Discovering Comfortable Driving Strategies Using Simulation-Based Multiobjective Optimization

Erik Dovgan and Tea Tušar
Jožef Stefan Institute, Jamova cesta 39, Ljubljana, Slovenia
Jožef Stefan International Postgraduate School, Jamova cesta 39, Ljubljana, Slovenia
E-mail: {erik.dovgan, tea.tusar}@ijs.si, http://dis.ijs.si

Matija Javorski
Faculty of Mechanical Engineering, University of Ljubljana, Aškerčeva cesta 6, Ljubljana, Slovenia
E-mail: matija.javorski@fs.uni-lj.si, http://www.fs.uni-lj.si

Bogdan Filipič
Jožef Stefan Institute, Jamova cesta 39, Ljubljana, Slovenia
Jožef Stefan International Postgraduate School, Jamova cesta 39, Ljubljana, Slovenia
E-mail: bogdan.filipic@ijs.si, http://dis.ijs.si

Keywords: driving strategies, multiobjective optimization, traveling time, fuel consumption, driving comfort

Received: May 28, 2012

Driving a vehicle along a route consists of control actions applied to the vehicle by taking into account the vehicle and route states. Control actions are usually selected by optimizing the traveling time and the fuel consumption. However, the resulting vehicle behavior can be uncomfortable for the driver/passengers. The comfort is measured as the change of acceleration, i.e., jerk. To obtain more comfortable driving strategies, we introduce comfort as an objective to the Multiobjective Optimization algorithm for discovering Driving Strategies (MODS), thus obtaining the Multiobjective Optimization algorithm for discovering Comfortable Driving Strategies (MOCDS). The two algorithms are compared on a real-world route. The results show that MOCDS finds more comfortable driving strategies than MODS, while not significantly deteriorating their traveling time and fuel consumption. The most significant improvement in comfort is achieved on driving strategies with low fuel consumption, which are highly uncomfortable and therefore have the most room for improvement. On the other hand, the driving strategies found by MODS with short traveling time are already comfortable and therefore cannot be additionally improved.

Povzetek: Prispevek predstavlja algoritem za iskanje strategij vožnje, ki ne optimira le časa vožnje in porabe goriva, temveč tudi udobje za voznika/potnike.

1 Introduction

When driving a vehicle along a route, two objectives are usually optimized: the traveling time and the fuel consumption. However, the algorithms optimizing only these objectives usually find pulse-and-glide driving strategies [10, 12]. Such driving strategies repeatedly exchange high throttle percentage and zero throttle percentage. Therefore, the acceleration continuously changes which significantly reduces the driving comfort [14]. Low driving comfort is unacceptable from the user point of view, even though such driving strategies efficiently reduce the traveling time and the consumed fuel. Consequently, the driving comfort has to be taken into account when discovering driving strategies.

In our previous work we designed and implemented the Multiobjective Optimization algorithm for discovering Driving Strategies (MODS) [2, 3], which searches for driving strategies by modeling a real vehicle driving on a real route as a black box, and optimizing the traveling time and the fuel consumption. The obtained driving strategies are better than the driving strategies found by optimization algorithms used so far [4], i.e., predictive control [16] and dynamic programming [7, 8]. However, MODS fails to find comfortable driving strategies, especially with low fuel consumption. In order to obtain comfortable driving strategies, we introduce the third objective, i.e., the comfort that has to be maximized, or equivalently, the discomfort that has to be minimized. To quantify the discomfort, Nilsson [14] suggested to measure the magnitude of the jerk. This measure was added to the dynamic programming algorithm presented in [7] and the obtained algorithm found more comfortable driving strategies. Wu et al. [19] presented a car-following model focused on passenger comfort. A comfortable vehicle driving was achieved by limiting the jerk. The same measure was also used by Haj-Fraj et al. [6] who searched for the optimal control of gear shift operations. In order to obtain a comfortable shifting, a dynamic programming algorithm was implemented which minimizes the jerk.

Comfort can be defined in various other ways, too. For example, a comfortable driving strategy may be a strategy that does not change the control actions frequently. In addition, it is not compulsory to consider the comfort as an objective in the algorithm to obtain comfortable driving strategies. For example, Gerdts [5] and Kirches et al. [9] developed single objective algorithms that search for the optimal double-lane-change manoeuver on a short (140 m) horizontal route and minimize the traveling time. The same problem was tackled by Logist et al. [13], who developed a multiobjetive algorithm that minimizes the traveling time and the fuel consumption. Although none of them optimizes the comfort, they obtained comfortable driving strategies that do not change the control actions frequently. However, such driving strategies were obtained by using model-based approaches, which cannot be applied when a black-box simulator is used. In addition, the algorithms were tested only on a short horizontal artificially generated route. Therefore, it is not clear if these algorithms would produce comfortable driving strategies also on data from (longer) real-world routes with inclined route segments and velocity limits.

In this paper we present the two-level Multiobjective Optimization algorithm for discovering Comfortable Driving Strategies (MOCDS) that minimizes the traveling time, fuel consumption and discomfort, i.e., jerk. The lower-level algorithm is based on breadth-first search [17] and Nondominated Sorting Genetic Algorithm (NSGA-II) [1]. The best input-parameter values for the lower-level algorithm are found by the upper-level evolutionary algorithm. MOCDS returns a set of nondominated [1] driving strategies and leaves the selection of the preferred driving strategy to the user.

The paper is further organized as follows. The MOCDS algorithm is described in Section 2. Section 3 presents the experiments and the obtained results. Finally, Section 5 concludes the paper with ideas for future work.

2 The Algorithm for Discovering Comfortable Driving Strategies

This section presents the two-level algorithm for discovering comfortable driving strategies (MOCDS) that minimizes the traveling time t, the fuel consumption c, and the driving discomfort d.

2.1 Strategy representation and evaluation

A driving strategy is a set of connections between the vehicle and route states, i.e., the state space, on the one hand, and the weights used to select the control action that is applied to the vehicle during the driving simulation on the other hand. The vehicle state is defined with the vehicle velocity, while the route state is defined with the inclinations and the velocity limits of the current and the next segments, and the route to the next segment. The control action is defined with the throttle and braking percentage ε_{V} and the gear g_{V}, while the weights are the consumption weight ω_{c} and time weight ω_{t}. The state space, control actions and weights are discretized in advance. The subspaces obtained by the state space discretization are called hypercubes [18]. Each hypercube stores a consumption weight and a time weight. These data are used to select the appropriate control action when the vehicle and route states correspond to the hypercube.

The driving strategy is evaluated with a black-box vehicle driving simulator that was implemented based on the vehicle description from [11, 15] and is described in [4]. The simulator receives the control action for the vehicle, simulates the vehicle driving for one route step, where the length of a step is Δs, and returns the spent time, the consumed fuel, the driving discomfort, and the new vehicle and route states. The new vehicle and route states are then used to select the current hypercube, and consequently to find the new control action that is used for the simulation of the next route step. This process continues until the traveling along the entire route has been simulated, i.e., until $\sum_1^x \Delta s = s$, where s is the length of the route and x is the number of already simulated route steps.

2.2 Lower-level algorithm

The lower-level algorithm is a deterministic multiobjective algorithm for discovering comfortable driving strategies that minimizes the traveling time, the fuel consumption and the driving discomfort (see Figure 1). It starts with a single driving strategy with empty hypercubes. Then it simulates the vehicle driving for several route steps with several driving strategies until the driving along the entire route has been simulated (Main procedure in Figure 1). If the current hypercube of a driving strategy at a route step is empty, the driving strategy is cloned for each discrete set of weights $\{\omega_{\mathrm{c}}, \omega_{\mathrm{t}}\}$ and this data is stored in the hypercube. More precisely, the driving strategy is cloned when the vehicle and route states correspond to the current hypercube for the first time during the driving simulation. When the current hypercube stores the weights, these data are used to select the most preferred control action as shown in Figure 1. The control action is selected by predicting the vehicle driving for N_{P} prediction steps ahead for each possible discrete control action $\{\varepsilon_{\mathrm{V}}, g_{\mathrm{V}}\}$. Afterwards, the spent time t, the consumed fuel c and the driving discomfort d are com-

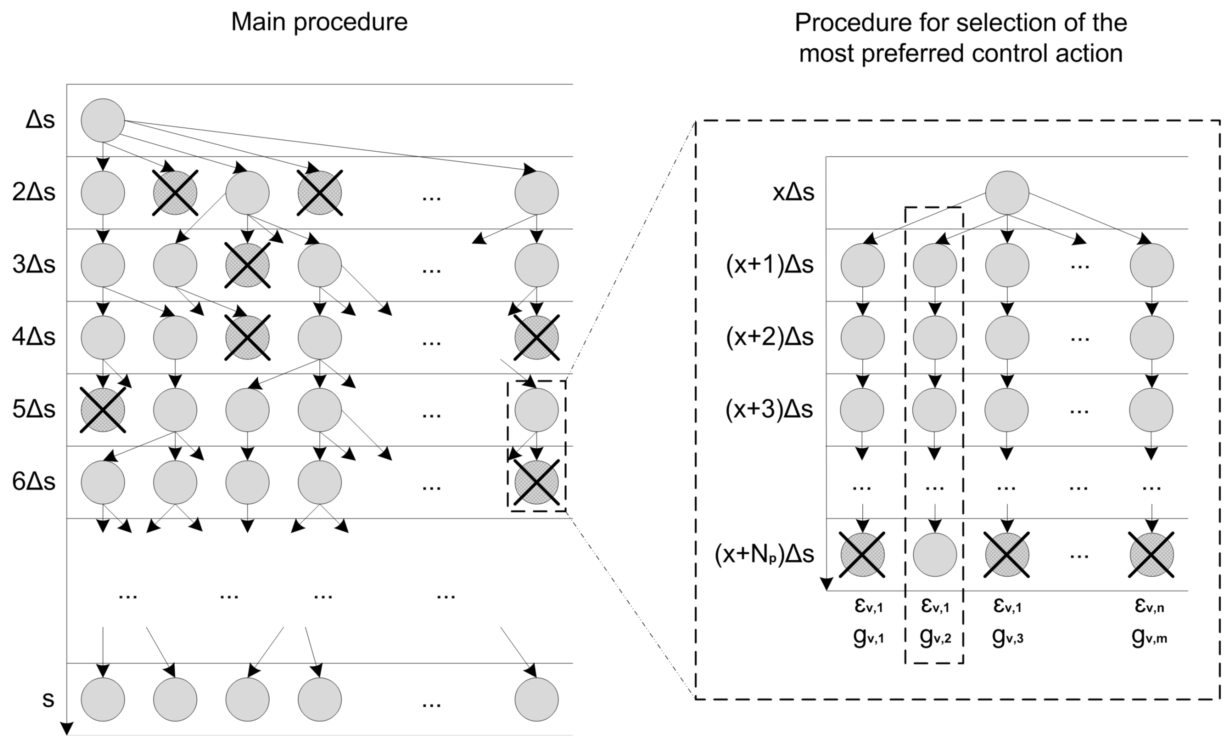


Figure 1: The lower-level algorithm for discovering driving strategies.

bined into the cost function f:

$$f = \omega_c c + \omega_t t + (1 - \omega_c - \omega_t)d, \tag{1}$$

and the control action that minimizes f is selected for one step simulation (in the Main procedure in Figure 1). The driving discomfort is calculated by summing up the magnitudes of the jerk, i.e., the differences in acceleration a denoted as Δa, during the driving simulation as follows:

$$d = \sum_{x\Delta s}^{(x+N_P)\Delta s} |\Delta a| \tag{2}$$

Since the driving strategies are cloned, the number of driving strategies grows exponentially. To reduce their number, fast nondominated sort and crowding distance mechanisms from the Nondominated Sorting Genetic Algorithm (NSGA-II) [1] are used at each route step to select the most promising driving strategies with respect to the objectives and maintain a constant number of driving strategies. The non-promising driving strategies are deleted and are marked with "$\times$" in the Main procedure in Figure 1. When the vehicle driving along the entire route has been simulated, the algorithm returns a set of nondominated driving strategies.

The lower-level algorithm requires the following input parameters:

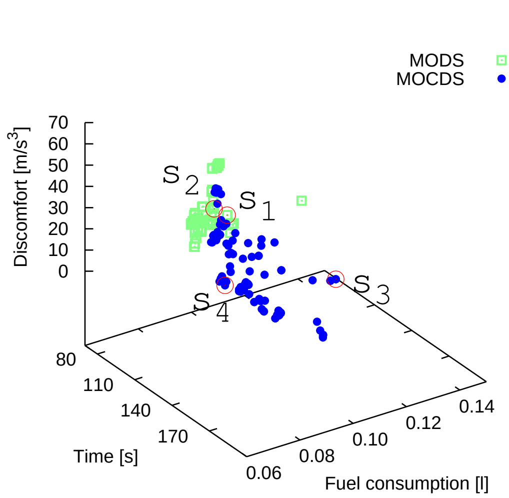
- discretization of vehicle and route state space,
- discretization of control actions,
- discretization of weights, and
- number of prediction steps N_P.

2.3 Upper-level algorithm

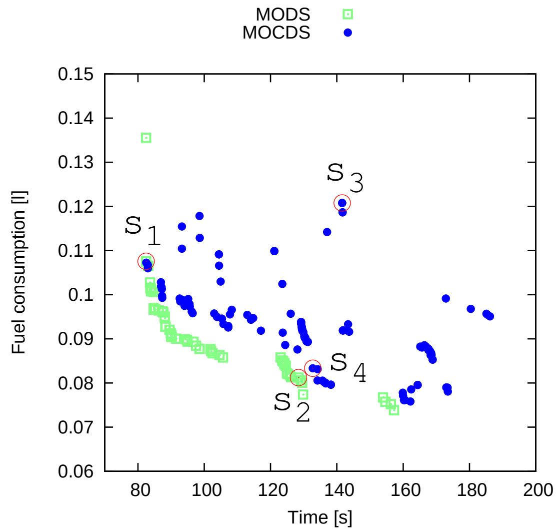
The upper-level evolutionary algorithm searches for the best sets of input-parameter values for the lower-level algorithm and maximizes the hypervolume [20]. A set of input-parameter values is an upper-level solution. The upper-level algorithm applies evolutionary principles, i.e., selection, crossover and mutation, to the set of upper-level solutions through several generations [1]. The evaluation of an upper-level solution is carried out as follows. Firstly, the lower-level algorithm finds the nondominated driving strategies using the input-parameter values stored in the upper-level solution. Finally, the hypervolume covered by the driving strategies is calculated. For more details see [3, 4].

3 Experiments and Results

MOCDS was tested on data describing a real-world route and the obtained driving strategies were compared to the driving strategies obtained by MODS in order to determine the influence of the comfort as an objective. The selected route was an urban road of around 1100 m that includes a few uphills and downhills. Its characteristics are summarized in Figure 2.



(a)



(b)

Figure 3: Driving strategies found by MODS and MOCDS.

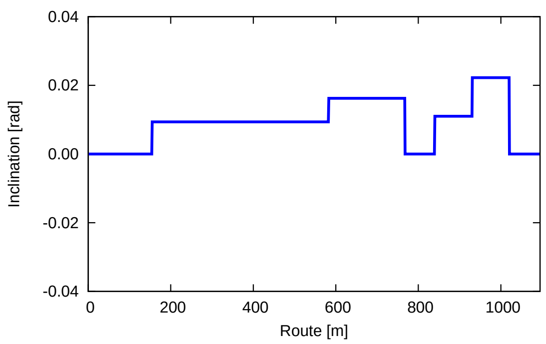


Figure 2: Inclinations of the testing route; the velocity limit is 50 km/h along the entire route.

Figure 3 shows the nondominated driving strategies found by MOCDS and MODS. Specifically, Figure 3(a) shows the driving strategies in the objective space of all three objectives, while Figure 3(b) shows a projection of the driving strategies to the objective space with only the objectives t and c. Ideally, MOCDS should find all the driving strategies obtained by MODS. However, Figure 3 shows that MOCDS does not find (all) these driving strategies. This is due to the fact that the number of driving strategies grows exponentially at each route step and, therefore, several driving strategies have to be deleted as described in Subsection 2.2. More precisely, MOCDS and MODS maintain the same number of driving strategies but MOCDS has a larger search space due to the additional objective. Consequently, several driving strategies that are promising in the values of t and c are deleted by MOCDS since a larger search space has to be covered by the same number of driving strategies. Those driving strategies are not deleted by MODS and therefore MODS better optimizes the traveling time and fuel consumption. Nevertheless, the results show that MOCDS is able to find, in addition to the comfortable driving strategies, driving strategies similar to the ones found by MODS in terms of traveling time and fuel consumption.

Four interesting driving strategies were further analyzed. They are marked in Figure 3 as follows:

– s_1 is a driving strategy with short traveling time found by MODS;

– s_2 is a driving strategy with low fuel consumption found by MODS;

– s_3 is the driving strategy with the highest comfort but also long traveling time and high fuel consumption found by MOCDS; and

– s_4 is a driving strategy found by MOCDS, which has similar traveling time and fuel consumption but significantly higher comfort than s_2.

The objective values of these driving strategies are shown in Table 1. Moreover, the vehicle behavior obtained by applying these driving strategies can be seen in Figures 4 and 5. These figures show the control actions, i.e., the throttle and braking percentage and the gear, the vehicle velocity and the jerk along the entire route. The results show that in order to obtain highly comfortable driving strategies (e.g., s_3), the control actions must rarely change. Consequently, the vehicle velocity slowly changes and the jerk is low along the entire route. On the other hand, when the comfort is not taken into account (e.g., s_1 and s_2), the control actions change frequently and consequently the jerk is higher. Finally, Figure 5 shows the vehicle behavior obtained by applying the driving strategies that are similar

Driving strategy	t [s]	c [l]	d [m/s^3]
s_1	82.46	0.1076	12.784
s_2	128.41	0.0812	44.573
s_3	141.60	0.1208	1.298
s_4	132.75	0.0833	9.518

Table 1: The objective values of the driving strategies marked in Figure 3.

in terms of traveling time and fuel consumption, but significantly differ in comfort (see also Table 1). More precisely, it shows that a driving strategy of the same quality in terms of traveling time and fuel consumption but significantly more comfortable can be obtained by reducing the changes of control actions. Such driving strategy can be obtained by MOCDS but not by MODS.

Although the MOCDS driving strategies change the control actions, such as the gear, less frequently than the MODS driving strategies, the number and frequency of changes remains high when nondominated driving strategies in terms of traveling time and fuel consumption are taken into account, e.g., s_4 (see Figure 5). Nevertheless, MOCDS also finds driving strategies with a significantly lower number and frequency of changes, see the driving strategy with the highest comfort, s_3 (see Figure 4). To even further reduce the number and frequency of changes of control actions, the comfort should be redefined, e.g., by penalizing the changes in control actions, or the search space should be limited, for example, by restricting the changes of control actions.

Figure 6 shows the driving strategies found by MODS and MOCDS which are nondominated with regard to objectives t and c. These driving strategies found by MOCDS are the most interesting ones since they are similar to the driving strategies obtained by MODS in terms of traveling time and fuel consumption. The figure shows that MOCDS does not find more comfortable driving strategies than MODS when traveling time is short (the driving strategies outside the dashed rectangle in Figure 6), since MODS finds driving strategies with short traveling time that are already comfortable and cannot be improved in comfort anymore. However, MOCDS finds significantly more comfortable driving strategies than MODS when fuel consumption is low (the driving strategies inside the dashed rectangle in Figure 6). This is due to the fact that MODS finds driving strategies with low fuel consumption that are highly uncomfortable and, therefore, have the most room for improvement. In summary, the results show that MOCDS finds more comfortable driving strategies than MODS, while not significantly deteriorating the other objectives, especially when the fuel consumption is reduced.

Finally, the computation and simulated times are shown in Table 2. It shows that the average computation time per driving strategy is longer than the simulated traveling times of driving strategies but still in the order of minutes.

4 Conclusion

We presented a two-level multiobjective optimization algorithm for discovering comfortable driving strategies (MOCDS). The lower-level algorithm is a deterministic multiobjecitve algorithm that searches for comfortable driving strategies, while the upper-level algorithm is an evolutionary algorithm that searches for the best input-parameter values for the lower-level algorithm. The obtained driving strategies were compared to the driving strategies found by the algorithm that does not optimize the comfort, i.e., MODS. The results show that comfortable driving strategies either rarely change the control actions or reduce the changes of the control actions. Moreover, when comparing the driving strategies with low fuel consumption, those found by MOCDS are significantly more comfortable than those found by MODS. However, when comparing the driving strategies with short traveling time, there is no significant difference in comfort between those found by MOCDS and those found by MODS, since both are already comfortable and cannot be improved anymore.

In the future work, we will test other approaches for increasing the driving comfort. These approaches will include an objective other than jerk. However, the comfortable driving strategies may be obtained by not including the third objective but limiting the search space, e.g., restricting the changes of control actions. It would be also interesting to include the third objective in the algorithms used so far, i.e., predictive control and dynamic programming, and/or limit the search space of these algorithms to compare the obtained driving strategies with those found by MOCDS.

Acknowledgement

We would like to thank Matjaž Gams for his contribution to the research presented in this paper.

References

[1] K. Deb, A. Pratap, S. Agrawal and T. Meyarivan (2002) A fast and elitist multiobjective genetic algorithm: NSGA-II, *IEEE Transactions on Evolutionary Computation*, IEEE Computational Intelligence Society, vol. 6, no. 2, pp. 182–197.

[2] E. Dovgan, M. Gams and B. Filipič (2011) A multiobjective optimization algorithm for discovering driving strategies, *GECCO'11 proceedings*, ACM, New York, pp. 751–754.

[3] E. Dovgan, M. Javorski, M. Gams and B. Filipič (2011) A two-level approach for discovering driving strategies according to conflicting objectives, *Proceedings of the 14th International Multiconference Information Society*, Jožef Stefan Institute, Ljubljana, pp. 41–44.

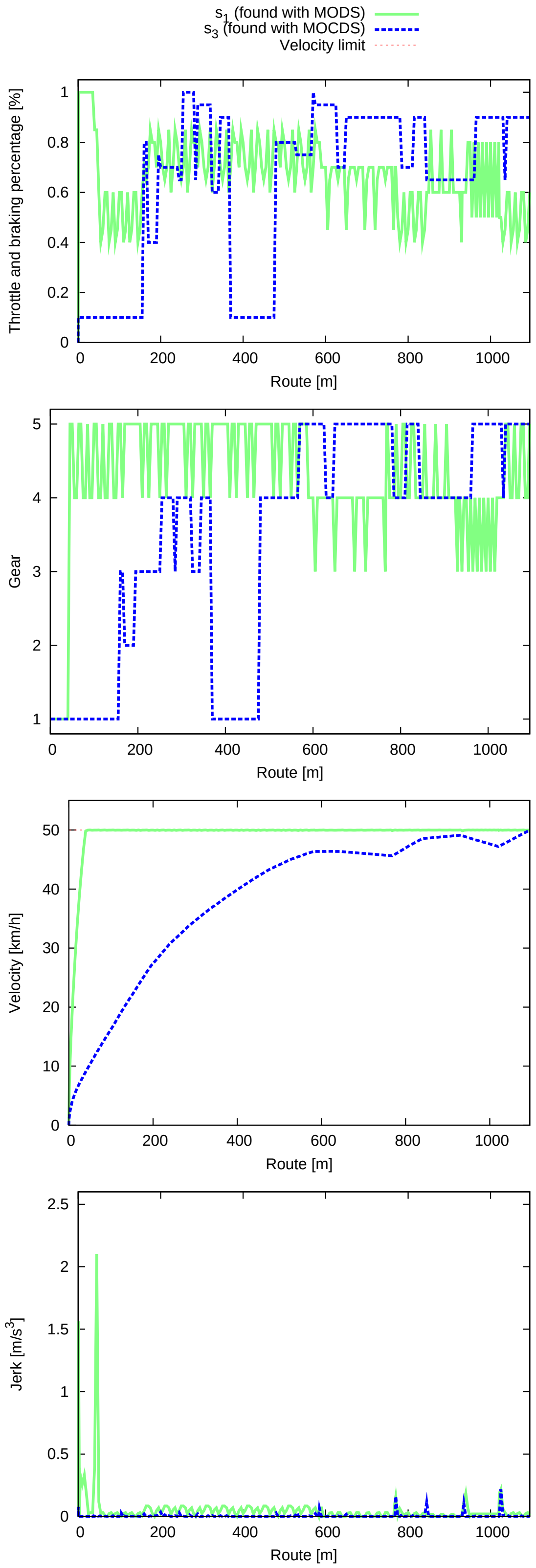


Figure 4: Examples of vehicle behavior obtained by applying the driving strategies with high fuel consumption (s_1 and s_3 from Figure 3).

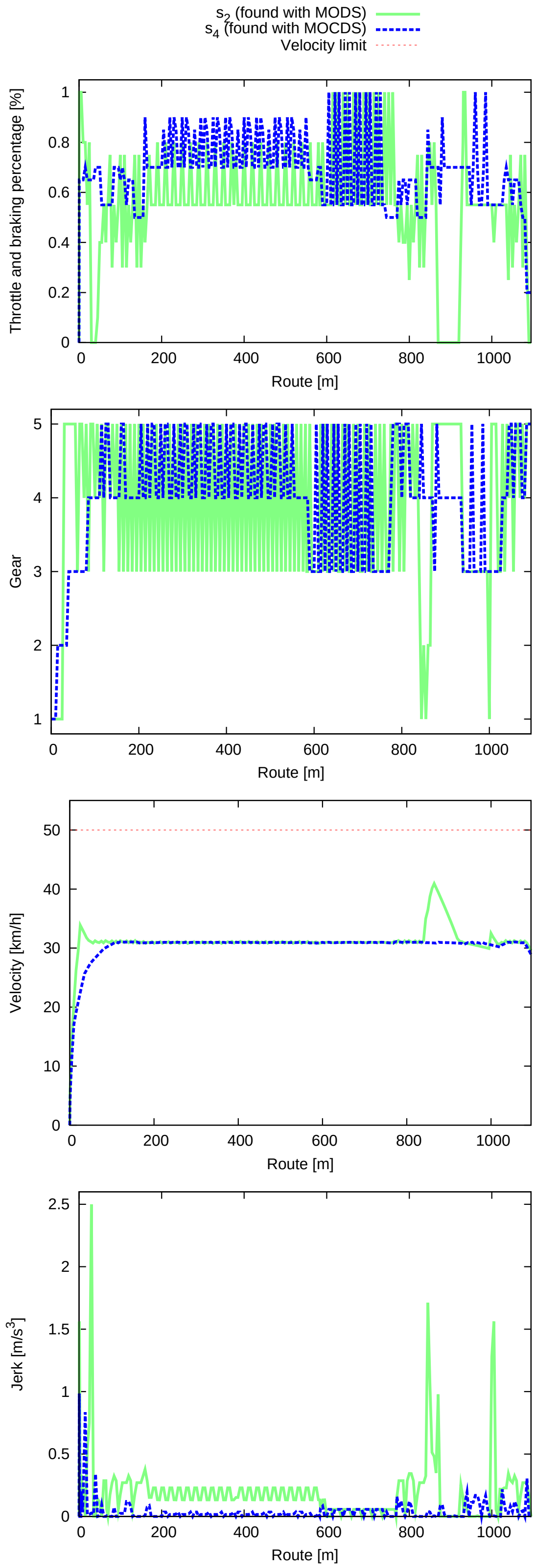


Figure 5: Examples of vehicle behavior obtained by applying the driving strategies with low fuel consumption (s_2 and s_4 from Figure 3).

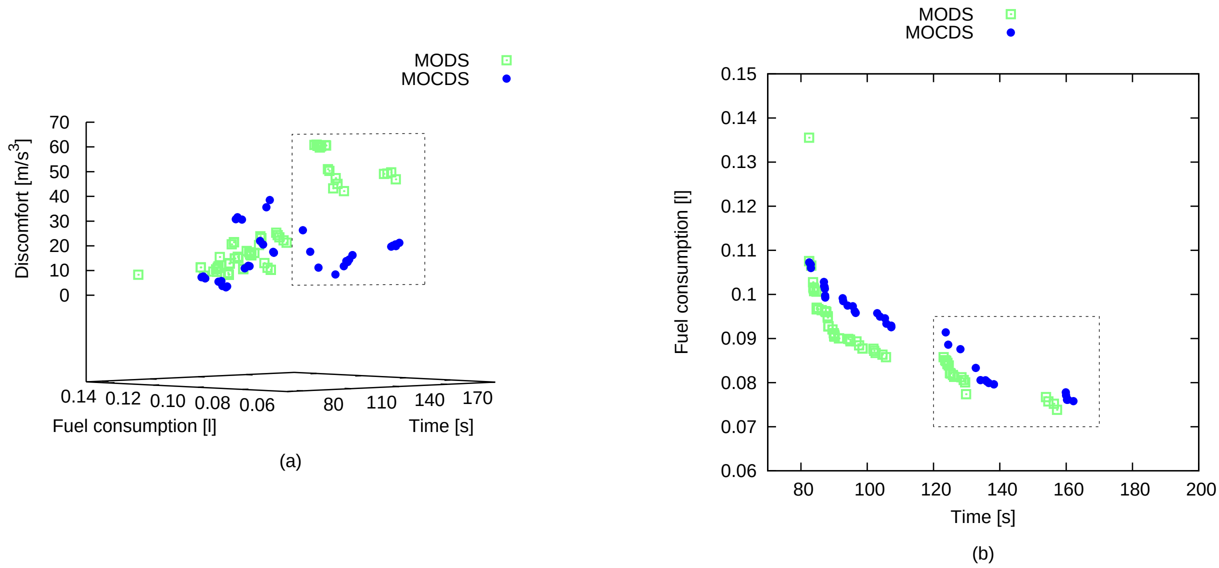


Figure 6: Nondominated driving strategies in the objective space with only the objectives t and c found by MOCDS and MODS. The dashed rectangle denotes the driving strategies with low fuel consumption.

Algorithm	MODS	MOCDS
Total computation time (h:m:s)	6:41:49	23:18:48
Number of found nondominated driving strategies	90	119
Average computation time per driving strategy (m:s)	4:28	11:45
Minimal simulated traveling time of driving strategies (m:s)	1:22	1:22
Maximal simulated traveling time of driving strategies (m:s)	2:37	3:06

Table 2: Comparison of computation and simulated times of MODS and MOCDS.

[4] E. Dovgan, M. Javorski, T. Tušar, M. Gams and B. Filipič (2012) Discovering Driving Strategies with a Multiobjective Optimization Algorithm, submitted for publication.

[5] M. Gerdts (2005) Solving mixed-integer optimal control problems by branch&bound: a case study from automobile test-driving with gear shift, *Optimal Control Applications and Methods*, John Wiley & Sons, vol. 26, no. 1, pp. 1–18.

[6] A. Haj-Fraj and F. Pfeiffer (2001) Optimal control of gear shift operations in automatic transmissions, *Journal of the Franklin Institute*, Elsevier, vol. 338, pp. 371–390.

[7] E. Hellstrom, J. Aslund and L. Nielsen (2010) Design of an efficient algorithm for fuel-optimal look-ahead control, *Control Engineering Practice*, Elsevier, vol. 18, no. 11, pp. 1318–1327.

[8] E. Hellstrom, M. Ivarsson, J. Aslund and L. Nielsen (2009) Look-ahead control for heavy trucks to minimize trip time and fuel consumption, *Control Engineering Practice*, Elsevier, vol. 17, no. 2, pp. 245–254.

[9] C. Kirches, S. Sager, H. G. Bock and J. P. Schloder (2010) Time-optimal control of automobile test drives with gear shifts, *Optimal Control Applications and Methods*, John Wiley & Sons, vol. 31, no. 2, pp. 137–153.

[10] D. Kroushl (2005) *Prius marathoners top 100 mpg*, available online: `http://www.toyota.com/html/hybridsynergyview/2005/fall/marathon.html`.

[11] G. Lechner and H. Naunheimer (1999) *Automotive Transmissions: Fundamentals, Selection, Design and Application*, Springer.

[12] J. Lee (2009) *Vehicle inertia impact on fuel consumption of conventional and hybrid electric vehicles using acceleration and coast driving strategy*, Virginia Polytechnic Institute and State University.

[13] F. Logist, S. Sager, C. Kirches and J. F. Van Impe (2010) Efficient multiple objective optimal control of dynamic systems with integer controls, *Journal of Process Control*, Elsevier, vol. 20, no. 7, pp. 810–822.

[14] A. Nilsson (2009) *Fuel and ride comfort optimization in heavy vehicles*, Linkoping University.

[15] T. Randolph (2007) *Waste heat regeneration systems for internal combustion engines*, available online: http://www.heat2power.net/downloads/GPC2007/20070618_heat2power_GPC_WHR_presentation.pdf.

[16] L. Del Re, F. Allgower, L. Glielmo, C. Guardiola and I. Kolmanovsky (2010), *Automotive Model Predictive Control: Models, Methods and Applications*, Springer.

[17] S. J. Russell and P. Norvig (2010) *Artificial Intelligence: A Modern Approach*, Prentice Hall.

[18] E. Vareilles, M. Aldanondo and P. Gaborit (2009) How to take into account piecewise constraints in constraint satisfaction problems, *Engineering Applications of Artificial Intelligence*, Elsevier, vol. 22, no. 4–5, pp. 778–785.

[19] Z. Wu, Y. Liu and G. Pan (2009) A smart car control model for brake comfort based on car following, *IEEE Transactions on Intelligent Transportation Systems*, IEEE Intelligent Transportation Systems Society, vol. 10, no. 1, pp. 42–46.

[20] E. Zitzler and L. Thiele (1999) Multiobjective evolutionary algorithms: A comparative case study and the strength Pareto approach, *IEEE Transactions on Evolutionary Computation*, IEEE Computational Intelligence Society, vol. 3, no. 4, pp. 257–271.