# DEALING WITH COMFORT AS AN OBJECTIVE IN MULTIOBJECTIVE OPTIMIZATION OF DRIVING STRATEGIES


*Erik Dovgan* (1, 2), *Matija Javorski* (3), *Tea Tušar* (1, 2), *Bogdan Filipič* (1, 2)
(1) Department of Intelligent Systems, Jožef Stefan Institute, Jamova cesta 39, 1000 Ljubljana, Slovenia
(2) Jožef Stefan International Postgraduate School, Jamova cesta 39, 1000 Ljubljana, Slovenia
(3) Faculty of Mechanical Engineering, University of Ljubljana, Aškerčeva cesta 6, 1000 Ljubljana, Slovenia
Tel: +386 1 4773393; fax: +386 1 4773131
e-mail: erik.dovgan@ijs.si



**ABSTRACT**

When a person drives a vehicle along a route, he/she optimizes the traveling time and the fuel consumption. The same problem is tackled by the Multiobjective Optimization algorithm for discovering Driving Strategies (MODS) which we designed and implemented. However, the driving strategies found with MODS change the control actions frequently (more frequently than humans) and, therefore, the driving comfort is reduced. To improve the driving comfort, we introduced it as an objective in MODS, thus obtaining the Multiobjective Optimization algorithm for discovering Comfortable Driving Strategies (MOCDS). The two algorithms were compared on data from a real-world route and the results show that MOCDS finds highly comfortable driving strategies, especially when the fuel consumption is reduced. On the other hand, when the traveling time is reduced, MODS already finds comfortable driving strategies that cannot be additionally improved.


## 1 INTRODUCTION

Comfort is important when driving a vehicle along a route. Nevertheless, it is not explicitly optimized by humans, i.e., the goal of vehicle driving is not to feel as comfortable as possible. Usually, two other goals are pursued: minimization of the traveling time and minimization of the fuel consumption. Nevertheless, human driving strategies are mostly comfortable. However, this is not the case when driving strategies are found with optimization algorithms, since they do not implicitly optimize the driving comfort. Therefore, the comfort has to be explicitly introduced in the optimization algorithms as the third objective in order to find comfortable driving strategies acceptable from the user point of view.

In our previous work we designed and implemented the Multiobjective Optimization algorithm for discovering Driving Strategies (MODS) [1] which searches for driving strategies by optimizing the traveling time and the fuel consumption. Although it finds good driving strategies (better than relared algorithms [1], such as predictive control [2] and dynamic programming [3]), it fails to find comfortable driving strategies. To overcome this shortage, we introduced the third objective, i.e., the comfort that has to be maximized, or equivalently, the discomfort that has to be minimized, to MODS, thus obtaining the Multiobjective Optimization algorithm for discovering Comfortable Driving Strategies (MOCDS) [4]. The discomfort is measured as the magnitude of the jerk, i.e., the magnitude of changes in acceleration [5]. In this paper we compare the driving strategies obtained with MODS with the driving strategies obtained with MOCDS. The comparison focuses on weaknesses of MODS. More precisely, we analyze the cases where MODS fails to find comfortable driving strategies, i.e., driving strategies similar to the driving strategies found with MOCDS in terms of driving comfort.

The paper is further organized as follows. The MODS and MOCDS algorithms are described in Section 2. Section 3 presents the experiments and the obtained results. Finally, Section 4 concludes the paper with ideas for future work.

## 2 MULTIOBJECTIVE DISCOVERY OF (COMFORTABLE) DRIVING STRATEGIES

MODS and MOCDS are two-level algorithms where the algorithm at the lower level is a deterministic algorithm that searches for driving strategies, while the upper-level algorithm is an evolutionary algorithm that searches for the best input parameter values for the lower-level algorithm.

### 2.1 The lower-level algorithm

The lower-level algorithm is a deterministic multiobjective algorithm that searches for driving strategies by minimizing the traveling time (MODS and MOCDS), the fuel consumption (MODS and MOCDS) and the discomfort (MOCDS only). MODS and MOCDS have very similar lower-level algorithms. The only difference is that the lower-level algorithm of MOCDS deals with an additional objective, i.e., the discomfort, while the core of the algorithms remains the same.

The driving strategies are sets of hypercubes [6], where hypercubes are defined with discretization of the vehicle and route state space. Hypercubes store the fuel consumption weights (MODS and MOCDS) and discomfort weights (MOCDS only) that are used to select the best control action, i.e., throttle and braking percentage and gear, when the vehicle and route state correspond to the hypercube. The algorithm searches for the best driving strategies by starting with a single driving strategy with empty hypercubes. Next, it simulates the vehicle driving by steps with several driving strategies until the driving along the entire route has been simulated. At each step, the current hypercube is checked and if it does not contain the weight(s), i.e., the hypercube has not been "visited" yet, the driving strategy is cloned for each discrete value of fuel consumption weight (MODS and MOCDS) and discomfort weight (MOCDS only), and the weights are stored in the hypercubes of cloned driving strategies. When the weight(s) is/are determined, the control action is selected by predicting the vehicle driving for a number of prediction steps ahead and selecting the control action which minimizes the weighted sum of spent time (MODS and MOCDS), consumed fuel (MODS and MOCDS) and driving discomfort (MOCDS only). The cloning of driving strategies increases the number of driving strategies exponentially. Therefore, in order to maintain a constant number of most promising driving strategies, the fast nondominated sorting and the crowding distance mechanisms from the Nondominated Sorting Genetic Algorithm (NSGA-II) [7] are used at each route step.

### 2.2 The upper-level algorithm

The upper-level algorithm searches for the best input parameter values for the lower-level algorithm, i.e., the discretization of vehicle and route state space, the discretization of weight(s) and control actions, and the number of prediction steps. The algorithm is an evolutionary algorithm [8] that applies selection, crossover and mutation on a population of sets of input parameter values through several generations, and maximizes the hypervolume [9] covered by the driving strategies found with the lower-level algorithm. For more details see [1].

## 3 EXPERIMENTS AND RESULTS

MODS and MOCDS were tested on data describing a real-world urban route of about 1100 m. The route characteristics are shown in Figure 1.

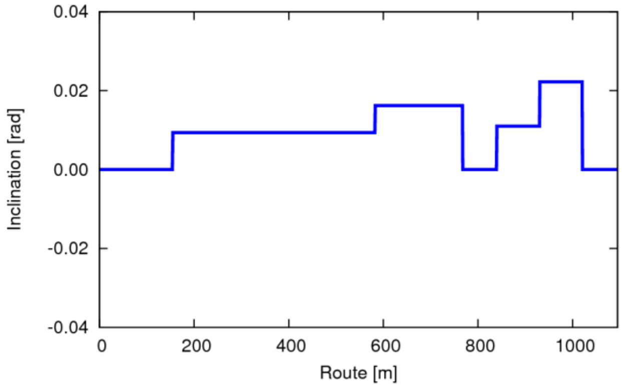


Figure 1: *Inclinations of the test route; the velocity limit is 50 km/h along the entire route.*

Figure 2 shows the driving strategies found with MODS and MOCDS. More precisely, the figure shows only the driving strategies that are nondominated in terms of traveling time and fuel consumption, since such driving strategies are the most interesting ones. The driving strategies on the left side of Figure 2 have short traveling time and high fuel consumption. On the other hand, the driving strategies on the right side of Figure 2 have long traveling time but low fuel consumption. The results show that both algorithms find similar driving strategies in terms of the driving comfort when traveling time is minimized (left side of Figure 2). More precisely, the MODS driving strategies are already comfortable when the traveling time is minimized and cannot be additionally improved by MOCDS. On the other hand, MOCDS finds significantly more comfortable driving strategies than MODS when the fuel consumption is minimized (right side of Figure 2, the driving strategies inside the dashed rectangle). The driving strategies that are significantly more comfortable are the most interesting ones, therefore, four of them labeled with $s_1$, $s_2$, $s_3$ and $s_4$ in Figure 2 were additionally analyzed. Strategies $s_1$ and $s_2$ were obtained with MODS, while $s_3$ and $s_4$ with MOCDS. $s_1$ and $s_3$ are similar in terms of the traveling time and the fuel consumption. The same holds for $s_2$ and $s_4$. Figures 3 and 4 show the control actions and the vehicle behavior obtained by applying these driving strategies. The figures show that in order to increase the driving comfort, i.e., decrease the jerk, the control actions have to change less frequently (see throttle and braking percentage, and gear of $s_3$ and $s_4$). Consequently, the vehicle velocity obtained by applying $s_3$ and $s_4$ is more constant than the vehicle velocity obtained by applying $s_1$ and $s_2$. Finally, the jerk obtained by applying $s_3$ and $s_4$ is lower than the jerk obtained by applying $s_1$ and $s_2$ along the entire route.

## 4 CONCLUSION

This paper compares the Multiobjective Optimization algorithm for discovering Driving Strategies (MODS) and Multiobjective Optimization algorithm for discovering Comfortable Driving Strategies (MOCDS). Both algorithms are two-level algorithms, where the lower-level algorithm is a deterministic multiobjective algorithm for discovering driving strategies, and the upper-level algorithm is a single objective evolutionary algorithm that searches for the best input parameter values for the lower-level algorithm. The only difference between these two algorithms is an additional objective used in MOCDS. More precisely, MODS minimizes the traveling time and the fuel consumption, while MOCDS additionally minimizes the discomfort. The algorithms were tested on data from a real-

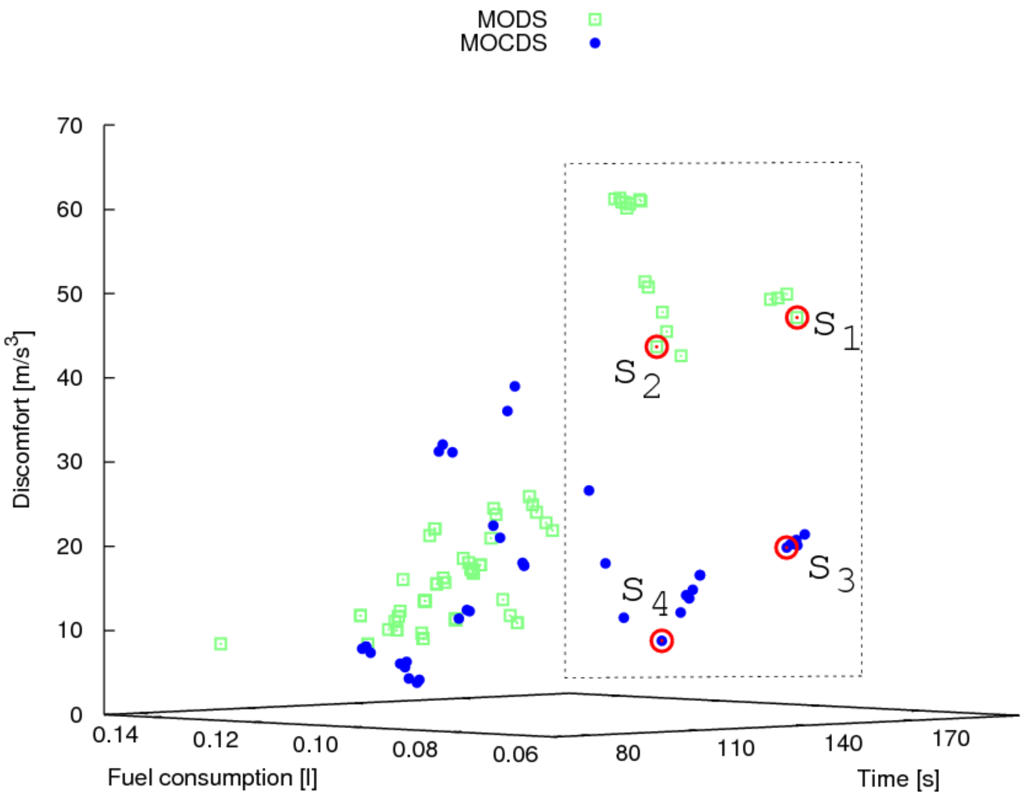


Figure 2: *Nondominated driving strategies in terms of traveling time and fuel consumption obtained with MODS and MOCDS. The dashed rectangle denotes the driving strategies with low fuel consumption.*

world route. The results show that MODS and MOCDS find similar driving strategies in terms of driving comfort when the traveling time is minimized. On the other hand, MOCDS finds significantly better driving strategies in terms of driving comfort than MODS when fuel consumption is minimized. The highly comfortable driving strategies are the most interesting, therefore, four of them were additionally analyzed. The analysis shows that in order to increase the driving comfort, the control actions have to change less frequently.

The future work will include testing additional routes. Moreover, we will test other functions for calculating the comfort. In addition, it would be also interesting to include the third objective in the algorithms of other authors and compare the obtained driving strategies with the MOCDS driving strategies.

## References


[1] E. Dovgan, M. Javorski, T. Tušar, M. Gams, B. Filipič. Discovering Driving Strategies with a Multiobjective Optimization Algorithm. *Submitted for publication*. 2012.

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

[3] E. Hellstrom, J. Aslund, L. Nielsen. Design of an efficient algorithm for fuel-optimal look-ahead control. *Control Engineering Practice*. Vol. 18. No. 11. pp. 1318–1327. 2010.

[4] E. Dovgan, T. Tušar, M. Javorski, B. Filipič. Discovering Comfortable Driving Strategies Using Simulation-Based Multiobjective Optimization. *Submitted for publication*. 2012.

[5] A. Nilsson. *Fuel and ride comfort optimization in heavy vehicles*. Linkoping University. 2009.

[6] E. Vareilles, M. Aldanondo, P. Gaborit. How to take into account piecewise constraints in constraint satisfaction problems. *Engineering Applications of Artificial Intelligence*. Vol. 22. No. 4–5. pp. 778–785. 2009.

[7] K. Deb, A. Pratap, S. Agarwal, T. Meyarivan. A fast and elitist multiobjective genetic algorithm: NSGA-II. *IEEE Transactions on Evolutionary Computation*. Vol. 6. No. 2. pp. 182–197. 2002.

[8] L. Davis. *Handbook of Genetic Algorithms.* Van Nostrand Reinhold. New York. 1991.

[9] E. Zitzler, L. Thiele. Multiobjective evolutionary algorithms: A comparative case study and the strength Pareto approach. *IEEE Transactions on Evolutionary Computation*. Vol. 3. No. 4. pp. 257–271. 1999.

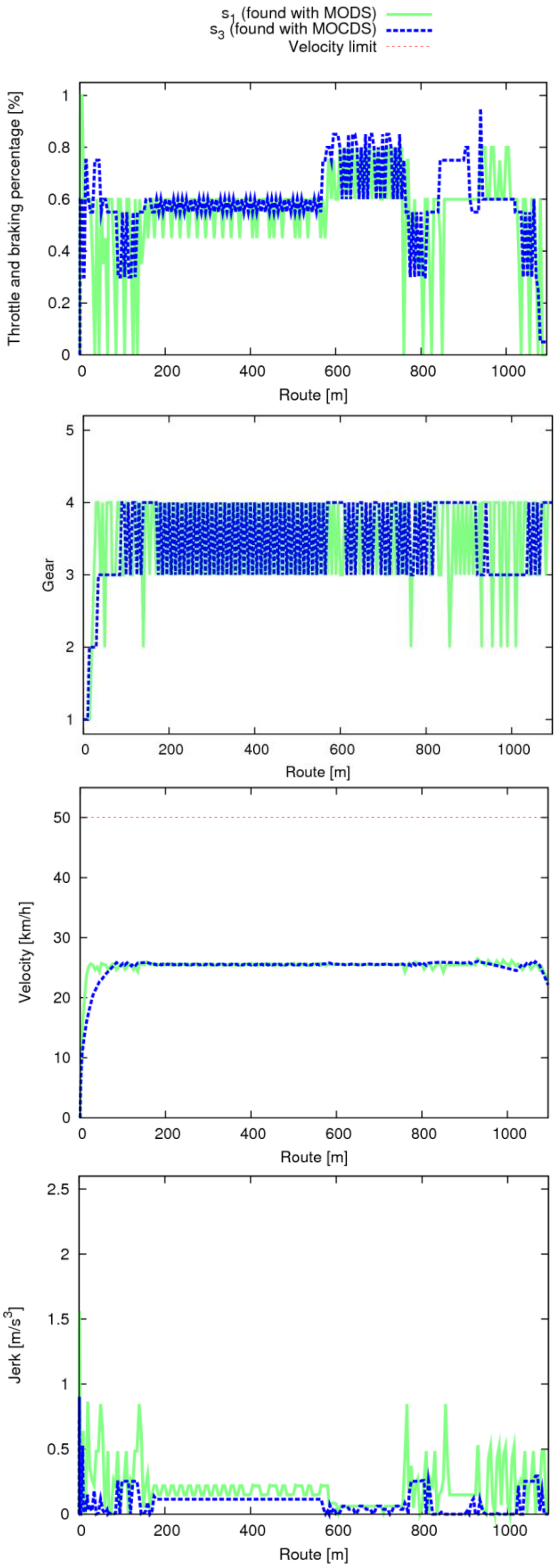


Figure 3: *Examples of vehicle behavior obtained by applying the driving strategies $s_1$ and $s_3$ from Figure 2.*

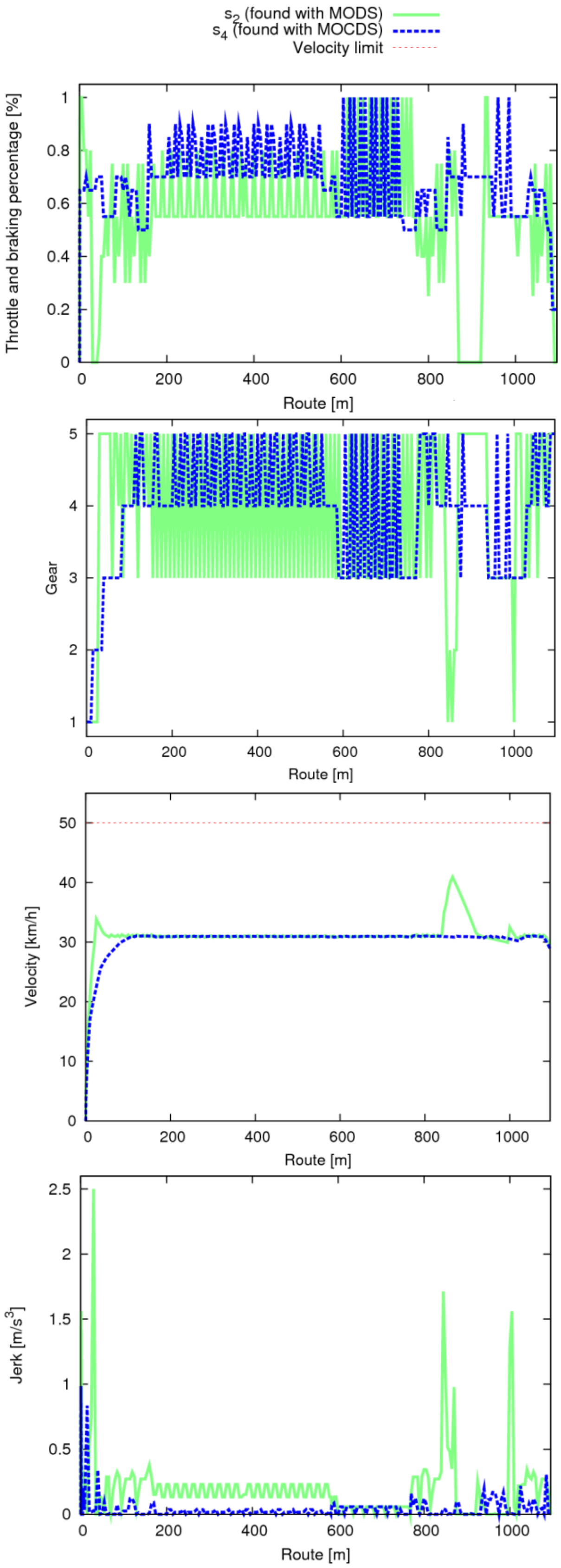


Figure 4: *Examples of vehicle behavior obtained by applying the driving strategies $s_2$ and $s_4$ from Figure 2.*