International Journal of
INTELLIGENT SYSTEMS AND APPLICATIONS IN ENGINEERING
ISSN:2147-6799 www.ijisae.org Original Research Paper

Multi-Objective Travel Route Optimization Using Non-Dominated Sorting Genetic Algorithm

Jyoti Singh[1], Shahnaz Fatima[2], Alok Singh Chauhan*[3]

Submitted: 23/04/2023 **Revised**: 24/06/2023 **Accepted**: 04/07/2023

Abstract: At times, optimization procedures are required to address practical issues. A single goal may be the focus of some of these issues, while others may include competing priorities. An issue is said to be a single-objective optimization problem if there is only one goal to achieve and a multi-objective optimization problem if there are two or more. Public transportation has been generally acknowledged as a viable approach to ameliorate transportation-associated issues including traffic congestion and air pollution as demand for transportation rises in most major cities across the globe. The development of an efficient public transportation system is a priority. In this study, we structure the trip route issue as a multi-objective optimization problem with the aim of reducing users' financial outlays, journey times, and carbon footprints. The proposed Non-dominated Sorting Genetic Algorithm-based approach provides environmentally preferable travel choices and allows the traveller to choose between the slower but more eco-friendly bus journey and the faster but more "eco-unfriendly" air plane.

Keywords: *Multi-criterion decision-making, Genetic algorithms, Multi-objective optimization, Pareto-optimal solutions*

1. Introduction

The vast majority of the issues that are faced in the actual world are multi-goal difficulties. Finding a unique answer that will work for every situation is challenging [1]. The matter becomes much more complicated if there are trades-offs between competing goals; for example, if one goal is reduced while another is maximized. The goal of multi-objective optimization (MOO) algorithms is to strike a compromise between competing goals [2], [3]. Due to its ability to consider all relevant factors, the MOO algorithms provide a workable set of solutions to issues with multiple goals. As an alternative to a single optimum solution, MOO algorithms provide a collection of options that meet the criteria, known as Pareto optimal solutions. The Pareto optimum set of solutions includes all of the possible outcomes while being non-detrimental to any of the other possible outcomes [2], [3]. The choice is made by picking one of the feasible options presented.

Over the course of the last decade, a number of different multi-objective evolutionary algorithms (MOEAs) have been proposed [4], [5]. Specifically, they may find many Pareto-optimal solutions with a single run. Since it is difficult to find a single solution that simultaneously maximizes all objectives, a multi-objective formulation of a problem is necessary [6]. A method that produces numerous solutions sitting on or near the Pareto-optimal front is particularly helpful in practice.

Assume that all objective functions are to be reduced since maximizing of one objective function may be recast as the minimizing of another. To define a multi-objective optimization problem, we need to provide a collection of objective functions and a solution space.

Consider the solution space to be S. An example of a multi-objective optimization problem is:

$$\min f1(s), \ldots, \min fk(s),\ s \in S$$

where f1, f2,..., fk are objective functions where fi: $S \rightarrow R$.

The cost vector of solution s is the vector [f1(s),..., fk(s)]. To discover a solution with the best possible cost vector is, thus, to solve a multi-optimization issue. In most cases, the goal functions are incompatible, thus minimizing one will require sacrificing another. Therefore, there is no solution that achieves a minimum of all goals. Hence, the cost vector ($\min_{s\in S}$ f1(s), ..., $\min_{s\in S}$fk(s)) is not present in any solution. Rather, it is necessary to make a compromise between the goals. To that end, we may define an optimum cost vector as one that strikes the best balance between competing goals. In a perfect compromise, reducing one goal any more would have a negative impact on another. The dominance relation is used to formally describe this.

If at every index i in both vectors, x is less or equal than y, then x dominates y. The notation for this is $x \prec y$. In other words, given $x \neq y$, xi must be strictly less than yi for some i and this must hold true for at least one i. It is important to keep in mind that dominance is a partial order relation, therefore, $(1, 2, 2) \prec (1, 2, 3)$, but $(1, 2, 3) \nprec (1, 3, 2)$ and (1,

[1] Amity Institute of Information Technology, Amity University, Lucknow, India
[2] Amity Institute of Information Technology, Amity University, Lucknow, India
[3] School of Computing Science & Engineering, Galgotias University, Greater Noida, India
* Corresponding Author Email: alokchauhan.1983@gmail.com
ORCID: https://orcid.org/0000-0002-9144-4381

3, 2) $\nprec$(1, 2, 3). The non-dominated portion of a collection of vectors may be determined using the dominance relation.

For each vector set X, the non-dominated subset is characterized as

$$\text{nondom}(X) = \{x \in X \mid \nexists x' \in X, x' \prec x\}$$

In this way, we may specify the solutions that map to the set of cost vectors that is not dominated by any other set in the solution space.

When solving a MOP, the best solutions will have a cost vector that is not dominated by any other cost vectors in the MOP. From here on, we shall refer to a solution that is inside the Pareto front as a non-dominated solution. A MOP often has several non-dominated solutions [7]. As dominance is a partial order relation, a decision maker needs to supply extra preference information to choose one answer. When extra preferences are needed are used to classify multi-objective optimization techniques. To function properly, interactive algorithms need extra preference information at various stages of the algorithm's execution. Using the preferences, we may tweak the optimization by, for instance, reordering the priority of objectives [8].

Discovering the best possible path is the focus of this research. The algorithms utilized for multi-objective optimization in this research look for ways to improve on three different metrics at once.

Each part of the research is labeled as follows: Section 2 examines the work of other scholars who have come to similar conclusions. The suggested approach is described in Section 3. Section 4 details the problem formulation that will be used in the experimental analysis that follows. In this part, we also discuss the outcomes. The paper is finished with Section 5.

2. Literature Review

Path planning involves not just identifying a collision-free path from a starting point to an endpoint, but also a path that reduces or maximizes a set of important goals.

Jinjun Tang et al.[9] propose data-driven bus timetable improvements. We initially created a bi-objective optimization model to calculate the bus company's departure schedule and passengers' waiting times. By combining bus GPS trajectories with Smart Card passenger data, optimization model variables such as time-dependent travel duration, bus dwell time, and passenger volume are determined. Finally, a unique encoding approach enabled a faster Non-dominated Sorting Genetic Algorithm-II (NSGA-II) to find Pareto optimal solutions. To test the proposed method, a Beijing bus route is tested. Compared to empirical scheduling and GA-based single-objective optimization, the proposed methodology may quickly create high-quality and acceptable schedule schemes for urban transportation system administrators.

Using an elitist non-dominated sorting genetic algorithm and a unique kind of multi-objective differential evolution called heterogeneous multi-objective differential evolution, the paths of mobile robots are designed in [10]. Numerical simulations are used to evaluate the effectiveness of the suggested optimization methods. The demonstrated superiority of the offered methods for this issue has been confirmed by the findings. The Vehicles route problems (VRP) with time windows (VRPTW) were studied by Fei Tan et al.[11] The authors developed a model for robust multi-objective VRPTW (RMOVRPTW) and suggested a MOEA/D-based robust optimization method (R-MOEAD-VRP) to minimize both the overall distance travelled and the number of vehicles needed to complete the journey. The experimental findings demonstrate the effectiveness of their suggested algorithm in producing answers that are both more resilient and less influenced by uncertainty.

Researchers Fergal Stapleton et al. [12] examined the impact of five distinct objectives on neuroevolution for trajectory prediction in autonomous vehicles using the popular Non-dominated Sorting Genetic Algorithm-II. The researchers found that the objectives can have either a beneficial or detrimental impact on the process. The open green vehicle routing issue was solved sustainably by Joydeep Dutta et al. [13] using the cluster primary-route secondary approach. Reducing operating costs and service vehicle fuel emissions are two realistic aims that occasionally conflict. This paradigm lets the decision-maker pick the best option from a set of choices, unlike multi-objective problems. First, it employs a modified k-means algorithm to sort all users into discrete groups. Residents use one automobile per building group. Then, a multi-objective evolutionary algorithm finds the best sub route to service all cluster users. The approximated fronts were calculated using the expanded Strength Pareto Evolutionary Algorithm (SPEA2) and the Non-dominated Sorting-based Genetic Algorithm (NSGA-II). When compared to SPEA2, NSGA-II has lower efficacy.

3. Material and Methods

In this paper, we offer a Non-dominated Sorting Genetic Algorithm-based solution approach for determining the trade-off levels between competing goals. The approach to the answer includes three steps: The first stage involves generating a large number of candidate routes with corresponding frequencies; the second stage involves assigning trips to routes for a given transit network and demand matrix and then evaluating the network based on that assignment; and the third stage involves using a Genetic Algorithm-based optimization process to find the trade-off levels between the conflicting objectives.

The pseudocode of the proposed algorithm is presented below:

Proposed Multi-Optimization Algorithm

P creates a "parent population" of N individuals.

while Iteration <MaxIteration do

C <- Empty child population

while the number of individuals C in < N do

Use tournament selection approach for selecting parent1(P1)

Use tournament selection approach for selecting parent2(P2)

Get child1(C1), child2(C2) through the Binary Crossover (P1,P2)

Polynomial Mutation (C1, C2)

Evaluate C1, C2 for their fitness values

Insert C1, C2 into C

end while

U<- Combine P and C to get 2N individuals

Rank the union set U using the nondominated sorting.

P <- N front individuals in U by the crowded comparison selector.

end while

Return plausible non-dominated solutions.

4. Evaluation and Results

To illustrate and evaluate the performance of the proposed methodology, an experimental transit network is constructed and tested. An all-India travel plan was prepared including 12 cities i.e. 'Lucknow', 'New Delhi', 'Mumbai', 'Pune', 'Kolkata', 'Patna', 'Chennai', 'Guwahati', 'Chandigarh', 'Jaipur', 'Ahmedabad', and 'Hyderabad'. Journey to be started from Lucknow. Any one of two modes of travel i.e bus and flight is to be selected based on three objective criteria fare, time, and CO_2 emission. For bus travel time calculation 60KM/Hr speed of the bus has been considered. For flight time and fare calculation, the average of the best flights in the month Feb 2023 has been considered. As per a study[14], the CO_2 emission of the bus is 515.2 g/KM. Considering the average of 50 passengers in a bus, CO_2 emission of 10.304 g/KM per person is used in this study. 36.6 g/km CO_2 emission has been considered for Airplane travel [15]. The fare calculation of the bus has been done as Rs. 4 per kilometer.

Data used for the experimental evaluation of the proposed methodology is illustrated in tables 1-6.

Table 1: Travel time by Bus

	Lucknow	**NewDelhi**	**Mumbai**	**Pune**	**Kolkata**	**Patna**	**Chennai**	**Guwahati**	**Chandigarh**	**Jaipur**	**Ahmedabad**	**Hyderabad**
Lucknow	99999	530	1570	1417	1017	507	1985	1315	695	505	950	1250
NewDelhi	530	99999	1398	1477	1501	991	2215	1800	317	264	913	1327
Mumbai	1570	1398	99999	160	2059	1766	1032	2629	1350	1175	534	764
Pune	1417	1477	160	99999	1996	1666	1117	2535	1696	1208	654	584
Kolkata	1017	1501	2059	1996	99999	611	1663	757	1741	1540	2080	1456
Patna	507	991	1766	1666	611	99999	1907	859	1614	1104	1671	1486
Chennai	1985	2215	1032	1117	1663	1907	99999	2030	2479	2119	1693	669

	Lucknow	New Delhi	Mumbai	Pune	Kolkata	Patna	Chennai	Guwahati	Chandigarh	Jaipur	Ahmedabad	Hyderabad
							99					
Guwahati	1315	1800	2629	2535	757	859	2030	999999	1469	1907	1421	2223
Chandigarh	695	317	1350	1696	1741	1641	2479	1469	999999	492	1139	1840
Jaipur	505	264	1175	1208	1540	1104	2119	1907	492	999999	538	1442
Ahmedabad	950	913	534	654	2080	1671	1693	1421	1139	538	999999	1152
Hyderabad	1250	1327	764	584	1456	1486	669	2223	1840	1442	1152	999999

Table 2: Travel time by Flight

	Lucknow	**NewDelhi**	**Mumbai**	**Pune**	**Kolkata**	**Patna**	**Chennai**	**Guwahati**	**Chandigarh**	**Jaipur**	**Ahmedabad**	**Hyderabad**
Lucknow	999999	80	125	480	115	75	463	105	85	120	119	120
New Delhi	80	99999	115	125	130	105	165	135	60	60	85	125
Mumbai	125	115	999999	275	150	150	100	195	140	130	65	90
Pune	480	125	275	999999	145	135	105	165	135	110	75	120
Kolkata	115	130	150	145	999999	135	135	65	140	155	165	160
Patna	75	105	150	135	135	999999	145	135	115	300	130	135
Chennai	463	165	100	105	135	145	999999	180	335	155	145	80
Guwahati	105	135	195	165	65	135	180	999999	225	165	420	360
Chandigarh	85	60	140	135	140	115	335	225	999999	85	105	150
Jaipur	120	60	130	110	155	300	155	165	85	999999	80	115

Ahmedabad	119	85	65	75	165	130	145	420	105	80	999999	115
Hyderabad	120	125	90	120	160	135	80	360	150	115	115	999999

Table 3: CO2 emission by Bus

	Lucknow	**NewDelhi**	**Mumbai**	**Pune**	**Kolkata**	**Patna**	**Chennai**	**Guwahati**	**Chandigarh**	**Jaipur**	**Ahmedabad**	**Hyderabad**
Lucknow	999999	6	17	15	11	6	21	14	8	6	10	13
NewDelhi	6	999999	15	16	16	11	23	19	4	3	10	14
Mumbai	17	15	999999	2	22	19	11	28	14	13	6	8
Pune	15	16	2	999999	21	18	12	27	18	13	7	7
Kolkata	11	16	22	21	999999	7	18	8	18	16	22	16
Patna	6	11	19	18	7	999999	20	9	17	12	18	16
Chennai	21	23	11	12	18	20	999999	21	26	22	18	7
Guwahati	14	19	28	27	8	9	21	999999	16	20	15	23
Chandigarh	8	4	14	18	18	17	26	16	999999	6	12	19
Jaipur	6	3	13	13	16	12	22	20	6	999999	6	15
Ahmedabad	10	10	6	7	22	18	18	15	12	6	999999	12
Hyderabad	13	14	8	7	16	16	7	23	19	15	12	999999

Table 3: Co2 emission by flight

Emission_flight	**Lucknow**	**NewDelhi**	**Mumbai**	**Pune**	**Kolkata**	**Patna**	**Chennai**	**Guwahati**	**Chandigarh**	**Jaipur**	**Ahmedaba**	**Hyderaba**

											d	d
Lucknow	999999	20	58	52	38	19	73	49	26	19	35	46
New Delhi	20	999999	52	55	55	37	82	66	12	10	34	49
Mumbai	58	52	999999	6	76	65	38	97	50	44	20	28
Pune	52	55	6	999999	74	61	41	93	63	45	24	22
Kolkata	38	55	76	74	999999	23	61	28	64	57	77	54
Patna	19	37	65	61	23	999999	70	32	60	41	62	55
Channai	73	82	38	41	61	70	999999	75	91	78	62	25
Guwahati	49	66	97	93	28	32	75	999999	54	70	53	82
Chandigarh	26	12	50	63	64	60	91	54	999999	19	42	68
Jaipur	19	10	44	45	57	41	78	70	19	999999	20	53
Ahmedabad	35	34	20	24	77	62	62	53	42	20	999999	43
Hyderabad	46	49	28	22	54	55	25	82	68	53	43	999999

Table 5: Travel cost by bus

	Lucknow	**New Delhi**	**Mumbai**	**Pune**	**Kolkata**	**Patna**	**Channai**	**Guwahati**	**Chandigarh**	**Jaipur**	**Ahmedabad**	**Hyderabad**
Lucknow	999999	2120	6280	5668	4068	2028	7940	5260	2780	2020	3800	5000
New Delhi	2120	999999	5592	5908	6004	3964	8860	7200	1268	1056	3652	5308
Mumbai	6280	5592	999999	640	8236	7064	4128	10516	5400	4700	2136	3056
Pu	566	590	64	999	798	666	446	101	6784	483	2616	2336

ne	8	8	0	999	4	4	8	40		2		
Kolkata	4068	6004	8236	7984	999999	2444	6652	3028	6964	6160	8320	5824
Patna	2028	3964	7064	6664	2444	999999	7628	3436	6456	4416	6684	5944
Chennai	7940	8860	4128	4468	6652	7628	999999	8120	9916	8476	6772	2676
Guwahati	5260	7200	10516	10140	3028	3436	8120	999999	5876	7628	5684	8892
Chandigarh	2780	1268	5400	6784	6964	6564	9916	5876	999999	1968	4556	7360
Jaipur	2020	1056	4700	4832	6160	4416	8476	7628	1968	999999	2152	5768
Ahmedabad	3800	3652	2136	2616	8320	6684	6772	5684	4556	2152	3999996	4608
Hyderaba	5000	5308	3056	2336	5824	5944	2676	8892	7360	5768	4608	999999
d												

Table 6: Travel cost by flight

	Lucknow	**New Delhi**	**Mumbai**	**Pune**	**Kolkata**	**Patna**	**Chennai**	**Guwahati**	**Chandigarh**	**Jaipur**	**Ahmedabad**	**Hyderabad**
Lucknow	999999	4000	7433	8946	9438	7480	7723	4250	6484	6361	4211	4411
New Delhi	3562	999999	3434	2985	4840	2915	4878	4297	1914	2489	2290	4083
Mumbai	7433	3434	999999	4434	5338	4023	1728	6399	4727	2526	1414	2946
Pune	8946	2985	4434	999999	3624	5628	4358	5919	5809	5085	2501	2414
Kolkata	9438	4840	5338	3624	999999	7175	6913	2949	7734	8292	4549	5123
Patna	7480	2915	4023	5628	7175	999999	7252	6899	5729	5663	4318	3871
Chenn	772	487	172	435	691	725	9999	689	5729	566	4318	3871

ai	3	8	8	8	3	2	99	9		3		
Guwahati	4250	4297	6399	5919	2949	5454	6899	999999	7101	7847	6501	5920
Chandigarh	6484	1914	4727	5809	7734	4034	5729	7101	999999	3623	3540	7707
Jaipur	6361	2489	2526	5085	8292	6824	5663	7847	3623	999999	3727	5295
Ahmedabad	4211	2290	1414	2501	4549	6042	4318	6501	3540	3727	999999	5524
Hyderabad	4411	4083	2946	2414	5123	7115	3871	5920	7707	5295	5524	999999

The suggested technique incorporates CO2 emissions, trip time, and overall cost into its fitness function. This procedure takes in the route (e.g., [0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 0]) and the BF as inputs (e.g. [0, 0, 0, 1, 1, 1, 0, 0, 0, 1, 1, 1]). Where 1 represents the bus and 0 represents the airplane, BF is a binary array. In this study, the local search function two opt () was used. When it comes to local searches for the traveling salesman issue, the 2-opt operator is by far the most popular and successful choice. The basic concept is to reverse the sequence of a route that now doubles back on itself.

The most efficient method for computing 2-opt involves comparing just the distance of the swapped edges before and after the transformation. In certain cases, the 2-opt operator may be used, which is computationally wasteful since it compares the whole distance. Given the availability of both bus and plane, as well as a bus than plane and plane then bus, there are a total of four permutations to think about.

Visualizations of Pareto approximation sets in three dimensions are produced using the suggested method (fig-1).

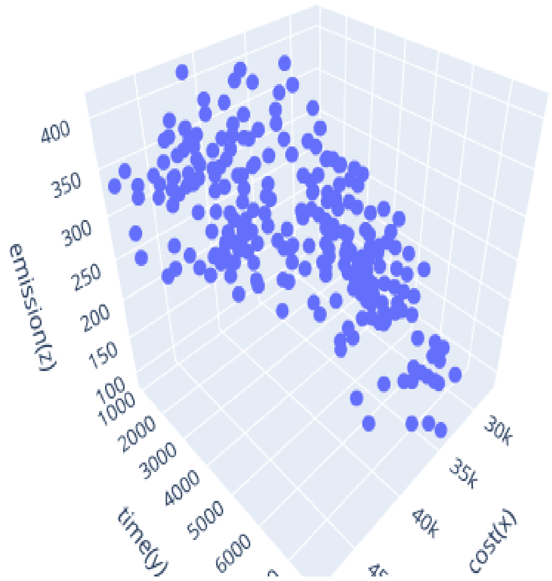

Fig. 1: Pareto Visualization

Additionally, the most economical, expedient, and environmentally friendly route combination is identified (table 7).

Table 7: cheapest, the fastest, and the greenest set of routes

cost	time	carbon	route	BF	sum
27525.0	4344.0	225.0	0,5,7,4,3,11,6,2,10,9,1,8,0	B,B,F,F,F,B,F,F,B,B,B,B	32094.0
28642.0	4655.0	173.0	0,5,4,7,6,11,3,2,10,9,1,8,0	B,B,F,F,B,B,B,F,B,B,B,B	33470.0
29183.0	4141.0	248.0	0,9,8,1,10,2,6,11,3,4,7,5,0	B,B,F,F,B,F,B,F,F,F,B,B	33572.0
29735.0	3775.0	260.0	0,9,8,1,10,2,6,11,3,4,7,5,0	B,B,F,F,F,F,F,F,F,B,B,B	33770.0
28326.0	5505.0	191.0	0,5,7,4,3,11,6,2,10,9,1,8,0	B,B,B,F,F,B,F,B,B,B,B,B	34022.0

Best route is visualized in fig 2.

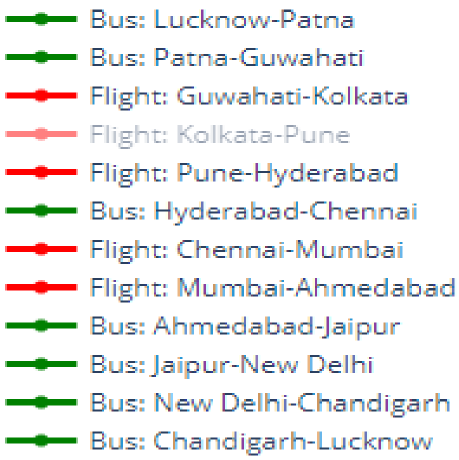

Fig. 2: Best route (cheapest, the fastest, and the greenest)

5. Conclusion

Public transportation system expansion potential has received a lot of attention as of late because of the rising demand for transportation services in most major cities throughout the globe. There are several factors to consider while organizing a trip, such as the best way to get there and how long it will take. We present a computationally efficient elitist multi-objective evolutionary algorithm that uses a non-dominated sorting strategy to solve this issue. The model maximizes a Pareto front (trade-off) between reducing capital expenditures and minimizing network delays while pursuing dual target functions.

Afterward, we conduct a thorough evaluation of a suggested method. The data demonstrate that the suggested method yields optimum Pareto front solutions, with strong convergence to the genuine optimal front and a characteristic of uniform distribution. A useful tool for real-world planning, the suggested method can be implemented.

Author contributions

Jyoti Singh 1: Conceptualization, Methodology, Software, Writing-Reviewing. **Shahnaz Fatima 2:** Data curation, Investigation**. Alok Singh Chauhan 3:** Methodology, Visualization, Software, Writing-Reviewing and Editing.

Conflicts of interest

The authors declare no conflicts of interest.

References

[1] R. Tiwari, M. Husain, S. Gupta, and A. Srivastava, "Improving ant colony optimization algorithm for data clustering," in ICWET 2010 - International Conference and Workshop on Emerging Trends in Technology 2010, Conference Proceedings, 2010. doi: 10.1145/1741906.1742026.

[2] J. Liang et al., "A Survey on Evolutionary Constrained Multi-objective Optimization," IEEE Transactions on Evolutionary Computation, 2022, doi: 10.1109/TEVC.2022.3155533.

[3] S. Petchrompo, D. W. Coit, A. Brintrup, A. Wannakrairot, and A. K. Parlikad, "A review of Pareto pruning methods for multi-objective optimization," Comput Ind Eng, vol. 167, p. 108022, May 2022, doi: 10.1016/J.CIE.2022.108022.

[4] R. Li, W. Gong, and C. Lu, "Self-adaptive multi-objective evolutionary algorithm for flexible job shop scheduling with fuzzy processing time," Comput Ind Eng, vol. 168, p. 108099, Jun. 2022, doi: 10.1016/J.CIE.2022.108099.

[5] X. Yao, W. Li, X. Pan, and R. Wang, "Multimodal multi-objective evolutionary algorithm for multiple path planning," Comput Ind Eng, vol. 169, p. 108145, Jul. 2022, doi: 10.1016/J.CIE.2022.108145.

[6] R. G. Tiwari, M. Husain, V. Srivastava, and K. Singh, "A hypercube novelty model for comparing E-commerce and M-commerce," in ACM International Conference Proceeding Series, 2011. doi: 10.1145/1947940.1948068.

[7] Y. Liu et al., "multi-objective optimal scheduling of automated construction equipment using non-dominated sorting genetic algorithm (NSGA-III)," AutomConstr, vol. 143, p. 104587, Nov. 2022, doi: 10.1016/J.AUTCON.2022.104587.

[8] W. Deng et al., "An enhanced fast non-dominated solution sorting genetic algorithm for multi-objective problems," Inf Sci (N Y), vol. 585, pp. 441–453, Mar. 2022, doi: 10.1016/J.INS.2021.11.052.

[9] J. Tang, Y. Yang, W. Hao, F. Liu, and Y. Wang, "A Data-Driven Timetable Optimization of Urban Bus Line Based on Multi-Objective Genetic Algorithm," IEEE Transactions on Intelligent Transportation Systems, vol. 22, no. 4, pp. 2417–2429, Apr. 2021, doi: 10.1109/TITS.2020.3025031.

[10] V. Sathiya and M. Chinnadurai, "Evolutionary Algorithms-Based Multi-Objective Optimal Mobile Robot Trajectory Planning," Robotica, vol. 37, no. 8, pp. 1363–1382, Aug. 2019, doi: 10.1017/S026357471800156X.

[11] F. Tan, Z. Chai, and Y. Li, "Multi-objective evolutionary algorithm for vehicle routing problem with time window under uncertainty," Evolutionary Intelligence 2021, pp. 1–16, Oct. 2021, doi: 10.1007/S12065-021-00672-0.

[12] F. Stapleton, E. Galván, G. Sistu, and S. Yogamani, "Neuroevolutionary multi-objective approaches to trajectory prediction in autonomous vehicles," GECCO 2022 Companion - Proceedings of the 2022 Genetic and Evolutionary Computation Conference, pp. 675–678, Jul. 2022, doi: 10.1145/3520304.3528984.

[13] J. Dutta, P. S. Barma, A. Mukherjee, S. Kar, and T. De, "A hybrid multi-objective evolutionary algorithm for open vehicle routing problem through cluster primary-route secondary approach," International Journal of Management Science and Engineering Management , vol. 17, no. 2, pp. 132–146, 2022, doi: 10.1080/17509653.2021.2000901.

[14] T. V. Ramachandra and Shwetmala, "Emissions from India's transport sector: Statewise synthesis," Atmos Environ, vol. 43, no. 34, pp. 5510–5517, Nov. 2009, doi: 10.1016/J.ATMOSENV.2009.07.015.

[15] "Aviation emissions." https://www.carbonindependent.org/22.html (accessed Jan. 28, 2023).

[16] Juan Garcia, Guðmundsdóttir Anna, Johansson Anna, Maria Jansen, Anna Wagner. Machine Learning for Decision Science in Healthcare and Medical Systems. Kuwait Journal of Machine Learning, 2(4). Retrieved from http://kuwaitjournals.com/index.php/kjml/article/view/210

[17] Sharma, R., & Dhabliya, D. (2019). Attacks on transport layer and multi-layer attacks on manet. International Journal of Control and Automation, 12(6 Special Issue), 5-11. Retrieved from www.scopus.com

[18] Juan Garcia, Guðmundsdóttir Anna, Maria Jansen, Johansson Anna, Anna Wagner. Exploring Decision Trees and Random Forests for Decision Science Applications. Kuwait Journal of Machine Learning, 2(4). Retrieved from http://kuwaitjournals.com/index.php/kjml/article/view/211