Competitive Coevolution versus Objective Fitness for an Autonomous Motorcycle Pilot

Dana Vrajitoru
Intelligent Systems Laboratory
Computer and Information Sciences
Indiana University South Bend
South Bend, IN 46634, USA
danav@cs.iusb.edu

Abstract

Evolution in the context of genetic algorithms is driven by the fitness function. For some applications, this factor is not easy to compute and coevolution represents an alternate solution. Thus, competition between individuals in the population can be used as a performance measure instead of an objective function, when the nature of the problem allows it. In this paper we explore the impact of such a choice on the overall performance of the solutions, as compared to the classic approach. We apply this model to a problem of configuring a multi-agent autonomous pilot for motorcycles.

1 Introduction

Generally, coevolution [5] describes evolutionary models where the performance of any individual or candidate solution depends on other individuals in the population. There are two possible forms of coevolution: competitive, where individuals are tested against each other [10], and cooperative, where individuals must collaborate to solve the problem [12]. The competition/collaboration can concern individuals from the same population or from populations evolving in parallel.

Competitive coevolution seems especially well suited for game playing [6]. This class of problems can include any situation where the individuals or candidate solutions can be seen as opponents competing to achieve a mutually exclusive goal. In other terms, the co-evaluation of several individuals leads to a winner and one or more losers. The question we are asking in this paper is, how does this kind of evaluation compare to the evolution drive of an objective fitness function.

In this experiment we are using a genetic algorithm to configure a multi-agent probabilistic pilot for single track vehicles (STV) like motorcycles. The application aims to control the vehicle in a non-deterministic way inspired from the behavior of a human driver and using similar perceptual information to make decisions.

The genetic algorithms (GAs) and other evolutionary approaches have been successfully been applied to related areas such as path-find for robots [3] and scheduling [1]. Several approaches have applied multi-agent models to the simulation of autonomous drivers [8] and our application is based on the model presented in [13].

Most of the research on autonomous pilots is directed toward piloting aircrafts [7], and cars [9]. Our approach targets motorcycles which have not yet been studied as extensively as the other types of vehicles and which represent a more challenging modeling problem.

2 Competitive Coevolution

Competitive coevolution is usually employed as an alternative to an objective fitness function and consists in playing two or more potential solutions against each other, resulting in a winner and a loser. The competing solutions often belong to parallel populations, but that may not always be the case.

More precisely, problems like game playing and strategy involve opponents competing to achieve a mutually exclusive goal that can only be reached by one of them. In the example of a predatory-prey scheme [10], two populations evolve separate strategies that approach the problem from a different perspective. In the case of a backgammon or soccer strategy, the competing solutions need not belong to different populations and a different competing scheme is necessary.

In the case of individuals competing within the same population, one possible evaluation approach is the *Sin-*

gle Elimination Tournament [2] inspired from real sportive events. A tournament pairs the opponents in each round, runs them against each other, then promotes the winners to the next round. Thus, the game takes the shape of a tree with a global winner as the root. Such a scheme has the disadvantage of possibly eliminating a good solution early on, which is an issue that real sportive tournaments address by not pairing the teams in an entirely random fashion. Strong teams are prevented from playing against each other in early stages of the game based on a performance estimation from previous games.

In our study we employed the *K-Random Opponents* strategy [11]. According to this method, each individual in the population has a chance to compete against a chosen number of opponents defined by the parameter k. Thus, to evaluate an individual, we select k random opponents and run a match against each of them, incrementing the score of the winner every time.

In our situation the problem to be solved is driving a motorcycle over a circuit without stranding, crashing, and as fast as possible. In the competitive setting, two pilots drive their vehicles simultaneously over the circuit. If one of them is stranded outside of the circuit or crashes by losing balance, then the opponent is declared a winner. If none of the pilots ends the run in a failure, then the pilot finishing the circuit the first is the winner.

The main question we ask in this paper is how effective is such an evaluation as compared to an objective individual fitness function. In our case, the objective performance can be measured by running the pilot alone over the circuit in the same conditions. In case of a failure by crashing or stranding out of the circuit, the fitness is measured as the percentage of the circuit that was covered before the crash. In the case where the pilot finishes the circuit successfully, a component is added to the fitness favoring a faster completion of the circuit. More details are provided in Section 4.

To enable the comparison, the pilot is evolved both using uniquely the objective fitness function and using the competitive evaluation. In the latter situation, at the end of each run of the GA we evaluate the best solution evolved competitively using an objective function so that we have a common ground for the comparison.

3 Physical Model of the Motorcycle and Autonomous Pilot

The physical model of the motorcycle has been more extensively described in [13] and is close to [4]. The motorcycle or STV, is modeled as a system composed of several elements with various degrees of freedom that can be driven through several control units. Figure 1 shows the components of the physical model for a motorcycle.

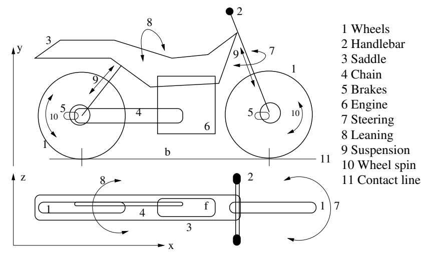

Figure 1. A motorcycle with control units and degrees of freedom

An STV is a non-holonomic dynamic system with six degrees of freedom: the rotation of the wheels around an axis parallel to Oz, the rotation of the handlebar and of the front wheel around the fork axis (steering), the front and back translation along the suspension axis, and the rotation of the whole vehicle around the Ox axis in a system of coordinates relative to the motorcycle where the origin is in the center of the vehicle, on the ground level. The driver can control the vehicle through five inputs: the handlebar steering, leaning the vehicle laterally, the throttle, and the two brakes, front and back.

The STV is modeled as a reduced state system of continuous variables. The generalized coordinates of the vehicle at a particular moment are given by

$$q = (s, \alpha, \theta)^T \tag{1}$$

where $s(t) = (x(t), z(t))$ represents the *spatial position* of the STV, α the *leaning angle*, and θ the *orientation angle* determining the *direction of movement* $d = (\cos\theta, \sin\theta)$.

The vertical component of both s and d is determined by the altitude and by the slope of the road considering the current position and orientation of the vehicle. In this paper we consider the road to be close enough to the sea level such that the gravitational acceleration is the constant $g = 9.8\ m/s^2$. Let $\sigma(s,d)$ be the angle made by the contact line of the vehicle with the horizontal plane (x,z).

The driver's input into the system is defined by the tuple $u = (\tau, \beta_f, \beta_r, \phi, \alpha)$ where τ is the component of the acceleration tangent to the direction of movement d and β_f, β_r represent the front and rear brakes respectively.

Let $v = s'$ be the momentary speed or velocity in the direction of movement, and $a = v' = s''$ the momentary acceleration in the direction of movement. We modeled the motion of the vehicle using Newtonian mechanics. The position and velocity of the vehicle at $t + \Delta t$ are defined by

$$s(t+\Delta t) = s(t) + \Delta s, v(t+\Delta t) = v(t) + \Delta v \tag{2}$$

where:

$$\Delta s = d\left(v \cdot \Delta t + a\frac{\Delta t^2}{2}\right), \quad \Delta v = a \cdot \Delta t \tag{3}$$

The acceleration is defined by the gravity, the friction, the drag, and the throttle. The brakes do not act as a simple negative acceleration, but contribute to the friction force instead.

3.1 The Autonomous Pilot

In this section we present the multi-agent autonomous pilot for our motorcycle and the perceptual information it uses.

The autonomous pilot uses perceptual information to make decisions about the vehicle driving. This information is inspired from the perceptual cues that a human driver would also be paying attention to while driving a vehicle. In our application, the pilot is aware of the following measures:

The *visible front distance*, denoted by $front$, defined as the distance to the border of the road from the current position in the direction of movement, scaled by the length of the vehicle, or *horizon*.

The *front probes*, denoted by $frontl$ and $frontr$, are defined as the distances to the border of the road from the current position of the vehicle in directions rotated left and right by a small angle from the direction of movement.

The *lateral distances*, denoted by $leftd$ and $rightd$, are measures of the lateral distance from the vehicle to the border of the road, at a short distance ahead of the vehicle, simulating what the pilot might be aware of without turning their head to look.

The *slope* is a perceptual version of σ, discretized to simulate the intuitive notion of road inclination that a human driver would have, approximated by the values almost flat, slightly inclined up or down, or highly inclined up or down. This simulates the fact that a human pilot is not aware of the precise value of σ.

Figure 2 shows an example of the geometrical definition of these measures.

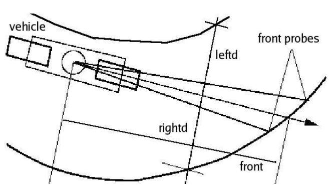

Figure 2. Perceptual information used by the autonomous pilot

The motorcycle is driven by several control units (CUs). Each of them is controlled by an independent agent with a probabilistic behavior. The agents are not active during the computation of each new frame simulating the evolution of the vehicle on the road, but only once in a while in a non-deterministic manner. This simulates the behavior or a human driver that may not be able to respond instantly to the road situation and requires some reaction time.

The current control units focus on the gas (throttle), the brakes, the handlebar/leaning. Each of these CUs is independently adjusted by an agent whose behavior is intended to drive the motorcycle safely in the middle of the road at a speed close to a given limit. In our case, the agents controlling the throttle and the handlebar are in general more active than the agent controlling the brakes.

The agents behave based on a set of equations relating the road conditions to action. The full set of equations is described in [13]. Here we will briefly describe each of the agents. The equations comprise a fair number of coefficients and thresholds. The configuration of each agent uses independent values for the coefficients.

The Throttle. This agent controls the amount of gas supplied to the engine and thus the speed of the vehicle.

The agent uses a minimal speed threshold v_{low}, a maximal speed threshold over which the speed is considered unsafe, and the given speed limit v_{limit}. The agent aims to keep the vehicle speed above v_{low} and below the maximal one, and also close below the v_{limit}.

The agent detects a turn in the road by testing $leftd$ and $rightd$ and if needed, cuts the gas to allow for a safe turn. A similar rule is applied to the visible distance in front of the driver: a low value for $front$ indicates an unsafe road situation requiring a reduced speed. In any other situation it attempts to keep the speed close to v_{limit}.

The Brakes. The agent controlling the brakes presents a similar behavior to the throttle agent but can only decrease the speed. This agent acts when a more dramatic change is necessary.

The Steering / Leaning Agent. The motorcycle can achieve a change in direction either by steering using the handlebar, or by leaning. The autonomous pilot has been tested under three conditions: when the motorcycle is driven entirely by steering, when the motorcycle is driven entirely by leaning, and a combined strategy, consisting of steering at low speed (bellow a threshold) and leaning at a speed above the threshold. This emulates the general strategy employed by human drivers.

Alerting Agent. Beside all the agents that are in direct control of the motorcycle, the pilot comprises a fourth agent that does not perform any action on the vehicle. While the other agents are active only occasionally, this agent is probing the vehicle and road conditions for every new frame and is capable of activating one of the other agents if the situation requires special attention. Such situations include the speed of the vehicle being too high or too low, or the visible front distance being too short.

4 Pilot Configuration by Genetic Algorithms

All of the agents composing the autonomous pilot are governed by equations comprising a set of thresholds and constants that can be configured to adjust its behavior and optimize its performance.

To apply the GAs to this problem, we chose a representation where each configurable coefficient is assigned 10 binary genes, and the chromosome results by concatenating all of the coefficients. Thus, we worked with 32 coefficients for the leaning and steering modes, and with 36 for the combined mode, which means that the chromosome is of a length of 320 and 360 respectively.

We used the one-point crossover for our experiments with a probability of 0.8 and a probability of mutation of 0.01. We employed an elitist reproduction preserving the best individual from each generation to the next.

A chromosome is evaluated by running the motorcycle in a non-graphical environment once with the pilot configured based on values obtained by decoding the chromosome over a test circuit presenting various turning and slope challenges. To compute the *objective fitness* we marked 50 reference points on the road and counted how many of them were almost touched by the motorcycle. The fitness is computed as follows:

$$F(x) = \begin{cases} \frac{d_m}{d_t} + \frac{1}{1+t_m} & \text{if the circuit was completed} \\ \frac{d_m}{d_t} + \frac{1}{5+t_m} & \text{if the circuit was not completed} \end{cases} \tag{4}$$

where d_m is the number of points crossed by the motorcycle, d_t is the total number of points, and t_m is the total time taken until either the circuit was completed, or until a failure condition was detected.

Thus the objective fitness reflects both how much of the circuit the motorcycle completed, and how fast it was capable of finishing the track. In general, a fitness higher than 1 is an indication of completion of the circuit.

A failed circuit can be caused by one of the following three situations: a crash due to a high leaning angle, an exit from the road with no immediate recovery, or crossing the starting line without having reached all the marks, as when the vehicle takes a turn of 180 degrees and continues backward.

To compute the *competitive fitness*, we initialize the score of each individual to 1, such that none of them will have a probability of 0 to be selected by the fitness-proportionate scheme. After each match of the k-random opponents competition, the score of the winning pilot is increased by 1. The final score counts as the individual's fitness value.

Table 1. Average competitive fitness in 100 generations

k	Steer	Lean	Combined
2	5.24	5.27	5.26
5	9.55	9.59	9.53
10	12.91	15.41	15.27

4.1 Experiments

We performed 100 runs of the GAs for each of the piloting modes, steering, leaning, and combined. Separate experiments were done using uniquely the objective fitness, and the competitive fitness for three values of k. At the end of each competitive run we evaluated the best individual in the population with the objective fitness. The population size was of 20 in all cases and we limited the evolution to 500 generations. These parameters are justified by the high cost of the fitness evaluation which requires from a few seconds to a few minutes.

For each experiment we selected the best pilot evolved from the trials and performed 100 runs over the circuit in a full graphical environment. These experiments give us a better appreciation of how the pilots evolved using each model compared over a greater number of probabilistic trials. We also compare these results with the performance of the human players as reported in [13] and presented in Table 3.

The statistics we considered as measures of performance are the total time to complete the circuit, the average and maximal speed, the total distance covered by the pilot which is an indication of how efficiently the circuit was completed, and the lateral balance. The latter takes values between 0 and 1, 0 indicating the center of the road, 1 the extreme borders of the road, and lower values showing a better behavior. The last row shows the percentage of completed circuits in each case.

Table 1 presents the average performance of the best individual in the last generation according to the competitive evaluation for the three values of the parameter k that we used. Table 2 presents the average objective evaluation of the best individual in the last generation for the same experiments. The row denoted by ObjEv represents the experiments where we used the objective evaluation exclusively without a competitive component. In general, values of the objective fitness higher than 1 signify completed circuits.

From Table 2 we can notice a big difference in performance between the three modes, using steering, leaning, or a combination, as well as between the competitive evolution and the objective fitness evolution. We performed a T-Test

Table 2. Average objective fitness in 100 generations

k	Steer	Lean	Combined
2	*0.970348*	0.921057	0.930663
5	*0.979599*	0.926643	0.933911
10	***1.10882***	0.930716	**0.928135**
ObjEv	***1.90845***	**0.943309**	***1.00504***

Table 3. Performance of the human players

	Human 1	Human 2
Total time	97.4	79.2
Total distance	2312.05	2316.83
Average speed	6.19	8.94
Maximum speed	8.75	12.26
Lateral balance	0.29	0.36
Completed circuits	100%	100%

to determine which of these results were significantly different from the ones in the row above, marked in bold, and from the next best result on the same row, marked in italics. For example, the result for steering more, $k = 10$ is significantly better than both the steering mode where $k = 5$ and than the leaning mode for $k = 10$.

We can note that all results from the objective evolution are significantly better than the ones obtained by competitive evolution. Also, the steering mode systematically produces better results than the two other modes.

Tables 4 and 5 show the performance of the pilot configured based on the parameters derived by the GA, for the completed and incomplete circuits respectively. In the competitive mode we selected the pilot showing the best objective fitness among all the trials, which in all cases was achieved for $k = 10$.

These statistics show some interesting facts. In the steering mode, the best competitive pilot performs better than the best pilot evolved with an objective fitness, even though the objective evolution showed a better average performance. It completes the circuit successfully significantly faster, but it also fails to complete the circuit 5% of the time due to stranding out of the track for too long. It seems that the competitive evolution generates a more efficient but also more reckless road behavior.

In the leaning and combined modes, the best competitive pilots also drive faster than the ones evolved by an objective fitness. In these cases the behavior is not profitable because

Table 4. Results of the GA configured pilot, completed circuits

	Steer	Lean	Combined
	Objective fitness		
Total time	47.49		107.333
Total distance	2295.41		2359.51
Speed	2.08982		0.990959
Max speed	3.1264		2.19597
Lateral balance	0.36593		0.296454
Completed circuits	100%	0%	3%
	Competitive evolution		
Total time	19.7895		
Total distance	2495.61		
Speed	5.49419		
Max speed	8.19704		
Lateral balance	0.38513		
Completed circuits	95%	0%	0%

the pilot leans too far and crashes for all of the circuits. Notice that in the combined mode, the objectively evolved pilot could complete the circuit 3 times using a much lower speed than its competitive colleagues.

Overall the performance of the pilots in the steering mode is close to that of the human players, and the competitive pilot is even more efficient than the human players. This means that the program can be used to provide challenging opponents to a human player in a competitive game scenario.

5 Conclusions

In this paper we presented a comparison of an evolution through genetic algorithms using an objective fitness with a competitive coevolution based on the k-random opponents model, as described in Section 2. Both of these schemes were applied to configure a multi-agent autonomous pilot for motorcycles, described in Section 3.

Our experiments presented in Section 4 show that the pilot evolved through a competitive scheme can outperform the one evolved with an objective fitness based on some criteria such as the average speed and the time efficiency in completing the circuit. The objectively evolved pilots on the other hand show a higher robustness and are capable of completing the circuit more often.

Table 5. Results of the GA configured pilot, incomplete circuits

	Steer	Lean	Combined
		Objective fitness	
Total time		4.33	27.6289
Total distance		126.825	661.058
Speed		1.01564	0.946201
Max speed		3.09039	1.76654
Lateral balance		0.55627	0.35331
Incomplete circuits	0%	100%	97%
		Competitive evolution	
Total time	24	1.7	1.48
Total distance	3717.95	121.76	131.69
Speed	3.36772	2.54087	3.52169
Max speed	4.92465	4.91532	6.83542
Lateral balance	0.290996	0.461192	0.384132
Incomplete circuits	5%	100%	100%

The difference in performance between the two schemes can be explained by the features of the pilots that are rewarded by each evaluation method. The competitive fitness encourages faster pilots that can finish the circuits before their peers, while the objective evaluation encourages the completion of the circuit more than how fast it is done.

Both of these schemes produce pilots that can represent a real challenge to a human player. They can be used to generate a variety of autonomous opponents in a multi-player situation and enhance the playing experience for a human user.

References

[1] D. Abramson, G. Mills, and S. Perkins. Parallelisation of a genetic algorithm for the computation of efficient train schedules. In *Proceedings of the Parallel Computing and Transputers Conference*, pages 139–149, 1993.

[2] P. Angeline and J. Pollack. Competitive environments evolve better solutions for complex tasks. In S. Forest, editor, *Proceedings of the Fifth International Conference on Genetic Algorithms (ICGA)*, pages 264–270. Morgan Kaufmann, 1993.

[3] R. R. Cazangi, F. J. Von Zuben, and M. F. Figueiredo. Autonomous navigation system applied to collective robots with ant-inspired communication. In *Proceedings of the Genetic and Evolutionary Computation Conference*, pages 121–128, Washington, D.C., 2005. ACM Press.

[4] N. Getz. Control of balance for a nonlinear nonholonomic no-minimum phase model of a bicycle. In *American Control Conference*, Baltimore, June 1994.

[5] J. Koza. Genetic evolution and co-evolution of computer programs. In *Artificial Life II*, pages 603–629, 1992.

[6] A. Lubberts and R. Miikkulainen. Co-evolving a go-playing neural network. In R.K. Belew and H. Juillé, editors, *Proceedings of the Genetic and Evolutionary Computation Conference*, pages 14–19, 2001.

[7] I. Martinos, T. Schouwenaars, J. De Mot, and E. Feron. Hierarchical cooperative multi-agent navigation using mathematical programming. In *Proceedings of the Allerton Conference on Communication, Control and Computing*, 2003.

[8] T. Al-Shihabi and R.R. Mourant. Toward more realistic behavior models for autonomous vehicles in driving simulators. *Transportation Research Record*, (1843):41–49, 2003.

[9] R.R. Mourant and S. Marangos. A virtual environments editor for driving scenes. In *Proceedings of the International Conference on Computer, Communication and Control Technologies*, volume IV, pages 379–384, Orlando, Florida, 2003.

[10] S. Nolfi and D. Floreano. Coevolving predator and prey robots: Do “arms races” arise in artificial evolution? *Artificial Life*, 4(4):311–335, 1998.

[11] L. Panait and S. Luke. A comparative study of two competitive fitness functions. In W.B. Langdon et al., editor, *Proceedings of the Genetic and Evolutionary Computation Conference (GECCO)*, pages 503–511. Morgan Kaufmann, 2002.

[12] M. Potter and K. De Jong. Cooperative coevolution: an architecture for evolving coadapted subcomponents. *Evolutionary computation*, 8(1):1–29, 2000.

[13] D. Vrajitoru and R. Mehler. Multi-agent autonomous pilot for single-track vehicles. In *Proceedings of the IASTED Conference on Modeling and Simulation*, Oranjestad, Aruba, 2005.