Cihan University-Erbil Scientific Journal (CUESJ)

REVIEW ARTICLE

On the Development of a Multilayered Agent-based Heuristic System for Vehicle Routing Problem under Random Vehicle Breakdown

Anees M. Abu-Monshar[1], Ammar F. Al-Bazi[1]*, Qusay H. Al-Salami[2]

[1]Faculty of Engineering, Environment and Computing, Coventry University, Coventry, United Kingdom, [2]Department of Business Administration, Cihan University-Erbil, Kurdistan Region, Iraq

ABSTRACT

With the recent technological advancement, the Dynamic Vehicle Routing Problem is becoming more applicable. However, almost all of the researches in this field limited the source of dynamism from the order side instead of the vehicle and the adoption of inflexible tools that are mainly designed for the static problem, considering multiple random vehicle breakdowns complicate how to adapt and distribute the workload to other functioning vehicles. In this ongoing PhD research, a proposed multilayered agent-based model (ABM) and a modeling framework on dealing with such disruptive events in a continuous reactive manner. The model is partially constructed and experimented, with a developed clustering rule, on two randomly generated scenarios for validation. The rule achieved reasonable order allocation to vehicles and reacted to different problem sizes by rejecting orders over the model capacity. This allocation shows a promising path in fully adopting the ABM model in this dynamic problem.

Keywords: Agent-based modeling, dynamic, vehicle routing, breakdown, heuristic

INTRODUCTION

Technological advancements in computing and communication are believed to influence how businesses are operated significantly. The Internet of Things concept is one of the main drivers in revolutionizing business processes by supplying physical objects with electronic devices that will allow for real-time control and monitoring of such objects.[1] The "Industrie 4.0" concept further supports such a revolution by providing more customized services efficiently.[2] The transportation and logistics fields are far from these transformational concepts as they are one of the main themes for potential integration.[3]

Vehicle Routing Problem (VRP) is one of the well-known logistical problems extended from the Travelling Salesman Problem (TSP) to accommodate additional constraints. The problem was first seen in 1959 by Dantzig and Ramser,[4] concerned about providing vehicles to visit customers' locations starting and ending at a depot. Other variants of the problem were later introduced: Capacitated vehicles, time window constraint, multiple depots, and pick-up before delivery.[5] VRP is proven to be NP hard,[6] and solutions adopted are mainly (meta)-heuristics that provide near-optimal routes; for example, Solomon[7] adopted an insertion heuristic for VRP with time window and Schneider[8] adopted Tabu search (TS).

Although VRP problems have been well-explored, it is only the static type of the problem that has been researched and recent research interested in shifting toward the online and dynamic problem.[9] Several pieces of research have been done on the Dynamic Vehicle Routing Problem (DVRP); however, it only focused on updates to customer orders, new orders, or cancellation, rather than disruptions in vehicle operations. Furthermore, solutions adopted in such dynamic problems are still inspired by the classical and static VRP, making them inflexible in adopting changes. Therefore, the agent-based modeling (ABM) approach is proposed to DVRP under breakdown due to the responsive and flexible approach in producing solutions in such a dynamic context.

This paper addresses this dynamic breakdown problem due to its applicability to recover optimal schedule and routes of a logistical operation where vehicle failures occur or when

Corresponding Author:
Ammar F. Al-Bazi, Faculty of Engineering, Environment and Computing, Coventry University, Coventry, United Kingdom.
E-mail: aa8535@coventry.ac.uk

Received: Feb 26, 2021
Accepted: May 07, 2021
Published: May 20, 2021

DOI: 10.24086/cuesj.v5n1y2021.1-10

drivers have problems.[10] The main contribution of this paper is to introduce a novel multilayered agent-based heuristic system for the breakdown VRP problem that solves the breakdown instant by first clustering customers then generating vehicle routes.

The rest of the paper is organized as follows: Section II states the problem under study. Section III provides a selection of literature papers relevant to this work in the area of DVRP. Section IV justifies using the ABM approach in DVRP and presents related work. Section V presents the two-layered ABM approach, and the modeling framework specifically adapted to the dynamic vehicle breakdowns in the routing problem. Section VI provides preliminary results for the partially developed ABM model under two problem scenarios. Finally, Section VII draws the conclusion and future research recommendation.

PROBLEM STATEMENT

In a classical VRP with both delivery and parcel collections, vehicles are routed to specific customer locations starting and ending at their representative depots. However, one vehicle or more of the operating fleet might face random disruptions at any time and location while in service. As a result, an issue would arise of sharing the disrupted vehicle workload with the remaining in-service vehicles. This workload is represented by parcels initially supposed to be delivered to/collected from customers by the disrupted vehicle. Figure 1 presents this type of vehicle disruption problem.

Figure 1 shows a set of in-service vehicles scheduled routes, represented by the solid arrows. These scheduled routes have been determined by considering vehicle capacity constraint, customer time window availability, demand quantity of either delivery or collection, and the required servicing time. In terms of distance and time, these original routes' costs depend on the sequence of customers in the routes.

However, a major vehicle breakdown might occur randomly during the scheduled routes and result in some unserved customers. To mitigate the problem, the other in-service vehicles' routes could be rescheduled to visit the location of the disrupted vehicle that is considered here as a collection point to collect only the delivery load and consider the new reschedule route aiming to minimize these unserved customers. The dotted arrows indicate the rescheduled route by considering the visit to the disrupted vehicle, taking its workload, fully or partially, as well as serving its remaining customers.

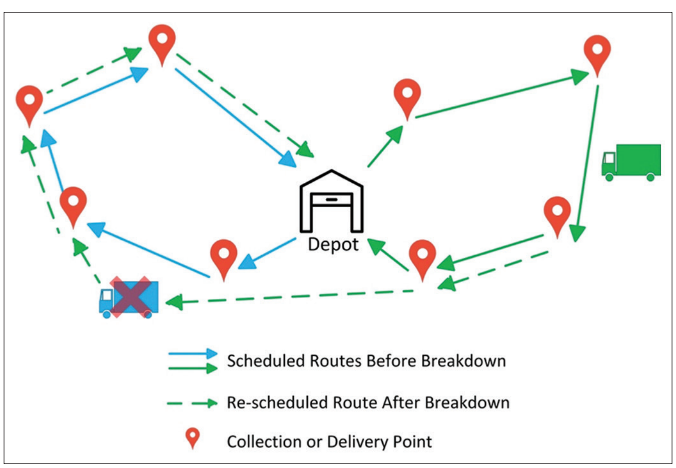

Figure 1: Problem visualization

Considering a case with more than 2 vehicles complicates the problem even more, especially when this unpredicted disruption of more than 1 vehicle happens continuously overtime, deciding which and how many of the operating vehicles to perform the visit to the disrupted vehicle(s) as well as how many of the load to burden given the original problem constraints. Further complications will arise if the breakdown event occurs again after producing the new rearranged routes.

PREVIOUS WORK ON DVRP

Since the evolvement of DVRP from the static and traditional problem, it has focused mainly on the source of dynamism from the customer's side, meaning orders that are revealed dynamically and require immediate replanning. An example of such a problem can be seen in Gendreau *et al.*,[11] where they adopted an insertion heuristic to accommodate newly revealed customers, and TS would improve the overall solution. A good review of dynamic order VRP has been provided by Pillac *et al.*[9] and their adopted solution strategies. This section, however, will provide studies of VRPs or similar that encountered vehicle disruptions.

A few studies have been conducted on VRP under breakdown; however, they are still limited in terms of how these disruption events are introduced and modeled. Vehicle breakdown was first seen in a scheduling context in studies done by Li *et al.*[12] and Mirchandani and Borenstein[13] as vehicle rescheduling problem (VRSP). They considered that the disrupted vehicle has to be visited by another vehicle to collect its load. They proposed a decision support system (DSS) for human schedulers to follow when a disruption occurs. The DSS generates possible feasible networks then selects the most suitable ones, in terms of delay minimization, using an auction algorithm. In their later work,[14] they adopted a Lagrangian relaxation approach along with column generation (CG) that resulted in better solutions compared to the previously developed DSS. The same Lagrangian approach was also applied to a simple routing problem (VRP).[15] However, their latest proposed approach faced performance limitation when applied to large-scale problems. Furthermore, they only introduced a vehicle disruption once during the model run and were predetermined earlier, making it not dynamic.

Pandi *et al.*[15] introduced a breakdown problem to a Dial-a-Ride problem, where people are transported from pickup to delivery locations, and proposed a graphics processing unit computed extensive neighborhood search that is triggered every time a breakdown occurs to minimize the fleet size. The approach resulted in improved vehicle utilization and reduced operational costs under disruption with efficient computing compared to CPU-based approaches.

Mu *et al.*[16] based their work on Li *et al.*;[15] however, it deals with delivering a single commodity that does not require a visit to the disrupted vehicle. They adopted a method of heuristic insertion of the disrupted customers, resulted from the disrupted vehicle, and then improved the solution using TS. Compared to the exact method, the approach provided a slight cost increase to minimize the number of vehicles, and total distance traveled from the optimal solution. However, the problem understudy was also limited to 1 time pre-specified vehicle disruption.

Minis also adopted another heuristic/meta-heuristic approach, Mamasis and Zeimpekis,[17] for a VRP under breakdown with an objective adopted from the team orienteering problem that aims at maximizing customers served. Disrupted customers are sought to be routed by an insertion heuristic then compared to an extensive genetic algorithm (GA) approach that allows more time to generate feasible solutions. In a later work,[18] the authors consider a single commodity delivery with a possibility of visiting the disrupted customers for load replenishment. In both papers, the only heuristic approach has slightly costly deviation compared to develop GA making the heuristic approach more responsive. However, an assumption has been made that only one breakdown occurs and is pre-specified, and accordingly, the optimization is done once per problem.

Van der Merwe *et al*.[19] studied the problem of wildfire responses in the form of VRP under a vehicle breakdown to maximize coverage and minimize deviations from the original plan. They proposed a biobjective Mixed Integer Programming (MIP) adapted to the problem after the breakdown and limited the solution time for 30 min. Experimental problems with 30, 40, 50, and 60 nodes were considered, each with 10 vehicles. Solution for problem sizes of 40 and above could not be produced as MIP need more time to search for the optimal solution. The MIP approach is limited, and, in the case of multiple random breakdowns, the method would not be applicable.

Seyyedhasani and Dvorak[20] tackled a VRP problem in an agricultural context where land is needed to be harvested using vehicles, and the aim is to minimize the time needed. They considered a problem with three vehicles with a breakdown introduced in one of them at 25%, 50%, or 75% completion of the land. TS is considered an initial solution provided by a saving heuristic.[21] Although the approach provided an optimal solution, the problem considered is small, and an assumption of pre-specifying when the breakdown would occur limits the problem. Moreover, the solution adopted is restricted to such 1 time optimization per problem.

It is necessary to differentiate between vehicle routing and scheduling problems. Scheduling explicitly specifies when a vehicle should arrive at a node while routing does not.[22] This specification makes the routing problem much complex as it searches for a more extensive solution space. All the papers mentioned previously are routing problems (VRP), except for Borenstein and Mirchandani,[12] Li *et al*.,[13] and Li *et al*.,[14] as they are rescheduling problems (VRSP).

Van Lieshout *et al*.[23] extended the work in Li *et al*.[14] for VRSP by "softening" the time window constraint to maximize the served customers. They proposed iterative neighborhood exploration, resulting in 60% reduction in orders cancellation. However, the problem under study is still limited to one predetermined breakdown.

Guedes and Borenstein[24] also considered VRSP, however, with multiple depots and heterogeneous vehicles aiming to minimize travel costs and the deviations from the original plan due to driver's familiarity with the routes. A heuristic framework approach has been proposed with truncated CG that does not repeatedly generate columns when the solution is not improving. The approach was tested on large-sized problems (up to 2500 nodes) then applied on a real-life bus transit case study and produced a good quality solution for such a large-sized problem in <3 min. On the other hand, the event of disruption is explicitly specified at a time of the day. The disruption affects up to three vehicles simultaneously.

Dávid and Krész[25] were the only authors to consider the vehicles' disruption events to be fully randomized across the whole period of the modeling, making the problem fully dynamic. The problem considered is a scheduling problem (DVRSP) routing aiming to minimize the deviation from the original plan. They studied two heuristic approaches, recursive search and local search, and they both resulted in solutions less than 15% in plan deviations. Although the problem considered random vehicles' breakdown over the operating time horizon, it is a scheduling problem and not routing.

Based on this review, all the previous studies only consider 1 time rescheduling or rerouting per problem and do not consider continuous time optimization for any random breakdown instant introduced to the problem this paper addresses.

AGENT-BASED IN DVRP

Most of the tools and techniques used in VRP are adopted from the traditional OR techniques: Exact, heuristics, and meta-heuristics.[26] On the other hand, Fischer *et al*.[27] argued that such static methods are not suitable to the dynamic problem and not flexibility adapted due to the need to react to system events while running; they propose the ABM to DVRP.

Mes *et al*.[28] further supported the use of ABM in DVRP, arguing that the traditional OR techniques are sensitive to newly emerged information and time consuming. ABM is a powerful tool because it models the entities of the system as agents that have rational sense and autonomy for making rule-based decisions.[29]

ABM has been applied previously in DVRP. Kuhn *et al.*[30] were the first to adopt such an approach in this problem context with agents of orders and trucks governed by a bidding rule. Their work has been taken further by different authors. Fischer *et al*.[27] introduced cooperation rule across companies (depots) for order and capacity sharing. Kohout and Erol[31] added a verifier agent and adopted Solomon's insertion heuristic and bidding rule. Zeddini *et al*.[32] adopted a continuous optimization strategy in which the model reacts instantly when a dynamic event occurs. Mes *et al*.[28] adopted a more decentralized approach by involving companies (depots) less in the bidding process or rule. Barbucha and Jędrzejowicz[33,34] developed an agent architecture that has a solution manager as an agent and adopted a heuristic approach for optimization, moreover, in their later work Barbucha[35] and Barbucha,[36] they adopted meta-heuristics along with ABM. Barbucha[37] proposed a multiagent system for DVRP with a meta-heuristic improvement approach of guided local search. Vokřínek *et al.*[38] applied prioritizing rules on the arrived customer orders considering the capacity constraint, while in a later work,[39] rule based on time window has been applied. Maciejewski and Nagel[40] adopted a more centralized ABM approach where routing is optimized centrally by an optimizer agent. Gath *et al*.[41] involved depth-first branch-and-bound method within each vehicle agent as a TSP for bid calculation. Finally, Nambiar and Indicula[42] proposed a different architecture of agents; data agent that deals with information update in the system, control agent who deals with the optimization using both k-means clustering and ant colony optimization, in addition to the vehicle agents that execute routes.

All of the ABM approaches proposed for DVRP previously are limited to orders as the only source of dynamism in the system. No one has adopted the approach in a dynamic vehicle disruption context.

MULTILAYERED AGENT-BASED HEURISTIC SYSTEM

This section explains the proposed methodology to solve the DVRP under vehicle disruption. The main challenge here is how to deal with such disruptions and is addressed by developing a modeling framework that is flexible enough to accommodate the resource changes.

Input-Process-Output (IPO)

Before developing the framework, an IPO model has been adopted to define the scope of the model first. The IPO is mainly concerned about the given problem inputs and the type of process performance indicators represented in the output. Figure 2 provides a general overview of such IPO considering the two-layered ABM that will be explained in the following subsection.

Agent input data will be provided from the inputs to determine each agent's unique attributes. First, order agents will have a demanded quantity, location, time window, and servicing time and type (collection/delivery). Second, vehicle agents will have a specific type, constrained capacity, home depot location, and availability from the operating shifts. Finally, drivers have a specific skill that dictates which type of vehicle to drive, home location, availability, and previous experience. Each agent has a unique ID. The inputs will be fed first to the first agent-based layer where a representative agent instance of each order, vehicle and driver will be initiated, then a set of rules will govern how drivers will be assigned to the vehicle, and another set of rules will dictate how orders will be allocated to the previously assigned vehicles and drivers. Such rules will be explained later in this paper. This layer will result in clusters; each consists of orders with one driver and one vehicle.

Clusters data resulting from the first layer will be used to initiate cluster agents in the second. This layer will be concerned about routing orders and optimizing the allocation across other cluster agents based on a predetermined set of rules. This layer will calculate the time advancement in the model by marking which orders have been served. Provided the nature of the problem, the ABM layer 2 is assigned to model the random and continuous disruptions of vehicles during their run. The framework in the following subsection will deal with such a disruption event.

The output of the second layer and the process output will be the clustering solutions and key performance indicators (KPI) to evaluate the developed approach. The output would be customers grouping and sequencing, the allocation of drivers and vehicles, in addition to KPIs: Number of vehicles utilized, each vehicle utilization, each driver idle time, and the unsatisfied customers.

ABM Framework Under Vehicle Breakdown

When a vehicle faces a breakdown, it has to be withdrawn from the model for the breakdown period. As a result, its previous cluster allocation will be invalid and requires modification. Figure 3 illustrates a proposed modeling framework specializing in dealing with breakdown and its required modeling changes.

It has been mentioned in the previous section that breakdowns are faced in the second layer during the model time progression. This breakdown needs to be dealt with by updating the corresponding vehicle agent in the first layer, represented by the red arrow, to update the cluster agent that utilizes the disrupted vehicles seen by the blue dotted arrow between the first layer and the cluster agent. This latest update will be concerned about changing the cluster's optimization strategy from a two-way exchange strategy, which allows it to give and take orders to/from other clusters before the breakdown to only one way allows it to give its orders to other clusters. This strategy change will be put into action once reported to the second layer of the ABM, represented by the blue dotted arrow from the cluster agent. Figure 4 illustrates how the implementation is done for the two strategies.

Both exchange policies are governed by a set of optimizing and predefined rules. If the cluster is emptied from orders, the second layer will report to the cluster agent to be deactivated, represented by the blue dotted arrow on the right in Figure 3. If the cluster was not emptied, it would remain in the second layer with a one-way exchange policy for possible future consideration of its orders by other clusters, and in case, these remaining orders are not allocated, they noted at unserved. After considering the reallocations, the model is ready to be rerun and serve the new allocation form. The model will also be flexible enough to accommodate another random vehicle breakdown event.

Sequence Messaging

To implement the ABM model, sequencing of agents' collaboration is needed, and its logic can be illustrated using a UML sequence diagram, also known as agent messaging.[43] The messaging diagram for the developed model is shown in Figure 5.

The super-agent initiates the sequence by allowing vehicles to seek drivers and then receiving driver assignment. Accordingly, possible orders allocations are sought then the

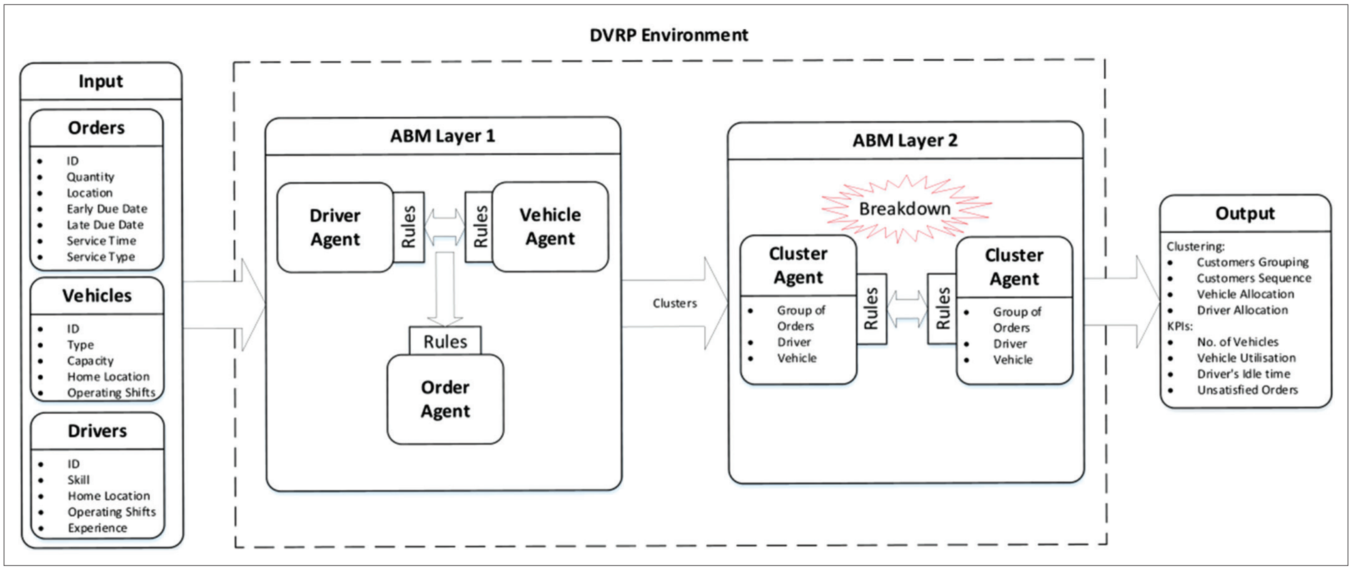

Figure 2: The proposed input-process-output model

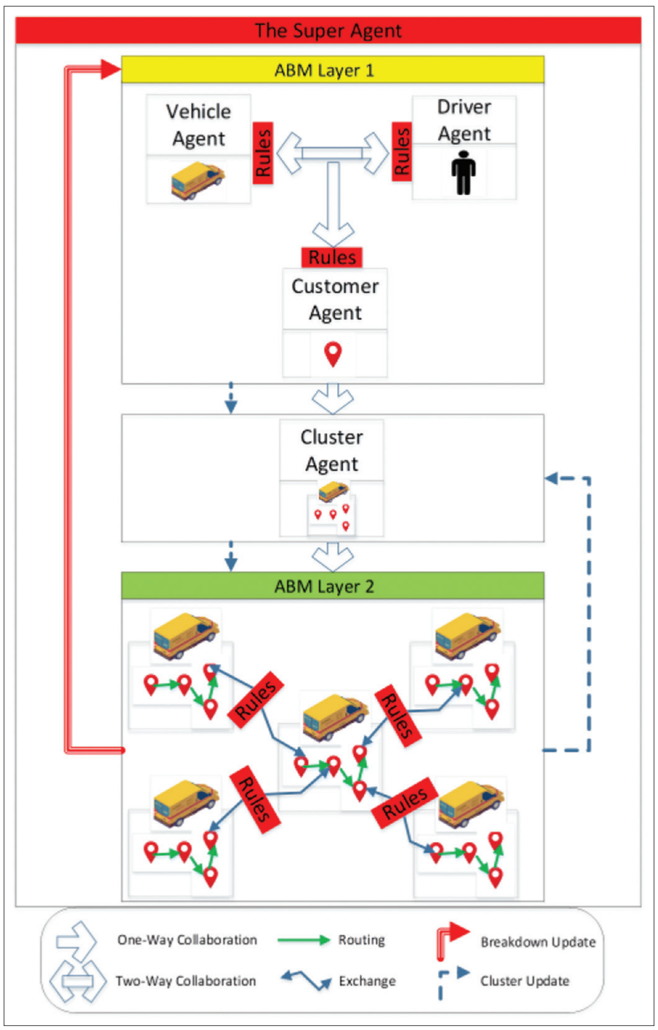

Figure 3: Agent-based model modeling framework under vehicle breakdown

vehicle agent reports the vehicle-driver-orders grouping to the super-agent to initiate the second layer's cluster agents.

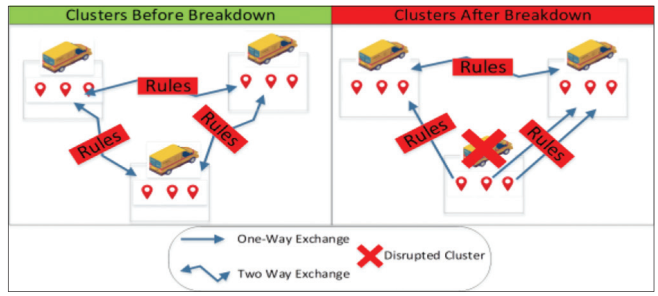

Figure 4: Cluster strategy change before and after a breakdown

On forming the cluster agents, each will perform the two-way exchange for optimization purposes and determine its next order to serve. Once the next node has been approved, the driver agents are notified to drive the vehicle to the order location then perform the order service to be then marked as complete. In case of a random breakdown, a reporting message from the cluster agent to the super-agent to dismiss the vehicle and driver agents affected by the disruption, in addition to cluster policy change to one-way exchange.

Agent Collaboration Rules

It has been previously stated that a set of rules will be governing the decisions in this ABM framework. This section will explicitly state the proposed rules categorized based on the stage of its use. Table 1 shows the proposed ABM rules in each of their categories. This work is still ongoing research; therefore, ABM rules are still not fully developed to comprehend the logic of solving the dynamic vehicle VRP.

The first category of rules is the allocation rules that deal with allocating drivers to vehicles. Five rules have been proposed: The first rule looks into the driver with the most availability and is allocated first to a vehicle. The second rule seeks vehicles with the most availability to be selected first. The third rule may prioritize experienced high drivers in the allocation due to their route and order servicing familiarity.

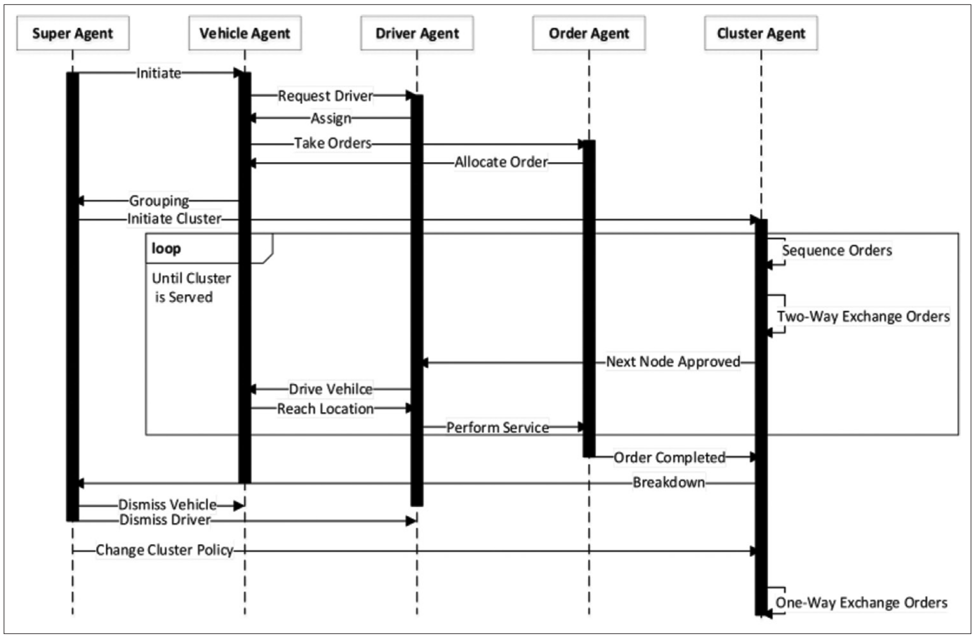

Figure 5: Agents messaging sequence

Table 1: Rules categories

Allocation rules	1. Driver with longest operating shift
	2. Vehicle with longest operating shift
	3. Driver with the most experience
	4. Driver with the earliest operating shift
	5. Vehicle with the earliest operating shift
Clustering rules	1. Earliest early time window
	2. Nearest vehicle-driver
	3. Both with early time window priority
Routing rules	1. Earliest early time window
	2. Nearest order
	3. Earliest arrival time

The last two rules in allocation are trying to utilize the earliest operating drivers and vehicles first.

Clustering rules, on the contrary, deal with orders assignment to the vehicle-driver allocations. Such assignment may be based on prioritizing orders with their earliest early time window, location to the vehicle-driver, or both. Adopting both rules can be seen in the pseudocode represented in Figure 6. From this pseudo-code, it can be seen that it prioritizes the order with the earliest early time window to be allocated first to a near vehicle, considering the vehicle's time and capacity constraints. This process will be repeated until either the orders are assigned or the vehicles are fully utilized.

Notations:
- oi: ith order
- qi: ith order demanded quantity
- (ei, li): ith order time window
- O: orders list

Step1: Sort O ascending based on their earliest time-window
Step2: For each o_i in O
________Sort V ascending based on vehicles distance from o_i
__________For each v_i in V
___________If $ca_i + q_i <= c_i$ and $ei < b_i$
_____________Allocate o_i to v_i and update $ca_i = ca_i + q_i$.
_____________Stop V loop

Figure 6: Clustering rule based on earliest time window and distance

- vi: ith vehicle-driver
- ci: ith vehicle constraint capacity
- cai: ith vehicle allocated capacity
- (ai, bi): ith vehicle-driver operating shift
- V: vehicle-driver entities list

Routing rules are concerned with making decisions with the cluster agent to serve next. A decision could be based on the earliest early time window, nearest to the current driver-vehicle location, or with the earliest possible arrival time. The latest rule is represented by a pseudocode shown in Figure 7, where it performs calculations of possible arrival time for each order in the cluster list then sorting them based on the earliest arrival time. The rule then allocates the earliest order arrival if it satisfies the time window constraint.

Notations:
- oi: ith order
- (ei, li): ith order time window
- Oc: remaining orders in the cluster list
- to: time needed to reach the ith order
- ct: current time
- arri: arrival time at the ith order

As this is still ongoing research, more work has to be done to develop rules in each category. Another category will be added concerned with the exchange rules among cluster agents.

EXPERIMENTATION AND RESULTS ANALYSIS

The ABM system has been partially developed, without vehicle breakdown, representing only order and vehicle agents using Python to represent these agents. Only one rule has been tested: The pseudocode shown in Figure 6 represents orders allocation with early time window and location priorities.

Two scenarios have been randomly generated; the first is with 200 customers while the other is 500. Both problems have 25 vehicles. Performing experiments with many customers is to validate the developed rule. The problem map of both of the problems is represented in Figures 8 and 9, respectively. For each problem, an allocation solution is provided by generating a color-coded allocation map where each color represents a group of orders and one vehicle, number of customers allocated to vehicles chart and vehicle capacity utilization chart.

Step-1: For each o_i in O_c
__________$arr_i = ct + t_o$
Step-2: Sort O_c ascending based on their arrival time arr_i
Step-3: For each o_i in O_c
__________if $e_i <= arr_i <= l_i$
___________Assign as the next order to serve
___________Stop O_c loop

Figure 7: Routing rule based on earliest arrival time

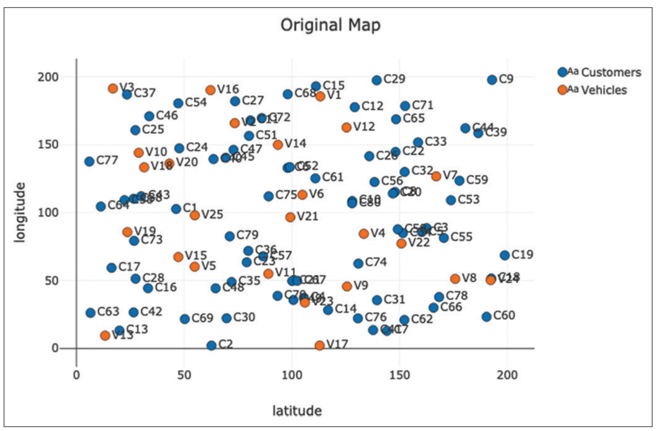

Figure 8: Problem 1 map

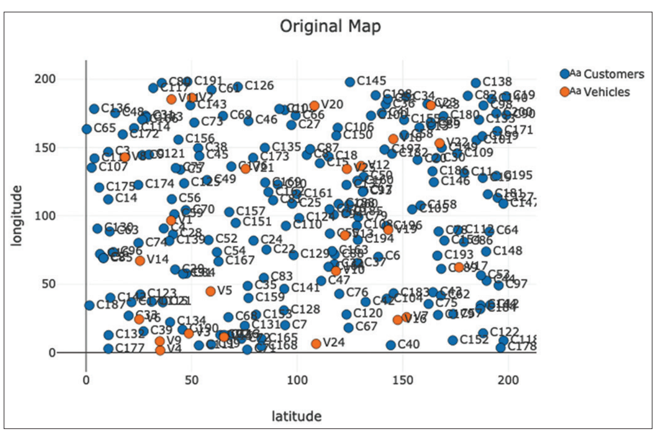

Figure 9: Problem 2 map

For Problem 1, the system has generated an allocation solution shown in the color-coded map (Figure 10). It can be seen that this rule works successfully in providing reasonable allocation of orders concerning their location, although the rule prioritizes the early time window first. All customers have been allocated for this particular problem, and there are no missing customers.

Figure 11 illustrates how many orders have been allocated to each vehicle. This allocation indicates the amount of workload needed from each vehicle, given that the vehicle has to travel across all its allocated customers. For example, Vehicle 1 has been allocated five customers meaning it might be obliged to travel 6 times given that it should end at the depot.

Figure 12 shows the capacity utilization of each vehicle. Since each customer has a specific demand to be either collected or delivered; hence, it is vital to consider the vehicle's capacity in the allocation process. It can be seen that most of the vehicles are highly utilized, above 60%, while only three vehicles have utilization below 60%. This utilization can be judged to be a higher quantity loaded problem.

Contrary to the previous problem, the number of customers generated is increased to 500 to test the validity of the proposed clustering rule. Figure 9 shows the problem

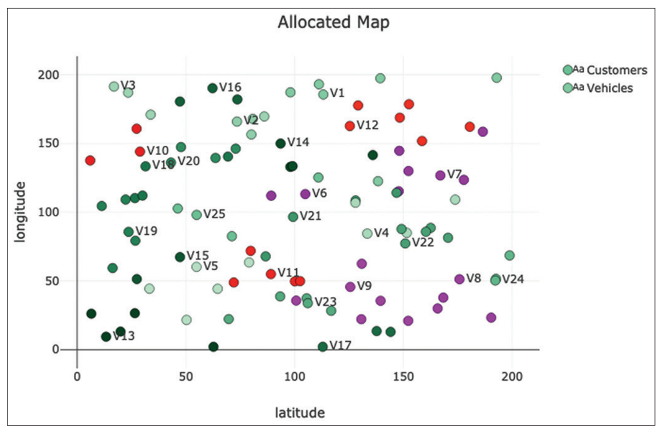

Figure 10: Problem 1 allocated map

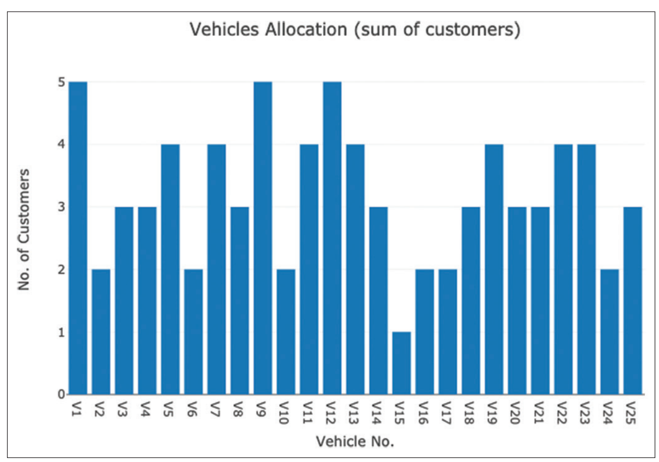

Figure 11: Problem 1: Number of customers allocated to each vehicle

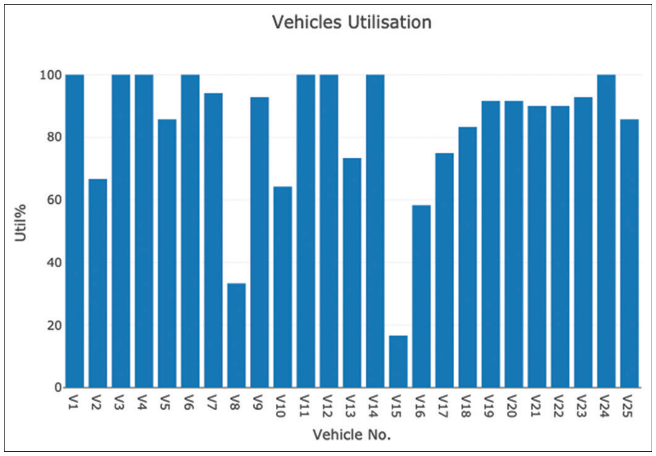

Figure 12: Problem 1: Each vehicle capacity utilization

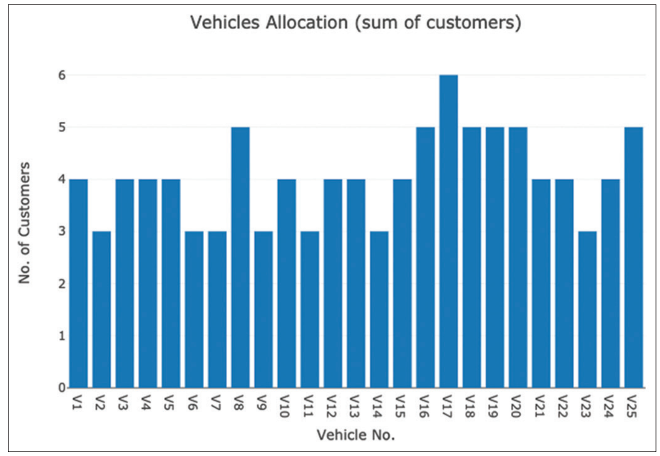

Figure 14: Problem 2: Number of customers allocated to each vehicle

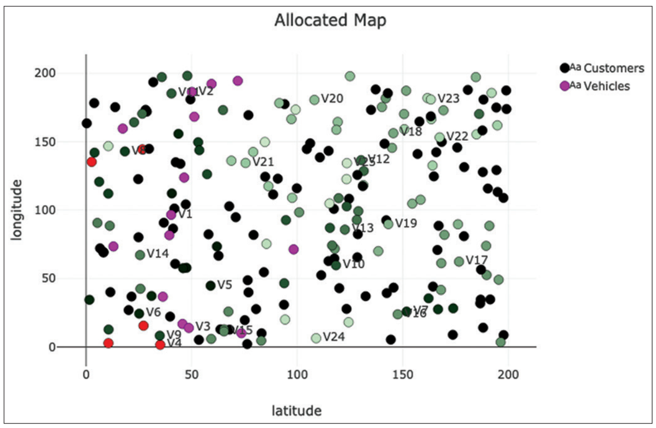

Figure 13: Problem 2 allocated map

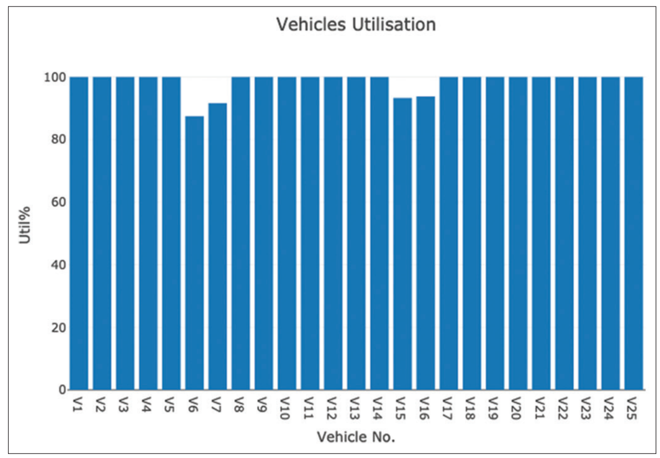

Figure 15: Problem 2: Each vehicle capacity utilization

2 map, which is highly dense due to the over generation of customers within this limited map setting.

Figure 13 illustrates the allocation of each order to which vehicle. It can be noticed that there are 99 unallocated customers, represented in black, this is explained because of the massive customers' increase, and the number of vehicles is the same compared to the previous problem.

The number of allocated customers per vehicle has increased, as shown in Figure 14, as the increased demand pushed the clustering rule for more allocations to vehicles.

For each vehicle capacity, the vehicles in this problem have been pushed to nearly 100% capacity utilization. Figure 15 shows the utilization of each vehicle in this problem. Only four vehicles did not reach a full 100% utilization.

It can be noted that the proposed clustering rule is functioning as indented within the partially developed ABM model with the only customer and vehicle agents. It provides simple allocation considering the capacity and time constraints of the vehicles and their possible adoption to different problem sizes by rejecting the customers over the overall capacity.

CONCLUSION AND FUTURE WORK

This paper adopted the ABM approach to propose a framework that deals with VRP under dynamic vehicle breakdowns in continuous time. Modeling this problem is a promising opportunity to be more agile and responsive to dynamic, disruptive events in transportation and logistics.

The proposed ABM model consists of two layers: The first is to generate clusters consisting of group of order agents allocated to vehicle and driver agents, the second deal with running the model, routing, and optimization by representing the clusters using clusters agents. A set of rules has also been proposed to govern the agents' interaction in the model. As this work is still ongoing research, the model proposed here is partially developed that represent only vehicle and customers agents without breakdowns, and only one rule, a clustering rule, has been programmed and tested on two different problem scenarios of different numbers of customers, 200 and 500. The rule has produced good order allocation results by utilizing the vehicles time availability and capacity and neglecting customers who are considered over the overall capacity. The resulted average customer allocation is 3.2 for the small problem compared to 4.04 for the bigger problem. On the other hand, vehicle utilization is less the small problem, with only 8 out of 25 are fully utilized compared to 21 in the bigger problem.

Future work aims to develop the ABM model by further programming the agents with random breakdowns and further rules development to provide routes that are adaptive to the breakdown event optimally.

REFERENCES

1. M. G. Speranza. Trends in transportation and logistics. *European Journal of Operational Research*, vol. 264, no. 3, pp. 830-836, 2018.
2. H. Lasi, P. Fettke, H. G. Kemper, T. Feld and M. Hoffmann. Industry 4.0. *Business and Information Systems Engineering*, vol. 6, no. 4, pp. 239-242, 2014.
3. J. Gubbi, R. Buyya, S. Marusic and M. Palaniswami. Internet of things (IoT): A vision, architectural elements, and future directions. *Future Generation Computer Systems*, vol. 29, no. 7, pp. 1645-1660, 2013.
4. G. B. Dantzig and J. H. Ramser. The truck dispatching problem. *Management Science*, vol. 6, no. 1, pp. 80-91, 1959.
5. G. Laporte. The vehicle routing problem: An overview of exact and approximate algorithms. *European Journal of Operational Research*, vol. 59, pp. 345-358, 1992.
6. J. K. Lenstra and A. H. G. Kan. Complexity of vehicle routing and scheduling problems. *Networks*, vol. 11, no. 2, pp. 221-227, 1981.
7. M. M. Solomon. Algorithms for the vehicle routing and scheduling problems with time window constraints. *Operational Research*, vol. 35, no. 2, pp. 254-265, 1987.
8. M. Schneider. The vehicle-routing problem with time windows and driver-specific times. *European Journal of Operational Research*, vol. 250, no. 1, pp. 101-119, 2016.
9. V. Pillac, M. Gendreau, C. Guéret and A. L. Medaglia. A review of dynamic vehicle routing problems. *European Journal of Operational Research*, vol. 225, no. 1, pp. 1-11, 2013.
10. P. C. Guedes and D. Borenstein. Real-time multi-depot vehicle type rescheduling problem. *Transportation Research Part B: Methodological*, vol. 108, pp. 217-234, 2018.
11. M. Gendreau, F. Guertin, J. Y. Potvin and É. D. Taillard. Parallel tabu search for real-time vehicle routing and dispatching. *Transportation*, vol. 33, no. 4, pp. 381-390, 1999.
12. J. Q. Li, D. Borenstein and P. B. Mirchandani. A decision support system for the single-depot vehicle rescheduling problem. *Comput. Oper. Res.*, vol. 34, no. 4, pp. 1008-1032, 2007.
13. J. Q. Li, P. B. Mirchandani and D. Borenstein. The vehicle rescheduling problem: Model and algorithms. *Networks*, vol. 50, no. 3, pp. 211-229, 2007.
14. J. Q. Li, P. B. Mirchandani and D. Borenstein. A Lagrangian heuristic for the real-time vehicle rescheduling problem. *Transportation Research Part E: Logistics and Transportation*, vol. 45, no. 3, pp. 419-433, 2009.
15. J. Q. Li, P. B. Mirchandani and D. Borenstein. Real-time vehicle re-routing problems with time windows. *European Journal of Operational Research*, vol. 194, no. 3, pp. 711-727, 2009.
16. R. R. Pandi, S. G. Ho, S. C. Nagavarapu, T. Tripathy and J. Dauwels. Disruption management for dial-a-ride systems. *IEEE Intelligent Transportation Systems Magazine*, vol. 12, no. 4, pp. 219-234, 2020.
17. Q. Mu, Z. Fu, J. Lysgaard and R. Eglese. Disruption management of the vehicle routing problem with vehicle breakdown. *Journal of the Operational Research Society*, vol. 62, no. 4, pp. 742-749, 2011.
18. I. Minis, K. Mamasis and V. Zeimpekis. Real-time management of vehicle breakdowns in urban freight distribution. *The Journal of Heuristics*, vol. 18, no. 3, pp. 375-400, 2012.
19. K. Mamasis, I. Minis and G. Dikas. Managing vehicle breakdown incidents during the urban distribution of a common product. *Journal of the Operational Research Society*, vol. 64, no. 6, pp. 925-937, 2013.
20. M. Van der Merwe, M. Ozlen, J. W. Hearne and J. P. Minas. Dynamic re-routing of vehicles during cooperative wildfire response operations. *Annals of Operations Research*, vol. 254, no. 1-2, pp. 467-480, 2017.
21. H. Seyyedhasani and J. S. Dvorak. Dynamic re-routing of a fleet of vehicles in agricultural operations through a dynamic multiple depot vehicle routing problem representation. *Biosystems Engineering*, vol. 171, pp. 63-77, 2018.
22. G. Clarke and J. W. Wright. Scheduling of vehicles from a central depot to a number of delivery points. *Operational Research*, vol. 12, no. 4, pp. 568-581, 1964.
23. L. Bodin and B. Golden. Classification in vehicle routing and scheduling. *Networks*, vol. 11, no. 2, pp. 97-108, 1981.
24. R. Van Lieshout, J. Mulder and D. Huisman. The vehicle rescheduling problem with retiming. *Operations Research and Computers*, vol. 96, pp. 131-140, 2018.
25. B. Dávid and M. Krész. The dynamic vehicle rescheduling problem. *Central European Journal of Operations Research*, vol. 25, no. 4, pp. 809-830, 2017.
26. G. Laporte. Fifty years of vehicle routing. *Transportation Science*, vol. 43, no. 4, pp. 408-416, 2009.
27. K. Fischer, J. Müller and M. Pischel. Cooperative transportation scheduling: An application domain for DAI. *Applied Artificial Intelligence*, vol. 10, no. 1, pp. 1-34, 1996.
28. M. Mes, M. van der Heijden and A. van Harten. Comparison of agent-based scheduling to look-ahead heuristics for real-time transportation problems. *European Journal of Operational Research*, vol. 181, no. 1, pp. 59-75, 2007.
29. M. Wooldridge and N. R. Jennings. Agent theories, Architectures, and languages: A survey. In: M. J. Wooldridge and N. R. Jennings, (Eds.), *Intelligent Agents*. Vol. 890. Berlin, Heidelberg: Springer, pp. 1-39, 1995.
30. N. Kuhn, J. P. Müller and J. Müller. Simulating cooperative transportation companies. In: J. Biethahn, W. Hummeltenberg, B. Schmidt and T. Witte, (Eds.), *Simulation als Betriebliche Entscheidungshilfe*, Wiesbaden: Vieweg+Teubner Verlag, pp. 263-274, 1994.
31. R. Kohout and K. Erol. *In-time Agent-based Vehicle Routing with a Stochastic Improvement Heuristic*. Menlo Park, California: American Association for Artificial Intelligence, pp. 864-869, 1999.
32. B. Zeddini, M. Temani, A. Yassine and K. Ghedira. An Agent-Oriented Approach for the Dynamic Vehicle Routing Problem. In: *2008 International Workshop on Advanced Information Systems for Enterprises*, pp. 70-76, 2008.
33. D. Barbucha and P. Jedrzejowicz. An agent-based approach to vehicle routing problem. *International Journal of Computer and Information Technology*, vol. 1, no. 2, p. 6, 2007.
34. D. Barbucha and P. Jędrzejowicz. Agent-Based Approach to the Dynamic Vehicle Routing Problem. Berlin, Heidelberg: *7th International Conference on Practical Applications of Agents and Multi-Agent Systems* (*PAAMS 2009*), pp. 169-178, 2009.
35. D. Barbucha. Agent-based guided local search. *Expert Systems with Applications*, vol. 39, no. 15, pp. 12032-12045, 2012.
36. D. Barbucha. Experimental Study of the Population Parameters Settings in Cooperative Multi-agent System Solving Instances of the VRP. In: N. T. Nguyen, (Ed.), *Transactions on Computational Collective Intelligence I* × . Vol. 7770. Berlin, Heidelberg: Springer Berlin Heidelberg, pp. 1-28, 2013.
37. D. Barbucha. *Multi-agent Approach to the DVRP with GLS Improvement Procedure*. Singapore: Intelligent Decision Technologies, pp. 117-126, 2020.
38. J. Vokřínek, A. Komenda and M. Echouček. Agents towards Vehicle Routing Problems. In: *Proceedings of the 9th International Conference on Autonomous Agents and Multiagent Systems*, pp. 773-780, 2010.
39. P. Kalina and J. Vokřínek. Parallel Solver for Vehicle Routing and Pickup and Delivery Problems with Time Windows Based

on Agent Negotiation. *2012 IEEE International Conference on Systems Man and Cybernetics*, pp. 1558-1563, 2012.

40. M. Maciejewski and K. Nagel. Towards Multi-Agent Simulation of the Dynamic Vehicle Routing Problem in MATSim. In: R. Wyrzykowski, J. Dongarra, K. Karczewski and J. Waśniewski, (Eds.), *Parallel Processing and Applied Mathematics*. Vol. 7204. Berlin, Heidelberg: Springer Berlin Heidelberg, pp. 551-560, 2012.
41. M. Gath, S. Edelkamp and O. Herzog. Agent-based Dispatching in Groupage Traffic. Singapore, Singapore: *2013 IEEE Symposium on Computational Intelligence in Production and Logistics Systems* (*CIPLS*), pp. 54-60, 2013.
42. S. K. Nambiar and S. M. Idicula. A Multi-agent Vehicle Routing System for Garbage Collection. In: *2013 5th International Conference on Advanced Computing* (*ICoAC*), Dec, pp. 72-76, 2013.
43. A. Pan, S. Y. S. Leung, K. L. Moon and K. W. Yeung. Optimal reorder decision-making in the agent-based apparel supply chain. *Expert Systems with Applications*, vol. 36, no. 4, pp. 8571-8581, 2009.