Journal of Engineering Science and Technology
Vol. 15, No. 3 (2020) 1972 - 1983

SOLVING LATE FOR RELIEF EVENT IN BUS CREW RESCHEDULING USING MULTI AGENT SYSTEM

AHMAD FADZLI NIZAM ABDUL RAHMAN[1],
ABDUL SAMAD SHIBGHATULLAH[2]*, ZURAIDA ABAL ABAS[1],
TILLAL ELDABI[3], CHIT SU MON[2], AMIR 'AATIEFF AMIR HUSSIN[4]

[1]Optimization, Modelling, Analytics and Simulation (OptiMAS), Centre of Advanced Computing Technology, Fakulti Teknologi Maklumat dan Komunikasi, Universiti Teknikal Malaysia Melaka (UTeM), Hang Tuah Jaya, 76100 Melaka, Malaysia
[2]School of Information Technology, Faculty of Business & Information Science, UCSI University, 56000 Kuala Lumpur, Malaysia
[3]Faculty of Arts and Social Sciences, University of Surrey, Guildford GU2 7XH, United Kingdom.
[4]Kuliyyah of Information and Communication Technology, IIUM Gombak, Kuala Lumpur, Malaysia
*Corresponding Author: abdulsamad@ucsiuniversity.edu.my

Abstract

Unpredictable event is an event which happens anytime without notice that will disrupt bus services. Bus crew is one of the causes of the unpredictable event as if a bus crew comes late - s/he will cause certain bus to depart late. In this paper, three types of bus crew lateness are defined which is Late For Sign-On (LFSO), Late For Relief (LFR), and Late For Second Work (LFSW). However, this paper will only discuss the solution for LFR type. When LFR happens, the schedule needs to be rescheduled. Currently, there is no automated mechanism to handle LFR issue especially in Internet of Thing (IoT) environment. Most real time rescheduling approaches are not supported due to static schedules constraint. Mathematical approaches require extensive computational power, therefore delaying real-time results. Manual rescheduling by supervisor is likely to have an errors and not an optimize solution. This paper presents a new approach for rescheduling the bus crew's timetable in the event of LFR. The multi agent system will adapt quickly to the dynamic environments to find the best and optimize solutions. The experiment of LFR is conducted by using the AgentPower simulation tool. The result concluded that the proposed technique can produce quick rescheduling the for bus crew schedule in the event of LFR.

Keywords: Bus crew scheduling, Late for relief, Multi-agent system, Unpredictable event.

1. Introduction

An unpredictable event (UE) can happen at any time because of bus breakdown, bus crew absenteeism, a variation of demand, and temporary traffic congestion [1-4]. Bus crew's discipline is one of the causes of the unpredictable event as it will cause trouble to the bus timetable. In this paper, we will focus to the UE that caused by Late For Relief (LFR). There are two more types of UE from bus crew perspective which is Late For Sign-On (LFSO) and Late For Second Work (LFSW) [4]. When LFR occurred, the current bus crew schedules will be affected. A bus crew schedule is a report of duties that have been pre-assigned to the bus crew according to the bus schedule for a certain scheduling period to avoid delay in bus travelling [5, 6]. The bus operators will be penalised if the delay of the bus is caused by the bus crew problems or mechanical issues faced by the bus operator. If the delay is due to the traffic congestions, they are not usually being penalised [7-9]. Therefore, majority of the bus operators will put more efforts to manage their vehicles maintenance and the bus crew scheduling to avoid any services disruption and penalised by the authority.

One method to overcome UE is through "crew rescheduling" [10-12]. If UE occurs, bus schedules will remain the same, but the assignment of the bus crew will be changed accordingly based on other crew availability. Current bus operator's approaches for UE are based on the static schedules [3] which have incapability of rescheduling in a real time scenario. When UE happened, a new complete schedule is reproduced without concerning the real time situation [13-15]. Therefore, there is still a room of improvement that can be done, which is being highlighted in this paper. The integration part of control, communications, and processing of the information across various transportation system (infrastructures, vehicles and drivers/user) can be easily done by the introduction of Internet of Thing (IoT) [16]. In the process of solving this problem, the definition of LFR needs to be understood.

LFR occurs when the bus crew is late for his/her relief between 15 to 60 minutes time frame due to several reasons such as trapped in traffic congestion or other emergencies. The data for the experiment are taken randomly from anonymous bus company in London based on different schedule - large, medium and small schedule. Two assumptions were made for this experiment based on informal discussions with the bus operator and regulation of driving hour in European Council (EC). First assumption, the maximum driving hours for each bus crew is nine (9) hours in a day and each bus crew is given 45 minutes for relief time. For a second assumption, 5 minutes extra buffer time would be given to each bus crew to get ready for work after sign-on, to start relief, to finished relief, to start second work and to sign-off after work. The second assumption is to avoid any lateness and delay case, but it is not totally a privilege [16]. This paper discussed how MAS can be implemented to solve the LFR event in crew rescheduling.

2. Late For Relief

In the event of LFR, the duties timetable should need reschedule to ensure the availability of bus crew at the bay. The available bus crew could be from a list of crew that finished his/her relief, the signed-on crew but has not started driving, or crew that has finished his/her duty and do not signed-off yet [16]. The existing manual system in case of bus crew late for relief choose another crew that has the starting duty time nearest to the ready time of the late-crew (arrival time plus 5 minutes). This will ensure that the late-crew does not need to wait long and will take over the original crew's duty that have cover him/her. Figure 1 shows the manual system that a supervisor assigned a specific crew to the task due to other crew LFR.

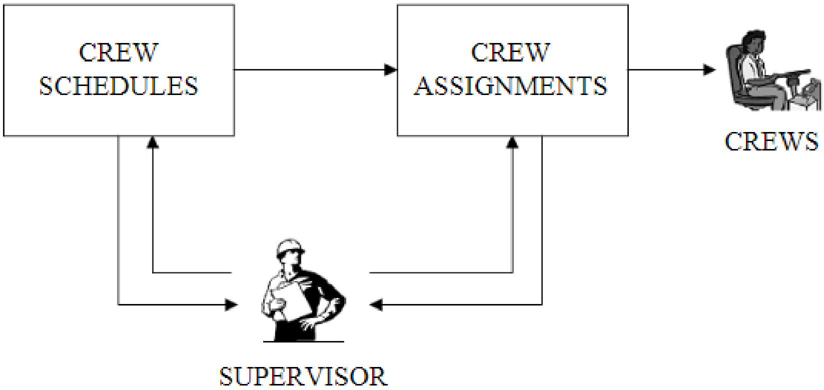

Fig. 1. Manual system to assign specific crew to the task.

The manual process is prone to errors and tedious until the match is found. The best solution with least involvement of crew is accepted as a principle to minimise the effect of rescheduling when more than one rescheduling is needed. As shown in Table 1, crew B who has a duty No 16 is supposed to finish his/her first duty at 11:10. In actual real time, crew B finished his/her first duty at 11:25 which is 15 minutes late and told the supervisor at the bay. By considering the relief time (45 minutes) plus another 5 minutes preparation for each stage from finish duty 1 to the start next duty 2, the supervisor will assign another bus crew to start duty 2.

Table 1. Scheduled time and actual real time for Crew B.

	Finish Duty 1	Start (Relief)	Finish (Relief)	Start Duty 2
Scheduled Time (a) (Duty No 16)	11:10:00	11:15:00	12:05:00	12:10:00
Actual Real Time (b)	11:25:00	11:30:00	12:20:00	12:25:00
Time Difference (b)-(a)	00:15:00	00:15:00	00:15:00	00:15:00

From Table 1, we acknowledged that, crew B will finish his/her relief time at 12:20 and will starts his/her second duty at 12:25. In other words, we can conclude that crew B is 15 minutes late. In this situation, the company supervisor must have assigned another crew to take over original crew B duty 2 scheduled time at 12:10:00 to avoid any delay of the bus. Another crew will be assigned to take over the remaining duty of the crew that took over crew B's duty (if there is a consequences). The outcome of the manual rescheduling done by the supervisor is shown in Table 2(a) and 2 (b). From both tables, we can also see new columns are introduced which is ready time, new finish relief, reassignment and waiting time for late crew. Ready time is the finished relief time minus the buffer time given (5 minutes). New finished relief is the time of next assigned duty minus buffer time. Reassignment is a column that indicate either the crew has another duty to fulfil or not. Lastly, waiting time for late crew is how long the LFR crew late to take over the duty left (if any).

Table 2(a). Rescheduling of LFR for Start Duty 2 (Duty No 16) - First round.

Duty No	Crew ID	Start Relief	Ready Time	Finish Relief	New Finish Relief	Start Duty 2	Need More Reassignment	Waiting Time for Late-Crew
26	H	11:20:00	12:05:00	12:10:00	12:05:00	12:15:00	Y	00:00:00

Table 2(b). Rescheduling of LFR for Start Duty 2 (Duty No 16) - Second round.

Duty No	Cre w ID	Start Relief	Ready Time	Finish Relief	New Finish Relief	Start Duty 2	Need More Reassignment	Waiting Time for Late-Crew
19	E	11:27:00	12:10:00	12:15:00	12:10:00	12:20:00	Y	00:05:00

We can conclude that in order to solve the LFR problem create by crew B, the supervisor at bay needs to have two round of rescheduling involving crew H and crew E. In the first round, only crew H with ready time of 12:05:00 is ready to take over the second duty of crew B. The buffer time for finished relief in crew H is withdrawn. Crew H original duty need to starts his/her duty after relief at 12:15:00 so another crew or crew B need to take over this duty. Since crew B ready time is at 12:15 too so crew B cannot take over duty 2 that is left by crew H. A second round of rescheduling need to be executed which identified crew E to take over duty 2 of crew H. Since the duty 2 for crew E is at 12:20:00, the ideal candidate to take over crew E's duty 2 is crew B. Crew B will have his/her ready time at 12:15:00 and will take over duty 2 of crew E. In conclusion after second round of rescheduling, crew H will take over duty 2 of crew B, crew E will take over duty 2 of crew H and lastly, crew B will take over duty 2 crew E. In practice, factors such as capabilities, past experiences, and common sense of a supervisor will determine the success of this manually crew rescheduling. Sometimes these skills are blended inconsistently in an unorganised, and sometimes are not well-understood [17]. Some bus supervisors as reported in Taiwan [18] usually manage the abnormal conditions in ad-hoc manner by using their knowledge and intuition. It is more or less a common practice for the rest of the world when handling a rescheduling manually. From the literatures also, it is argued that manual crew rescheduling has many deficiencies that leading to difficulty in rescheduling. Thus, in this paper an automated bus crew rescheduling is proposed to overcome these deficiencies during rescheduling. A multi-agent system (MAS) will be used to implement the proposed system.

3.Methods

This section explains the methods for automated bus crew rescheduling system using MAS. MAS is defined as a system that have a population of autonomous agents which interact with each other to reach a common objective, while simultaneously pursuing their individual objectives [19, 20]. MAS has been successfully used in scheduling for several problem domains such as logistics management scheduling, manufacturing scheduling, and meeting scheduling [21-23]. It has become more important in many areas that realised computer science can be very helpful to solve problems. MAS solving the issues of distributed intelligence and interaction. In other words, they represent a new way of analysing, designing, and implementing complex software systems [23, 24]. The general advantages of MAS are extensibility, fault tolerance, scalability and able to capture autonomy in distributed systems and dynamic environment [25, 26]. This research proposes MAS as an approach to do automated crew rescheduling when dealing with UE problems. Agents in MAS can adapt their behaviour dynamically when the environments are changing and a quick solution will be produced through the negotiations and cooperation between these agents. The proposed MAS system is designed to help the supervisors in managing UE and minimising the effect of UE to crew schedules. This will consequently reduce the amount of disruptions to the bus operation. The architecture of MAS for the proposed system has two types of agents that are known as the Crew Agent or CA and the Duty Agent or DA. The basic architecture for crew bus scheduling is shown in Fig. 2.

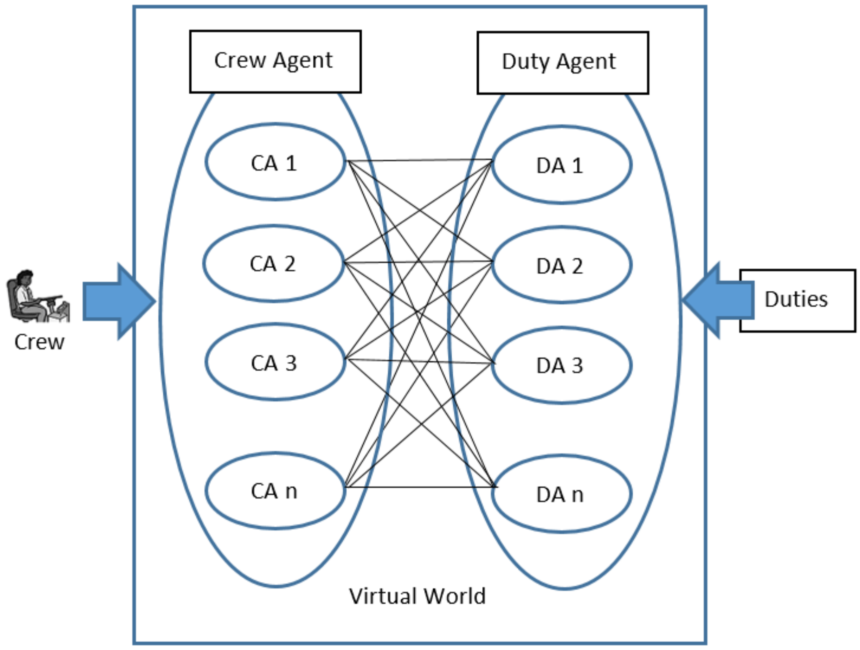

Fig. 2. Basic architecture for crew rescheduling using MAS.

The virtual world is the platform where all the agents can interact, communicate and negotiate. DA will represent all the tasks/duties need to be done and CA represent the available crews that can fulfil the tasks. Each CA and DA is given a dedicated number to distinguish them. In this virtual world, CA agents will have to negotiate with DA agents simultaneously until a satisfactory matching is reached. When agent receives a message, it will use a dedicated pre-set reasoning procedures related to the message in order to get appropriate action or answer based on its domain specific knowledge. MAS attain its objectives from countless interaction messages between these CA agents and DA agents. Each message delivers certain semantics meaning that associated to a specific task and the matching process is initiated by DA. The algorithm of MAS is summarized and presented in the flowchart shown in Fig. 3.

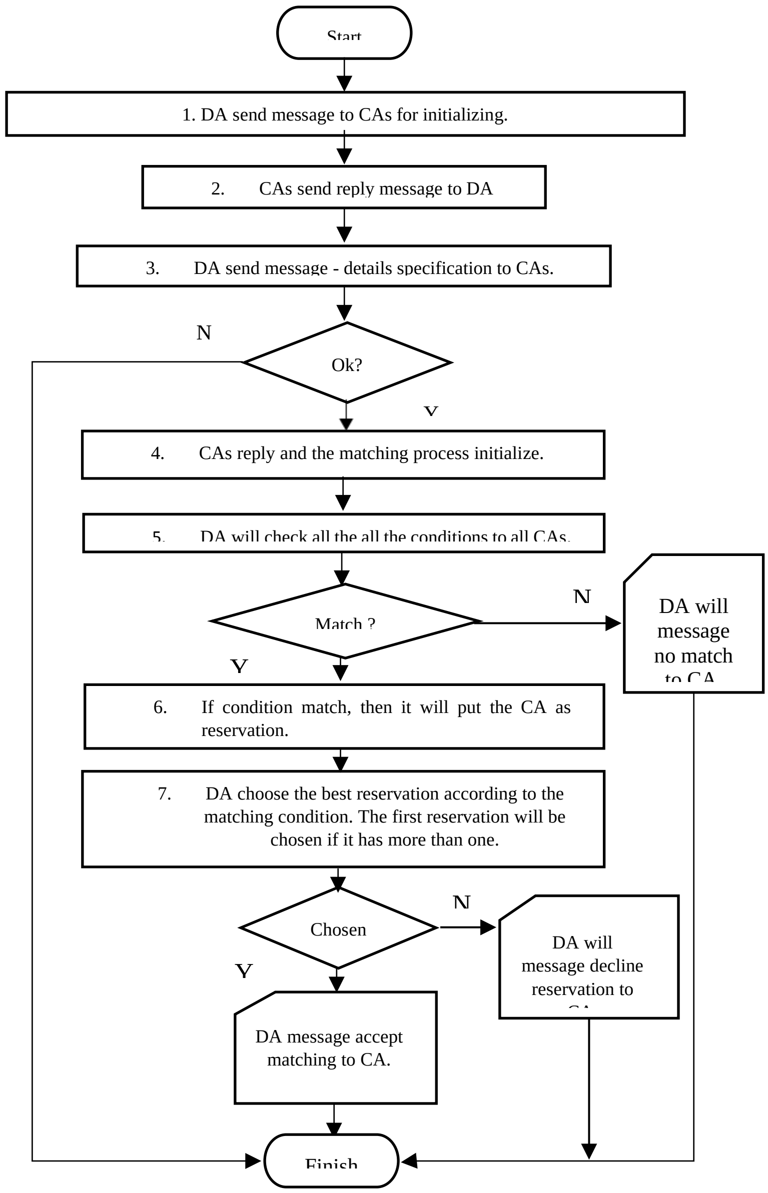

Fig. 3. The flowchart for late for relief in crew rescheduling using MAS.

The experiments will be conducted using AgentPower simulation tool with 90 random data (large, medium and small schedules) collected from the bus company in London. The name of the bus company will be remained anonymous. The MAS is implemented by using AgentPower software. The components in AgentPower consists of Computer Interface, Human Computer Interface, Virtual World Ontology, and Multi-Agent Engine. The agents will communicate, interact, and negotiate in by using virtual relation as shown in Fig. 4. In virtual world in AgentPower software, crew resource is noted as CA and duty demand is noted as DA.

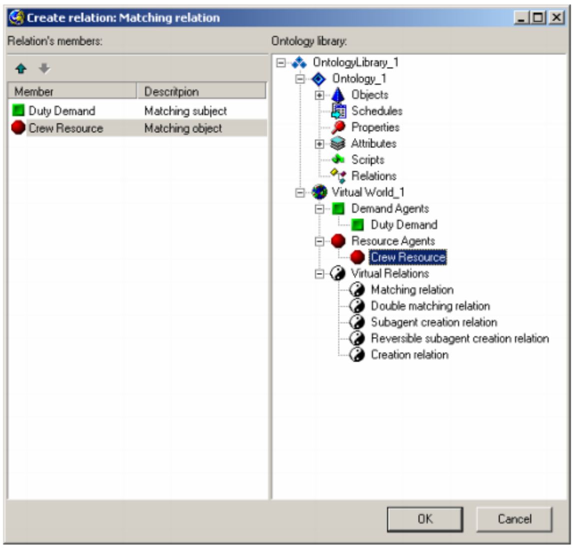

Fig. 4. The resource agent (CA) and demand agent (DA) in virtual world.

A task or work need to be done is represented by DA or demand agent. CA or resource agent will represent the crew that work for the bus company. CA main responsibility is to follow the prescribed schedule to drive a bus. There are five activities for CA in real situation. Sign On is the activity when crew report for a duty. Drive activity is when the crew starting the engine bus and drive. Relief is the activity for time break for crew and usually it will take at least 45 minutes. Sign-Off is activity when crews finish their duty and there no more duty on that day. Lastly, StandBy activity is when the crew is in standby mode. There are a few rules need to be adhere such as the crew cannot drive the bus continuously for more than 4.5 hours. For the time relief, it will must be at least 45 minutes for each crew and lastly the total driving hours in a day should be at most 10 hours. DA's represent all a duties need to be done resulting from UE events and its objective is to find available crew that can take the duty. In the proposed MAS architecture, DAs and CAs interact in a virtual world until a satisfactory matching is attained. Table 3 shown a list of interaction messages that will happen in virtual world.

DA will initialize the matching process by giving a *reqDriver* message to all possible CAs as shown in Figs. 5 and 6. All CAs will respond back to DA with *respond* message when it received the message. Then DA will reply to all CAs with *detailsSpecs* message for the corresponding duty. In return, available CAs for the duty will send back the *beginMatching* message to DA. If CAs is not available, no message will be return back to DA and the interaction stop here. DA will again send *reserved* message to CA that match the condition and put this CA into the reservation list. The process of sending *reserved* message to the other CA that send *beginMatching* message will continue for all CAs. After all potential CAs is in the reservation list, the negotiation will be start and DA will make the decision to choose the best option among the CAs in the reservation list. The chosen CA will get an *acceptMatch* message while the rest CAs in the reservation list will get a *declineReservation* message. Figure 5 shows the activity of interactions between DA and CAs that indicate case of matching scenario is granted.

Table 3. Message interaction between DA and CA.

Message Type	From Sender	To Receiver	Remarks
reqDriver	DA	CA	Message sent whenever a duty needs a crew.
respond	CA	DA	Message sent as soon as a crew received a request from a duty.
detailsSpecs	DA	CA	It conveys information about the details specification of a duty.
beginMatching	CA	DA	Message sent to initiate a negotiation
noMatch	DA	CA	Message sent to inform that there is no match because the crew does not fulfil the duty's requirement.
reserved	DA	CA	Message sent to inform that the crew is reserved to take the duty.
acceptMatch	DA	CA	Message sent to inform that the crew is accepted to take the duty.
declineReservation	DA	CA	Message sent to inform that the crew reservation is rejected because there is other crew that is more suitable to take the duty.

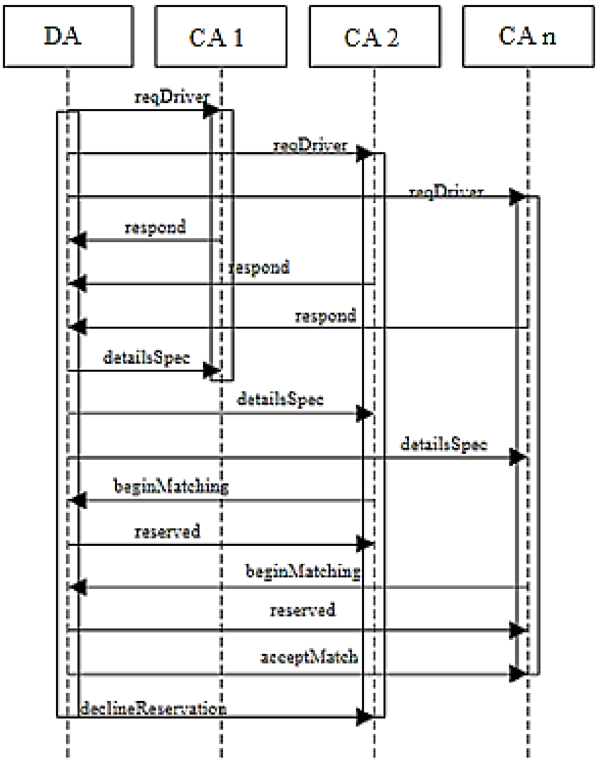

Fig. 5. Sequence of messages when match is found.

Figure 6 in the other hand shows the scenario of no match is found. When this happened, DA will send *noMatch* message to that particular CAs. The process of interaction, communication and negotiation will be done in the virtual world and will involve lots of DA and CA agents. This will be difficult if it is done manually by the supervisor and became worst if the supervisor is not having enough experiences when dealing with UE events.

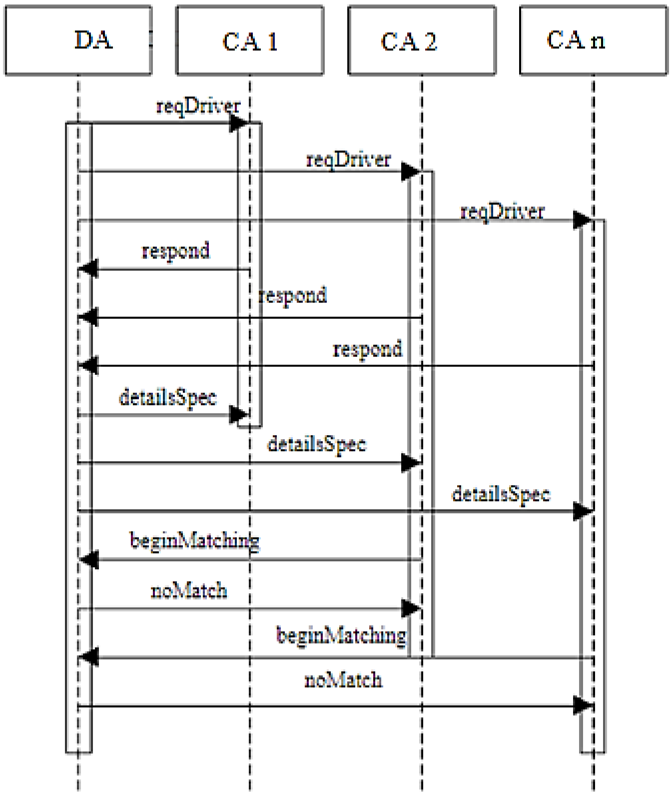

Fig. 6. Sequence of messages when match is not found.

4. Results and Discussion

The matching simulation is done by using AgentPower simulation tool. The rescheduling capability is usually measured the number of matched and the time taken to achieve the match. The best result will definitely be depending on the maximum number of matched found with the lowest time taken. Figure 7 illustrates the experiment results of solving LFR event using MAS in different schedule types (large, medium, and small) and the types of duty distributions (maximum, median, and minimum).

The analysis in Table 4 and Fig. 7 demonstrate that in rescheduling successfully, the major factor is the duty distribution. From the results show in Fig. 7, we can conclude that two cases in large schedule (large-maximum duty, large-medium duty), one case each in medium schedule (medium-maximum duty) and small schedule (small-maximum duty) obtained a 100% success of rescheduling of crew. In the medium schedule (medium-median duty), even though the total experiments and total matched number is the same, it is not considering a success as it does not involve any rescheduling. This is because of the duration time for the relief is enough to absorb the delay. Table 4 will show clearly regarding to this case.

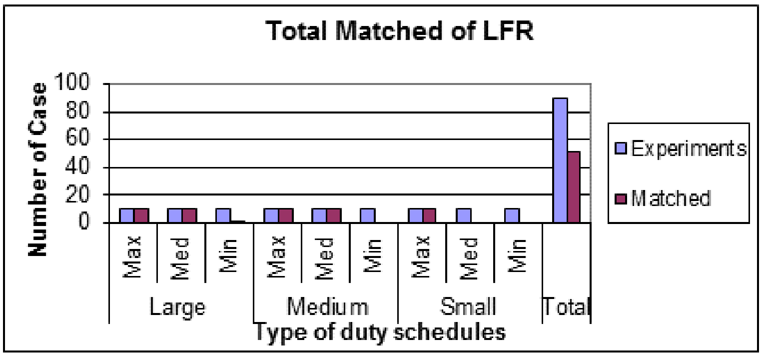

Fig. 7. Experiment result using MAS to solve LFR for different duty schedules.

Table 4. Rescheduling analysis for solving LFR using MAS.

	Large Schedule			**Medium Schedule**			**Small Schedule**		
	Max	**Med**	**Min**	**Max**	**Med**	**Min**	**Max**	**Med**	**Min**
Total Experiments	10	10	10	10	10	10	10	10	10
Total Successful Matched	10	10	1	10	10 *	0	10	0	0
Percentage of Successful Matched (%)	100	100	10	100	100	0	100	0	0
Total of Time for Successful Matched (S)	19.41	23.40	-	12.86	-	-	4.19	-	-
Average Time for Successful Matched	1.94	2.34	-	1.29	-	-	0.77	-	-
Total of Minutes Late	0	44	-	0	-	-	79	-	-
Average Minutes Late	0	4.4	-	0	-	-	9.88	-	-

** No need for rescheduling*

The average time for every rescheduling depends on the type of schedule as the results show 1.94 to 2.43 seconds for large schedule, 1.29 seconds for medium schedule, and 0.77 seconds for small schedule. The average minutes late are according to duty distribution as expected (0 minutes for large-maximum and medium-maximum, 4.4 minutes for large-median, and 9.88 for small-maximum duty).

5. Conclusions

In this works, MAS is proposed as a tool to implement the automated crew rescheduling system because it provides a quick solution in real-time and uncertain environments. The proposed architecture consists of two types of agents that are the DA and the CA. CA represents a bus crew, and DA corresponds to a duty that needs to find a crew. The agents perform the rescheduling process through negotiation between them. Based on the experiments in *AgentPower* simulation tool using 90 random data (large, medium and small schedules) collected from the bus companies in London, it can be concluded that MAS is suitable for automating the crew rescheduling process and is capable of quick rescheduling in the event of LFR. It also reveals that the distribution of a duty plays a major role in determining rescheduling success.

Acknowledgements

The authors would like to thank Universiti Teknikal Malaysia Melaka (UTeM) and Centre of Advanced Computing (C-ACT) for sponsoring this work.

Abbreviations

CA	Crew Agent
DA	Duty Agent
IoT	Internet of Technology
LFR	Late For Relief
LFSO	Late For Sign-On
LFSW	Late For Second Work
MAS	Multi Agent System
UE	Unpredictable Event

References

1. Dongqing, J.; and Qunxiong, Z. (2014). Grain emergency vehicle scheduling problem with time and demand uncertainty, *Mathematical Problems in Engineering*, Article ID 971861, 8 pages.
2. Dragoicea, M.; Salehpour, S.; Novoa, H.; and Oltean, V.E. (2017) Towards a proposal for the sustainability through institutions in public transport services in times of emergency. Lecture Notes in Business Information Processing, Springer, 279, 355-369.
3. Fu, L.; and Liu, Q. (2014). Real-time optimization model for dynamic scheduling of transit operations, *Journal of Transportation Research Record*, 1857(1), 48-55.
4. Shibghatullah, AS, Safei, S, Abas, Z Abal, Abidin, Z Zainal, Musa, H, Rahmalan, H (2017). Automated Bus Crew Rescheduling for Late for Sign-On (LFSO) Event using Multi-Agent System. *International Journal of Human and Technology Interaction* (*IJHaTI*), 1(1), 67-72.
5. Ibarra-Rojas, O.J.; Delgado, F.; Giesen R.; and Muñoz J.C. (2015). Planning, operation, and control of bus transport systems: a literature review, *Transportation Research Part B: Methodological*, 77, 38-75.
6. Tahmasseby, S.; and Van Nes, Rotterdam R (2010). Improving service reliability in urban transit networks. *European Transport Conference*, Glasgow, Scotland, UK.
7. Shibghatullah, A.S.; Eldabi, T.; and Kuljis, J. (2006). A proposed multiagent model for bus crew scheduling. *WSC 2006 Winter Simulation Conference*, 1554-1561.
8. Shibghatullah, A.S.; Eldabi, T.; and Rzevski, G. (2006). A framework for crew scheduling management system using multiagents system. *28th IEEE International Conference on Information Technology Interfaces*, 379-384.
9. Shibghatullah, A.S.; Eldabi, T; and Rzevski, G. (2006). The requirements for a dynamic bus crew scheduling system. *Proceedings of the 10th International Conference on Computer-Aided Scheduling of Public Transport*.
10. Veelenturf, L.P.; Wagelmans, A.P.M.; Potthoff, D.; Huisman, D.; Kroon, L.G.; and Maróti, G. (2016). A quasi-robust optimization approach for crew rescheduling. *Transportation Science*, 50(1), 204-215.
11. Wei, M.; Sun, B.; and Jin, W. (2013). A bi-level programming model for uncertain regional bus scheduling problems. *Journal of Transportation Systems Engineering and Information Technology*, 13(4), 106-113.

12. Xie, L.; and Suhl, L. (2015). Cyclic and non-cyclic crew rostering problems in public bus transit. *OR Spectrum*, 37(1), 99-136.
13. Fu, W.; and Hao, X. (2010). Improved ant colony algorithm for multi-depot bus scheduling problem with route time constraints. *8th World Congres. Intell. Control Automation*, 4050-4053.
14. Belén, M.; García, V.; Zanón, B. B.; and Rodríguez, E. C. (2012). Combining metaheuristic algorithms to solve a scheduling problem. *Lecture Notes in Computer Science*, 7209. 381-391.
15. Mahmud, K.; Town, G.E.; Morsalin, S.; and Hossain, M.J. (2018). Integration of electric vehicles and management in the internet of energy. *Renewable and Sustainable Energy Reviews*, 82, 4179-4203.
16. Shibghatullah, A.S. (2008). *A multi-agent system for a bus crew rescheduling system*. Ph.D. dissertation, School of Information Systems, Computing and Mathematics, Brunel University, London, UK.
17. Carosi, S.; Gualandi, S.; Malucelli, F.; and Tresoldi, E. (2015). Delay management in public transportation: service regularity issues and crew re-scheduling. *Transportation Research Procedia*, 11, 483-492.
18. Cheng, J.H.; and Chang, Y.H. (1999). Application of a fuzzy knowledge base on bus operations under uncertainty. *Fuzzy Systems Conference*, Seoul. IEEE International, 1355-1360.
19. Ferber, J. (1999). *Multi-agent systems: An Introduction to distributed* artificial *intelligence*. London: Addison-Wesley.
20. Wooldridge, MJ. (2002). *An introduction to multiagent systems*. (2nd ed.). Chichester: J. Wiley.
21. Jia, H.Z.; Ong, S.K.; Fuh, J.Y.H.; Zhang, Y.F.; and Nee, A.Y.C. (2004). An adaptive and upgradable agent-based system for coordinated product development and manufacture. *Robotics and Computer-Integrated Manufacturing*, 20(2), 79-90.
22. Karageorgos, A.; Mehandjiev, N.; Weichhart, G.; and Hammerle, A. (2003). Agent-based optimisation of Logistics and Production Planning. *Engineering Applications of Artificial Intelligence*, 16(4), 335-348.
23. Lee, C.; and Pan, C. (2004). An intelligent fuzzy agent for meeting scheduling. *Decision Support System, Fuzzy Sets and Systems*, 142(3), 467-488.
24. Jennings, N.R.; Sycara, K.; and Wooldridge, M. (1998). A roadmap of agent research and development. *Autonomous Agents and Multi-Agent Systems*, 1(1), 7-38.
25. Dimitrov, S.; Ceder, A. (Avi); Chowdhury, S.; and Monot, M. (2017). Modeling the interaction between buses, passengers and cars on a bus route using a multi-agent system. *Transportation Planning and Technology*, 40(5), 592-610.
26. Balbo, F.; Bhouri, N.; and Pinson, S. (2016), Bimodal Traffic Regulation System: A Multi-Agent approach. *International Journal of Web Intelligence and Agent Systems*, 14(2), 139-151.