

A Proposed Metaheuristic for Solving Multi-Objective Heterogeneous Vehicle Routing Problems with Time Windows

Eric Wibisono, Natashya Suryani Tiro, Dina Natalia Prayogo

Department of Industrial Engineering, University of Surabaya, Indonesia
ewibisono@staff.ubaya.ac.id

Abstract

Multi-objective optimization (MOO) has gained popularity and research interests due to its closeness in real-life applications. One of the applications of MOO is in routing problems such as the vehicle routing problems (VRP). The heterogeneity of fleet raises the complexity in VRP and a number of research has been devoted to solve the multi-objective heterogeneous vehicle routing problems (MO-HVRP) and its variants such as MO-HVRP with time windows (MO-HVRPTW). One study has attempted to develop an algorithm for MO-HVRPTW but the reported 5-6 hours of computation time is considered not practical especially in logistics problems that require fast solutions even at the cost of optimality, such as in the health sector (distribution of vaccines, blood, etc.). This paper aims to remedy the situation by improving the previous algorithm. The hybridization of elitist non-dominated sorting genetic algorithm (NSGA-II) and genetic algorithm (GA) is maintained, but a better memory management is developed in the new algorithm by keeping track of infeasible chromosomes and allowing soft time windows instead of hard time windows. Four scenarios were tested, alternating the hard and time windows and also the mutation probabilities. Compared to the results from the previous algorithm, the new algorithm reduces the computation time by 68.37% and the number of infeasible splitting by 52.84%.

Keywords: Multi-objective; Vehicle Routing Problems; Time Windows; NSGA-II; Genetic Algorithm.

1. Introduction

The increasing computing power encourages researchers to study mathematical models with increased complexity to obtain solutions that are closer to real-life situation. Included in this example is the routing problem which is part of the logistics problem often encountered in daily life, both in personal and industrial uses. For vehicle routing problems (VRP), the complexity is due to practical variants such as the nature of the route (one-way/delivery only, round-trip/delivery and pickup, etc.), the nature of the service point (time window, demand, and deterministic or probabilistic service time), the nature of the fleet (homogeneous, heterogeneous), the nature of the depot (one, many), etc. [1]. In addition to the above variants, VRP can be approached from the methodology standpoint, which are exact, heuristic, meta-heuristic, or simulation methods. The choice of method depends on the type of problem. For long-term strategic decisions, the exact method can produce the

best decision and therefore more desirable even if it takes long computation time. However, for decisions that must be obtained quickly, short computation time even with sub-optimal results is more preferable. Examples of cases like this include the distribution of daily logistics [2], selection of tourist routes [3], or determination of service areas resulting from natural disasters.

Another perspective that has attracted more attention today is the multi-objective approach to accommodate more than one problem objective. This approach is becoming more popular because many optimization problems actually have multiple objectives and it is only because of simplification that previous models only considered one objective (e.g., total cost vs. driver load, total distance vs. satisfaction). Many meta-heuristics for multi-objective optimization have been developed. However, considering that the existing methods are still new, there is still room for development to optimize the existing methods or even design new and more efficient methods.

One of the comprehensive quantitative studies is the multi-objective evolutionary algorithm with application in maritime logistics collaboration reported in [4]. The proposed model has two objectives minimizing: (1) the total cost, and (2) the difference in the proportion of costs of the two companies. The proposed method integrating the elite non-dominated sorting genetic algorithm (NSGA-II) [5] and genetic algorithm [6]. The algorithm was then tested on shipping data in the Indonesian archipelago consisting of 1 depot and 13 ports with different demands and time windows. The case belongs to heterogeneous vehicle routing problem with time windows (HVRPTW). The algorithm managed to get acceptable solutions, but it requires long computation times of 5–6 hours. This long computation time is not practical especially given only small number of nodes. The authors have identified potential problems causing the long computation time such as the high number of infeasible splitting that occurs during chromosome construction. The motivation of this research is to remedy the problems by shortening the computation time in the algorithm.

2. Literature review

A literature review on the multi-objective routing problem (MORP) can be found in [7]. Since that publication, there has not been comprehensive review even though there have been quite a number of articles related to MORP published after 2008. From [7] it can be seen that the trend of approaches in MORP in general leads to the search for a set of alternative solutions that do not dominate each other, or known as Pareto set of non-dominated solutions (or Pareto front). This approach does not give *a priori* weight to the objective function (which is practically difficult to do), but looks for several alternative solutions to then be submitted to the decision maker. Decision makers in this case are considered as parties who have

general knowledge of the problem so they do not need to be burdened with mathematical exercise in determining the weight of the objective function. Some of the popular methods for finding Pareto fronts are the Elite Non-dominated Sorting Genetic Algorithm (NSGA-II), the Strength Pareto Evolutionary Algorithm (SPEA2), the Pareto Envelope-based Selection Algorithm (PESA), the Vector Evaluated Genetic Algorithm (VEGA), Multi-Objective Genetic Algorithm (MOGA), etc. [8].

VRP can also be developed into a multi-objective problem. Following publications listed in [7], there are more than 20 new publications in this family (this figure is based on Science Direct publishers only). Various new methods are offered, especially hybrid methods that combine the concepts of some basic methods. In addition, several recent studies accommodate not only two objective functions, but three, for example, total travel costs, number of routes, and minimization of uncollected profit [9]; total energy consumption, number of vehicles, and customer satisfaction [10]; and the total cost of the trip, the number of vehicles, and the maximum collected prize [11]. The multi-objective methods commonly used for comparison are NSGA-II [9, 10] and SPEA2 [11] in addition to other methods in other studies. Other constraints such as the nature of the fleet and the time window generally also determine the model variant. Some authors explicitly mention the variant, e.g., heterogeneous vehicle routing problem with time windows (HVRPTW) [10, 12], but some others express these variants only implicitly in the mathematical formulation. For data sets, the Solomon data set is the most popular, next to the Christofides, Taillard, and Golden data sets, or random data.

MO-VRP is also widely applied in the health sector, such as in the distribution of vaccines and blood. Vaccine distribution has become a hot topic in recent times given the

pandemic. A number of recent research in this field indicates the development of the VRP to solve vaccine distribution problems, taking into account uncertainties in time, demand, etc. [13]. Other research worth mentioning are concerning the distribution of the Covid-19 vaccine at ultra-low temperatures [14]; design of an innovative multi-period model of vaccine allocation to streamline delivery [15]; and vaccine distribution with multiple objectives, namely, to reduce the number of deaths and distribution costs [16]. Other applications in the healthcare sector is in the distribution of blood. Optimal distribution of blood is very important because it involves human life [17]. Many approaches have been tried in this case, for example the vendor-managed inventory [18] by taking principles in inventory management in general. In relation to the research being made, a multi-objective approach was tried to be applied in this case study to study its characteristics first, before testing with a new algorithm.

To be able to produce a meaningful solution, the applied MO method must be able to produce alternative solutions as close as possible to the Pareto front. The effectiveness of an MO method is evaluated based on these criteria. The more complex the problem, for example in the case of HVRPTW, the longer it will take to approach the Pareto front. In other words, the complexity of the MO method will increase the computation time. There is still room for development in this research areas, which is to increase the effectiveness of the MO method in producing alternative solutions that are useful for decision makers. A hybrid algorithm as demonstrated in [4] is a novel approach, however, 5–6 hours of computation time call for further improvement in the algorithm. A faster algorithm could be useful in some sectors such as the health care, where sub-optimal solutions can be obtained within reasonable amount of time, rather than waiting for the optimal solutions that are not practical timewise.

The objective of this research is therefore to improve the algorithm proposed in [4] by investigating factors that could lead in reducing the overall algorithm's computation time.

3. Research methodology

In this study, the data set is replicated from [4]. The data set has 14 nodes consisting of one depot and 13 ports. Data of ports include distances between ports, demand in each port, and time window visiting each port. Data of vessels include capacity, sailing speed, and fixed cost and variable cost of the vessels. A methodology combining NSGA-II and a particular version of GA with effective splitting of chromosomes was established and used to solve the problem. The algorithm is explained below in brief, but readers are invited to read [4] for more detailed information. The initial algorithm covers the following principles:

1. The scope of the problem is maritime logistics collaboration. The problem seeks out how to fairly share the costs between two collaborating liner companies. Therefore, cost minimization is not the only objective, but a second objective is added, i.e., to minimize the deviation of cost proportion between the two companies.
2. A metaheuristic-based method combining NSGA-II and a particular version of GA is developed. The chromosomes are constructed as giant tour without trip delimiter, but an optimal solution of each chromosome can be found using a procedure called Split.
3. Population is constructed by randomly generating the order of cities (or the genes) in the chromosomes. However, there are chromosomes generated by heuristics (nearest neighbor and sweep) so that some good solutions are included in the population and the search process

does not start from completely random search space.

4. Forming a new population is carried out using the NSGA-II principles, i.e., members in the first layer (the outermost) of the Pareto front is included first, followed by members in the second layer, and so on. If not all members in one layer can be included, the crowding-distance tournament is used as a tri-breaker.
5. For crossover, order crossover (OX) operator is used, whereas for mutation, a 9-step local search procedure is employed. The mutation rates are set to be aggressive with a probability (p_m) of over 20%.

The next sections will report the proposed improvement of the methodology without further reciting the initial algorithm.

Efforts to improve the algorithm are carried out by studying the causes of long computation time in the initial algorithm. One of the main causes is the high number of infeasible splitting cases, i.e., cases where the chromosomes of giant tour generated from the GA operator, either crossover or mutation, are failed to be split into feasible routes due to constraints on the number of fleets and heterogeneity in terms of capacity (infeasible splitting does not occur in the case of VRP with an unlimited number of fleets). To fix this, a newly developed memory management algorithm with two functions is proposed.

The first function is to record the chromosomes that are failed to split in a block of memory called trash. When a new chromosome is generated, the algorithm will check if this chromosome is in the trash and if that is case, further processes need not be carried out. This improvement saves a considerable amount of computation time, since one of the modules in the previous algorithm that consumes processing time is mutation with local search. With the initial

tagging of the infeasible chromosomes, those chromosomes will not undergo mutation.

The second function is concerning the time windows. The previous algorithm used hard time windows thus solutions that did not meet the time window constraints were immediately rejected. This led to a high number of chromosomes rejected in the process. This condition is corrected in the proposed algorithm by changing the hard time windows to soft time windows with 6 hours tolerance. If the hard time window is not met but the arrival time is within 6 hours or still within the soft time window, the chromosome is still accepted, but placed at the bottom of the population. This does not affect the quality of the solution because the Pareto front is usually divided into several layers in the population, so the lower-ranked solutions have small chance of being selected. On the other hand, preserving such chromosome maintains diversity in the search space and reduces computation time because rejection occurs less frequently.

Another amendment is related to the value of distance limit which is returned to a constant value of 1. This was actually suggested in [6] but in [19] the authors tried to vary the value during the GA iteration with the value changing up and down to explore the search space, but from the initial research, this assumption was not proven.

The new algorithm is coded in Python compared to originally in MATLAB. Under Python open-source environment, various libraries and modules can be utilized. Computing time is also accelerated a little bit although it is not as significant as other changes. The algorithm is tested on Intel Core i7 11th Gen CPU with a RAM capacity of 16 GB running in Windows 10.

Four scenarios are run and compared, each in 10 replications. The scenarios are the 2×2 experiment by alternating two parameters, i.e., the time windows and the mutation rates. The four scenarios are explained as follows.

1. Scenario 1: using memory management and constant distance limit (DL = 1); hard time windows; mutation probability 0.05.
2. Scenario 2: using memory management and constant distance limit (DL = 1); hard time windows; mutation probability 0.20.
3. Scenario 3: using memory management and constant distance limit (DL=1); soft time windows (Twin + 6 hours); mutation probability 0.05.
4. Scenario 4: using memory management and constant distance limit (DL = 1), soft time windows (Twin + 6 hours); mutation probability 0.20.

The proposed new algorithm is detailed step by step in Figure 1.

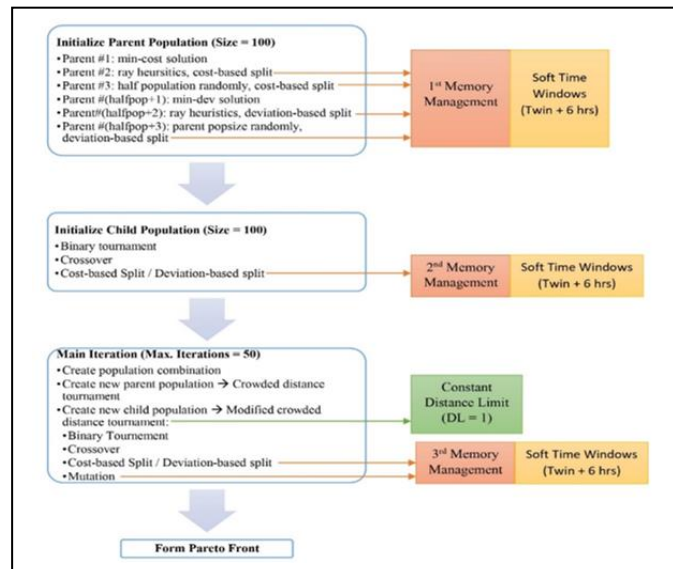


Figure 1. The proposed new algorithm for MO-HVRPTW

4. Results and discussion

In scenario 1, the average Pareto points (solutions that are not dominating each other) is 9.5. The average number of duplicate chromosomes identified by the memory management is 25,659. The average number of infeasible splitting is 1289. The average total computing time is 1.85 hours.

In scenario 2, the average Pareto points is 11.3. The average number of duplicate chromosomes identified by the memory management is 100,970. The average number of infeasible splitting is 1,458. The average total computing time is 1.80 hours.

In scenario 3, the average Pareto points is 9.8. The average number of duplicate chromosomes identified by the memory management is 25,866. The average number of infeasible splitting is 649. The average total computing time is 2.07 hours.

In scenario 4, the average Pareto points is 10.2. The average number of duplicate chromosomes identified by the memory management is 96,982. The average number of infeasible splitting is 608. The average total computing time is 1.93 hours.

Based on the above results, scenario 4 is considered as the best algorithm.

Comparison of the number of duplicate chromosomes in memory management is visualized in boxplots. From Figure 2, it can be seen that each scenario has a very high number of duplicate chromosomes. This proves that in the original algorithm, one of the causes of inefficiency and high number of infeasible splitting is the generation of clone chromosomes that forces the algorithm to re-examine the duplicate chromosomes.

However, there is a difference in the number of duplicated chromosomes between scenarios. Scenarios with aggressive mutation rate ($p_m = 0.20$), i.e., scenarios 2 and 4 have an average of significantly more chromosome duplicates compared to scenarios with a smaller mutation rate ($p_m = 0.05$), i.e., scenarios 1 and 3. This is because the scenarios with high mutation rate lead to higher mutation activity. This mutation activity creates new chromosomes repeatedly so that it increases the chances of the generation of chromosomes that are

already in the trash and are infeasible (forming duplicate chromosomes). Therefore, scenarios 2 and 4 have more duplicate chromosomes than scenarios 1 and 3, although they have similar characteristics.

When viewed from the comparison of scenarios with the same mutation rate, scenarios with low mutation rates do not have a significant difference in the number of trash duplicates compared to aggressive mutation rates. This indicates that memory management does not have a different effect on scenarios with normal time windows and soft time windows at low mutation rates. Meanwhile, scenario 4 which uses soft time windows has a lower number of trash duplicates and is negatively correlated to scenario 2 which uses normal time windows. Therefore, from the two above analysis, we can conclude that the use of soft time windows significantly reduces the number of duplicate chromosomes if high mutation rate is applied.

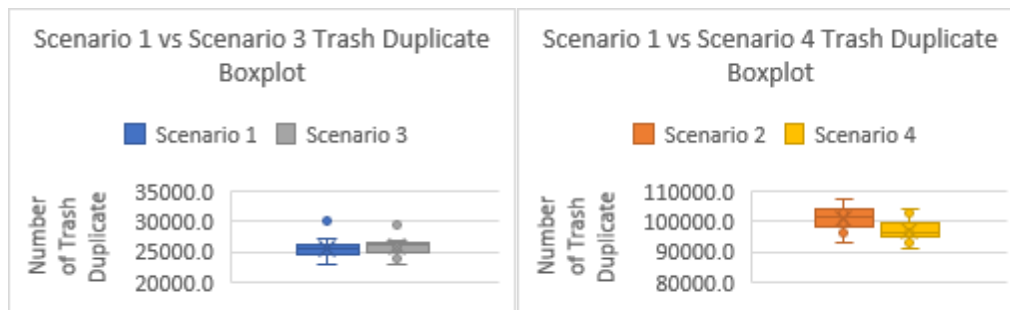


Figure 2. Box plots comparing trash duplicate between scenarios

The comparison of the number of infeasible splitting can be seen in Figure 3. The average total infeasible splitting from scenarios 1 to 4 is very low compared to the results with the original algorithm. This is because the original algorithm used increasing (1 to 5) or decreasing (5 to 1) distance limit mechanisms. A large value of distance limit causes many chromosomes to become infeasible (does not pass the distance limit criteria) so that in the original

algorithm, infeasible splitting occurs with an average of 2156 times. In scenarios 1 to 4, the constant distance limit is set with a value of 1. By comparing the numbers of infeasible splitting between the old (original) and new algorithms, it can be concluded that the new algorithm successfully reduces the average number of infeasible splitting by 52.841%.

From Figure 3 we can also see that scenarios 3 and 4 have lower numbers of

infeasible splitting than scenarios 1 and 2. This can be correlated by the fact that scenarios 3 and 4 use soft time windows. The relaxation of the time window constraint makes it easier to produce feasible chromosomes. The low numbers of infeasible splitting in scenarios 3 and 4 is also assisted by the use of memory management as previously explained. The average number of infeasible splitting in scenarios 3 and 4 is 51.97% lower compared to that of scenarios 1 and 2.

From the perspective of mutation probabilities, it can be seen that in normal time windows (scenarios 1 and 2), higher mutation rate leads to more infeasible splitting. Meanwhile, in scenarios with soft time windows (scenarios 3 and 4) there is no effect of mutation rate on the number of infeasible splitting. This shows that the 6 hours of time windows tolerance can still deal with aggressive mutation rate of 0.20, without causing an increase in infeasible splitting.

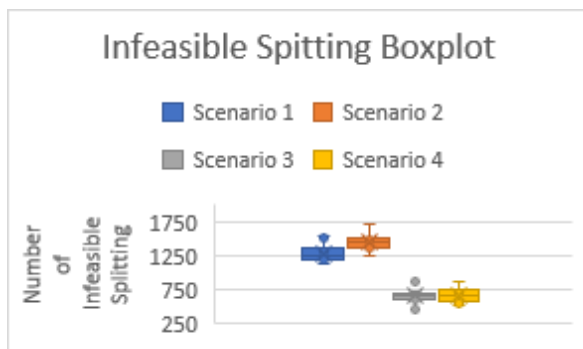


Figure 3. Box plots comparing numbers of infeasible splitting

Computation time is the main issue in MO-HVRPTW. Having successfully implemented all of the above improvements, the new algorithm significantly reduces the computation time from an average of 6.03 hours to 1.91 hours or an improvement of 68.37% from the initial results reported in [4].

As shown in Figure 4, there are differences in computation time due to the characteristics of each scenario. Scenarios 3 and 4 (scenarios with soft time windows) have a higher computation time than scenarios 1 and 2, even though they have lower numbers of infeasible splitting as detailed in the preceding paragraphs. It can be concluded that there seems to be a trade-off between the two criteria. The logic of the argument is because invoking the memory management module too often worsen the computation time, although it could prevent infeasible chromosomes being kept in the population thus also prevent subsequent process such as mutation being run.

Figure 4 also suggests that there is no significant difference in computation time between scenarios 1 and 2, which means that in scenarios with normal time windows, the mutation rate does not affect the computation time. However, in the application of soft time windows, scenario 4 with aggressive mutation rate actually has a lower computation time than scenario 3. Based on this, it can be concluded that the application of soft time windows is more efficient at aggressive mutation rate.

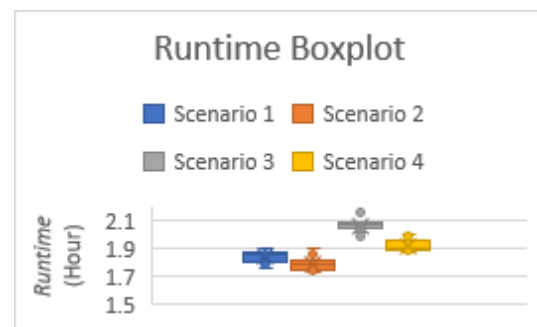


Figure 4. Box plots comparing computation times

6. Conclusion

This research proposes an improved methodology to solve multi-objective heterogeneous vehicle routing problems with time windows (MO-HVRPTW). A hybrid algorithm combining NSGA-II and a

particular version of GA with effective splitting of chromosomes has been proposed in the literature, however, the computation time of 5-6 hours in the original algorithm are considered unsatisfactory and impractical in logistics problems that prefer faster solutions to optimality. A new algorithm is proposed in this research to deal with the high number of infeasible splitting that caused long computation time in the original algorithm. The new algorithm features better memory management by tagging infeasible chromosomes, such that when they are recreated in the next iteration, they will be immediately discarded to avoid future reprocessing, e.g., in the mutation process. Another feature is the introduction of soft time windows that gives six hours of tolerance for visiting time. In the initial algorithm, failing to abide the time windows results in the rejection of the chromosome, causing the algorithm to create new chromosome for replacement, hence adding the computation time. Another improvement is returning the distance limit value to a constant value of 1 since varying the value in the other research has not been proven effective.

By implementing all the above improvement points, the new algorithm is able to reduce the computation time by 68.37% and the number of infeasible splitting by 52.84%. These significant improvements pave the avenue for applications. Some potential applications in the agenda, for example, in the health sector, i.e., distribution of vaccines and/or blood, or in humanitarian logistics to determine service areas impacted by natural disasters.

7. Acknowledgements

The authors wish to thank The Ministry of Education, Culture, and Research and Technology of The Republic of Indonesia for funding this research under the scheme Penelitian Dasar Unggulan Perguruan

Tinggi (PDUPT) contract no. 059/SP-Lit/LPPM-01/KemendikbudRistek/Multi/FT/III/2022.

References

- [1] K. Braekers, K. Ramaekers and I. van Nieuwenhuyse, "The vehicle routing problem: State of the art classification and review," *Computers & Industrial Engineering* 99: 300–313 (2016)
- [2] E. Wibisono, A. Santoso and M. A. Sunaryo, "Improving delivery routes using combined heuristic and optimization in a consumer goods distribution company." *Proceedings of International Conference on Informatics, Technology, and Engineering*, (2017) August 24–25; Bali, Indonesia
- [3] I. Hapsari and I. Surjandari, "Tourism mobile recommender systems: A survey." *Proceedings of the 6th IEEE International Conference on Advanced Logistics and Transport* 75–80, (2017) July 24–27; Bali, Indonesia
- [4] E. Wibisono and P. Jittamai, "Multi-objective evolutionary algorithm for a ship routing problem in maritime logistics collaboration," *International Journal of Logistics Systems and Management* 28(2): 225–252 (2017)
- [5] K. Deb, S. Agrawa, A. Pratap and T. Meyarivan, "A fast elitist non-dominated sorting genetic algorithm for multi-objective optimization: NSGA-II," *Proceedings of the Parallel Problem Solving from Nature VI (PPSN-VI)* 849–858 (2000)
- [6] C. Prins, "Two memetic algorithms for heterogeneous fleet vehicle routing problems," *Engineering Applications of Artificial Intelligence* 22(6) 916–928 (2009)
- [7] N. Josefowicz, F. Semet and E.-G. Talbi, "Multi-objective vehicle routing problems," *European Journal of Operational Research* 189(2): 293–309 (2008)

- [8] K. Deb, *Multi-Objective Optimization using Evolutionary Algorithms*, New York: Wiley (2008)
- [9] A. García-Nájera, J. A. Bullinaria and M. A. Gutiérrez-Andrade, M. A., “An evolutionary approach for multi-objective vehicle routing problems with backhauls,” *Computers & Industrial Engineering* 81: 90–108 (2015)
- [10] S. F. Ghannadpour and A. Zarrabi, “Multi-objective heterogeneous vehicle routing and scheduling problem with energy minimizing,” *Swarm and Evolutionary Computation* 44: 728–747 (2019)
- [11] J. Long, Z. Sun, P. M. Pardalos, Y. Hong, S. Zhang, and C. Li, “A hybrid multi-objective genetic local search algorithm for the prize-collecting vehicle routing problem,” *Information Sciences* 478: 40–61 (2019)
- [12] D. M. Miranda, J. Branke and S. V. Conceição, “Algorithms for the multi-objective vehicle routing problem with hard time windows and stochastic travel time and service time,” *Applied Soft Computing* 70: 66–79 (2018)
- [13] B. Li, H. Guo and S. Peng, “Impacts of production, transportation and demand uncertainties in the vaccine supply chain considering different government subsidies,” *Computers and Industrial Engineering*, 169(March) (2022)
- [14] J. Sun, M. Zhang, A. Gehl, B. Fricke, K. Nawaz, K. Gluesenkamp, B. Shen, J. Munk, J. Hagerman and M. Lapsa, “COVID-19 vaccine distribution solution to the last mile challenge: Experimental and simulation studies of ultra-low temperature refrigeration system,” *International Journal of Refrigeration* 133(September 2021) 313–325 (2022)
- [15] M. Fadaki, A. Abareshi, S. M. Far and P. T. W. Lee, “Multi-period vaccine allocation model in a pandemic: A case study of COVID-19 in Australia,” *Transportation Research Part E* 161(March) 102689 (2022)
- [16] M. Mohammadi, M. Dehghan, A. Pirayesh and A. Dolgui, “Bi-objective optimization of a stochastic resilient vaccine distribution network in the context of the COVID-19 pandemic,” *Omega* 113 102725 (2022)
- [17] S. Mahdi, H. Motlagh, M. Reza, G. Samani and S. Homaei, “Blood supply chain management: *robust* optimization, disruption risk, and blood group compatibility (a real-life case),” *Journal of Ambient Intelligence and Humanized Computing* 0123456789 (2019)
- [18] W. Liu, G. Y. Ke, J. Chen and L. Zhang, “Scheduling the distribution of blood products: A vendor-managed inventory routing approach,” *Transportation Research Part E* 140(December 2019) (2020)
- [19] E. Wibisono, I. Martin, and D. N. Prayogo, “Genetic Algorithm with Adaptive Diversification and Intensification for the Vehicle Routing Problem,” *AIP Conference Proceedings* 2470 050012 (2022)