travelled by a fleet of capacitated delivery vehicles to reach destination node of each geographically decentralized customer. Variants of VRP include the capacitated (multi-trip, cumulative) [1], [37], green VRP [38], heterogeneous fleet [39], split delivery [40], multiple depot [41],[42], periodic [38], multi-attribute [4] and stochastic [43] as well as those associated with pick-up and delivery (mixed and simultaneous) [3],[42], backhauls [44]-[46], cross-dock selection [47] and time windows (open and dynamic) [31],[48].

Among all, VRP with Time Windows (VRPTW) remains the most widely studied mainly because it is the real engine of distribution management and its NP-hard nature [23],[35]. VRPTW involves finding and traversing a best optimal vehicular paths such that a large set of sparsely-distributed customers is visited once by a number of limited capacity vehicles within a given pre-determined time window. However, the time window can be said to be a time-stamped boundary having lower early arrival time and upper late arrival time delimiters, $[e_t, l_t]$, within which a customer is to be visited [8],[49].

From a graph-theoretic point of view, VRPTW can be defined as follows [48]: given a digraph $G = (V, A)$, where $V = (0,1, \dots ,m)$ *is the vertex set and* $A = \{a_1(i_1, j_1), a_2(i_2, j_2), \dots, a_n(i_n, j_n)\}$ *is the arc set*: $\forall\, a_i(i_i, j_i) \in A, (i \leq n), a(i,j) = \{d_{ij}, t_{ij} > 0\}$ where d_{ij} is the travel cost and $t_{ij} > 0$ is the total travel and service times at vertex i. However, for every vertex $i \in V$, $(i \leq m), i = \{\,|q_i|, [e_i, l_i]\,\}$ where q_i is a non-negative demand and $[e_i, l_i]$ represents a time window to visit vertex i, with e_i and l_i being the earliest and latest times, respectively. For this definition, it is assumed that the triangle inequality theory is satisfied by matrices d_{ij} and t_{ij}, the time windows is a hard constraint and as such cannot be violated and that the vehicle capacity is a soft constraint. Hence, given a group of x similar vehicles of capacity C located at the depot with a request to meet the demand, D_{cust_i}, of each customer, a vehicle route $R = (0. i_1, \dots, i_r, 0)$ with $r \geq 1$, is a closed and complete path in digraph G, traversing the depot, visiting vertices $V(R) = \{0. i_1, \dots, i_r\}$, $V(R) \subseteq V$, such that:

i. $D_{total} < C$, where D_{total} is the total demand of visited customers and C is the vehicle capacity.

ii. a vehicle exits the depot 0 at the earliest time e_0, reaches each customer in $V(R)$ within its supposed time window, and arrives at the depot 0 before the latest time I_0.

iii. if the vehicle leaves the depot 0 to reach vertex at $i \in V(R)$ before its earliest time e_i, the service will be delayed to the time e_i.

iv. $R_{cost} = \sum_{i=0}^{n} A_i(R_i)$ where R_{cost} is the total travel cost of the arc set, $A(R)$, covered by route R.

v. for any prototype VRPTW, each customer is visited only once.

However, VRPTW can be formulated as a linear optimization problem such that [50],[51]:

$$\min Z = \sum_{i=1}^{n} \sum_{j=1}^{n} \sum_{k=1}^{m} C_{ij} X_{ijk} + C_0 \sum_{i=1}^{n} \max(t_{e_i} - t_i, 0) + C_p \sum_{i=1}^{n} \max(t_i - t_{l_i}, 0) \quad (1)$$

subject to

$$\sum_{i=1}^{n} \omega_i y_{ik} \leq W \quad k \in [1, m] \quad (2)$$

(vehicle capacity / load) constraint

$$\sum_{k=1}^{n} y_{ik} = 1 \quad i \in [1, n] \quad (3)$$

(delivery service identified by clients)

$$\sum_{i=1}^{n} X_{ijk} = y_{ik} \quad j \in [1, \ldots, n], \forall k \quad i \in [1, \ldots, n], \forall k \quad (4)$$

(same route constraint)

$$t_{ei} \leq t_i \leq t_{li} \text{ (time windows constraint)} \quad (5)$$

$$X_{ijk}, y_{ik} = \begin{cases} 1 & \textit{vehicle k from customer i to customer j} \\ 0 & \textit{otherwise} \end{cases} \quad (6)$$

$$t_i = \sum_{i=0}^{n} \sum_{k=0}^{m} X_{ijk} \left(t_i + t_{ij} + t_j \right) \quad j = 1, \ldots, n \quad (7)$$

(delivery time of goods at customer depot)

where *min Z* is the objective function (total distance travelled), C_{ij} *is the travel time (cost) from node i to node j*, t_{e_i} *is the earliest time to node i, and* t_{l_i} *isthe latest time to reach node i.*

For an extensive review of VRP variants, kindly refer to the work of Gautham *et al.* [36] and Toth and Vigo [52]. In the same vein, application areas of VRP and basic components (especially the road network, customers, vehicles, drivers, routes and the global transportation cost) were discussed by Tuomas [2]. Beresneva and Avdoshin [53] also provided a detailed analysis and mathematical formulations of VRP variants and appropriate approaches for their solutions. The exact algorithms (for example, the branch and bound approaches, guided local search, set segmentation techniques, vehicle flow formulations, commodity flow formulations, set partitioning, dynamic programming and integer programming algorithms), traditional heuristic algorithms (for example, savings, sweep, two-phase), heuristic algorithms (for example, Tabu search, GA, iterated local search, SA) and the metaheuristic algorithms (for example, variable neighborhood search, ant colony, neural network and artificial bee colony) as well as hybrid algorithms combining some of these approaches have been successfully applied to solving VRP and its variants. Avirup *et al.* [23] contains a comprehensive survey of some metaheuristic solutions to VRPTW. However, the serial and parallel algorithms for solving VRP have been discussed by Christopher [54]. Sequel to the extensive review and experimental evaluations of some exact algorithms for solving several classes of VRP, these methods are computationally inefficient especially for large VRP space [48],[52],[55]. Similarly, due to large solution space requirement for VRP, it would be most impractical to attempt especially under time limitations [16].

Inexact methods, often based on heuristic approaches, are claimed to be more practically applicable to handling such larger VRP situations in a more computationally efficient manner. More often than not, heuristics or approximate algorithms usually produce near optimal solutions to combinatorial problems as exact solutions are not guaranteed [30], [32], [56]. Three (3) variants of heuristics identified in literatures are constructive, classical and the improvement heuristics [57]-[59]. The constructive heuristics build stepwise solutions while maintaining the immediate cost at an acceptable minimum. An example is the Nearest Neighbour. On the other hand, the classical heuristics like the *intraroute* and the *interroute* methods are computationally time-efficient but tends to produce largely inaccurate results [13]. However, an exhaustive survey and analysis of several VRP solutions have been conducted by Sheng-Hua, Ji-Ping, Fu-Hao, Liang and Li-Jian [28]. Also, a comprehensive review of exact and approximate algorithms for solving VRP can be found in the work of Jean-Francois *et al.* [13]. More recent approaches to solving VRP and its variants are those adopting metaheuristic-related approaches, this is because they are relatively simple and applicable to diverse range of optimization problems [20]. For large and dynamic routing problems, only metaheuristic algorithms is best suited to offer speed, scalability, autonomy and adaptation features majorly required to achieve global convergence and optimality of solutions [5],[60]. Kindly refer to the work of Anna, Bertha and Gilang [61] for a comparative study of some metaheuristic algorithms for solving delivery problems. Moreso, hybrid versions of these metaheuristic algorithms are becoming more evident today [41].

III. RELATED WORKS

Blanton and Wainwright [37] applied GA to solving multiple capacitated VRPTW. Berger *et al.* [62] developed a route-directed GA to solve VRPTW ensuring a partial constraint relaxation maintenance between any two unique populations. Minimizations of total distance traversed and violations of the constraints of the time windows were the respective task of each population which was pursued independently. This approach proved efficient when the utilization of a reduced number of vehicles is the major objective. Aslaug [63] developed a Uniform Crossover GA (UC-GA) with Steepest

Improvement Crossover to optimize the number of vehicles serving a number of customers from a central destination in a Capacitated VRP (CVRP). The local search technique used was based on simple random crossover and mutation operators. The results indicate that Steepest Improvement algorithm is best for small problems while UC-GA with Steepest Improvement Crossover produced is best fit for larger problems. John and Patrick [34] applied ACO algorithm to addressing the route construction, the trail updating and the route improvement strategies of a prototype VRP. Experimental evaluation involved single and multiple ant colonies and were applied to three different optimization problems. Yingjie and Michael [46] solved a VRP with Backhauls and Time Windows (VRPBTW) by using a guided local search algorithm. The objectives were to reduce both overall total distance of all evolving routes and the number of routes. The solution evolved in two phases. In the first phase, initial infeasible solution was generated using an adapted sweep algorithm while a path planning technique was used to enhance feasibility of routes in the second phase. Active Guided Evolution Strategies [64] and Scatter Search [65] have also been developed to solving VRPTW. Ho *et al.* [22] introduced a hybrid GA concept to solve the multi-depot VRP. Nie and Yue [27] merged self-optimization concept of Particle Swarm Optimization (PSO) with the evolving individuals' concept of GA. Aziz [7] developed a hybrid metaheuristic technique to solve VRP that combines ACO and local search algorithms. Considering a graph problem, the objective of the technique was to group nodes in the neighborhood to same branch of the minimum spanning tree. ACO was introduced within a cluster of client nodes to improve the optimality of solutions and to update the associated weights of the graph arcs at each iteration after a route is returned by the local search algorithm.

Variable Neighborhood Search with GA and Route-Nearest Neighborhood with Tabu search have also been applied to solving large scale VRP [43], [66], [67]. Thangiah, Nygard and Juell [68] developed a GA-based heuristic technique called GIDEON to solve VRPTW via a genetic sectoring approach. In this approach, a corresponding service time is allotted to a local post-optimization improvement strategy for the core purpose of minimizing the total travel time of the vehicle relative to its capacity, arrival time and instantaneous travel time among other constraints to reach each customer to be served. Adaptively, via a GA, genetic sectoring method search for sector rays through which the customers are partitioned into clusters before being served. Rita, Cláudio, Valério, François and Saïd [69] applied a pseudo-polynomial network flow model to develop an iterative exact algorithm for solving VRPTW with multiple routes. Lingling and Ruhan [15] combined Nearest Neighbor search (NNS) with Tabu Search in a two-stage process to realize an efficient hybrid metaheuristic algorithm for Large-Scale VRP. The VRP was divided into two phases via a decomposition approach. In the first phase, the initial route was constructed using the Nearest Neighbor algorithm while both intra and inter (cross-exchange) routes were optimized in the second phase by Tabu search. Similarly, decomposition strategies such as the divide and conquer approach and the POPMUSIC framework have been successfully applied to managing large size restrictions in VRP [70].

Zhang and Wang [71] developed a Hybrid Nearest Neighbor Heuristic (NNH) and ACO Algorithm tagged "HAA" for solving VRP. In the HAA solution space, initial solution was generated using NNH before a more optimal solution was obtained using ACO via a 4-stage procedure namely construct_solutions, mutation operation, 2-opt heuristics and update_pheromone. It was reported that HAA can find feasible solutions and avoid premature convergence in the search space. ACO, ACO+2-opt and the HAA algorithm were evaluated over nine (9) benchmark problems in terms of speed of convergence and HAA performs best in this regard. Min and Ping [14] developed an improved ACO for optimal vehicle routing path planning. Factor and visibility functions were added via a transition probability function as well as path weight and save matrices. The accuracy of the route search was maintained using a penalty function assigned to update new pheromone. 3-opt method was adopted to reduce the complexity of the search route to help realize optimized path length. More specifically, the improved ACO algorithm was developed to overcome problem of slow ad local minimum convergence and stagnation of ACO in VRP. Wenxue, Li and Guomin [72] combined improved GA and improved ACO to solve VRPTW with performance requirement specifications including shortest distance, shortest path length and minimum number of vehicles.

Kourank, Hejazi and Mirmohammadi [40] developed a hybrid metaheuristic solution for VRP with delivery time cost based on electromagnetism and SA algorithms. In their approach, electromagnetism algorithm was used to generate diverse set of solution populations while SA was used to achieve global convergence. A branch-and-price approach was proposed by Andrea *et al.* [42] to address a prototype VRP with relaxed time windows. Masrom, Abidin, Omar, Nasir and AbdRahman [73] combined a PSO operator with mutation and crossover operators of GA with prior dynamic parameterization to solve VRPTW. Sheng-Hua *et al.* [28] addressed VRPTW via a particle real number encrypting method to decide the path to reduction in computational complexities while allowing for a leverage between local and global exploration strengths. The authors addressed premature convergence to local minimum by using a linear decreasing function integrated with the crossover operator of GA. Peiqing, Jie, Dunyong, Yongsheng and Chenhao [74] developed an improved GA based on a penalty strategy to solve VRPTW. Cagric, Tolga, Ola and Gilbert [60] developed a hybrid GA based on population-based search and adaptive large neighborhood search for heterogeneous fleet VRPTW. Suresh and Ramasamy [5] addressed a VRPTW in a 3PL network. The authors employed GA to solve a prototype e-Commerce supplier site pickups having distribution and logistics challenges. The algorithm was modified using a random insertion-based crossover technique such that the suppliers' site is visited by buyer's vehicle to pick up some ordered items within specified suppliers' time windows and approximate travel time. This approach was also adopted by Oyediran, Fagbola, Olabiyisi and Omidiora [75] to developing an ant-optimized mobile agent migration pattern in complex distributed networks. It is generally defined such that for any given distributed network, the cost approximate for a given route is:

$$R_i = \{v_0, v_1, \ldots, v_{k+1}\} \tag{8}$$

where $R_1, \ldots, R_m$ is a subset of V (vertex set) depicting the routes of the vehicles to reach all the customers; $v_j \epsilon$ V and $v_0 = v_{k+1} = 0$ (where 0 depicts the depot), is given by:

$$Cost(R_1) = \sum_{j=0}^{k} C_j, j + 1 + \sum_{j=0}^{k} \delta_j \qquad (9)$$

while the cost of the generated problem solution given by S is:

$$F_{VRP}(S) = \sum_{i=1}^{m} Cost(R_1) \qquad (10)$$

Martin [76] approached VRPTW using a modified GA and state space search. In the new GA, the mutation operator was replaced by a neighbour generation function. Cao, Yang and Ren [77] introduced the cooperative hunting strategy of wolves to devise a solution-space model for a VRP with multiple fuzzy time windows. The authors modified the migration modes and summoning behavior of the conventional Wolf Pack Algorithm (WPA) by introducing drift and wave operators. These operators are to ensure that the entire search space is reached and that there is steady information flow during a fierce wolf raid process. A self-adaptive dynamic adjustment factor mechanism was also introduced to reinforce the local search ability. GA, Tabu search and a local search were combined by He and Li [3] to generate a near-optimal solution for a VRP with partly simultaneous pickup and delivery. Borzou, Guy, Fausto and Andrea [78] addressed a CVRP characterized by statistically-correlated and uncertain travel times (CST) via a mean-variance approach that penalizes paths with high difference in travel time. The capacitated constraint is to ensure that the capacity of each vehicle is not exceeded. Basically, CVRP-CST was developed to devise vehicular paths with reliable travel times only. The authors proposed two optional set-partitioning models for the resulting parametric binary quadratic system by developing an exact branch-price-and-cut algorithm that allows for sub-problem and column generation master problem-based derivation of solution(s) to the quadratic component of each model. Evaluation results presented indicate that CVRP-SCT exhibits significant improvement in time variability over conventional CVRP solutions. Homero, Leandro, Claudia and Maria [79] developed a Variable MIP Neighborhood Descent (VMND) algorithm by embedding a local search heuristic into a branch-and-bound algorithm to solve a multi-vehicle, multi-period VRP with due dates.

Pratiwi *et al.* [31] developed a hybrid Bat-SA algorithm and a crow search-based cat swarm optimization algorithm for solving VRPTW with large customer base. Experimental evaluations indicate that the crow search-based algorithm outperforms the hybrid Bat-SA for large VRPTW size. In the work of Malek, Sana, Hafiz and Salwani [80], bee algorithm was used to solve VRPTW and evaluated using Solomon's benchmark dataset. Gewen, Yanguang and Hao [81] developed a Multi-agent ACO (MACO) for solving VRP with Soft Time Windows and Road Factor (VRPSTWRF). The constraints of interest are fuel consumption, transport cost and customer satisfaction. In solving a prototype 40-customer VRPSTWRF, the authors introduced pheromone expectation and adaptive heuristic factors to obtain global convergence. A 3-opt strategy was also used to improve local search ability. Petr [41] used a deterministic approach to optimize ACO for solving a modified multi-depot VRP. Rubén, John and Mauricio [39] used eight intra- and inter-route local search strategies for optimizing travel routes for a Multi-Depot VRP with a Heterogeneous Fleet (MDHFVRP). Auxiliary graphs were employed for encoding and a modified GA was used to attain optimal quality of solution. Dedy, Herman and Buulolo [82] modeled the VRPTW with pick-up and delivery. The authors considered the fleet and driver as inclusion criteria into the optimization problem and also applied the direct search method to generate the solution space.

IV. METHOD

In this section, Genetic algorithm, Termite hill algorithm, the proposed hybrid TERMHIGEN, the Solomon's benchmark problems and the performance evaluation metrics are presented.

IV.I Termite-Hill Algorithm

Termites are social insects often characterized as autonomous, interdependent, adaptive and simple but with relatively small size and reduced number of neurons [83],[84]. Termite colonies possess sophisticated cooperative and self-organization abilities that make them capable of handling complex tasks in a well-coordinated, efficient and effective manner towards realizing targeted global objectives [85],[86]. Their adaptive cooperative behaviour has been conceptualized to solving many real-life distributed routing problems. Termite-inspired schemes have shown outstanding results when applied to areas including intelligent route maintenance, stagnation avoidance, quality of service delivery and optimized network performance in Mobile Ad-Hoc Networks (MANET) [86]-[88]. A summary of Termite-inspired Algorithms and applications is presented in Table 1. From the evaluation of intelligent routing methods on MANET, termite algorithm exhibits superior performance over standard routing techniques like Ad-hoc On-demand Distance Vector (AODV) in realistically more adverse environments with less overhead [86]. TinyTermite routing algorithm, a modified variant of the traditional termite algorithm, was developed by Mina and Josh [9] to reduce energy consumption requirement of wireless sensor networks and to secure same against replay attacks and selective forwarding. TinyTermite was implemented on TinyOS-based Intel Mote 2 platform and offered over 30% reduction in energy use and lowered packet loss by over 50%. Praveenkumar, Kiran and Ram [29] developed a novel termite algorithm, *Opt-Termite*, to enhance loading balancing strategy on MANETs using the stigmergy concept for self-organization. A mobile-aware termite algorithm that uses pheromone smoothing mechanism to find reliable route and mitigate measurement bias on MANETs was developed by Kiran and Ram [89].

In a doctoral research conducted at Harvard University by Petersen [56], termite-inspired robots were developed following the characteristic mound-building behaviour of termites. An error tolerant, locally-perceiving and scalable multi-robot framework, termed TERMES, was developed to allow for cooperative assemblage of large building structures by these robots.

Table 1. Termite-inspired Algorithms and Applications

S/N	Author(s)	Algorithm	Purpose	Application area
1	Zungeru *et al.* [85] and Zungeru *et al.* [94]	Termite-Hill and energy-aware Termite-Hill, respectively	Energy optimization and increased network lifetime of radio models, respectively.	Wireless Sensor Networks (WSN) and radio networks
2	Martin and Stephen [95]	*Termite*	optimized network performance	MANET
3	Sharvani [88]	Modified Termite Algorithm	quality of service, efficient and intelligent route maintenance	MANET
4	Martin and Stephen [86]	*Termite*	Routing behaviour modelling	MANET
5	Rajarajeswari *et al.* [93]	Energy Conserved - Supervised Termite Colony-based Role Assignment scheme (EC-STCRA)	Energy conservation	Wireless Sensor Networks
6	Petersen [56]	TERMES	error tolerance, local-perceptiveness and scalability	Multi-Robotic Systems
7	Mina and Josh [9]	TinyTermite	energy conservation and security	Wireless Sensor Networks
8	Praveenkumar *et al.* [29]	Opt-Termite	Load balancing for optimization	MANET
9	Sharad and Asha [96]	Termite Colony Optimization (TCO)	Optimal route finding and selection	Wireless Mesh Networks
10	Kiran and Ram [90]	Load Balanced (LB) - Termite	Load balancing, routing protocol	MANET
11	Abdul and Khaled [91]	Pheromone Termite	high throughput, fast and robust Routing, low latency	Wireless Sensor Networks
12	Kiran and Ram [92]	Bat-termite	Fast route finding, backup route maintenance, routing protocol	MANET
13	Kiran and Ram [89]	Mobility-aware termite	Reliable path finding	MANET
14	Sharvani, Ananth and Rangaswamy [87]	Modified Load Balancing Termite Algorithm	Efficient stagnation avoidance	MANET

Towards finding reliable routes to the destination and tackling stagnation problem on MANETs, Kiran and Reddy [90] developed a Load Balanced Termite, LB-Termite. The overall target is to optimize load balancing. Abdul and Khaled [91] also developed a mobility-aware, scalable Pheromone Termite (PT) model based on pheromone sensitivity and packet generation speed for all forms of links. PT model was developed to minimize latency and improve throughput in WSNs. Kiran and Ram [92] developed a hybrid Bat-termite routing protocol by consolidating the echo-location feature of mammal bats with the cooperative hill building nature of termites. This initiative was to realize rapid route maintenance and efficient multiple routes' management A supervised termite colony-based scheme tagged, EC-STCRA, was developed by Rajarajeswari, Karthikeyan and Deva [93] for energy conservation in wireless sensor networks. EC-STCRA performed significantly better than most local search and metaheuristically-enhanced routing schemes based on Packet Loss and Delivery Ratios, delay, throughput, delay and Residual Energy. Sharad and Asha [96] developed a Termite colony optimization algorithm for wireless mesh networks to improve optimal route finding strategy. In a related manner, Termite Hill Algorithm (THA) was developed by Zungeru *et al.* [85] to demonstrate the characteristic social behavioral patterns of termites for solving real-world routing problems in wireless sensor networks. More recently, an energy-aware Termite hill algorithm was developed by Zungeru, Chuma and Mangwala [94] to realize increased network lifetime of radio models. In a typical termite hill scenario, termite agents migrate as packets through the network and modifying routing parameters towards finding the most cost effective path to reach one or more destinations depending on the network size. The destination is a specialized code often referred to as the sink node. There are forward soldiers that gather information that are consequently kept in the pheromone table. This table appears in matrix form containing information regarding the destination and neighbor nodes. At each node, a table that tracks the amount of pheromone on each neighbor route is maintained. This is to allow for the determination of selecting probabilities of each neighbor by the values in the table. Generally, Termite-oriented algorithms are energy efficient with less latency, adaptable, scalable and robust with a characterized global optimum convergence [56], [93]. The termites evaluate the quality of each new route to a hill by the pheromone contents of the pebbles on the path. A pheromone table, containing set of entries $T_{n,h}$ can be described as a $n \times h$ matrix of neighbors and destinations such that n is a vector containing a set of neighbors $n_1, n_2, \ldots, n_x$ and h contains a set of hills (sink nodes) $h_1, h_2, \ldots, h_d$ where $x\, and\, d$ are the total number of intermediate nodes and destinations, respectively. However, updated pheromone value for a node is denoted as [85]:

$$T^1{}_{n,h} = T_{n,h} + \gamma \tag{11}$$

where $T_{n,h}$ is the initial pheromone value.

Simply put,

$$\gamma = \frac{N}{E-(\frac{E_{min}-N_j}{E_{avg}-N_j})} \quad (12)$$

where γ is the new information added to $T_{n,h}$ when a new packet is delivered through an immediate node, *n*, from an originating node *h*; *E* is the initial energy of the nodes while E_{min} and E_{avg} depict the minimum and average energy expended while the forward soldier moves towards the sink node. Route is determined and selected via the initialization of the routing tables of all nodes using a uniform probability distribution stated as:

$$P_{n,h} = \frac{1}{N} \quad (13)$$

This probabilistic function indicates if a forward soldier at any source node, *n*, will get to the sink node, *h,* and *N* is the total number of all nodes in the network.

IV.II Genetic Algorithm

Genetic Algorithm (GA), borne from the principle of natural evolution, has found wide application in a number of theoretical optimization problems and several industrial applications. It has been successfully adopted as a guided search technique and arbitrary functions' optimizer yielding approximate solutions. Given a problem, a typical GA repeatedly encodes members of a finite set of possible candidate solutions (*individuals*) via the principle of natural selection with strong genetic inheritance from a population of chromosomes using m*utation* and *crossover* operators. A concise description of a typical GA procedure is presented as Algorithm 1 [97]:

Algorithm 1: Pseudocode of the Conventional GA

i. Obtain an initial population of N solutions.
ii. While termination condition is false
  a. Devise a fitness function to evaluate each solution of the initial population
  b. Via randomness or probability scheme, select best-fit solutions to form new generation from the set of initial population
  c. Generate new generation from the parent solutions in (ii.b) via a crossover procedure.
  d. Modify the new generation randomly via a mutation probability.
  e. Repeat (ii) through (v) until a stopping criteria is satisfied.
iii. return best-fit solution

In GA, a vehicle is represented using a chromosome. A demarcation between two routes in the chromosome is represented by each vehicle identifier. However, consequent upon the need for GA to maintain a large population of solutions, it suffers from premature convergence, consumes several megabytes of memory and turns out as inefficient due to over-reliance on crossover [21],[24]-[26],[98]. These limitations resulted in the development of several modifications specifically tailored to address VRPTW [22],[28],[99].

IV.III The Proposed TERMHIGEN Algorithm

The proposed TERMHIGEN algorithm is presented in Algorithm 2 consisting of a modified Genetic Algorithm (GA) in stage 1, to offer optimized fitness measurement, evaluation and selection via a ranking approach, and the Termite Hill algorithm in stage 2, respectively. In the modified GA, the entire population is initially ranked such that each chromosome receives a fitness value due to the ranking with the worst having the least fitness and the best having the fitness value equivalent to the entire number of chromosomes that make up the population. This way, all the chromosomes have higher likelihood of being selected in a bid to eliminate premature convergence of GA.

Algorithm 2: The Proposed TERMHIGEN Algorithm

Stage 1: Optimized Fitness Measurement, Evaluation and Selection

Initialization:

i. $t \leftarrow 0$;; // at time 0
ii. NoOfRuns : maximum number of iteration;
iii. Iteration ← 0;
iv. InitializeEntirePopulation $[P(t)]$;
// *P*(*t*) = *N;* Given that *N* chromosomes form the initial population
v. EvaluateEntirePopulation $[P(t)]$; ;
//sort the initial population based on fitness values
vi. **while** not end do
vii. $P^1(t) \leftarrow Variation[P(t)]$;
//creates new entire solutions
viii. EvaluatePopulation [P^1 (*t*)];
//*evaluates new solutions via the application of fitness function and selection strategy*
//based on $f_{new}(S)$ as follows:

$$\begin{Bmatrix} f_1(S)=F_{VRP}(S)+\lambda.overcapacity(S)+\mu.overtime(S) \\ f_{eval}(S)=f_{max}-f_1(S) \end{Bmatrix} \quad (14)$$

where $F_{VRP}(S)$ is the sum of the total costs of all the distance, overcapacity(S) is the solution's capacity overhead subject to the maximum allowed value of each route, λ. Overtime(S) is the solution's time overhead subject to the maximum allowed weight for each route, μ. However, f_{eval} is the final evaluated fitness value and f_{max} is the maximum value obtainable.

$f_g = \frac{1}{x_g}$ where x_g is the objective function value with respect to each chromosome (inverse of the total cost where total cost depicts the fitness value).

$$f_{new}(S) = \frac{1}{2}\left(f_{eval}(S) + f_g\right) \quad (15)$$

//fitness function
//*evaluates new solutions via the application of ranking selection approach*

The ranking selection function used is expressed as:

$$\{p\ (k) = \frac{2k}{M(M+1)}\} \quad (16)$$

where k is the k^{th} individual in the rank and M represents the size of the population. In a case where $k=M$, the best individual emerges with a probability $\frac{2}{M+1}$ of being selected.

ix. $P(t+1) \leftarrow ApplyGeneticOperator\ s[P^i(t)]$ //next generation population
x. $t \leftarrow t+1$
xi. **end while**
xii. Obtain new population, $P^i(t)$

Stage 2: Termite-Hill Re-Routing Procedure

xiii. Initialize distribution parameters;
xiv. Obtain P^i; // $P^i < P$ is the new population
xv. **if** there exists one or more termite hills (sink nodes), $\bigcup_1^j j \in N$ where N is a set of natural numbers and j is the number of termite hills (sink nodes) depending on the size of the network
begin
a. **if** (customer_demand_available_for_transmission_at_originating_node_to_sink (ON)) == TRUE and (ON_valid_data_routing table_exist) == FALSE
b. **then** Generate a forward soldier and send it to all intermediate neighbors
c. **if** (intermediate_node_receives_forward_soldier == TRUE)
a. Acquire hill address (sink node / destination) from the forward soldier
b. **if** (search_for_ a_valid_ path_to_a_hill_in_the local_routing_table == SUCCESS) **then** Generate backward soldier packet with valid path address to the hill
if (the next hop is not the originating node of the forward soldier),
repeat,
a. send backward soldier as a unicast message to the next intermediate node via reverse routes
b. apply cross layer approach to detect and avoid routes with high packet loss
c. check feedback from link layer (MAC) to detect link failures
until the originating node (ON) of the forward soldier receives the backward soldier
endif
endif
endif
endif
d. update routing table on all intermediate nodes using equations (8 and 9)
e. identify best valid route to the hill using updated forward pointer information (equation 10)
f. convey message from backward soldier to the next intermediate node using reverse links
end
endif
xvi. **do while** (iteration < NoOfRuns)
repeat
stage 1, stage 2;
until no further improvement is observed.
end do
xvii. Output the solution

As a two-staged algorithm, the fitness and selection procedures in GA were modified to ensure rich global convergence and efficiency of the overall VRP solution using Equations (14 and 15) in the first stage. Ranking selection approach developed by Rakesh [100] was adopted to sort all individuals in the population using individual fitness value in such a way that an individual with a better fitness value $f(x)$ gets a higher rank based on the optimization criteria using Equation 16. This helps to realize quick convergence as good individuals become predominant in the population. In the second stage, two distinct termites' colonies were used and each termites' colony was saddled with an objective to optimally minimize the number of vehicles and the distance respectively. A sophisticated cooperative system among the colonies through a pheromone update allows for information exchange such that both colonies become re-activated and updated when a new packet is delivered to a node. The basic assumptions include:

i. each node is adjacently connected to one or more intermediate nodes in the network (neighbors);
ii. to establish communication among adjacent pair of nodes, a node may serve multiple purposes as source, as a router or as a destination;
iii. prior to packet transmission, information regarding the configuration of the network and routing table structure is unknown;
iv. energy requirement to transmit a message between any pair of adjacent nodes all through the network is equal.

IV.IV Solomon's Benchmark VRPTW Problems

Solomon's benchmark problems is a standard publicly-available VRPTW dataset developed by Solomon [101] and freely downloadable from http://neo.lcc.uma.es/vrp/vrp-instances/capacitated-vrp-with-time-windows-instances. This dataset describes a very considerable number of different real life VRPTW scenarios, and in turn, has been widely used in several studies including the works of Malek, Sana, Hafiz and Salwani [80], Avirup *et al.*, [23], Kumar and Panneerselvam [102], Azi, Gendreau and Potvin [103], Suresh and Ramasamy [5] and Yingjie and Michael [46] among others. The benchmark dataset contains 56 VRPTW with each having 100 customer instances. The dataset is organized into six (6) categories C1, C2, R1, R2, RC1 and RC2 based on customers' locations and the time windows. There are two categories for remotely-distributed customers (R1 and R2), clustered customers (C1, C2) and combinations of remotely-distributed and clustered customers (RC1 and RC2). Categories (C1, R1 and RC1) have vehicles with low capacities and short time windows while categories C2, R2 and RC2 have vehicles with higher capacities and longer time windows. Each of the 56 VRPTW's instances differ relatively based on the customer's location (*x* and *y* coordinates), customer's preferred time window to be served, service due time, quantity of demand, capacity of the vehicle, service time, vehicle travel time and total number of vehicles.

IV.V Performance Evaluation Metrics

i. *Traveled Distance (TD)*: This is the value of the total distance travelled by all the vehicles
ii. *Number of Vehicles (NV)*: This is the value of the total number of vehicles used
iii. *Execution Time*: This refers to the central processing unit runtime spent to generate solutions to each instance of the Solomon's benchmark problems.

V. EXPERIMENTAL RESULTS AND DISCUSSION

The algorithms (the modified GA, the Termite Hill and the TERMHIGEN) developed in this study were implemented in MATLAB 2015a environment and evaluated using Solomon's VRPTW benchmark problems on a HP 15 Notebook PC with 320 GB Hard drive, Windows 8.1 Pro 64-bit, Intel CPU N2820, 2.13GHz and 4MB L3 cache specification. The results produced by the modified GA, the Termite Hill and the TERMHIGEN algorithm for solving the 56 prototype problems are presented in Table 2. For each of the algorithms, an average value of TD and NV was generated based on 10 consecutive runs for each problem instance. However, best (least) values of TD and NV generated by TERMHIGEN from the 10 consecutive runs of each problem instance are also reported. The CPU time to generate the results were also computed and presented. The best known solutions to the 56 Solomon Benchmark problems presented in Table 2 were obtained from several studies including the works of Malek, Sana, Hafiz and Salwani [80], Munari and Morabito [104] and Sheng-Hua, Ji-Ping, Fu-Hao, Liang and Li-Jian [28], having apparently different methodological approaches. These aforementioned works presented more harmonized previous best-known solutions which are equally more optimal than those stated in some other works as well as new best-known solutions which are also considered in this study.

Based on the results of the developed TERMHIGEN algorithm, three (3) new best-known solutions, marked red and in bold format in Table 2, are reached for RC102, RC105 and RC106 as summarized in Table 3. Actually, there is a trade-off between the number of vehicles used and the total distance travelled.

Table 2. Results of the Total Distance Travelled obtained

Problem No.	Problem Instance	Best Known Solution		Modified Genetic Algorithm		Termite Hill		TERMHIGEN			
		Best Values		Average		Average		Best		Average	
		TD	NV	TD	NV	TD	NV	TD	NV	TD	NV
0	C1-01	827.30	10	843.26	10.10	842.49	10.10	827.30	10	842.45	10.00
1	C1-02	827.30	10	828.98	10.00	828.97	10.00	827.30	10	828.76	10.00
2	C1-03	826.30	10	847.26	10.00	832.45	10.00	826.30	10	828.04	10.00
3	C1-04	822.90	10	823.99	10.00	823.61	10.00	822.90	10	823.28	10.00
4	C1-05	827.30	10	849.82	10.00	843.43	10.10	827.30	10	839.95	10.00
5	C1-06	827.30	10	846.91	10.00	847.45	10.00	827.30	10	844.68	10.00
6	C1-07	827.30	10	842.70	10.10	842.35	10.20	827.30	10	841.96	10.10
7	C1-08	827.30	10	855.26	10.00	845.52	10.00	827.30	10	841.18	10.00
8	C1-09	827.30	10	889.07	10.10	841.76	10.10	827.30	10	835.79	10.10
9	C2-01	588.88	3	598.93	3.20	598.91	3.20	***589.10***	3	598.89	3.10
10	C2-02	588.88	3	628.65	3.30	610.34	3.10	***589.10***	3	592.11	3.10
11	C2-03	585.27	3	606.10	3.00	604.55	3.00	***591.12***	3	603.58	3.00
12	C2-04	584.49	4	689.62	3.10	665.78	3.00	***590.60***	3	628.37	3.00
13	C2-05	588.49	3	599.01	3.10	598.24	3.20	***588.88***	3	597.85	3.10
14	C2-06	588.49	3	600.49	3.00	589.21	3.00	588.49	3	588.32	3.00
15	C2-07	587.31	3	599.39	3.00	599.35	3.00	587.31	3	599.33	3.00
16	C2-08	588.32	3	612.05	3.00	603.89	3.00	588.32	3	599.62	3.00
17	R1-01	1483.57	16	1629.21	19.00	1633.45	19.30	***1628.31***	20	1639.67	19.40
18	R1-02	1355.93	14	1477.14	17.80	1476.58	17.00	***1471.91***	17	1476.23	17.00
19	R1-03	1133.35	12	1239.01	14.10	1239.98	13.40	***1227.67***	13	1241.16	12.20

20	R1-04	968.28	10	992.52	9.70	992.06	9.70	***979.28***	10	991.96	9.50
21	R1-05	1262.53	12	1363.42	15.00	1362.75	13.80	***1343.72***	14	1361.98	14.20
22	R1-06	1201.78	12	1247.29	12.10	1247.02	12.10	1201.78	13	1246.96	12.30
23	R1-07	1051.92	11	1088.74	10.70	1088.49	11.00	1051.92	11	1087.02	10.60
24	R1-08	948.57	9	958.82	9.70	957.32	9.30	948.57	9	955.65	9.20
25	R1-09	1110.40	12	1169.71	11.20	1165.60	11.20	1110.40	12	1164.70	12.20
26	R1-10	1080.36	11	1098.14	10.20	1098.11	10.70	1080.36	11	1098.08	10.60
27	R1-11	987.80	10	1059.14	10.60	1054.09	11.50	***1042.19***	12	1056.06	11.70
28	R1-12	953.63	10	988.58	10.50	975.74	10.00	***962.58***	10	967.92	9.60
29	R2-01	1148.48	6.5	1199.89	7.70	1182.64	6.10	1148.48	8	1178.26	8.20
30	R2-02	1049.74	3	1081.57	5.70	1080.72	4.90	1049.74	6	1079.83	5.60
31	R2-03	900.08	3	922.71	4.50	922.77	3.40	900.08	5	922.81	4.40
32	R2-04	772.33	2	826.47	2.30	810.72	2.20	772.33	3	799.86	3.10
33	R2-05	959.74	3	987.15	4.60	978.98	4.10	***966.74***	5	978.01	5.00
34	R2-06	898.91	3	903.89	3.20	903.85	3.10	898.91	4	903.83	4.00
35	R2-07	814.78	3	836.67	3.20	835.91	2.80	***814.82***	3	835.20	3.00
36	R2-08	715.37	2	725.53	2.00	725.51	2.70	715.37	3	725.43	2.40
37	R2-09	879.53	3	891.82	5.20	891.28	5.10	879.53	5	890.16	5.00
38	R2-10	932.89	3	943.89	6.30	940.62	4.70	932.89	5	937.66	4.10
39	R2-11	761.10	2	886.99	2.20	862.56	3.10	***801.63***	4	824.96	3.70
40	RC1-01	1619.80	15	1685.18	14.50	1654.89	14.80	1619.80	15	1638.56	14.70
41	RC1-02	1530.86	13	1487.10	13.50	1473.68	13.40	**1462.31**	14	1465.89	13.30
42	RC1-03	1221.53	10	1262.04	11.10	1261.93	11.00	***1248.67***	11	1261.63	11.00
43	RC1-04	1135.48	10	1135.89	10.00	1135.72	10.00	1135.48	10	1135.54	10.00
44	RC1-05	1629.44	13	1639.89	15.20	1625.73	14.60	**1617.32**	15	1624.65	15.20
45	RC1-06	1395.40	12	1392.80	12.20	1393.04	12.10	**1375.38**	12	1392.12	12.10
46	RC1-07	1230.50	11	1246.01	12.30	1232.51	12.20	1230.50	12	1232.37	12.10
47	RC1-08	1117.53	10	1133.24	10.50	1129.82	10.10	1117.53	10	1125.96	10.30
48	RC2-01	1249.00	4	1391.40	7.10	1391.10	4.40	***1337.62***	7	1390.63	6.30
49	RC2-02	1164.30	4	1243.34	3.30	1196.50	7.60	***1168.90***	8	1173.18	7.80
50	RC2-03	1049.62	3	1055.67	3.30	1052.04	3.20	***1051.62***	4	1051.99	4.10
51	RC2-04	798.41	3	798.67	3.10	798.60	3.00	798.41	3	798.52	3.16
52	RC2-05	1161.81	7	1262.51	6.50	1255.31	4.60	***1241.19***	7	1250.94	6.83
53	RC2-06	1059.89	3	1069.22	5.80	1063.57	3.60	1059.89	7	1061.07	6.01
54	RC2-07	976.40	3	1057.62	3.70	998.84	4.10	976.40	7	994.78	6.32
55	RC2-08	785.93	3	822.36	3.68	813.85	4.00	***793.87***	5	810.93	4.62

For RC102 and RC105, TERMHIGEN comparatively only showed improved performance over the previously best-known solutions in terms of total distance travelled at the expense of the number of vehicles. The latter solutions required lesser number of vehicles. However, for RC106, TERMHIGEN showed improved performance in terms of total distance travelled than the best-known solution while maintaining same number of vehicles used. Twenty-one (21) out of the 56 solutions generated by TERMHIGEN, marked bold and in italics, are very close to but could not reached the best-known solutions. However, it is able to generate similar previously best-known solutions to 32 out of the 56 Solomon Benchmark problems. As presented in Table 4, results obtained indicate that TERMHIGEN performs better than the modified Genetic algorithm and the Termite Hill algorithm in terms of the total and average total distance travelled for all the problem categories.

Table 3. New best-known solutions reached for RC102, RC105 and RC106

Problem	**Best Known Solution**			**TERMHIGEN Algorithm**	
	NV	**ND**	**Reference**	**NV**	**ND**
RC1-02	13	1539.86	Malek *et al*.[80]	14	1462.31
RC1-05	13	1629.44	Malek *et al*.[80]	15	1617.32
RC1-06	12	1395.40	Malek *et al*.[80]	12	1375.38

Table 4. Comparative Results of the Average Total Travelled Distance

Problem	**Number of Vehicles / Total Distance Travelled**	**Modified Genetic Algorithm**	**Termite Hill Algorithm**	**TERMHIGEN Algorithm**
		Average	**Average**	**Average**
C1	**NV**	10.03	10.06	10.02
	TD	847.47	838.67	836.23
C2	**NV**	3.09	3.06	3.04
	TD	616.78	608.78	601.01
R1	**NV**	12.55	12.42	12.38
	TD	1,192.64	1,190.93	1,190.62
R2	**NV**	4.26	3.84	4.41
	TD	927.87	921.41	916.00
RC1	**NV**	12.41	12.28	12.34
	TD	1,372.77	1,363.42	1,359.59
RC2	**NV**	4.56	4.31	5.64
	TD	1,087.60	1,071.23	1066.51
All	**NV**	448.28	438.90	455.34
	TD	56,762.73	56,322.18	56,106.32

Consequently, a slightly higher average number of vehicles was used by TERMHIGEN to solve R2 and RC2 problem categories than those required by the latter algorithms. Similarly, in comparison with some baseline VRPTW algorithms having previous best-known results as presented in Table 5, TERMHIGEN shows improved performance over GA+PSO [28], Bee algorithm [80], EC-MIN-DIS [105] and EC-MIN-VEH [105] in terms of the average number of vehicles used in all the solutions. In terms of the overall average distance travelled, TERMHIGEN performs better than EC-MIN-VEH, Bee and R-T [106] algorithms. It is important to mention that since the previously best-known solutions were produced by methods which were implemented on different machines having varying configurations, it cannot be said that a method is superior to another even if its results seem more significant but rather acceptable in a more relative sense.

In Table 6, the average execution times of the algorithms for the six (6) categories of Solomon's Benchmark problems are presented. As evident in all the evaluations conducted, TERMHIGEN produced the least average execution times, hence assumes the most computationally-efficient technique among the three, followed by Termite-Hill and the modified Genetic Algorithm in that order. This order is consistent for all the six categories of Solomon's Benchmark problems. The modified GA must have suffered high computational time complexity because of the incorporated ranking selection. Rank-based selection are usually computationally-expensive due to the additional sorting time requirement [100].

Table 5. Performance Comparison among VRPTW Algorithms

Algorithms	R-T		EC-MIN-VEH		EC-MIN-DIS		Bee Algorithm		GA+PSO		The proposed TERMHIGEN	
	NV	TD	NV	TD	NV	TD	NV	TD	NV	TD	NV	TD
Average Results	7.7	1030.6	8.2	1002.0	8.9	990.8	9.25	1020.08	8.33	1001.66	8.13	1001.9

Table 6. Results of the Average Execution Times (milliseconds) of the Algorithms

Problem Type	Modified Genetic Algorithm (ms)	Termite-Hill (ms)	TERMHIGEN (ms)
C1	118	100	63
C2	298	256	162
R1	131	114	73
R2	560	315	169
RC1	81	75	61
RC2	339	322	145

VI. CONCLUSION AND FUTURE WORK

In this paper, a new hybrid metaheuristic algorithm tagged "TERMHIGEN" was developed to solve VRPTW. TERMHIGEN is based on combined characteristics of Termite-Hill and a modified Genetic algorithm using ranking selection. With Solomon's Benchmark problems, TERMHIGEN was able to reach three (3) new best-known solutions for RC102, RC105 and RC106 in terms of the total distance travelled; however, it could not reach the recorded best-known number of vehicles used for RC102 and RC105. The developed algorithm also shows improved performance over the modified GA and Termite Hill in terms of the average total distance travelled and average execution time. However, further works can be directed to make TERMHIGEN better in terms of reduced number of vehicle requirement as a primary objective function. Similarly, future works can be directed towards the application of TERMHIGEN to solving other variants of VRP. In the same vein, variants of TERMHIGEN could be developed by combining Termite-Hill with other evolutionary algorithms like PSO, ACO and Bee algorithm and comparatively evaluate their performances for some general VRP instances.

REFERENCES

[1] Rojas-Cuevas, I-D., Caballero-Morales, S-O., Martinez-Flores, J-L. and Mendoza- Vazquez, J-R. (2018): 'Capacitated vehicle routing problem model for carriers', Journal of Transport and Supply Chain Management 12(0), a345. https://doi. org/10.4102/jtscm.v12i0.345.

[2] Tuomas Pellonpera (2014): Ant Colony Optimization and the Vehicle Routing Problem, A M.Sc. Thesis in the Department of Computer Science, School of Information Sciences, University of Tampere, pp. 1-51.

[3] He, P. and Li, J. (2018). Vehicle Routing Problem with Partly Simultaneous Pickup and Delivery for the Cluster of Small and Medium Enterprises. Archives of Transport, 45(1), 35-42. DOI: https://doi.org/10.5604/01.3001.0012.0940

[4] Vidal T., Crainic T. G., Gendreau M., and Prins C. (2013). "Heuristics for multi-attribute vehicle routing problems: a survey and synthesis". In: European Journal of Operational Research 231 (1), pp. 1–21.

[5] Suresh Nanda Kumar and Ramasamy Panneerselvam (2012). A Survey on the Vehicle Routing Problem and Its Variants, *Intelligent Information Management*, 4, pp. 66-74.

[6] Balseiro S. R., I. Loiseau and J. Ramone "An Ant Colony Algorithm Hybridized with Insertion Heuristics for the Time Dependent Vehicle Routing Problem with Time Windows," *Computers & Operations Research*, Vol. 2011, pp. 954-966.

[7] y E. (2010), "An Algorithm for the Vehicle Problem," International Journal of Advanced Robotic Systems, 7 (2), pp. 125-132.

[8] Deflorio F. P., Perboli G. and Tadei R. (2010). "Freight distribution performance indicators for service quality planning in large transportation networks". In: Flexible services and manufacturing journal 22.1-2 (2010), pp. 36–60.

[9] Mina Sartipi and Josh Patterson (2009): TinyTermite: A Secure Routing Algorithm on Intel Mote 2 Sensor Network Platform, Proceedings of the Twenty-First

Innovative Applications of Artificial Intelligence Conference, pp. 141-146.

[10] Kevin Spieser and Emilio frazzoli (2012). The Cow-Path Game: A Competitive Vehicle Routing Problem. 51st IEEE Conference on Decision and Control, December 10-13, Maui, Hawaii, USA, pp. 6513-6520.

[11] Haksever C., Render B., Russell R. and Murdick R. (2000). *Service Management and Operations*, 2nd ed. Prentice Hall: Upper Saddle River, pp. 476–497.

[12] Tonci Caric and Hrvoje Gold (2008), Vehicle Routing Problem, I-Tech, ISBN 978-953-7619-09-1,1-142.

[13] Jean-Francois Cordeau, Michel Gendreau, Alain Hertz, Gilbert Laporte and Jean-Sylvain Sormany (2004): New Heuristics for the Vehicle Routing Problem, Les Cahiers du GERAD, ISSN: 0711-2440.

[14] Min Huang and Ping Ding (2013). An Improved Ant Colony Algorithm and its Application in Vehicle Routing Problem. Mathematical Problems in Engineering, Volume 2013, ID 785383.

[15] Lingling Du and Ruhan He (2012). Combining Nearest Neighbor Search with Tabu Search for Large-Scale Vehicle Routing Problem. Physics Procedia, 25, pp. 1536-1546.

[16] Haitao Xu, Pan Pu and Feng Duan (2018): Dynamic Vehicle Routing Problems with Enhanced Ant Colony Optimization, Discrete Dynamics in Nature and Society, Hindawi publishers, 1295485, pp. 1-13.

[17] Malekly Hooman, Haddadi Behzad and Reza Tavakkoli-Moghadam (2009). A fuzzy random vehicle routing problem: The Case of Iran, International Conference on Computers and Industrial Engineering, 1070-1075.

[18] Zhou, R., Lee, H.P. and Nee, A.Y.C. (2008) 'Applying Ant Colony Optimization (ACO) algorithm to dynamic job shop scheduling problems', Int. J. Manufacturing Research, Vol. 3, No. 3, pp.301–320.

[19] Radcliffe N.J. (1991). Forma Analysis and Random Respectful Recombination. *ICGA*, pp. 222-229.

[20] Ibrahim El-Henawy and Nagham Ahmed Abdelmegeed (2018): Meta-Heuristics Algorithms: A Survey, International Journal of Computer Applications, 179 (22), pp. 45-54.

[21] Oyeleye C. Akinwale, Olabiyisi S. Olatunde, Omidiora E. Olusayo and Fagbola Temitayo (2014). Hybrid Metaheuristic Simulated Annealing and Genetic Algorithm for Solving Examination Timetabling Problem. International Journal of Computer Science and Engineering (IJCSE), India, 3 (5): pp 7-22.

[22] Ho, W., Ho, G. T., Jib, P., and Laub, H. W. (2008). A hybrid genetic algorithm for the multi-depot vehicle routing problem. Engineering Applications of Artificial Intelligence, 21(4), 548-557.

[23] Avirup Guha Neogi, Singamreddy Mounika, Salagrama Kalyani, Yogananda S.A. (2018): A Comprehensive Study of Vehicle Routing Problem With Time Windows Using Ant Colony Optimization Techniques, International Journal of Engineering & Technology, 7, pp. 80-85.

[24] Fagbola Temitayo Matthew, Babatunde Ronke Seyi and Oyeleye Akinwale (2013). Image Clustering Using a Hybrid GA-FCM Algorithm. International Journal of Engineering and Technology, UK, 3(2): pp 99-107.

[25] Fagbola Temitayo, Olabiyisi Stephen and Adigun Abimbola (2012): "Hybrid GA-SVM for Efficient Feature Selection in E-Mail Classification". Journal of Computer Engineering and Intelligent Systems, International Institute of Science, Technology and Education (IISTE), New York, USA, 3(3): pp 17-28.

[26] Olabiyisi Stephen O., Fagbola Temitayo M., Omidiora Elijah O. and Oyeleye Akin C. (2012): "Hybrid Metaheuristic Feature Extraction Technique for Solving Timetabling Problem". International Journal of Scientific and Engineering Research**,** USA**,** 3(8): pp 1-6.

[27] Nie Ru and Yue, J. A. (2008). Genetic Algorithm and Particle Swarm Optimization Based Hybrid Algorithm. In Proceedings of the IEEE Congress on Evolutionary Computation, Hong Kong, China, 1–6 June 2008; pp. 1047–1050.

[28] Sheng-Hua Xu, Ji-Ping Liu, Fu-Hao Zhang, Liang Wang and Li-Jian Sun (2015). A Combination of Genetic Algorithm and Particle Swarm Optimization for Vehicle Routing Problem with Time Windows. Sensors, 15, pp. 21033-21053.

[29] Praveenkumar G. Hoolimath, Kiran M and Ram Mohana Reddy G. (2012): Optimized Termite: A Bio-inspired Routing Algorithm for MANETS, International Conference on Signal Processing and Communications, pp. 1-5.

[30] Bhuvaneswari M., Sumathy Eswaran and Rajagopalan S. P. (2018): A Survey of Vehicle Routing Problem and its Solutions using Bio-Inspired Algorithms, International Journal of Pure and Applied Mathematics, 118 (9), pp. 259-264.

[31] Pratiwi A B., Pratama A, Sa'diyah I and Suprajitno H (2018). Vehicle routing problem with time windows using natural inspired algorithms. Journal of Physics: Conference Series: International Conference on Mathematics: Pure, Applied and Computation, 974 012025, pp. 1-10.

[32] Leonid A. Gladkov, Sergey N. Scheglov, Nadezhda V. Gladkova (2017): The application of bio-inspired methods for solving vehicle routing problems, Procedia Computer Science, 120, pp. 39–46.

[33] Doerner, K. F., Hartl, R.F., and Reimann, M.: Are CompetANTS competent for problem solving - the case of a transportation problem. POM Working Paper 01/2001.

[34] John E. Bell and Patrick R. McMullen (2004). Ant colony optimization techniques for the vehicle routing problem. Advanced Engineering Informatics, Volume 18, Issue 1, January 2004, Pages 4148. doi:10.1016/j.aei.2004.07.001.

[35] Rafael Mejía Santana (2016): Heuristic algorithms and Variants of the Vehicle Routing Problem for a Distribution Company: A Case Study, A Master's

Thesis of the European Master's Program in Computational Logic, Universidade Nova De Lisboa.

[36] Gautham Outtur Rajappa, Joseph Wilck and John Bell (2016). An Ant Colony Optimization and Hybrid Metaheeuristics Algorithm to Solve the Split Delivery Vehicle Routing Problem. International Journal of Applied Industrial Engineering, 3(1), pp. 55-73.

[37] Blanton, J.L. and Wainwright, R.L. (1993): Multiple Vehicle Routing with Time and Capacity Constraints Using Genetic Algorithms. In: Forrest, S., Ed., *Proceedings of the* 5*th International Conference on Genetic Algorithms*, Morgan Kaufmann Publishing, San Francisco, 452-459.

[38] Mirmohammadi S. H., Babaee Tirkolaee E., Goli A. and Dehnavi–Arani S. (2017): The Periodic Green Vehicle Routing Problem with Considering of Time-Dependent Urban Traffic and Time Windows, International Journal of Optimization in Civil Engineering, 7(1):143-156.

[39] Rubén Iván Bolaños, John Willmer Escobar and Mauricio Granada Echeverri (2018): A metaheuristic algorithm for the multi-depot vehicle routing problem with heterogeneous fleet, International Journal of Industrial Engineering Computations, Volume 9, pp. 1-18.

[40] Kourank Beheshti A., Hejazi S. R. and Mirmohammadi S. H. (2014): A Hybrid Metaheuristic Algorithm for the Vehicle Routing Problem with Delivery Time Cost, International Journal of Applied Operational Research, 4 (3), pp. 49-64.

[41] Petr Stodola (2018): Using Metaheuristics on the Multi-Depot Vehicle Routing Problem with Modified Optimization Criterion, Algorithms, 74 (11), pp. 1-14.

[42] Andrea Bettinelli, Alberto Ceselli and Giovanni Righini (2014): A Branch-and-Price Algorithm for the Multi-Depot Heterogeneous-Fleet Pickup and Delivery Problem with Soft Time Windows, Math. Prog. Comp., pp. 171–197.

[43] Duygu Tas, Nico Dellaert, Tom van Woensel and Ton de Kok (2012): Vehicle Routing Problem with Stochastic Travel Times including Soft Time Windows and Service Costs, Computers & Operations Research, pp. 1-29.

[44] Cagri Koc and Gilbert Laporte (2018). Vehicle Routing with Backhauls: Review and Research Perspectives, Computers and Operations Research, 91, pp. 79-91.

[45] Garcia-Najera Abel (2012). The Vehicle Routing Problem with Backhauls: A Multi-objective Evolutionary Approach. In: Hao JK., Middendorf M. (eds) Evolutionary Computation in Combinatorial Optimization, EvoCOP 2012. Lecture Notes in Computer Science, vol 7245, pp. 25-266, Springer, Berlin, Heidelberg.

[46] Yingjie Zhong and Michael H. Cole (2005): A Vehicle Routing Problem with Backhauls and Time Windows: A guided Local Search Solution. Transportation Research, 41, pp. 131-144.

[47] Mohammad Yousef Maknoon and Gilbert Laporte (2017). Vehicle Routing with Cross-dock Selection. Computers and Operations Research, 77, pp. 254-266.

[48] Baldacci, R., Mingozzi, A. and Roberti, R. (2012). Recent exact algorithms for solving the vehicle routing problem under capacity and time window constraints. European Journal of Operational Research, 218(1), 1-6. DOI:10.1016/j.ejor.2011.07.037.

[49] Beheshti A. K. and Hejazi S. R. (2015). "A novel hybrid column generation-metaheuristic approach for the vehicle routing problem with general soft time window". In: Information Sciences, vol. 316, pp. 598–615.

[50] Dhahri, A.; Zidi, K. and Ghedira, K. (2015). A Variable Neighborhood Search for the Vehicle Routing Problem with Time Windows and Preventive Maintenance Activities. *Electron. Notes Discret. Math.*, *47*, 229–236.

[51] Garcia-Najera, A. and Bullinaria, J.A (2011). An improved multi-objective evolutionary algorithm for the vehicle routing problem with time windows. *Comput. Oper. Res.*, *38*, 287–300.

[52] Toth P. and Vigo D. (2002). An Overview of Vehicle Routing Problems. In The Vehicle Routing Problem. SIAM, 2002.

[53] Beresneva E. and Avdoshin S. (2018): Analysis of Mathematical Formulations of Capacitated Vehicle Routing Problem and Methods for their Solution. *Trudy ISP RAN/Proc. ISP RAS*, vol. 30, issue 3, pp, 233-250.

[54] Christopher Groer (2008): Parallel and Serial Algorithms for Vehicle Routing Problems, a Doctoral thesis in the Faculty of the Graduate School of the University of Maryland, College Park.

[55] Roberto Roberti (2012): Exact Algorithms for Different Classes of Vehicle Routing Problems, a Doctoral Thesis in the Department of Control System Engineering and Operational Research Alma Mater Studiorum - University of Bologna.

[56] Petersen Kirstin Hagelskjaer (2014): Collective Construction by Termite-Inspired Robots. Doctoral dissertation, School of Engineering and Applied Sciences, Harvard University, Cambridge, Massachusetts.

[57] Petrica Claudiu Pop, Ioana Zelina, Vasile Lupse, Corina Pop Sitar and Camelia Chira (2011). Heuristic Algorithms for solving the Generalized Vehicle Routing Problem. International Journal of Computers, Communications and Control. 6 (1), pp. 158-165.

[58] Cordeau J-F, Gendreau M., Laporte G., Potvin J-Y. and Semet F. (2002): A Guide to Vehicle Routing Heuristics, The Journal of the Operational Research Society, 53(5), pp. 512-522.

[59] Laporte G, Gendreau M., Potvin J-Y. and Semet F. (2000): Classical and Modern Heuristics for the Vehicle Routing Problem. International Transactions in Operational Research, 7 (4-5), pp. 285-300.

[60] Cagric Koc, Tolga Bektas, Ola Jabali and Gilbert Laporte (2015). A Hybrid Evolutionary Algorithm for Heterogeneous Fleet Vehicle Routing Problems with

Time Windows. Computers and Operations Research, 64, pp. 11-27.

[61] Anna Maria Sri Asih, Bertha Maya Sopha, and Gilang Kriptaniadewa (2017): Comparison Study of Metaheuristics: Empirical Application of Delivery Problems, International Journal of Engineering Business Management, 9, pp. 1–12.

[62] Berger, J., Barkaoui, M., Bräysy, O. (2003). A route-directed hybrid genetic approach for the vehicle routing problem with time windows. *INFOR* 41, 179–194.

[63] Aslaug Soley Bjarnadottir (2015). Solving the Vehicle Routing Problem with Genetic Algorithms. Informatics and Mathematical Modelling, Technical University of Denmark, DTU, pp. 1-127.

[64] Mester, D. and Bräysy, O. (2005): Active Guided Evolution Strategies for Large-Scale Vehicle Routing Problems with Time Windows. Computers & Operations Research, 32, 1593-1614. http://dx.doi.org/10.1016/j.cor.2003.11.017.

[65] Russell, R.A. and Chiang W.-C. (2006): Scatter Search for the Vehicle Routing Problem with Time Windows. European Journal of Operational Research, 169, 606-622. http://dx.doi.org/10.1016/j.ejor.2004.08.018.

[66] He R., Xu W., Wang Y. and Zhan W. (2010). A Route-Nearest Neighbor Algorithm for Large-Scale Vehicle Routing Problem, 3rd International Symposium on Intelligent Information Technology and Security Informatics (IITST 2010), pp. 390-393.

[67] Kytojoki J., Nuortio T., Braysy O. and Gendreau M. (2007), "An efficient variable neighborhood search heuristic for very large scale vehicle routing problems," *Computers & Operations Research*, 34, pp. 2743-2757.

[68] Thangiah S. R., Nygard K. E. and Juell P. L. (1991). GIDEON: a genetic algorithm system for vehicle routing with time windows. In Proceedings of the Seventh IEEE Conference on Artificial Intelligence Applications, Miami, Florida, Volume I, pages 322-328.

[69] Rita Macedo, Cláudio Alves, Valério de Carvalho, François Clautiaux and Saïd Hanafi (2011): Solving the Vehicle Routing Problem with Time Windows and Multiple Routes exactly using a Pseudo-Polynomial Model, European Journal of Operational Research, 214, 536–545.

[70] Ostertag A., Doerner K. F., Hartl R. F., Taillard E. D. and Waelti P. (2008):."POPMUSIC for a Real World Large Scale Vehicle Routing Problem with Time Windows," *Journal of the Operational Research Society*, 60 (7), pp. 934-943.

[71] Zhang Xiao and Wang Jiang-qing (2012). Hybrid Ant Algorithm and Applications for Vehicle Routing Problem. Physics Procedia, 25, 1892-1899.

[72] Wenxue Ran, Li Liu and Guomin Yang (2013). A Hybrid Ant Colony Algorithm for Vehicle Routing Problem with Time Windows. Information Technology Journal, 12(20), 5701-5706.

[73] Masrom, S.; Abidin, S.Z.; Omar, N.; Nasir, K.; AbdRahman, (2015). Dynamic parameterization of the Particle Swarm Optimization and Genetic Algorithm hybrids for Vehicle Routing Problem with Time Window. *Int. J. Hybrid Int. Syst.*, *12*, 13–25.

[74] Peiqing Li, Jie He, Dunyong Zheng, Yongsheng Huang and Chenhao Fan (2015): Vehicle Routing Problem with Soft Time Windows Based on Improved Genetic Algorithm for Fruits and Vegetables Distribution, Discrete Dynamics in Nature and Society, Hindawi, 483830, pp. 1-9.

[75] Oyediran M. O., Fagbola T. M., Olabiyisi S. O. and Omidiora E. O. (2016): Development of an Optimized Mobile Agent Migration Pattern for Pull-All Data Strategy, British Journal of Mathematics and Computer Science, SCIENCEDOMAIN International, 16(6): pp 1-8.

[76] Martin Macho (2017): Vehicle Routing Problem with Time Windows and its solving Methods, A Bachelor's Thesis in the Department of Theoretical Computer Science, Faculty of Information Technology, Czech Technical University in Prague.

[77] Cao Q.K., Yang K.W. and Ren X.Y. (2017): Vehicle Routing Optimization with Multiple Fuzzy Time Windows based on Improved Wolf Pack Algorithm, Advances in Production Engineering & Management, 12 (4), pp 401–411.

[78] Borzou Rostami, Guy Desaulniers, Fausto Errico and Andrea Lodi (2018). The Vehicle Routing Problem with Stochastic and Correlated Travel Times. Data Science for Real-Time Decision making, Canada Excellence Research Chair, pp. 1-50.

[79] Homero Larrain, Leandro C. Coelho, Claudia Archetti and Maria Grazia Speranza (2018): Exact Solution Methods for the Multi-Vehicle Multi-Period Vehicle Routing Problem with Due Dates. Interuniversity Research Centre on Enterprise Networks, Logistics and Transportation (CIRRELT).

[80] Malek Alzaqebah, Sana Jawarneh, Hafiz Mohd Sarim and Salwani Abdullah (2018): Bees Algorithm for Vehicle Routing Problems with Time Windows, International Journal of Machine Learning and Computing, 8 (3), pp. 234-240.

[81] Gewen Huang, Yanguang Cai, and Hao Cai (2018): Multi-agent ant colony optimization for vehicle routing problem with soft time windows and road condition, MATEC Web of Conferences, SMIMA, 173, pp. 1-6.

[82] Dedy Juliandri, Herman Mawengkang, F. Bu'ulolo (2018): Discrete Optimization Model for Vehicle Routing Problem with Scheduling Side Constraints 4th International Conference on Operational Research (InteriOR), pp. 1-8.

[83] Stigmergy (2004):http://institute.advancedarchitecture.org/Research/Ants/Stigmergy.

[84] Marais E. (1948): *Die Siel van die Mier (The Soul of the Ant)*. Pretoria, South Africa: J.L. van Schaik.

[85] Zungeru A., Ang L. K. and Seng K. (2012). Termite Hill: Performance Optimized Swarm Intelligence based routing algorithm for wireless sensor networks. Journal

of Network and Computer Applications, 35(6), 1901-1917.

[86] Martin Roth and Stephen Wicker (2004): Asymptotic Pheromone Behavior in Swarm Intelligent MANETs: An Analytical Analysis of Routing Behavior, Sixth IFIP IEEE International Conference on Mobile and Wireless Communications Networks (MWCN).

[87] Sharvani G. S., Ananth A. G. and Rangaswamy T. M. (2012). Ant Colony Optimization-based Modified Termite Algorithm (MTA) with Efficient Stagnation Avoidance Strategy for MANETS. International Journal on Applications of Graph Theory in Wireless Ad-Hoc Networks and Sensor Networks (GRAPH-HOC), 4 (2-3), pp. 39-50.

[88] Sharvani G. S. (2012): Development of Swarm Intelligent Systems for MANET. A doctoral thesis in the Department of Computer Science and Engineering, Faculty of Engineering, Avinashilingam University for Women, Coimbatore.

[89] Kiran Manjappa and Ram Mohana Reddy Guddeti (2013): Mobility aware-Termite: A novel Bio-inspired Routing Protocol for Mobile Ad-Hoc Networks. The Institution of Engineering and Technology, 2 (4), pp. 188-195.

[90] Kiran Manjappa and Reddy Guddeti (2014a): Design and Evaluation of Load Balanced Termite: A Novel Load Aware Bio-Inspired Routing Protocol for Mobile Ad-Hoc Network. Wireless Personal Communications, 75 (4), pp. 2053-2071.

[91] Abdul Razaque and Khaled Elleithy (2014): Pheromone Termite (PT) Model to provide Robust Routing over Wireless Sensor Networks. Proceedings of IEEE Conference of the American Society for Engineering Education, pp. 1-6.

[92] Kiran Manjappa and Reddy Guddeti (2014b). Bat-termite: A Novel Hybrid Bio-inspired Routing Protocol for Mobile Ad-Hoc Networks. IJWMC, 7, 258-269.

[93] Rajarajeswari. P. L., Karthikeyan N. K. and Deva Priya M. (2015): EC-STCRA: Energy Conserved - Supervised Termite Colony-based Role Assignment scheme for Wireless Sensor Networks, International Conference on Computational Science, Procedia Computer Science, 57, pp. 830 – 841.

[94] Zungeru A. M., Chuma J. and Mangwala M. (2017). Performance Analysis of Energy-aware Routing Protocols for Wireless Sensor Networks using Different Radio Models. International Journal of Applied Engineering Research, 12 (20), 9298-9305.

[95] Martin Roth and Stephen Wicker (2006): Termite: A Swarm Intelligent Routing Algorithm for Mobile Wireless Ad-Hoc Networks. In: Stigmergic Optimization. Studies in Computational Intelligence, 31, Springer, Berlin, Heidelberg.

[96] Sharad Sharma and Asha Malik (2017). Routing in Wireless Mesh Networks based on Termites Intelligence, International Journal of Applied Metaheuristic Computing, IGI Global, 8(2), pp. 1-21.

[97] Holland, J. H. (1975). Adaptation in Natural and Artificial Systems. Cambridge, MA: MIT Press, First edition, University of Michigan Press.

[98] Gen, M., Cheng, R. and Lin, L. (2008) Network Models and Optimization: Multi-objective Genetic Algorithm Approach. Springer Science & Business Media, New York.

[99] Wah Ben and Chen Y.X. (2000). Constrained Genetic Algorithms and their Applications in Nonlinear Constrained Optimization. Proceedings of IEEE ICTAI. Vancouver, pp. 286-293.

[100] Rakesh Kumar (2012). Blending Roulette Wheel Selection & Rank Selection in Genetic Algorithm. International Journal of Machine Learning and Computing, 2(4), pp. 365-370.

[101] Solomon M.M. (1987): Algorithms for the Vehicle Routing and Scheduling Problems with Time Window Constraints, Operations Research, 35 (2), 254–265.

[102] Kumar, S. N. and Panneerselvam, R. (2015): A Time-Dependent Vehicle Routing Problem with Time Windows for E-Commerce Supplier Site Pickups Using Genetic Algorithm. Intelligent Information Management, **7**, pp. 181-194.

[103] Azi N., Gendreau M., Potvin J.-Y. (2010): An Exact Algorithm for a Vehicle Routing Problem with Time Windows and multiple use of Vehicles, European Journal of Operational Research, 202, 756–763.

[104] Munari P and Morabito R. (2018). A branch-price-and-cut algorithm for the vehicle routing problem with time windows and multiple deliverymen. TOP, 26:437.

[105] Sontrop H., Van der Horn P. and Uetz M. (2005). Fast Ejection Chain Algorithms for Vehicle Routing with Time Windows. In: Blesa M.J., Blum C., Roli A., Sampels M. (eds) Hybrid Metaheuristics. HM 2005. Lecture Notes in Computer Science, Vol 3636, Springer, Berlin, Heidelberg.

[106] Rochat Y. and ´E. D. Taillard (1995). Probalistic Diversification and Intensification in Local Search for Vehicle Routing. Journal of Heuristics, 1:147–167.