Evolutionary Automation of Coordinated Autonomous Vehicles

Allen Huang
Department of Computer Science
University of Cape Town, South Africa
allen@allenhuang.net

Geoff Nitschke
Department of Computer Science
University of Cape Town, South Africa
gnitschke@cs.uct.ac.za

***Abstract*—Recently, there has been increased research on adaptive control systems for vehicles that operate on autonomous vehicle only roads. Specifically, roads without current infrastructure constraints of traffic lights, stop signals at intersections or vehicle lanes. This study investigates controller automation for vehicles that must navigate and coordinate with each other on such autonomous vehicle only roads. We comparatively evaluate fitness-function (*objective*) versus behavior-based (*novelty search*) versus hybridized objective-novelty evolutionary search for synthesizing autonomous vehicle coordinated driving behavior. The goal of such evolved coordinated driving behavior is to maximize *effective* (safe) and *efficient* (expedient) autonomous vehicle traffic throughput for given roads. Results indicate that while novelty and hybrid search evolved effective and efficient driving behaviors, these behaviors did not generalize to new roads as well as driving behaviors evolved with objective-based search.**

I. Introduction

Recently there has been increasing academic and industry research attention on producing adaptive control systems for autonomous vehicles [1]. To accommodate such autonomous vehicles there have been proposals that current road and highway infrastructure undergo significant changes. For example, replacing traffic lights and stop signs and allowing autonomous vehicles to coordinate their own interactions so as to avoid collisions and safely navigate through intersections [2]. One method is to design vehicle controllers such that desired coordinated driving behaviors automatically emerge for vehicles driving and interacting on any given road [3], [4].

This study investigates the efficacy of evolutionary controller design methods for enabling effective and efficient coordinated driving behavior for autonomous vehicle traffic on roads built exclusively for autonomous vehicles. That is, roads and highways without the current road infrastructure of traffic lights, intersection stop signals and vehicle lanes [2].

In the theme of *autonomous vehicle only* roads in future metropolitan transportation systems, the *Autonomous Intersection Management* (AIM) control protocol was proposed for coordinating autonomous vehicle traffic through intersections without traffic signals [5]. AIM enabled many autonomous vehicles to concurrently and efficiently transit through intersections, effectively reducing the delay of vehicles by orders of magnitude compared to intersections with traffic signals [6].

However, a key limitation of AIM and related work [7], is it only managed autonomous vehicle traffic flow through intersections and with few exceptions did not account for uncertainty or dynamic obstacles such as pedestrians [2] or mechanical failures [8]. Perfect traffic conditions and sensory information was assumed, whereas incomplete and noisy sensory environments must be accounted for if autonomous vehicles are to be deployed on public roads and highways.

In this study, individual vehicle driving behavior was adapted so as all vehicles elicited *effective* and *efficient* coordinated driving behaviors for traversal of any given road. Effectiveness and efficiency were performance metrics, equating to the number of collisions and time taken to traverse a given road. Task difficulty was the number of vehicles (traffic density) and obstacles (static and dynamic) on the road. The goal was thus to evolve individual vehicle controllers such that when multiple vehicles interacted, desired coordinated driving behavior emerged for the given task environment (road configuration and other vehicles and obstacles on the road).

Previous work applied *neuro-evolution* methods for evolving coordinated vehicle driving on roads [3], and through intersections [4], without stop signals or traffic lights to assist vehicle coordination and navigation. In such cases, evolutionary search for optimal vehicle controllers was directed by objective-based search (fitness functions). This study's contribution was a comparative evaluation of *objective-based*, *non-objective* (behavioral diversity maintenance), and *hybrid* search methods for directing controller evolution and the synthesis of effective and efficient coordinated driving behavior.

This evolutionary search method comparison was motivated by indications that behavioral diversity maintenance improves evolved multi-agent behavior quality [9], [10]. Such work indicated behavioral diversity to be especially beneficial for evolving complex coordinated multi-agent behaviors in task domains such as *RoboCup* [11] and *swarm robotics* [9].

This study has two key differences to previous work [3] [4]. First, to demonstrate the efficacy of applying *objective-based* versus *novelty* [12] versus *hybrid* search, to direct the evolution of coordinated-driving behavior. Second, to evaluate the capacity for evolved coordinated behaviors to generalize to, effectively and efficiently operating vehicles, on a range of new test road-ways. These objectives were motivated by a current deficiency in research on the efficacy of comparative evolutionary search methods applied to evolve *autonomous vehicle* traffic behavior, where such evolved traffic behavior effectively and efficiently operates in more general task environments. For example, road networks necessitating more complex types of driving behaviors and coordination between vehicles.

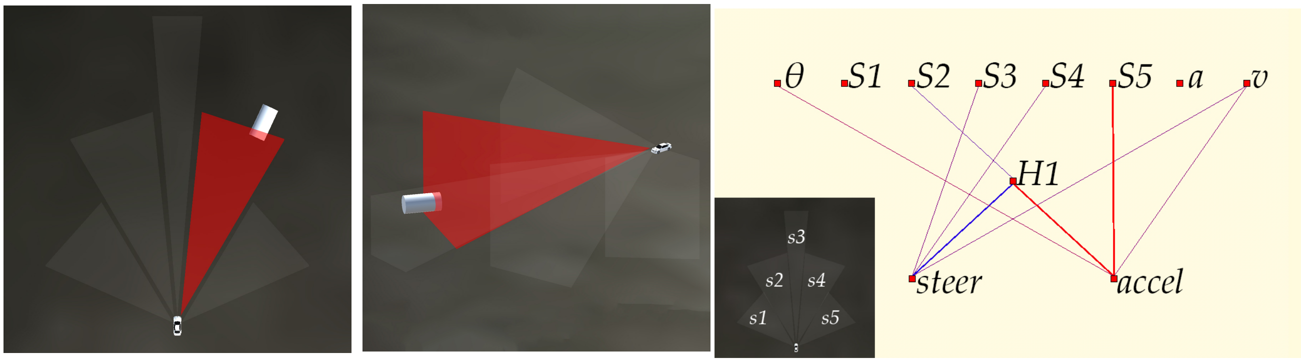

Fig. 1. *Left & Center:* Vehicles have five pyramidal sensors fanning out from the vehicle's front. An example of sensor 4 detecting an object is depicted in red. *Right:* Example ANN controller: vehicle sensors correspond to ANN input nodes ($S1$ to $S5$). Other inputs: bias input θ, angle to the next way-point a, current speed of the vehicle, v. This example has one hidden node, $H1$. ANN controller motor outputs: $steer$ and $acceleration$ nodes.

II. Methods

We applied the *Neuro-Evolution of Augmenting Topologies* (NEAT) [13] method to synthesize autonomous vehicle traffic behavior for given roads[1] assuming no stop signals or lanes to assist with vehicle coordination and navigation. NEAT was the vehicle controller design method, where multi-vehicle driving behavior emerged in response to task (minimizing traffic transit time and collisions) and environment (road structure) constraints. To direct NEAT controller evolution and synthesize coordinated vehicle driving behaviors, we used *objective*, *novelty* and *hybrid* evolutionary search (section II-A).

NEAT [13] is a direct encoding *neuro-evolution* method that has been demonstrated as suitable for multi-agent behavior evolution in a broad range of tasks [14], [4], [11]. NEAT evolves both connection weights and *Artificial Neural Network* (ANN) topologies and applies three key techniques to maintain a balance between ANN fitness and solution diversity.

First, NEAT assigns unique historical markers to new genes so crossover is only performed between pairs of matching genes. Second, it speciates the population so ANNs (genotypes) compete only within their own niches (identified by historical markers) instead of competing with the whole population. Third, NEAT initially evolves a population of simple ANNs with no hidden nodes but gradually adds new topological structure (nodes and connections) using two special mutation operators, *add hidden node* and *add link*. An advantage of NEAT is this complexifying process is likely to find a solution in lower dimension search spaces compared to relatively large search spaces corresponding to large fixed topology ANNs specified *a priori*. This *complexification* process also makes NEAT amenable for solving a broad range of problems.

NEAT was selected as it is an established method, that has been successfully applied to vehicle controller evolution in related work [15], [16]. However, with notable exceptions such as automating vehicle traffic for intersections [4], optimizing vehicle sensory configurations [3], and formation driving (*platooning*) [16], there has been relatively little research on evaluating various types of evolutionary search with NEAT for vehicle traffic optimization on *autonomous vehicle only* roads.

[1] *Road* and *track* are used interchangeably throughout this paper.

A. Vehicle Controller Evolution

Vehicle ANN controllers were evolved with one of three NEAT evolutionary search variants (sections II-A1, II-A2, II-A3). The goal was to maximize average distance traversed (measured by checkpoints passed, table I, section III-C) on a given track while minimizing collisions with static and dynamic obstacles (figure 2). Static obstacles represented unexpected objects on the road and dynamic obstacles represented other vehicles and pedestrians. An extension of *UnityNEAT*[2] [17], was used to simulate vehicles, sensors, roads and obstacles. Vehicle controller evolution used three vehicles (with identical controller and sensor configurations) traveling between static start and end-points on a given track (table I).

Controller evolution began with a population of minimally complex ANN controllers using eight sensory input nodes and two outputs (figure 1, right). During controller evolution, both the ANN inputs (vehicle sensors) and outputs (vehicle speed and turn angle) remained fixed and NEAT adapted the number of hidden layer nodes and connectivity between hidden and input and output and hidden nodes. The output nodes were *braking* and *steering*. High versus low activation values denoted the degree of braking and turning left versus right (labeled *accel* and *steer* in figure 1, right, respectively).

Given that each vehicle used the same controller, vehicle behavior differed only according to varying sensory inputs, resulting in different individual and thus coordinated driving behaviors (section II-A). Thus, the evolution of high task performance coordinated driving depended on individual vehicles effectively avoiding collisions via adapting speed and heading in response to the driving behaviors of other vehicles. Vehicle and controller homogeneity was selected as this allowed for more efficient controller evolution and evaluation.

1) Objective-based Evolution: Controller fitness was the number of checkpoints passed after 45 simulation iterations:

$$\text{fitness}(x) = \frac{1}{cars}\sum_{i=0}^{cars}\left(\frac{cp_{passed}}{cp_{total}} * 0.9^{coll}\right) \tag{1}$$

Where, $cars$ was the number of vehicles, cp_{passed}, checkpoints passed (section III-C), cp_{total} the total checkpoints on the track, and $coll$ the number of vehicle collisions.

[2] Unity is a multi-platform game development engine: http://unity3d.com

TABLE I. CONTROLLER EVOLUTION PARAMETERS AND CONTROLLER GENERALIZATION TEST PARAMETERS.

Vehicle Controller Evolution Parameters	
Parameter	**Value**
Number of vehicles	3
Number of runs / Generations per run	20 / 100
Task trials per generation / Task trial duration (seconds)	6 / 45
Evolutionary search methods (section II-A)	Objective, Hybrid, Novelty
Evolution tracks	1
Generalization Evaluation Parameters	
Parameter	**Value**
Generalization experiments (Fittest 20 evolved controllers per search method)	3
Number of vehicles	[1, 3, 5]
Number of runs per experiment (Fittest 20 evolved controllers)	20
Maximum trial duration (seconds)	100
Evolved controllers tested per test-track	60
Test tracks / Difficulty variations per test track	3 / 3

TABLE II. NEURO-EVOLUTION AND VEHICLE SIMULATION PARAMETERS.

Parameter	**Value**
Evolution and Test Track Starting Area	Initial 100 m^2
Evolution Track Length / Width	6 km / 45 m
Test Track 1, 2, 3 Width	40 m
Test Track 1, 2, 3 Length	5.6 km / 5.2 km / 6.4 km
Sensor Field of View (FOV) / Range	40° / 100 m
NEAT Population size / Species count / Complexity threshold	100 / 10 / 21
ANN Activation function	Steepened Sigmoid
Novelty Search archive size / Addition rate	Unbounded / 15 per generation
Novelty Search K-nearest neighbors	15
Behavior Characterization Sampling Rate	1/100th run length (100 Samples)
Hybrid Weighted-Sum Proportion (ρ)	0.5

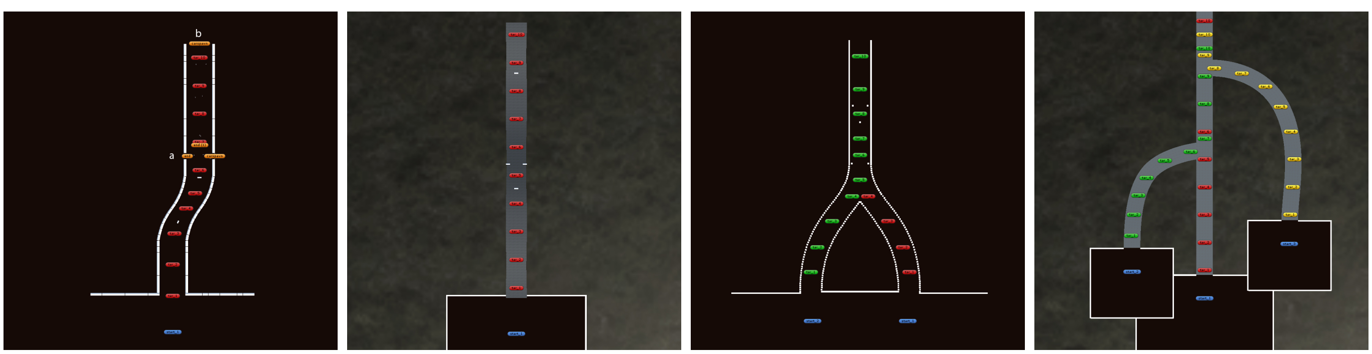

Fig. 2. Left: *Controller evolution track*: Two static obstacles were placed between the *third* and *forth* check-points, and *fifth* and *sixth* check-points. Dynamic (other vehicle) obstacles (*a* and *b*) crossing the road and oncoming traffic made this track difficult to complete. Center-left, Center-right, Right: *Test-tracks* to evaluate how well evolved controllers generalize to new environments. All tracks have 10 *checkpoints* (red, green and yellow) and a *starting-area* (blue).

Collisions (*coll*) caused an exponential decay to the fitness of a vehicle controller, where values lower than 0.9 resulted in slow and often stagnating evolution. This had the affect of rarely rewarding vehicles that rarely collided but penalizing such vehicles exponentially as collision counts increased.

2) Novelty-Search: Sparseness (equation 2) was used as a novelty metric [18]. Controller novelty was how different its behavior was compared to others in the population and novelty archive, which stored all *novel* controllers so previously novel behaviours were not lost. The K-nearest neighbors (behaviorally similar solutions in the novelty archive and population) were used to compute controller sparseness and novelty.

$$\text{Sparseness}(x) = \frac{1}{k}\sum_{i=0}^{k} dist(x, \mu_i) \tag{2}$$

Where, μ is the ith-nearest neighbor of x with respect to the novelty metric, and *dist* uses the Euclidean distance.

Novelty Archive: The 15 most novel solutions (of the current population, table II) at each generation were added to the archive. The novelty archive was unbounded meaning its maximum size was $15 \times n_{generations}$ at the final generation.

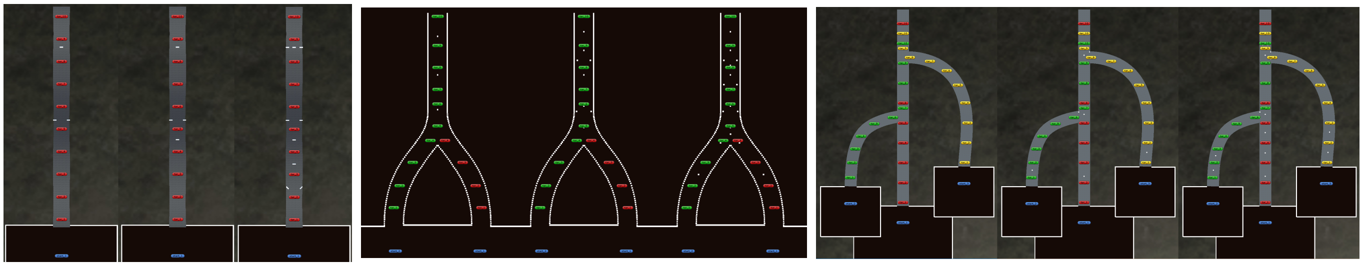

Fig. 3. Left, center, right: Least to most difficult test tracks. Left: *Test track 1*: Easiest, medium, difficult tracks used 3, 4, 9 obstacles. Center: *Test track 2*: Easiest, medium, difficult tracks used 3, 9, 17 obstacles. Right: *Test track 3*: Easiest, medium, difficult tracks used no obstacles, 7 and 16 obstacles.

Behavior Characterization (BC) and Behavior Sampling: Parameter tuning experiments compared three behavior characterizations: *speed*[3], *speed and cohesion* and *location* as potential behavioral dimensions to describe controller behavior and be used by the novelty metric to select for novel behaviors. In these experiments, *speed* yielded the highest mean fitness, so it was selected as the BC used in *hybrid* and *novelty* search directed controller evolution experiments (section III).

BC values were sampled at fixed simulation time-steps and controller behavior vectors were compared with each other to determine a controller's *sparseness* [18]. BC was a vector comprising values sampled at fixed intervals of 1/100th of total simulation time-steps per generation. For each behavior vector, 100 samples were collected (sampling rate, table II), where the vector of BC values were combined into a final vector for sparseness calculation (equation 2).

3) Hybrid-Search: This method linearly combined novelty and fitness to create a weighted sum [19], where the score that an individual controller (i) received was:

$$\text{score}(i) = \rho.\overline{fit}(i) + (1-\rho).\overline{nov}(i) \tag{3}$$

Where, $\rho = 0.5$, equally combining fitness and novelty for i:

$$\overline{fit}(i) = \frac{fit(i) - fit_{min}}{fit_{max} - fit_{min}}, \overline{nov}(i) = \frac{nov(i) - nov_{min}}{nov_{max} - nov_{min}} \tag{4}$$

Where, nov_{min}, fit_{min} were the lowest novelty and fitness values in the population, respectively and nov_{max}, fit_{max} were the highest. Previous work [19] indicated that large variations of ρ biased the results to either novelty or objective-based search. Other work using similar hybrid evolutionary search indicated $\rho = 0.5$ to be a suitable value [11].

III. Experiments

Two sets of experiments were conducted. First, controller evolution experiments and second controller generalization test experiments. Controller evolution experiments applied NEAT for vehicle driving behavior evolution directed by *objective*, *novelty* or *hybrid* search (section II-A), where average task performance was calculated over 20 runs. One experiment comprised NEAT controller evolution directed by one evolutionary search method. Each evolutionary run was 100 generations and each generation consisted of six simulation task trials that initialized three vehicles in random starting positions within the *starting area* of the evolution track (figure 2).

[3]Vehicle velocity was measured in meters per second (m/s).

Each generation, controller task performance was calculated as an average over six simulation task trials. A simulation task trial was completed when all vehicles passed the final checkpoint or after 100 simulation iterations. After each controller evolution experiment was concluded, the 20 controllers yielding the highest task performance, were selected at the end of each of three evolution (that is, defined by a different search method) experiment's 20 runs, and applied to the controller generalization test experiments. All experiment, simulation and evolutionary parameters are presented in tables I and II.

For controller generalization test experiments, the evolved controller yielding the highest average task performance (after 20 runs), was transferred to a non-evolutionary simulation test run using either *one*, *three* or *five* vehicles. Each generalization test run was the 20 best controllers (evolved by NEAT with either objective, novelty or hybrid search functions), run in a non-evolutionary task trial simulations on three increasingly difficult variations of three test tracks (figure 2).

These generalization experiments evaluate an evolved controller's ability to traverse previously unseen tracks. As for controller evolution experiments, each test track used checkpoints to measure vehicle task performance (figure 2). However, instead of penalizing controllers on collisions, the vehicle was immediately stopped upon collision, and if a vehicle collided with another, the other vehicle was also stopped. The rationale was to mimic real-world instances where vehicles should completely avoid collisions. Generalization experiments were non-evolutionary, though included dynamic obstacles, where controller task performance was calculated as the average task performance yielded by the fittest 20 evolved controllers for all variations of a given test track (figure 2). For each NEAT search method, an overall generalization task performance average was calculated as the average over all 20 controllers tested on the three variations of each test track.

A. Vehicle Simulation

Simulated vehicles had a maximum steering angle of $25°$ to left or right from the vehicle's current heading, and used five radar sensors on the vehicle's front (figure 1, left-center). A sensor's *Field Of View* (FOV) was a pyramidal shape, where the sensor reading was the inverse distance to the closest obstacle in the sensor's range (table II). This sensor reading was then fed into a corresponding ANN controller input node (figure 1). A vehicle's ANN controller also received as inputs: a bias value θ, an angle to the next way-point, and the vehicle's current velocity (figure 1, right). A detailed description of vehicle simulation can be found in previous work [18].

B. Controller Evolution and Generalization Test Tracks

Controller evolution and generalization experiments used separate evaluation and test task tracks. Figure 2 (left) illustrates the evolution track and figure 2 (left-center to right), presents the three tracks used for controller generalization experiments. Figure 3 presents the three variations of each of these three test tracks. Each test track simulated increasing task difficulty (number of obstacles on the track). Generalization experiments also tested *one*, *three*, and *five* vehicles to account for increasing traffic density. For added difficulty, test tracks also simulated merging lanes and height variances in tracks to introduce vehicle *blind-spots* in sensory coverage.

C. Checkpoints

Checkpoints were placed along each track to guide vehicles and determine the distance traveled by vehicles. Controller task performance was equated to the number of checkpoints all vehicles had passed relative to the total number of checkpoints (section II-A1, II-A2, II-A3). To ensure normalization across all tracks, each track used 10 checkpoints spread equally apart.

IV. Results and Discussion

This section presents comparative results for NEAT controller evolution directed by *objective-based*, *hybrid* and *novelty search*, (section II-A1, II-A2, II-A3, respectively) and results of the controller generalization tests. Figure 4 (left) presents average maximum fitness progression over 100 generations of controller evolution. This fitness progression indicates that controller evolution directed by hybrid search (section II-A3) was significantly more expedient at evolving effective controllers compared to objective-based and novelty search (*Mann-Whitney U*, $p \leq 0.05$). An average (normalized) fitness of approximately 0.75 was reached by hybrid search after 20 generations, compared to 0.43 and 0.46 yielded by objective-based and novelty search, respectively.

Figure 4 (right) presents box plots of the average fitness (over 20 runs) of controllers evolved by each search method. These controller evolution results indicate that all search methods yielded just above 60% of optimal task performance, where hybrid evolutionary search significantly out-performed (*Mann-Whitney U*, $p \leq 0.05$) the objective and novelty search methods. However, there was no significant difference between objective and novelty search directed controller evolution (figure 4, right). This result supports the benefits of hybrid evolutionary search in this coordinated driving task, and is supported by previous work demonstrating benefits of hybrid search in multi-agent behavior evolution [20], [10], [11].

To better elucidate the efficacy of comparative evolutionary search methods, we investigated search space exploration via visualizing portions of genotypes (controllers) evolved by each search method with an average fitness in a given range. Figure 5 presents, for *objective*, *hybrid*, and *novelty* search, heat-maps illustrating portions of genotypes within 0.20 ranges of normalized task performance: [0.0, 1.0]. Figure 5 (left) indicates for objective search, that approximately 70% of controllers yielded an average fitness of 0.20 between generations 5 and 65. A large portion of objective evolved controllers retained this low average fitness until generation 50 when approximately 35% of controllers yielded an average fitness of at least 0.40.

Objective search evolutionary runs yielded approximately 60% of evolved controllers in the fitness range: [0.2, 0.4] (figure 5, left), with an average maximum fitness of 0.77 (figure 4, right). Figure 5 (center) indicates that hybrid search evolved an approximate even-spread of solutions across the whole fitness range. This held except for between generations 10 to 20, where 40% of controllers had an average maximum task performance of 0.20. The efficacy of this broad controller search space exploration and thus near-optimal evolved controllers, is reflected in the average maximum fitness of 0.99, achieved by hybrid search (figure 4, right). Whereas, for all generations of novelty search (figure 5, right), 80% of evolving controllers were in the average fitness range: [0.0, 0.3]. This low fitness region exploration is reflected as the relatively low average maximum fitness of 0.73 (figure 4, right).

Figure 6 (left) presents the generalization test results for the fittest controllers evolved by the *objective*, *hybrid* and *novelty* search methods. Task performance averages were calculated over 20 runs for all vehicle group sizes (table I) and all test tracks (figure 3). Figure 6 (left) indicates that vehicle controllers evolved with objective search generalized best to the test tracks for all vehicle group sizes (figure 3), though controllers evolved with hybrid search were least well suited to generalize to the test tracks. *Mann-Whitney U* ($p \leq 0.05$) tests indicated a significant difference between generalization test average task performance results of each search method (figure 6, left). The fittest controllers evolved by objective, hybrid and novelty search yielded average task performances of 0.29, 0.20, and 0.24, respectively, in generalization tests.

To elucidate evolutionary mechanisms responsible for these results, we visualized the topological network complexity of each search method's fittest evolved controllers. Figure 6 presents the three fittest controllers evolved by NEAT with *objective*, *hybrid* and *novelty search*, where blue and red lines denote positive and negative weights, respectively, and line thickness denotes weight magnitude. Figure 6 indicates that the fittest controllers evolved by hybrid search were simple reactive ANN controllers with direct connections between sensory inputs and motor outputs and no hidden layers. In contrast, the fittest ANN controllers evolved by objective and novelty search were relatively complex given an increased number of hidden nodes and connections [11]. This difference in controller complexity was consistently evident between the fittest evolved controllers for all search methods and runs.

The efficacy of hybrid search is theorized to be a result of thorough controller (behavior) space search as evidenced in the approximately even spread of controllers and fitness values (figure 5). For example, at the final generation, 80% of hybrid search evolved controllers had fitness values in the range: [0.1, 0.8], where half of the controller population was in the fitness range: [0.5, 0.7]. This exploratory capability of hybrid search is also supported by related work [20], [10], [11]. In this study, broad search space exploration enabled the discovery of minimally complex controllers (figure 6) achieving significantly higher average fitness, compared to objective and novelty search evolved controller behaviors (figure 4). While this simple neural complexity was effective on the controller evolution track (figure 2), controller generalization experiments (figure 6, left), indicated that such simple controllers were ineffective across all test tracks (figure 3).

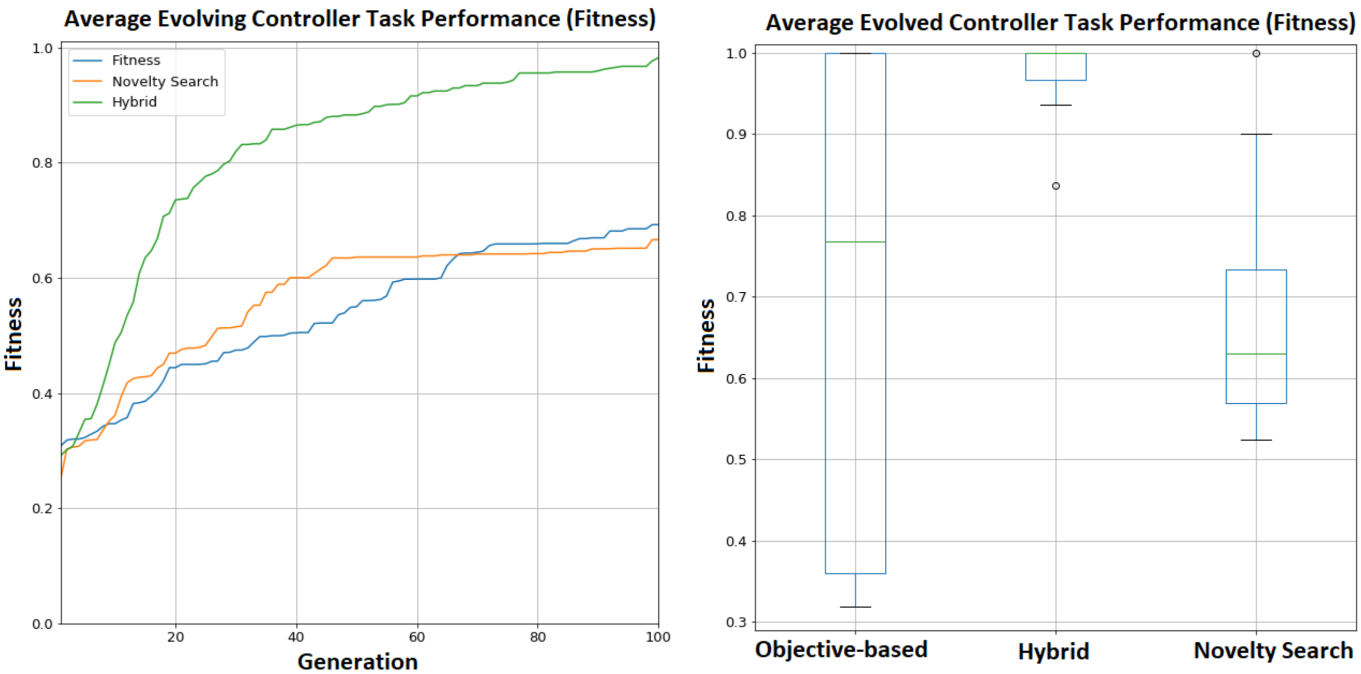

Fig. 4. Left: Average maximum fitness progression over vehicle controller evolution. Right: Box plots of average maximum evolved controller fitness. Averages are over 20 runs for *objective* and *hybrid* and *novelty* search based controller evolution on the evolution track (figure 2, left).

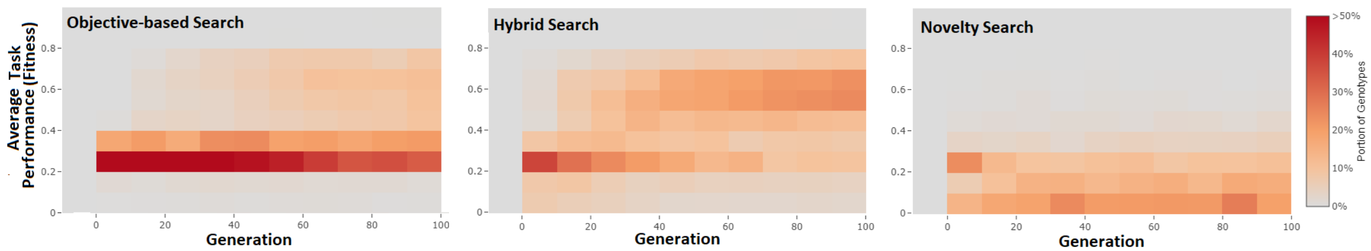

Fig. 5. Heat-maps presenting portions of genotypes evolved by *objective-based* (left), *hybrid* (center) and *novelty* search (right), within each 20 percentile of normalized task performance (fitness): [0.0, 1.0]. Darker shading indicates a higher portion of genotypes in the given fitness range.

The higher adapted neural complexity of the fittest objective and novelty search evolved controllers (figure 6) included necessary behavioral functionality. Such behavior enabled these controllers to yield significantly higher average task performance across all generalization test roads, for al vehicle group sizes (figure 6, left). Specifically, increased evolved controller complexity was detrimental given specific tasks (controller evolution track, figure 2), but generally beneficial to task performance across several tasks of varying difficulty (controller generalization test tracks, figure 3). This is supported by related work in multi-agent behavior evolution [11],

These results also indicate that hybrid search is least effective, whereas objective search is most effective for evolving coordinated driving behaviors capable of generalizing to a broader set of related task environments of varying difficulty (road configurations and vehicle group sizes). This indicates the coordinated driving task is not well suited to hybrid objective-novelty or novelty search directed controller evolution, due to strict task environment constraints [21], [22]. However, the impact of other behavioral characterizations in hybrid and novelty search and other evolutionary search variants on evolving generalized multi-agent behaviors, remains the topic of ongoing research. An end goal is to develop evolutionary methods to automate the design of controllers for coordinated autonomous vehicle traffic that operates effectively (safely) and efficiently (expediently) on *autonomous vehicle only* roads.

V. Conclusion

This study investigated controller automation for coordinated vehicle traffic on *autonomous vehicle only* roads. Results indicated that controller evolution directed by hybrid search evolved significantly more effective (high task performance) coordinated driving behaviors, compared to those evolved by objective (fitness function) and novelty search. However, the fittest hybrid search evolved coordinated driving behaviors did not generalize well to new test roads, compared the fittest driving behaviors evolved by objective and novelty search. These generalization test roads evaluated evolved controllers for various coordinated driving behavior tasks, vehicle numbers and road configurations. The relatively poor performance of the fittest hybrid evolved controllers in these generalization tests indicates that while hybrid search evolution was efficient and effective, the task constraints and test track variability were not conducive to controllers evolved by hybrid search.

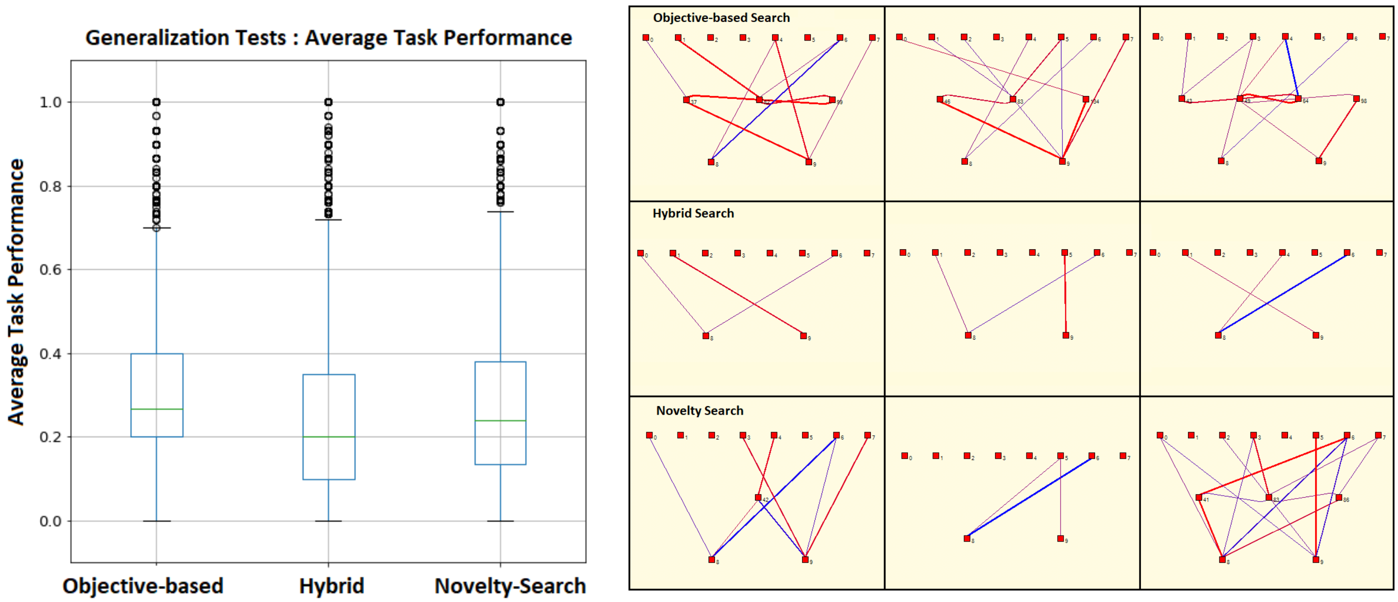

Fig. 6. Left: Average maximum (20 fittest evolved controllers) task performance from generalization tests, for all test tracks and vehicle group sizes (figure 3, table I). Right: Fittest controllers evolved by NEAT (after 20 runs) with *objective* (top), *hybrid* (middle) and *novelty* (bottom) search.

References

[1] P. Hancock, I. Nourbakhsh, and J. Stewart, "On the future of transportation in an era of automated and autonomous vehicles," *Proceedings of the National Academy Sciences*, vol. 116(16), pp. 7684–7691, 2019.

[2] T. Au, S. Zhang, and P. Stone, "Autonomous intersection management for semi-autonomous vehicles," in *Handbook of Transportation*, D. Teodorovic, Ed. London, UK: Taylor and Francis, 2015, pp. 88–104.

[3] A. Huang and G. Nitschke, "Evolving collective driving behaviors," in *Proceedings of the International Conference on Autonomous Agents and Multiagent Systems*. Sao Paulo, Brazil: ACM, 2017, pp. 1573–1574.

[4] A. Parker and G. Nitschke, "How to best automate intersection management," in *Proceedings of the IEEE Congress on Evolutionary Computation*. San Sebastian, Spain: IEEE Press, 2017, pp. 1247–1254.

[5] K. Dresner and P. Stone, "A multiagent approach to autonomous intersection management," *Journal of Artificial Intelligence Research*, vol. 31(1), pp. 591–656, 2008.

[6] D. Fajardo, A. Tsz-Chiu, T. Waller, P. Stone, and D. Yang, "Automated intersection control: Performance of a future innovation versus current traffic signal control," *Transportation Research Record: Journal of the Transportation Research Board*, vol. 2259(1), pp. 223–232, 2012.

[7] T. Tettamanti, A. Mohammadi, H. Asadi, and I. Varga, "A two-level urban traffic control for autonomous vehicles to improve network-wide performance," *Transportation Research Procedia*, vol. 72(1), pp. 913–920, 2017.

[8] T. Au, M. Quinlan, and P. Stone, "Setpoint scheduling for autonomous vehicle controllers," in *Proceedings of the IEEE International Conference on Robotics and Automation*. IEEE Press, 2012, pp. 2055–2060.

[9] J. Gomes, P. Urbano, and A. Christensen, "Evolution of swarm robotics systems with novelty search," *Swarm Intelligence*, vol. 7, no. 1, pp. 115–144, 2013.

[10] S. Didi and G. Nitschke, "Hybridizing novelty search for transfer learning," in *Proceedings of the IEEE Symposium Series on Computational Intelligence*. Athens, Greece: IEEE Press, 2016, pp. 2620–2628.

[11] G. Nitschke and S. Didi, "Evolutionary policy transfer and search methods for boosting behavior quality: Robocup keep-away case study," *Frontiers in Robotics and AI*, vol. 4, no. 62, pp. 1–25, 2017.

[12] J. Lehman and K. Stanley, "Abandoning objectives: Evolution through the search for novelty alone," *Evolutionary computation*, vol. 19, no. 2, pp. 189–223, 2011.

[13] K. Stanley and R. Miikkulainen, "Evolving neural networks through augmenting topologies," *Evolutionary Computation.*, vol. 10, no. 2, pp. 99–127, 2002.

[14] D. D'Ambrosio and K. Stanley, "Scalable multiagent learning through indirect encoding of policy geometry," *Evolutionary Intelligence Journal*, vol. 6, no. 1, pp. 1–26, 2013.

[15] L. Cardamone, D. Loiacono, and P. Lanzi, "Learning to drive in the open racing car simulator using online neuroevolution," *IEEE Transactions on Computational Intelligence and AI in Games*, vol. 2, no. 3, pp. 176–190, 2010.

[16] W. van Willigen, E. Haasdijk, and L. Kester, "A multi-objective approach to evolving platooning strategies in intelligent transportation systems," in *Proceedings of the Genetic and Evolutinary Computation Conference*. Amsterdam, the Netherlands: ACM, 2018, pp. 1397–1404.

[17] D. Jallov, *Evolve: Introducing a Novel Game Mechanic Based on the Indirect Control Of Evolving Neural Networks*. MSc Thesis. IT University of Copenhagen, Denmark, 2014.

[18] A. Huang, *Neuro-Evolution Search Methodologies for Collective Self-Driving Vehicles*. MSc Thesis. University of Cape Town, South Africa, 2019.

[19] C. Huang, D. Shorten, and G. Nitschke, "Searching for novelty in pole balancing," in *Proceedings of the IEEE Congress on Evolutionary Computation*. Sendai, Japan: IEEE, 2015, pp. 1792–1798.

[20] J. Gomes, P. Mariano, and A. Christensen, "Devising effective novelty search algorithms: a comprehensive empirical study," in *Proceedings of the Genetic Evolutionary Computation Conference*. Madrid, Spain: ACM, 2015, pp. 943–950.

[21] G. Cuccu and F. Gomez, "When novelty is not enough," in *European Conference on the Applications of Evolutionary Computation*. Torino, Italy: Springer, 2011, pp. 234–243.

[22] J. Doucette and M. Heywood, "Novelty-based fitness: An evaluation under the Santa Fe trail," in *European Conference on Genetic Programming*. Istanbul, Turkey: Springer, 2010, pp. 50–61.