Zachariah *et al.*
Journal of Electrical Systems and Inf Technol (2023) 10:35
https://doi.org/10.1186/s43067-023-00099-w

Journal of Electrical Systems and Information Technology

RESEARCH

Open Access

MRDPGA: a multiple restart dynamic population genetic algorithm for scheduling road traffic

Babangida Zachariah[1], Sanjay Misra[2*], Philip O. Odion[1] and Saidu R. Isah[1]

*Correspondence: sanjay.misra@ife.no

[1] Department of Computer Science, Nigerian Defence Academy, Kaduna, Nigeria
[2] Department of Applied Data Science, Institute for Energy Technology, Halden, Norway

Abstract
A genetic algorithm is a biologically inspired stochastic approach to finding solutions to optimization problems. However, unlike its deterministic counterpart, it cannot guarantee a globally optimal solution since it may be trapped within a local optimum of the search space. Most researchers have focused on proposing new techniques for various parameters of genetic algorithms. That is a mutation, crossover, or selection algorithm. This research proposes a modification to the standard genetic algorithm, which may serve as a framework that can integrate any of these parameters for their contribution to the final solution of the genetic algorithm. The multiple restart dynamic population genetic algorithm (MRDPGA) proposed in this research was used in training the parameters of the adaptive neuro-fuzzy inference system (ANFIS) for scheduling road vehicular traffic flows. The results of training the ANFIS models based on the different clustering methods showed that the MRDPGA-based ANFIS controller performed better with the mean square error (MSE) of 0.299 and root mean square error (RMSE) of 0.547 in the training phase; and MSE of 0.272 and RMSE of 0.521 in the testing phase. Using the controllers for traffic flow scheduling, the results showed that the MRDPGA-trained controllers performed better in terms of average waiting time (AWT) minimization and total arrived vehicles (TAV). The best-performing controller achieved 50.40% AWT minimization and 21.44% TAV improvement. Analyzing the results using a one-tailed *t*-test for paired two-sample means showed that the MRDPGA algorithm had a significant impact on the controllers. Particularly the FCM controller, where ($p=0.0038$) and ($p=0.0003$) for AWT and TAV at a 95% confidence level. Thus, MRDPGA algorithms are recommended for further assessment in other optimization problems to ascertain their performance in those problem domains.

Keywords: Genetic algorithm, ANFIS, Congestion, Roundabout, Optimization

Introduction

The concept of optimization, which is concerned with the effectiveness and efficiency of solutions to problems, has become an interesting research topic. It involves determining a mix of problem parameters in appropriate proportions so that the solution is either maximized or minimized as required [1]. Optimization may be deterministic or stochastic. A deterministic optimization approach often fails when the scale of the optimization problem becomes large and complex, or the internal working mechanism of the optimization is not known [2, 3]. Stochastic approaches can deal with these problems associated with deterministic approaches even though they also have the challenge of guaranteeing optimal global solutions [4, 5]. One of the most notable approaches to stochastic optimization is the evolutionary algorithmic technique [6, 7].

Evolutionary algorithms are biologically inspired techniques that essentially perform a search for a global solution within the possible solution space of the optimization [8]. The genetic algorithm (GA) is one of the most notable evolutionary algorithms. It is capable of dealing with large and complex problems, which are either discrete or continuous. It allows for the specification of adequate objective requirements and constraints as well as generates several possible optimal solutions. Therefore, GA has found several applications in diverse optimization problems. Examples include constraint function [6], two-dimensional rectangular packing problem [9], water distribution networks [10], classification problems [11], profit maximization [12], traffic flow control [13–17], and many other areas [18, 19].

Though GAs [20], in their several variants, have several advantages and have been applied in several optimization problems, they are often confronted with some fundamental challenges. Like any stochastic approach, GA has performance bottlenecks, which can be minimized with the application of parallel computing platforms. Also, GA may be trapped in local optima and unable to generate the global optima [4, 9]. Therefore, this research proposes a modified genetic algorithm technique that minimizes the probability of GA being trapped in local optima even with the increase in problem size. That is, the probability of attaining global optima is improved.

To assess the effectiveness of the proposed GA technique, the modified GA was used to tune the adaptive neuro-fuzzy inference system (ANFIS) for adaptive control of road vehicular traffic flows. Even though a classical problem, road traffic control remains a major problem in modern cities of the world [21]. Traffic congestions problems have been approached differently in terms of factors affecting free flows, from queueing [22], driver and driving factors [23], traffic rules violations [24], to signal optimization [25, 26] etc. However, desired results are yet attained in terms of minimization of vehicular waiting time (or delays), especially in modern cities. We have taken the city Kaduna of Nigeria to implement our model. Thus, it’s the case study used to assess the performance of the modified GA.

The major contributions of this work include:

- To develop a new genetic algorithm technique based on population modifications and multiple restarts.
- To apply the newly developed genetic algorithm technique for the optimization of ANFIS controller parameters.
- To estimate the vehicular hourly arrival rates at various roundabouts, and the enhancement of road traffic flows within a complex road network.

The remaining part of this paper is organized as follows: the review of the most relevant literature is presented in section "Literature review". In section "Methodology", the methodology describes the newly modified genetic algorithm, ANFIS controller, and simulation modeling. In section "Results and discussion", the results and discussion are presented. Section "Conclusion" presented the conclusion and recommended the future direction of the paper.

Literature review

The standard genetic algorithm (SGA) is a five-step technique that performs population initialization, evaluation, selection, crossover, and mutation [27]. Due to the limitations of SGA, some researchers have attempted to address the challenges by proposing varying modifications or techniques that suit specific problem domains; and a few have considered generic modifications to the standard GA. This section presents a review of the relevant literature survey and presents a summary in Table 1.

Dao et al. [4] proposed a new GA technique that involved adaptively restarting the GA. The proposed approach was tested on two benchmark functions, Hump with two dimensions and Rastrigin with fifteen dimensions. Results showed that the proposed method achieved better results at estimating the global optima of the functions compared to the other approaches. However, the study only performs population modification when restarting the genetic algorithm. Also, for every restart cycle, a new set of individuals from the population. These imply that the advantages of the best individuals in a given generation do not contribute to the next restart cycle. Also, close observation of the genetic algorithm reveals that though it may appear to have been trapped to local optima, at times, it can break out after several generations. Thus, jumping to restart the next cycle after there has not been an improvement in the best solution for a given number of generations may not necessarily be the best technique.

Potuzak [3] performed road traffic load balancing across multiple road intersections using distributed standard genetic algorithm. To perform the traffic load balancing, the average number of vehicles on each lane is obtained and used for traffic division into various sub-networks of the considered complex road network. Due to the computational bottlenecks of genetic algorithms, a parallel and distributed computing model was used. The results of simulating the proposed approach were reported to have shown significant minimization of computational time requirement and interesting results of load balancing. However, the use of a standard genetic algorithm implies that the proposed approach suffers from the general challenge of local optima associated with standard genetic algorithms. The author of another study combined parallel and distributed computing for the genetic algorithm with graph coarsening to solve the same problem of load balancing [28]. However, the challenges associated with the genetic algorithm and spectral and cut guarantees associated with graph coarsening are the study's limitations.

Luis et al. [13, 14] modeled complex isolated intersection designs and enhance vehicular throughput of the intersections. A model that calculated the intersection inflow's

Table 1 A summary of related literature

Author(s)	Objective(s)	Tools/techniques	Results	Limitations
Dao et al. [4]	Global optimization	Adaptive restart and elite chromosomes in genetic algorithm	Performed better than the benchmark approaches	Population size modification is only at a restart; a new population may not guarantee the early achievement of global optima
Potuzak [3]	Road traffic network load balancing	Distributed genetic algorithm computation and traffic division	Reduced computation time reported	Suffers from local optima challenge
Potuzak [28]	Road traffic network load balancing	Distributed genetic algorithm and graph coarsening	Reduced computation time reported	Suffers from local optima challenges as well as spectral and cut guarantees challenges associated with graph coarsening
Luis et al. [13, 14]	Throughput: traffic flow enhancement	Standard genetic algorithm and cellular automata simulations	Achieved between 9.21 and 36.98% throughput improvement	Associated challenges of standard genetic algorithms
Basak [29]	Minimize the effect of a constant control operator	Adaptive mutation approach in genetic algorithm	Better performance reported	Focused on a single parameter of the genetic algorithm
Muzid [30]	Global optimization	Fuzzy logic approach to determining crossover and mutation probability	Better results compared to the standard genetic algorithm technique	The choice of population size and boundaries remain a challenge. Non-adaptability of fuzzy logic is a problem
Villalba-morales and Ramírez-echeverry [31]	Steel trusses optimization in three-dimensional space	Multi-chromosome and self-adaptive parameters	35% weight minimization achieved	Hamming cliffs, uninformed precision, and uneven schema importance are challenges associated with binary-coded genetic algorithms
Mao et al. [15]	Traffic control optimization	Combined genetic algorithm and machine learning regression	Reported 43% minimization of average waiting time	Suffers from computational complexity and requires that the machine learning regression model be adequately trained
Al-Madi and Hnaif [32]	Minimizing congestion and duration	Human community-based genetic algorithm	Achieved 83% and 13% in congestion and duration minimization	Possibility and impact of ineffective modeling of problem constraints

points of conflict was proposed, and standard genetic algorithm techniques were used to optimize vehicular arrival rates. The implementation of the cellular automata simulator and simulation of various traffic scenarios showed that the proposed approach achieved improvement within the range of 9.21% and 36.98%. However, the utilization of the standard genetic algorithm technique suggests that the proposed approach may suffer from similar challenges to standard genetic algorithms.

Basak [29] considered that the limitations of SGA were due to constant control operators resulting in the proposed adaptive mutation based on rank order. The result reported showed better performance but had the limitation of focusing on a single operator. This may not yield the best result as genetic algorithms require a proper mix of various parameters. Also, in an attempt to address the same problem of constant control operator parameters, Muzid [30] proposed the use of fuzzy logic to determine the probabilities of mutation and crossover operators as well as population size based on fitness value. The results reported showed that the fuzzy logic-based approach to determining crossover and mutation probability showed better results compared to when the standard genetic algorithm was used. However, the challenge of the non-adaptability of fuzzy logic suggests a limitation of the study.

Villalba-morales and Ramírez-Echeverry [31] presented a weight optimization of a three-dimensional steel truss using a meta-heuristic search. The utilization of multi-chromosomal GA with self-adapting variables ensured a good mix of parameters that resulted in the minimization of the weight of three-dimensional steel. The authors reported that 35% minimization was achieved. However, the study may be said to have binary-coded genetic algorithm challenges that include hamming cliffs, uninformed precision, and uneven schema importance are challenges associated with binary-coded genetic algorithms.

Mao et al. [15] presented a single framework containing a genetic algorithm and machine learning for the optimization of traffic flow control. SGA was used for the determination of phase durations, and a machine learning regression technique was used to determine SGA operation parameters. The utilization of the new framework as a controller for traffic control problems with non-recurrent incidences was reported to have yielded promising results. However, the solution suffers from computational complexity and requires that the machine learning regression model be adequately trained.

Al-Madi and Hnaif [32] considered and apply a human community-based genetic algorithm to solve traffic congestion problems. The special-cased genetic algorithm can introduce constraints in the crossover and mutation operations to improve the chances of maintaining diversity in the population. The results of using this algorithm showed that it achieved shorter congestion durations by 13% and minimized congestion by 83% in the considered road traffic congestion scenarios. The challenge associated with the human community-based genetic algorithm is the effectual and adequate modeling of constraints. When constraints are not properly and adequately considered for a given problem, it may not perform as expected.

These studies focused on particular GA operations. Therefore, a generic technique that integrates these proposed techniques is proposed in this research. The new modified genetic algorithm is referred to multiple restart dynamic population algorithm.

Methodology

This section presents the modified genetic algorithm and the traffic control problem used to assess the performance of the genetic algorithm.

Proposed genetic algorithm

The multiple restart dynamic population genetic algorithm (MRDPGA) proposed in this research is presented in Algorithm 1. First, the general and local termination criteria are selected and used to control various iterations of the algorithm. The initial population size, crossover, and mutation percentages are chosen. The population size continually changes during the local or general iterations of the algorithm. To determine when population size is modified during the local termination criteria, a threshold criterion is chosen and used to assume when the algorithm is trapped within local optima. At the restart, the population size modification is performed and a subset of the best individuals from the previous cycle of the MRDPGA are migrated to form part of the newly initialized population of the restart cycle. This is to increase the chance of having the best solution by having the best individuals mixed with an entirely new random population.

Applying the developed algorithm to the particular problem of training the Adaptive Neuro-Fuzzy Inference System, the objective function used for fitness evaluation was Mean Square Error (MSE) and Root Mean Square Error (RMSE). The general and local termination criteria were chosen to be some iterations to be performed. The general termination criteria were either a minimum MSE of 0.0002 or five iterations ($C_{\mathrm{g}}=5$); while the local termination criteria were chosen to be 1000 iterations ($C_{\mathrm{l}}=1000$). The initial population size, p was fifty ($p=50$), and the threshold, T_{h} of local iterations without improvement in the local solution was ten iterations ($T_{\mathrm{h}}=10$).

Since this research focused on proposing a modified genetic algorithm technique, default Roulette-Wheel selection was used for population selection. Forty percent (40%) crossover rate and seventy percent (70%) mutation rate were used for the genetic operations. The trapped threshold value of ten (10) and initial population size of fifty (50) were used. To control population size growth and shrinkage, it was bounded above and below by fifty percent (50%). This was necessary to minimize performance (runtime) bottlenecks of the MRDPGA.

Algorithm 1: Multiple Restart Dynamic Population Genetic Algorithm

```
Input:
    Population size, p
    Global/General Termination Criteria, C_g
    Local/Sub Termination Criteria, C_l
Output:
    Global Best Solution, S_g
----------------------------------------------------------------
Begin
    Initialise counters c, l, n
    While c ≠ C_g
        if P_b = ∅
            Initialize random Population, P of size, p
        else
            Initialize random Population, P_i of size p/2
            Set P = Merge(P_b, P_i)
        endif
        Set P = EvaluateAndSort(P)
        Set S_g = P[1]
        While l ≠ C_l
            Set P_x = Crossover(P)
            Set P_m = Mutation(P)
            Set P = Merge(P, P_x, P_m)
            Set P = EvaluateAndSort(P)
            if P[1] ≥ S_l
                Set n = n + 1
                if n ≥ T_h
                    Modify population size, p
                    Set P = {P[j] : j = 1,2,3, ..., p}
                endif
            else
                Set S_l = P[1]
                Set n = 0
            endif
            Update l
        endwhile
        if S_l < S_g
            Set S_g = S_l
        Endif
        Set P_b = {P[j] : j = 1,2,3, ... , p/2}
        Modify population size, p
        Update c
    endwhile
end
```

ANFIS controller modeling

To assess the performance of MRDPGA, ANFIS controllers based on the clustering methods were modeled. The grip partition clustering (GPC), subtractive clustering (SC), and fuzzy C-means (FCM) clustering controllers modeled were trained using the default training algorithm, gradient descent, and least squares estimations. They were also trained using a standard genetic algorithm and multiple restart dynamic population genetic algorithm. The objective function for the genetic algorithm was the mean square error (MSE) and root means square error (RMSE). These parameters were used to determine performance variance between the ANFIS default training algorithm, SGA, and MRDPGA. The ANFIS controllers were modeled to have two inputs (waiting time

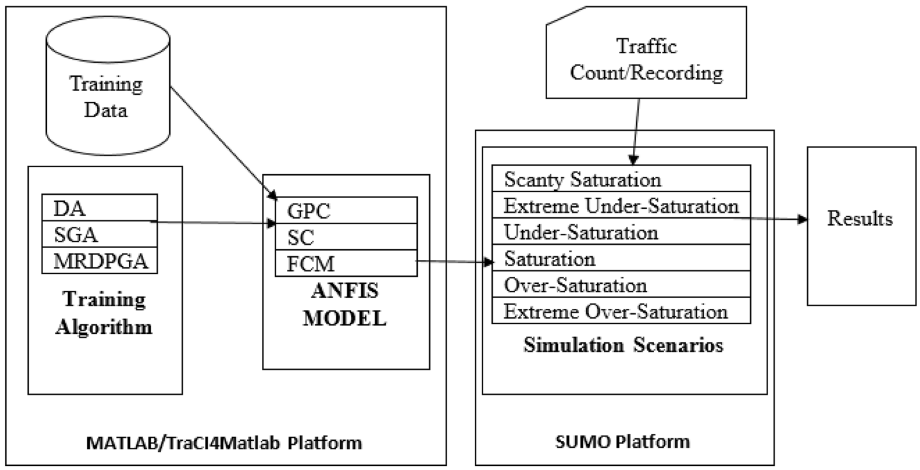

Fig. 1 Performed research activities framework

Table 2 Grid partition model training performances

Controller	Training		Testing	
	MSE	RMSE	MSE	RMSE
DefaultGridPartFIS	0.31706	0.56308	0.3169	0.56294
SGAGridPartFIS	0.32004	0.56572	0.30995	0.55673
MRDPGAGridPartFIS	0.30201	0.54955	0.32684	0.5717

and queue length) and one output (phase duration). The GPC controller was modeled to have a Gaussian membership function and five fuzzy sets for each of the inputs. The SC controller was modeled to have a radius of ten (10), and the FCM was modeled to have ten (10) clusters.

Simulation modeling

Simulation of traffic scenarios was used to assess the performance of the modeled ANFIS controllers trained using the default algorithm, SGA and MRDPGA. Particularly, microscopic simulation was implemented using the simulation of urban mobility (SUMO) open-source platform. To control SUMO objects such as traffic lights and access simulation parameters such as vehicular waiting time and flows queue lengths, a traffic control interface for matrix laboratory (TraCI4Matlab) was used.

The considered road traffic flow network in this research was the road section from Kalapanzi Army barrack and St. Gerrard's Catholic Hospital in Kakuri to General Hassan Katsina House in Kaduna, Nigeria. To capture a near-reality road network design, OpenStreetMap was used to retrieve the considered road network with little modification to include the new conversion of the Leventis roundabout to an underpass. The considered road network consists of eight roundabouts and one underpass with lanes ranging from one to three.

The traffic flow rates and vehicles per hour (VPH) were based on vehicular sample count performed at the two strategic roundabouts connecting the northern and southern parts of the metropolis. The peak period had an hourly average arrival rate of

Table 3 Subtractive clustering model training performances

Controller	Training		Testing	
	MSE	RMSE	MSE	RMSE
DefaultSubCluFIS	2.5284	1.5901	2.4621	1.5691
SGASubCluFIS	2.2908	1.5135	2.2936	1.5145
MRDPGASubCluFIS	0.7571	0.87011	0.78551	0.88629

Table 4 Fuzzy C-means clustering model training performances

Controller	Training		Testing	
	MSE	RMSE	MSE	RMSE
DefaultFCMFIS	0.62241	0.78893	0.63949	0.79968
SGAFCMFIS	0.81655	0.90363	0.78317	0.88497
MRDPGAFCMFIS	0.29886	0.54668	0.27152	0.52108

350VPH, and the off-peak period had 103VPH. These values were scaled to have different traffic scenarios considered to be representative of various traffic flow possibilities. The vehicles were into categories of light, moderate or heavy vehicles based on the Federal Highway Administration classification. Their speed specification in kilometers per hour (KMPH) was based on these categories, and as provided by Federal Road, Safety Corps recommended speeds for vehicles within cities. Light and moderate vehicles had a speed of 50KMPH, while heavy vehicles had a speed of 45KMPH. The traffic controller programmer was implemented such that a fraction of the assigned green phase duration for each traffic flow is used for the clearing of the roundabouts. This technique enhances the flows and minimizes the probability of deadlocks at the roundabouts. For this research, 40% of the green phase duration was used for the clearing of the roundabouts. That is, the green wave is assigned only to the circulating flows of the roundabouts.

In summarizing the research method, a framework of activities performed is presented in Fig. 1. That is, with traffic training data, the ANFIS models are trained using either a default algorithm (DA), a standard genetic algorithm (SGA), or a multiple restart dynamic population genetic algorithm. The ANFIS controllers resulting from the training of ANFIS models are then used for traffic scheduling. The considered traffic scenarios were based on the extrapolation of traffic count data and estimation performed at selected roundabouts of the considered road network.

Results and discussion

This section presents the results of training the ANFIS controllers using the default training algorithm, SGA and MRDPGA. The results of the simulation of traffic flow scenarios using the different controllers are also presented.

Controller training performance

In Table 2, the GPC controller training performance results of considered training algorithms are presented. SGA-based GPC-trained (SGAGridPartFIS) had the lowest

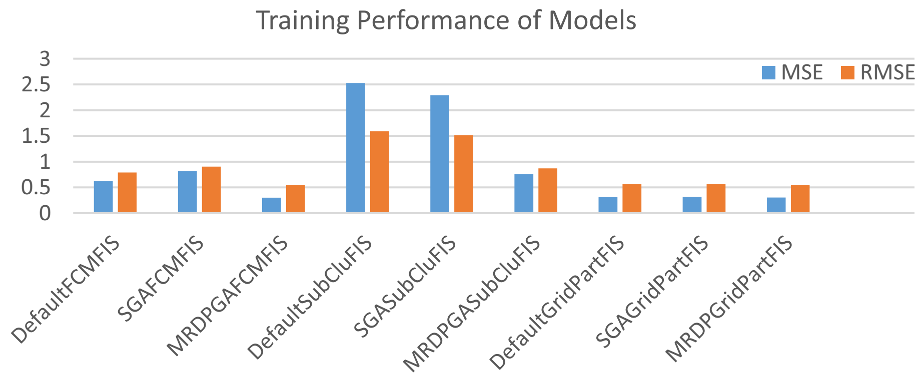

Fig. 2 Summary of controller training performance

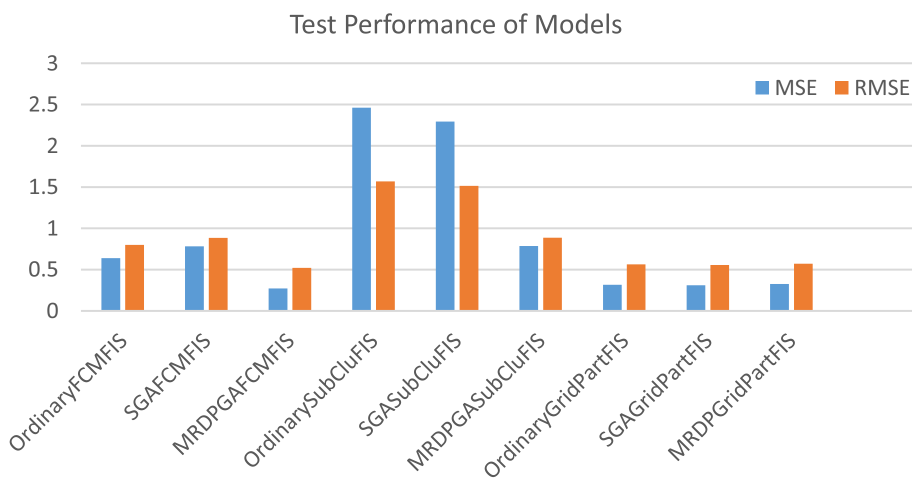

Fig. 3 Summary of controller testing performance

performance results in terms of MSE and RMSE in the training phase of the controller. The default-based GPC-trained (DefaultGridPartFIS) performed better than the SGA-based GPC-trained but below the MRDPGA-based GPC-trained (MRDPGAGridPartFIS). That is, the GPC controller outperformed the other two controllers to demonstrate its ability at finding the optimal solution to an optimization problem.

In Table 3, the SC controllers' training performance results are presented. The default-based SC-trained controller (DefaultSubCluFIS) underperformed compared to the other two controllers both in the training and testing phases. The MRDPGA-based SC-trained controller (MRDPGASubCluFIS) outperformed the other two controllers having the lowest values of MSE and RMSE both in the training and testing phases. This also shows that the MRDPGA training algorithm was better at searching for the optimal solution for the SC models.

In Table 4, the performance results of using the three considered algorithms for training the FCM model parameters are presented. The results showed that the SGA-based FCM-trained controller (SGASubCluFIS) underperformed compared to the other trained controllers. However, the MRDPGA-based FCM-trained controller (MRDPGASubCluFIS) performed best to outperform both the default-based FCM-trained and the SGA-based FCM-trained controllers. Having lower values of MSE and RMSE both

Table 5 Grid partitioned model simulation results

Scenarios	Default		SGA		Perf. of SGA over default		MRDPGA		Perf. of MRDPGA over default		Perf. of MRDPGA over SGA	
	AWT (s)	TAV	AWT (s)	TAV	% AWT	% TAV	AWT (s)	TAV	% AWT	% TAV	% AWT	% TAV
Scanty saturation	26.26	3858	26.26	3858	0	0	27.98	3778	−6.54	−2.12	−6.54	−2.12
Extreme under-saturation	278.01	2905	278.01	2905	0	0	188.01	4575	32.37	36.50	32.37	36.50
Under-saturation	482.32	3561	482.32	3561	0	0	279.38	4968	42.08	28.32	42.08	28.32
Saturation	760.42	4213	760.42	4213	0	0	389.93	5085	48.72	17.15	48.72	17.15
Over-saturation	624.16	4584	624.16	4584	0	0	415.98	5495	33.36	16.58	33.36	16.58
Extreme over-saturation	747.49	4032	747.49	4032	0	0	408.92	5665	45.29	28.83	45.29	28.83
Average					0	0			32.55	20.88	32.55	20.88

Table 6 Subtractive clustered model simulation results

Scenarios	Default		SGA		Perf. of SGA over default		MRDPGA		Perf. of MRDPGA over default		Perf. Of MRDPGA over SGA	
	AWT (s)	TAV	AWT (s)	TAV	% AWT	% TAV	AWT (s)	TAV	% AWT	% TAV	% AWT	% TAV
Scanty saturation	33.70	3791	27.64	3854	17.97	1.64	28.83	3778	14.45	− 0.34	− 4.29	− 2.01
Extreme under-saturation	504.56	2938	365.95	3097	27.47	5.13	171.78	4529	65.96	35.13	53.06	31.62
Under-saturation	484.64	3653	626.75	3408	− 29.32	− 7.19	275.62	5214	43.13	29.94	56.03	34.64
Saturation	549.40	4213	570.80	4186	− 3.90	− 0.65	451.49	5074	17.82	16.97	20.90	17.50
Over-saturation	629.61	4438	800.79	4462	− 27.19	0.54	550.55	5547	12.56	19.99	31.25	19.56
Extreme over saturation	836.35	4278	861.90	4088	− 3.06	− 4.65	261.87	5255	68.69	18.59	69.62	22.21
Average					− 3.00	− 0.86			37.10	20.05	37.76	20.59

in the training and testing phases demonstrates the ability of the MRDPGA algorithm to optimally tune the parameters of the FCM models.

The MRDPGA-based controllers showed that irrespective of clustering methods used, the algorithm generally performed better at finding the optimal solution to the optimization of ANFIS-based controllers. Comparing the results of the MRDPGA-based controllers shows that the MRDPGA algorithm performed best at tuning the parameters of FCM model, followed by the GPC model and then the SC model (Figs. 2 and 3). This may be a result of the ANFIS parameters used.

Simulation results analysis

The road traffic simulation results were assessed based on the vehicular average waiting for time (AWT) and the number of vehicles that have arrived at their destination [19]—total arrived vehicles (TAV). Five different traffic scenarios are considered in terms of hourly arrival rates.

In Table 5, the road traffic simulation results of the GPC controllers are presented. The results showed that there was no performance variance between the SGA-based and GPC-trained controllers. However, comparing the results of the MRDPGA-based GPC-trained controller to that of the default-based GPC-trained and SGA-based GPC-trained controllers, the results showed that the MRDPGA-based GPC-trained controller performed better in terms of both the AWT and TAV by approximately 33% and 21%, respectively.

In addition, analyzing the impact of MRDPGA algorithm on the performance of the implemented grid partitioned clustered controllers, $p = 0.0089$ was obtained from the *t*-Test for paired two sample means in the case of AWT results; and $p = 0.0053$ in the case of TAV results of GPC-trained and MRDPGA-trained controllers as well as for SGA-trained and MRDPGA-trained controllers. These p values show that at a 95% confidence level, there is a statistically significant difference between sampled means of the results obtained by the different controllers. This demonstrates the ability of MRDPGA at obtaining better global solutions to the problem at hand.

In Table 6, the road traffic simulation results using SC controllers are presented. The results showed that the SGA-based GPC-trained controller slightly underperformed compared to the default-based GPC-trained controller both in terms of the AWT and TAV by approximately 3% and 1%, respectively. However, the MRDPGA-based GPC-trained controller outperformed the default-based GPC-trained controller by approximately 37% in AWT and 20% in TAV. It also outperformed the SGA-based GPC-trained controller by approximately 38% and 21% in AWT and TAV, respectively.

In addition, analyzing the impact of MRDPGA on the performance of the controllers, $p = 0.0263$ was obtained AWT and $p = 0.0041$ for TAV from t-Test analysis performed between Default-trained and MRDGPA-trained controllers. In the case of SGA-trained and MRDPGA-trained controllers, $p = 0.0155$ for AWT results and $p = 0.0050$ for TAV results. These also showed that at a 95% confidence level, there is a significant difference between the sampled means of the results obtained using the controllers. A closer look at the p values in these cases showed that the MRDPGA-trained controller had only slight statistical significance variance in the case of AWT results obtained using

Table 7 Fuzzy C-means clustered model simulation results

Scenarios	Default		SGA		Perf. of SGA over default		MRDPGA		Perf. of MRDPGA over default		Perf. of MRDPGA over SGA	
	AWT (s)	TAV	AWT (s)	TAV	% AWT	% TAV	AWT (s)	TAV	% AWT	% TAV	% AWT	% TAV
Scanty saturation	25.62	3859	62.99	3377	− 145.81	− 14.27	31.04	3826	− 21.14	− 0.86	50.72	11.74
Extreme under-saturation	507.57	2647	400.48	3124	21.10	15.27	161.82	4538	68.12	41.67	59.59	31.16
Under-saturation	601.41	3459	776.20	3538	− 29.06	2.23	478.65	4677	20.41	26.04	38.33	24.35
Saturation	614.82	4170	577.32	4470	6.10	6.71	365.47	5475	40.56	23.84	36.69	18.37
Over-saturation	507.00	4530	595.97	4528	− 17.55	− 0.04	273.35	5854	46.09	22.62	54.13	22.65
Extreme over-saturation	530.55	4251	781.91	4240	− 47.38	− 0.26	289.74	5326	45.39	20.18	62.95	20.39
Average					− 35.43	1.61			33.24	22.25	50.40	21.44

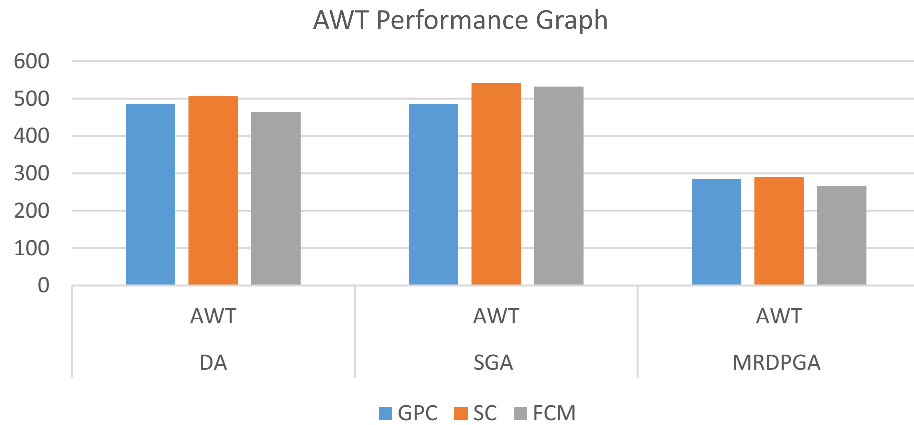

Fig. 4 Graphical representation of controllers' average waiting time

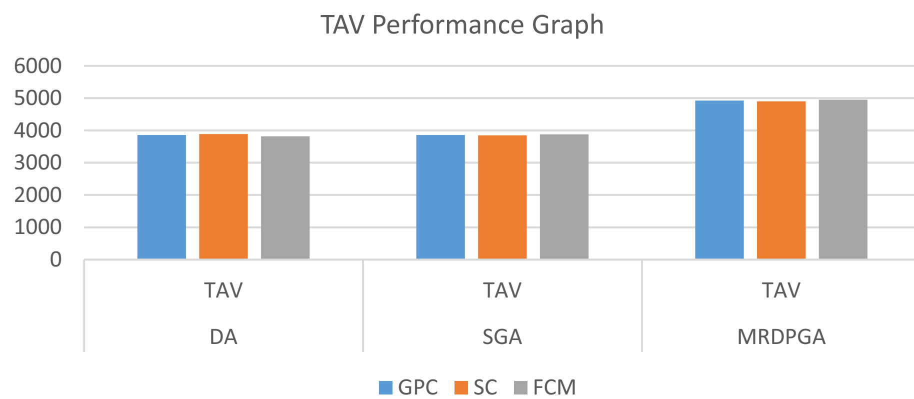

Fig. 5 Graphical representation of controllers' total arrive vehicles

SGA-trained and MRDPGA-trained controllers. However, it showed higher significance in the case of TAV for the controllers.

In Table 7, the road traffic simulation results of using FCM models as controllers are presented. Comparing the results of the SGA-based FCM-trained controller to that of the default-based FCM-trained controller, the results showed that the default-base FCM-trained controller significantly outperformed the SGA-based FCM-trained controller both in terms of the AWT and TAV by approximately 35% and 2%, respectively. Comparing the results of the MRDPGA-based FCM-trained controller to that of the default-based FCM-trained controller, the results showed that the MRDPGA-based FCM-trained controller outperformed the default-based FCM-trained controller by approximately 33% and 22% in terms of the AWT and TAV, respectively. Also, the results of the MRDPGA-based FCM-trained controller outperformed that of the SGA-based FCM-trained controller by approximately 50% in terms of the AWT and 21% in terms of the TAV.

In addition, analyzing the impact of the MRDPGA algorithm on the training of FCM controllers, $p=0.0053$ and $p=0.00396$ were, respectively, obtained for AWT and TAV in t-Test analysis between Default-trained and MRDPGA-trained controllers. In the case of SGA-trained and MRDPGA-trained controllers, $p=0.0038$ and $p=0.0003$ were obtained for AWT and TAV, respectively. These show that there are significant differences in the paired two sample means of the results obtained using the Default, SGA, and MRDPGA-trained controllers, which again demonstrate the impact of the MRDPGA algorithm in the training of the FCM-based controller.

These results showed the MRDPGA-based controller always outperformed the other controllers in their respects. This demonstrates the importance of effectively tuning ANFIS parameters, which the MRDPGA algorithm has distinguished itself in this research. That is, it considered the search space and effectively searched for the optimal results.

A comparison of the MRDPGA-based controllers based on the different ANFIS clustering methods showed that the GPC controller outperformed the SC controller by approximately 4% in terms of AWT but underperformed by approximately less than 1% in terms of TAV. FCM controller outperformed the GPC controller by approximately 11% in terms of AWT and by approximately 1% in terms of TAV. Also, the FCM controller outperformed the SC controller by approximately 12% and 1% in terms of AWT and TAV, respectively. That is, the FCM controller trained using the MRDPGA algorithm essentially demonstrated superiority in this research. This follows from the abilities of FCM to flexibly and adaptively assign data points to different clusters of varying shapes and sizes based on the degrees of the membership of each data point. In addition, FCM can determine the number of clusters as well as deal with irregularities in datasets, which may be a result of uncertainties. When trained with MRDPGA, it acquired the added advantage of searching through a possibly large solution space in such a manner as to enhance the chance of obtaining a global solution in terms of cluster shapes, sizes, and numbers.

The summary of the results obtained is presented in Figs. 4 and 5. In Fig. 4, the graphs of the average waiting times recorded when using the different controllers are presented. It shows that the MRDPGA-trained controllers for all ANFIS models recorded the lowest AWT, with the FCM model being the lowest of all controllers. This implies that the MRDPGA-trained controllers performed better in this respect. Also, in Fig. 5, the average total number of vehicles that have arrived at their destinations, called total arrived vehicles (TAV) for all controllers, are presented. The results showed that the MRDPGA-trained controllers recorded the highest TAV numbers. This implies that the MRDPGA-trained controllers performed better.

It should be noted that the computational complexity of MRDPGA may be higher compared to SGA, especially in cases where the population size of SGA is equal to the initial population size of MRDPGA. This is a result of the fact that MRDPGA runs multiple iterations based on the global and local termination criteria. However, when the number of iterations and population size differ, it may not be the case.

Conclusion

Optimization problems require techniques or algorithms that effectively mix problem parameters in other to guarantee optimal solutions to the problem. This requires searching through the solution space such that the chance of the algorithm being trapped within a local optimum and failing to obtain the global optima is reduced. This research modified a standard genetic algorithm, which has the above-mentioned challenge. The modified algorithm demonstrated its capability in training ANFIS models based on different clustering methods. It recorded the lowest MSE of 0.299 and RMSE of 0.547 in the training phase; and MSE of 0.272 and RMSE of 0.521 in the testing phase. Using the controllers for traffic flow scheduling, the results showed that the MRDPGA-trained controllers performed better in terms of average waiting time (AWT) minimization and total arrived vehicles (TAV). The best-performing controller achieved 50.40% AWT minimization and 21.44% TAV improvement over other controllers.

The modified algorithm is a framework that can integrate other parameter-based optimization of genetic algorithms. That is, the different adaptive population techniques may be used at the local and global population modification stages of MRDPGA. Also, different selection algorithms may be used instead of the used roulette-wheel selection algorithm. To further assess this modified genetic algorithm, it may be assessed with other known optimization problems used as benchmarks.

Acknowledgements
Authors are thankful to their departments for providing research environments and facilities to perform this research.

Author contributions
Babangida Zachariah (BZ), Sanjay Misra (SM), Philip O. Odion (POO), Saidu R. Isah (SRI). BZ, SM, POO, and SRI conceptualized the topic. BZ and SM are involved in methodology, investigation, and validation. SM, POO, and SRI supervised the whole work. All authors reviewed the manuscript.

Funding
No Funding received for this work.

Availability of data and materials
The data is available at https://doi.org/10.5281/zenodo.7109331 also cited the same in the reference list [33].

Declarations

Competing interests
Authors do not have any financial or non-financial interests that are directly or indirectly related to the work submitted for publication.

Received: 12 January 2023 Accepted: 30 May 2023
Published online: 17 July 2023

References

1. Habibi E, Salehi M, Yadegarfar G, Taheri A (2018) Optimization of ANFIS using a genetic algorithm for physical work rate classification. Int J Occup Saf Ergon. https://doi.org/10.1080/10803548.2018.1435445
2. Osigbemeh M, Onuu M, Asaolu O (2016) Design and development of an improved traffic light control system using hybrid lighting system. J Traffic Transp Eng 4(1):88–95. https://doi.org/10.1016/j.jtte.2016.06.001
3. Potuzak T (2016) Optimization of a genetic algorithm for road traffic network division using a distributed/parallel genetic algorithm. In: 9th international conference on human system interactions, pp 21–27. https://doi.org/10.1109/HSI.2016.7529603
4. Dao SD, Abhary K, Marian R (2015) An adaptive restarting genetic algorithm for global optimization. Proc World Congr Eng Comput Sci I:1–5
5. Zhang X, Fan X, Yu S, Shan A, Fan S, Xiao Y, Dang F (2022) Intersection signal timing optimization: a multi-objective evolutionary algorithm. Sustainability 14(1506):1–16. https://doi.org/10.3390/su14031506
6. Mir RN, Khan AA (2017) Optimization of constrained function using genetic algorithm. Comput Eng Intell Syst 8(2):11–15
7. Semet Y, Berthelot B, Glais T, Isberie C, Varest A (2019) Expert competitive traffic light optimization with evolutionary algorithms. In: Proceedings of the 5th international conference on vehicle technology and intelligent transport systems, (VEHITS 2019), pp 199–210. https://doi.org/10.5220/0007732701990210
8. Xiao H, Lee LH (2014) Simulation optimization using genetic algorithms with optimal computing budget allocation. SAGE Trans Soc Model Simul Int 90(10):1146–1157. https://doi.org/10.1177/0037549714548095
9. Li Y, Sang H, Xiong X, Li Y (2021) An improved adaptive genetic algorithm for two-dimensional rectangular packing problem. Appl Sci 11:1–20. https://doi.org/10.3390/app11010413
10. Sangroula U, Han K, Koo K, Gnawali K, Yum K-T (2022) Optimization of water distribution networks using genetic algorithm based SOP—WDN program. Water 14(851):1–24. https://doi.org/10.3390/w14060851

11. Yunanda FA, Wahyuningsih S, Situmorang Z (2020) The accuracies of ANFIS and genetic algorithm with tournament selection on classifying hepatitis data. J Phys Conf Ser 1566:1–12. https://doi.org/10.1088/1742-6596/1566/1/012121
12. Li J, Lu W (2014) Full truckload vehicle routing problem with profits. J Traffic Transp Eng (Engl Ed) 1(2):146–152. https://doi.org/10.1016/S2095-7564(15)30099-4
13. Luis C, Ivan M, Miguel LA (2019a) A variable-length chromosome genetic algorithm to solve a road traffic coordination multipath problem. IEEE Access. https://doi.org/10.1109/ACCESS.2019.2935041
14. Luis C, Miguel LA, Ivan M (2019) Automated optimization of intersections using a genetic algorithm. IEEE Access 7:15452–15468. https://doi.org/10.1109/ACCESS.2019.2895370
15. Mao T, Mihaita A, Chen F, Vu HL (2021) Boosted genetic algorithm using machine learning for traffic control optimization. ArXiv. https://doi.org/10.48550/arxiv.2103.08317
16. Qi A (2018) Research on timing optimization of regional traffic signals based on improved genetic algorithm. In: 3rd international workshop on materials engineering and computer sciences (IWMECS 2018): advances in computer science research, vol 78. pp 102–105
17. Zhou Z, Cai M (2014) Intersection signal control multi-objective optimization based on genetic algorithm. J Traffic Transp Eng (Engl Ed) 1(2):153–158. https://doi.org/10.1016/S2095-7564(15)30100-8
18. Wen-jing W (2018) Improved adaptive genetic algorithm for course scheduling in colleges and universities state of art. Int J Emerg Technol Learn 13(6):29–42
19. Muthee A (2021) The basics of genetic algorithms in machine learning. Engineering Education Community Website. https://www.section.io/engineering-education/the-basics-of-genetic-algorithms-in-ml. Accessed 8 Apr 2022
20. Gampa SR, Das D (2019) Simultaneous optimal allocation and sizing of distributed generations and shunt capacitors in distribution networks using fuzzy GA methodology. J Electr Syst Inf Technol 6(1):1–18
21. Hnaif AA, Al-Madi N, Abduljawad M, Ahmad A (2019) An intelligent road traffic management system based on a human community genetic algorithm. In: 2019 IEEE Jordan international joint conference on electrical engineering and information technology (JEEIT), pp. 554–559. IEEE
22. Pompigna A, Mauro R (2021) A multi-class time-dependent model for the analysis of waiting phenomena at a motorway tollgate. J Traffic Transp Eng (Engl Ed) 8(2):237–256. https://doi.org/10.1016/j.jtte.2020.09.001
23. Ulak MB, Ozguven EE, Moses R, Sando T, Boot W, AbdelRazig Y, Sobanjo JO (2019) Assessment of traffic performance measures and safety based on driver age and experience: a microsimulation based analysis for an unsignalized T-intersection. J Traffic Transp Eng (Engl Ed) 6(5):455–469. https://doi.org/10.1016/j.jtte.2018.05.004
24. Alrejjal A, Moomen M, Ksaibati K (2022) Evaluating the impact of traffic violations on crash injury severity on Wyoming interstates: an investigation with a random parameters model with heterogeneity in means approach. J Traffic Transp Eng (Engl Ed) 9(4):654–665. https://doi.org/10.1016/j.jtte.2021.05.003
25. Jiang Z, Yu D, Luan S, Zhou H, Meng F (2022) Integrating traffic signal optimization with vehicle microscopic control to reduce energy consumption in a connected and automated vehicle environment. J Clean Prod 371:133694. https://doi.org/10.1016/j.jclepro.2022.133694
26. Liang Y, Ren Z, Wang L, Liu H, Du W (2021) Surrogate-assisted cooperative signal optimization for large-scale traffic networks. Knowl Based Syst 234:107542. https://doi.org/10.1016/j.knosys.2021.107542
27. Abdulhamed AA, Tawfeek MA, Keshk AE (2018) A genetic algorithm for service flow management with budget constraint in heterogeneous computing. Futur Comput Inform J 3(2):341–347. https://doi.org/10.1016/j.fcij.2018.10.004
28. Potuzak T (2018) Division of road traffic network based on genetic algorithm and graph coarsening. In: 2018 11th international conference on human system interaction (HSI), pp 484–490. IEEE
29. Basak A (2020) A rank-based adaptive mutation in genetic algorithm. Int J Comput Appl 175(10):49–55
30. Muzid S (2020) An adaptive approach to controlling parameters and population size of evolutionary algorithm. J Phys Conf Series 2nd Int Conf Comput Sci Eng Technol 1430:1–8. https://doi.org/10.1088/1742-6596/1430/1/012048
31. Villalba-Morales JD, Ramírez-Echeverry S (2021) An approximation to the use of self-adaptive genetic algorithms in weight optimization of 3-D steel trusses. Ing y Compet Rev Cient y Technol. https://doi.org/10.25100/iyc.23i1.7337
32. Al-Madi NA, Hnaif AA (2022) Optimizing traffic signals in smart cities based on genetic algorithm. Comput Syst Sci Eng 40(1):65–74. https://doi.org/10.32604/csse.2022.016730
33. Zachariah B, Misra S, Odion OP, Saidu RI (2022) Intersection vehicular traffic flow scheduling datasets [Data set]. Zenodo. https://doi.org/10.5281/zenodo.7109331

Publisher's Note