Influence diagrams for speed profile optimization: computational issues

Jiří Vomlel and Václav Kratochvíl
Institute of Information Theory and Automation
Czech Academy of Sciences
vomlel@utia.cas.cz, velorex@utia.cas.cz

Abstract

Influence diagrams were applied to diverse decision problems. However, the general theory is still not sufficiently developed if the variables are continuous or hybrid and the utility functions are nonlinear. In this paper, we study computational problems related to the application of influence diagrams to vehicle speed profile optimization and suggest an approximation of the nonlinear utility functions by piecewise linear functions.

1 Introduction

In this paper, we use an example inspired by a real problem – a car moving on a road – to study various issues related to computations with influence diagrams. The modeled car is equipped with an automatic transmission and its speed is controlled using the throttle and the brakes. There are various speed limits on the road (e.g., 130 km/h on a highway or 50 km/h in an urban area). The goal is to find an optimal strategy for passing the road while minimizing (*i*) time spend on the road, (*ii*) the fuel consumption, or (*iii*) a mixture of both.

There are two principal ways for representing the solution:

speed profile – a function that assigns a speed value to all points on the road,

control policy – a function that assigns control values of the throttle and the brakes for every possible speed and to every point on the road.

The *control policy* is more general. In case of the *speed profile* the vehicle uses an additional regulator that follows the speed profile by controlling the car acceleration using the throttle and the brakes. In the *control policy*, the control signals are already precomputed for all admissible speed values. This becomes especially handy in real situations when the driver has to suddenly slow down or even to stop due to an unexpected traffic situation and the precomputed speed profile becomes obsolete.

Since all variables (speed, acceleration, throttle, brakes) are continuous by their nature, it would be natural to work with continuous or hybrid influence diagrams. Unfortunately, the theory of continuous influence diagrams is not sufficiently developed (especially for nonlinear utility functions). In this paper we perform experiments with discrete influence diagrams. One of our goals is to analyze the shape of nonlinear relations and propose good approximations.

An influence diagram (Howard and Matheson, 1981) is a Bayesian network augmented with decision variables and utility functions. Influence diagrams were applied to diverse decision problems. Recently, we introduced influence diagrams to the problem of optimization of a vehicle speed profile. We performed computational experiments in which an influence diagram was used to optimize the speed profile of a Formula 1 race car at the Silverstone F1 circuit (Kratochvíl and Vomlel, 2015).

In this paper we split the vehicle path into n segments of the same length s. For each segment of the vehicle path there are two random variables V_i and V_{i+1}, one decision variable U_i, and one utility potential f_{i+1}. In Figure 1, we present the structure of a part of the ID corresponding to one segment of the path. The values of i are from the set $\{1, 2, \ldots, n-1\}$. The physical model of the vehicle is given in

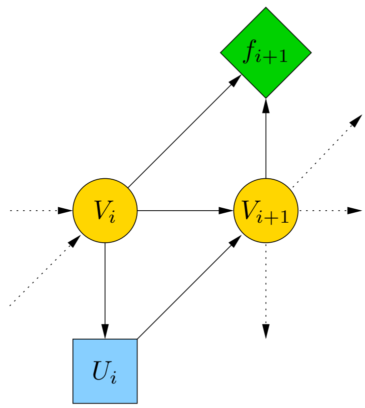


Figure 1: A part of the influence diagram for one path segment

Section 2. It is used to define the probability and utility functions of the influence diagram.

In this paper, we generally allow variables to be discrete or continuous and the main theoretical results presented in the paper are valid for both types of variables. However, experiments were performed with discrete variables only. For the sake of brevity we do not discuss related work in this paper – we refer interested readers to Kratochvíl and Vomlel (2015).

2 Vehicle physics

We model the vehicle behavior using the laws of physics. To model the engine behavior and the fuel consumption we assume the vehicle to be a passenger car and we follow the approach of Chang and Morlok (2005). The values of variables[1] describing the car state are defined by the following formulas[2].

Velocity at the coordinate $i+1$

$$v_{i+1} = v(a_i, v_i) = \sqrt{(v_i)^2 + 2 \cdot s \cdot a_i} \ , \tag{1}$$

where a_i and v_i is acceleration and velocity at the coordinate i, respectively. Let a_t^{max} be the maximum tangential acceleration of the vehicle,[3] a_t^{min} be the maximum tangential deceleration,[4] Engine acceleration at segment i is defined by the following equation:

$$a_i^e = a^e(u_i, v_i) = \begin{cases} u_i \cdot (a_t^{max} - c_a \cdot v_i) & \text{if } u_i > 0 \\ u_i \cdot a_t^{min} & \text{otherwise.} \end{cases} \tag{2}$$

In this paper we will consider a vehicle with values $a_t^{max} = 4$, $c_a = 0.06$, and $a_t^{min} = 5$. u_i is the control at the coordinate i. It has values from $\langle 0, 1 \rangle$ where negative ones correspond to braking, positive ones to using throttle. Deceleration caused by friction forces and aerodynamic drag is

$$a_i^d = a^d(v_i) = c_r + c_v \cdot (v_i)^2 \tag{3}$$

where $c_r = 0.1273$ and $c_v = 0.000257$ for the considered vehicle. Acceleration at segment $[i, i+1]$ is

$$a_i = a(u_i, v_i) = a^e(u_i, v_i) - a^d(v_i) \ . \tag{4}$$

By putting equations (1)–(4) altogether we get

$$\begin{aligned} v_{i+1} &= v'(u_i, v_i) \\ &= \begin{cases} \sqrt{(v_i)^2 + 2 \cdot s \cdot \Big(u_i \cdot (a_t^{max} - c_a \cdot v_i) - c_r - c_v \cdot (v_i)^2\Big)} & \text{if } u_i > 0 \\ \sqrt{(v_i)^2 + 2 \cdot s \cdot \Big(u_i \cdot a_t^{min} - c_r - c_v \cdot (v_i)^2\Big)} & \text{otherwise.} \end{cases} \end{aligned} \tag{5}$$

Time spent at the path segment $[i, i+1]$

$$t_{i+1} = t(v_i, v_{i+1}) = s \cdot \left(\frac{v_i + v_{i+1}}{2}\right)^{-1} \ . \tag{6}$$

[1] We use the symbol without subscript to denote the function that specifies the variable's value.

[2] Note that the relations between variables follow the edges of the influence diagram from Figure 1

[3] It is a property of the vehicle engine (without considering the aerodynamic drag and friction forces). The real maximum acceleration is lower.

[4] It is a property of the vehicle brakes (without considering the aerodynamic drag and friction forces). The real maximum deceleration is higher.

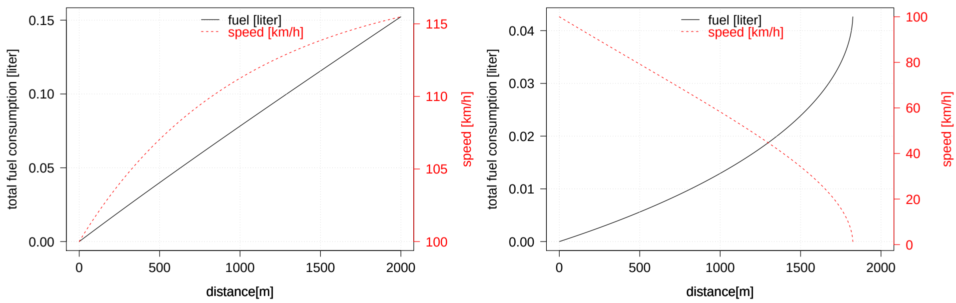


Figure 2: The total fuel consumption and the speed with the initial speed 100 km/h and the control $u = 0.2$ and $u = 0$, respectively.

When modeling the fuel consumption, we assume it is proportional to the work done by the engine (Chang and Morlok, 2005), which is the acceleration multiplied by the vehicle mass and by the distance s, plus a low fuel consumption constant per time:

$$\begin{aligned} g_{i+1} &= g(v_i, v_{i+1}) \\ &= c_g \cdot s \cdot m \cdot \max\left\{0, \frac{(v_{i+1})^2 - (v_i)^2}{2s} + a_d(v_i)\right\} + g^{min} \cdot t(v_i, v_{i+1}) \ , \end{aligned} \quad (7)$$

where the considered constants are the vehicle mass in kilograms $m = 1759$, the fuel rate in liter per one Joul of energy $c_g = 10^{-7}$, and the constant fuel consumption in liter per second $g^{min} = 1/3600$. The vehicle behavior in terms of the fuel consumption and its speed is illustrated in Figure 2.

3 Speed constraints in the model

We assume that a maximum speed v_i^{max} and a minimum speed v_i^{min} is given in advance at each path coordinate $i = 1, \ldots, n$. Let $\mathcal{V}_i$ denote the set of admissible speed values at i and let the admissible set at the end of the path be

$$\mathcal{V}_n = \{v \in \mathcal{V}, v_n^{min} \leq v \leq v_n^{max}\} \ . \quad (8)$$

We apply the constraints during optimization process where we allow to select only those control signals $u_i \in \mathcal{U}$ that lead to $v_{i+1} = v'(u_i, v_i)$ belonging to $\mathcal{V}_{i+1}$. We define functions $\mathcal{U}_i(V_i)$ that for each value v_i of variable V_i provide the set of admissible control values:

$$\mathcal{U}_i(v_i) = \{u \in \mathcal{U} : v'(u_i, v_i) \in \mathcal{V}_{i+1}\} \ . \quad (9)$$

This set inductively defines the set of admissible speed values at i for which there exist an admissible control value:

$$\mathcal{V}_i = \{v \in \mathcal{V} : v_i^{min} \leq v \leq v_i^{max}, \ \mathcal{U}_i(v) \neq \emptyset\} \ . \quad (10)$$

This, again, inductively defines set $\mathcal{U}_{i-1}(v_{i-1})$. This process is repeated until $i = 1$.

4 Expected utility of a control policy

In the sequel we will use the following abbreviations

$$\begin{aligned}\sum_{V_i} \varphi(V_i, \cdot) &= \sum_{v_i \in \mathcal{V}_i} \varphi(V_i = v_i, \cdot) \text{ and} \\ \max_{U_i} \psi(U_i, V_i) &= \max_{u_i \in \mathcal{U}_i(v_i)} \psi(U_i = u_i, V_i = v_i) \ .\end{aligned}$$

$\mathcal{M}$ will be a generalized marginalization operation. The operator $\mathcal{M}$ acts differently for a discrete random variable A, a continuous random variable B, and a decision variable U of a (probability or utility) potential ψ:

$$\mathop{\mathcal{M}}_{A} \psi(A, \ldots) = \sum_{A} \psi(A, \ldots), \qquad \mathop{\mathcal{M}}_{B} \psi(B, \ldots) = \int \psi(B = b, \ldots)\, db,$$
$$\mathop{\mathcal{M}}_{U} \psi(U, \ldots) = \max_{U} \psi(U, \ldots) \ .$$

The control of the vehicle speed will be realized by means of the control policy.

Definition 1. Control policy is a set of functions

$$\delta = \left\{\delta(U_i|V_i) : \ i \in \{1, \ldots, n-1\}, v_i \in \mathcal{V}\right\}$$

such that for all $i = 1, \ldots, n$ and all $v_i \in \mathcal{V}$ it maps $u_i \in \mathcal{U}$ to values from $[0, 1]$ and it holds that

$$\sum_{u_i \in \mathcal{U}} \delta(U_i = u_i | V_i = v_i) = 1 \ . \tag{11}$$

Definition 2. A control policy δ is deterministic if for all $i = 1, \ldots, n$ and all $v_i \in \mathcal{V}$ it holds that there is a function $u_i : \mathcal{V} \to \mathcal{U}$ such that for all $u \in \mathcal{U}$

$$\delta(U_i = u | V_i = v_i) = \begin{cases} 1 & \text{if } u = u_i(v_i) \\ 0 & \text{otherwise.} \end{cases} \tag{12}$$

Remark 1. In this paper, all considered policies will be deterministic.

Definition 3. The expected value E_f of a deterministic control policy δ specified by functions u_i is the sum or the integral over all possible configurations of random variables of the products of the probability and the criteria value of that configuration:

$$E_f(\delta) = \mathop{\mathcal{M}}_{V_1, \ldots, V_n} P(V_1, \ldots, V_n) \cdot f(V_1, \ldots, V_n) \tag{13}$$

where

$$P(V_1, \ldots, V_n) = P(V_1) \cdot \prod_{i=1}^{n-1} P(V_{i+1} | U_i = u_i(v_i), V_i) \tag{14}$$

$$f(V_1, \ldots, V_n) = \sum_{i=1}^{n-1} f(V_i, V_{i+1}) \ . \tag{15}$$

The criteria to be optimized will be the expected value E_f of a deterministic control policy.

Definition 4. An optimal deterministic policy δ^* is a deterministic policy such that it holds for all control policies δ that

$$E_f(\delta) \quad \leq \quad E_f(\delta^*) \ . \tag{16}$$

We will use symbol u_i^* to denote the function $u_i : \mathcal{V} \rightarrow \mathcal{U}$ that specifies the optimal deterministic policy δ^* according to Definition 2. The symbol $u_i^*(V_i)$ denotes the set of functions u_i^* for all values v_i of variable V_i.

Using the recursive application of the commutative and distributive laws we get the following theorem that specifies a computationally efficient algorithm for finding an optimal decision policy. Note that our algorithm is just a special case of general inference methods for influence diagrams (Jensen et al., 1994; Shenoy, 1992; Shachter and Peot, 1992). But since our influence diagram has a simple structure it is useful to derive a simple inference algorithm tailored for the task we solve. Note that, in this case, the algorithm does not involve divisions. The computations can be also viewed as a special case of dynamic programming (Bellman, 1957).

Theorem 1.

$$E_f^* \quad = \quad E_f(\delta^*) \quad = \quad \mathop{\mathcal{M}}_{V_1} P(V_1) \cdot \psi(V_1) \ , \tag{17}$$

where $\psi(V_1)$ is computed recursively for $i = 1, \ldots, n-1$ as

$$\psi(V_i) \quad = \quad \max_{U_i} \mathop{\mathcal{M}}_{V_{i+1}} P(V_{i+1}|V_i, U_i) \cdot \Big(f(V_i, V_{i+1}) + \psi(V_{i+1}) \Big) \ . \tag{18}$$

with the recursion terminal values being $\psi(V_n) = \mathbb{0}(V_n)$, where $\mathbb{0}(V_n)$ stands for the vector taking for all states of variable V_n value zero.

The proof can be found in Appendix A.

Remark 2. In each step $i = 1, \ldots, n$, an optimal deterministic policy is specified (according to Definition 2) by a function $u_i : \mathcal{V} \rightarrow \mathcal{U}$ such that $u_i(v_i) = u_i^*(v_i)$, where $u_i^*(v_i)$ is a value of U_i that maximize formula (18) for a given v_i .

5 Deterministic continuous model for the total time minimization

In this section we will present a special case for which it is easy to find an optimal speed profile even if all variables are continuous. The optimality criteria will be the total time $\sum_{i=1}^{n-1} t(v_i, v_{i+1})$ and the goal will be to minimize it.

Definition 5. Let $v'(u_i, v_i)$ be the function specified in (5). If for $i = 1, \ldots, n-1$ it holds that

$$P(V_{i+1} = v_{i+1} | U_i = u_i, V_i = v_i) \quad = \quad \begin{cases} 1 & \text{if } v_{i+1} = v'(u_i, v_i) \\ 0 & \text{otherwise.} \end{cases}$$

then we say that the vehicle behavior is deterministic.

Next we present a corollary of Theorem 1 that specifies an algorithm for the case of a deterministic vehicle behavior.

Corollary 1. *Assume that the vehicle behavior is deterministic. Then*

$$E_f^* \quad = \quad E_f(\delta^*) \quad = \quad \mathop{\mathcal{M}}_{V_1} P(V_1) \cdot \psi(V_1) \ , \tag{19}$$

where $\psi(V_1)$ is computed recursively for $i = 1, \ldots, n-1$ and for all $v_i \in \mathcal{V}$ as:

$$\psi(v_i) \quad = \quad f\left(v_i, v'\big(\max\ \mathcal{U}_i(v_i), v_i\big)\right) + \psi\left(v'\big(\max\ \mathcal{U}_i(v_i), v_i\big)\right) \ . \tag{20}$$

The recursion terminal values are defined as $\psi(v_n) = 0$ for all $v_n \in \mathcal{V}$.

Proof. Formula (20) follows from (18) - the considered criteria is the minimization of the total time. Therefore $\max_{U_i}$ corresponds to picking the highest value from $\mathcal{U}_i(v_i)$. Also, note that for the deterministic vehicle behavior and for any potential $\xi(V_i, V_{i+1})$ it holds for all $u_i \in \mathcal{U}, v_i \in \mathcal{V}$ that

$$\mathop{\mathcal{M}}_{V_{i+1}} P(V_{i+1}|U_i = u_i, V_i = v_i) \cdot \xi(V_i = v_i, V_{i+1}) \quad = \quad \xi(V_i = v_i, V_{i+1} = v'(u_i, v_i)) \ .$$

□

From Corollary 1 we derive computationally efficient Algorithm 1 that can be used to compute efficiently the optimal speed profile of the vehicle satisfying the speed constraints. We will use function $w(u_i, v_{i+1})$ that gives the initial speed v_i such that after driving distance s with the control u_i the speed is v_{i+1}. The idea behind the algorithm is that the function f, which is is to be maximized, implies that the best policy for any $v_i, i = 1, \ldots, n-1$ is to speedup as much as possible to be able to slow down by maximum allowed decceleration to satisfy that $v_j^* \leq v_j^{max}$ for all $j > i$.

First, the maximal speed profile is constructed from the speed constraints and the maximum decceleration of the vehicle. Second, the best policy is found with the maximum acceleration until the speed meets the maximum profile constructed in the first stage of the algorithm.

6 Experiments

In the experiments, we considered the speed and control variables to be discrete, i.e. sets $\mathcal{V}, \mathcal{U}$ are finite with the discretization steps being d_V, d_U, respectively. In

```
input  : v_i^max, i = 1,...,n – maximal speed values
output: v_i^*, i = 1,...,n – speed values maximizing E_f (see Definition 3)
v_n^* = v_n^max;
for i = n − 1,...,1 do
    v_i^* = w(−1, v_{i+1}^*);
    if (v_i^* > v_i^max) then
        v_i^* = v_i^max;
    end
end
for i = 1,...,n − 1 do
    v_{i+1} = v'(+1, v_i^*);
    if (v_{i+1} < v_{i+1}^*) then
        v_{i+1}^* = v_{i+1};
    end
end
```

Algorithm 1: Optimal speed profile construction for the deterministic vehicle behavior.

this case we use linear approximations of utility values of $v_i = v(u_{i-1}, v_{i-1})$, $v_i \notin \mathcal{V}$ by a mixture of utility values $\underline{v}_i \leq v_i$ and $\overline{v}_i \geq v_i$ that are the closest values from $\mathcal{V}$ to v_i. The mixture weights are the probabilities that are defined as

$$P(V_i = v | U_{i-1} = u_{i-1}, V_{i-1} = v_{i-1}) \quad = \quad \begin{cases} 1 - \dfrac{|v - v_i|}{d_V} & \text{for } v = \underline{v}_i, \overline{v}_i \\ 0 & \text{otherwise.} \end{cases} \tag{21}$$

To get an into the problem we performed the following computational experiment. Assume a road section of length 2 km in a flat area and the speed limit of 90 km/h in the whole section and with three short subpaths with the speed limit of 50 km/h. Let $s = 20$ m. The speed limit profile of the road can be seen in the upper part of Figure 3. The area of forbidden speeds is highlighted. The black line illustrates a speed profile of a car starting with initial speed of 80 km/h, following the control policy calculated using Theorem 1. The probability potentials were defined as in (21) and $\mathcal{V}$,$\mathcal{U}$ had 100 values.

Using deterministic relation between variables, we are inevitably working with states of zero probability. If the task is minimization of a criteria the zero probability values may lead to wrong solutions. Therefore we formulate the problem as a maximization task. Instead of the minimization of a specific mixture of the fuel consumption and the total time, we maximize the *savings* with respect to the worst performance. As the optimality criteria we use a mixture of the normalized total time savings and the normalized fuel savings. The normalized utility functions for the time and fuel savings at segment $[i, i+1]$ are defined as

$$f_{i+1}^t = f^t(v_i, v_{i+1}) = 1 - \frac{t(v_i, v_{i+1})}{t^{max}}, \qquad f_{i+1}^f = f^f(v_i, v_{i+1}) = 1 - \frac{g(v_i, v_{i+1})}{g^{max}},$$

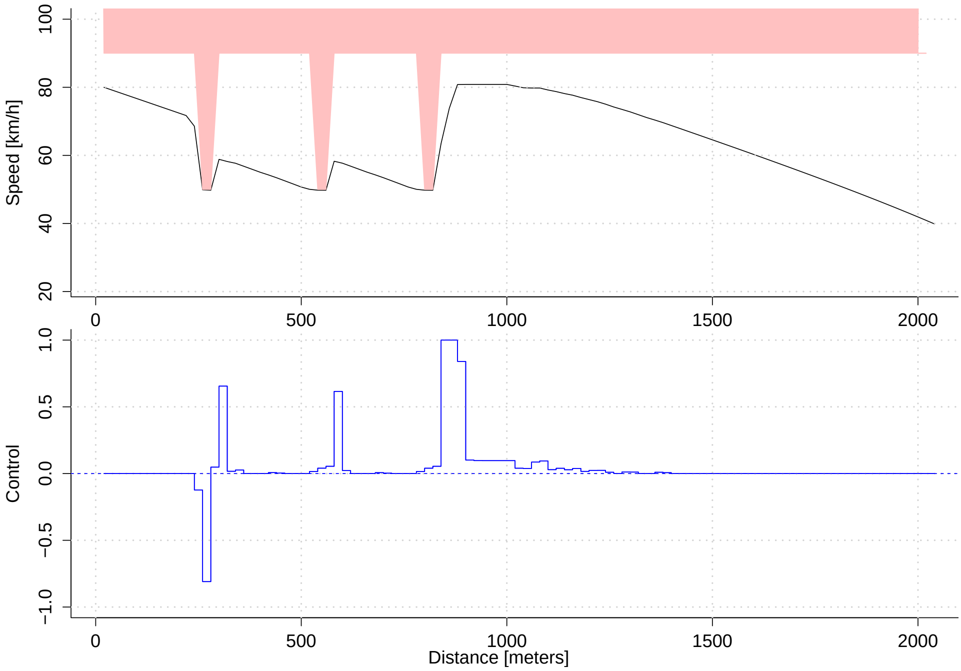


Figure 3: Generated speed profile and corresponding control profile

where $t(v_i, v_{i+1})$ and $g(v_i, v_{i+1})$ are defined by formula (6) and (7), respectively. t^{max} and g^{max} are the maximum possible time spent and fuel consumption in one segment. In the experiments we used utility function f defined (for $\alpha = 0.5$)

$$f = \sum_{i=1}^{n-1} \alpha f_i^t + (1-\alpha) f_i^f \quad . \tag{22}$$

Remark 3. For the speeds close to zero the values of $t(v_i, v_{i+1})$ and $g(v_i, v_{i+1})$ are very high. This would imply high values of t^{max} and g^{max}. Consequently, for most of other speed values the functions f_{i+1}^t and f_{i+1}^f would provide values close to one. This may cause rounding errors. To avoid this problem we disregard speeds lower than 4 km/h for the definitions of t^{max} and g^{max}.

In Figure 3, we present results of our numerical experiment. In the upper part the computed optimal speed profile is presented. The corresponding values of the control variable (the throttle or the brakes) are depicted in the lower part of the figure. It is interesting to note that most of the time the car is in a so called *flying mode*, which is driving with the neutral gear with no throttle or brakes. In case of a longer road without speed limits, the optimal speed stabilizes (for this settings) around 80 km/h - see the road section around 900 m. Because there is no requirement on the speed at the end - the algorithm decided to enter the flying mode, similarly, as in the case when the vehicle is approaching 50 km/h speed limits.

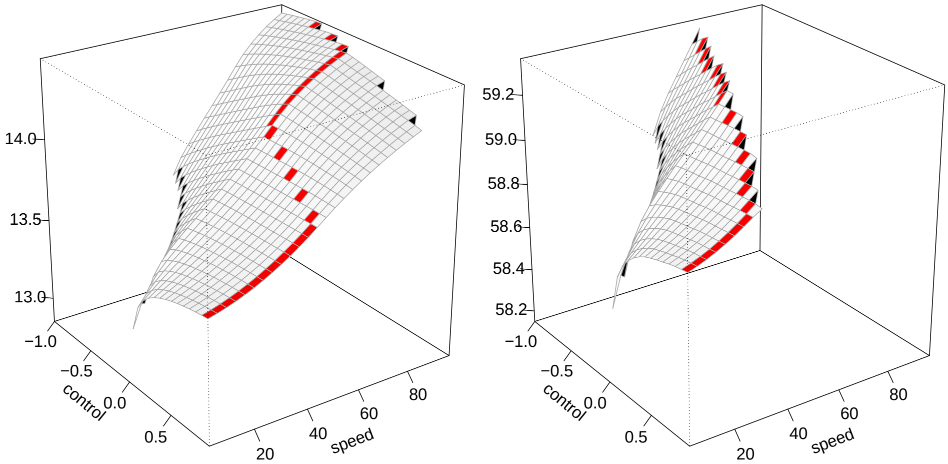
According to Theorem 1, the calculations are performed in the direction from the road's furthermost point backwards. The development of the values of the expected utility function in two randomly selected points of the path can be seen in Figure 4. Axis x and y correspond to speed and control, axis z refers to the expected utility. Figure 4a corresponds to the iteration 15 of the algorithm ($i = 87$), while Figure 4b corresponds to the iteration 62 ($i = 39$). Forbidden combinations of speed and control are not depicted. In Figure 4 the highlighted facets corresponds to maximal expected utility for the given speed. For every value of speed, we store respective control as the optimal deterministic policy in this points - see Remark 2.

From Figure 4a we can deduce that the best strategy for a low speed is to use the full throttle (first, to speed up and than to use the flying mode). For speeds of about $50 - 60$ km/h it starts to be better to use the flying mode immediately. For very high speeds, the optimal strategy has to be to use the brakes in order to satisfy the speed limits. The overall view of the image suggests that global optimum is at the highest speeds. It is logical, because with a high initial speed a lot of the fuel and time can be saved. Figure 4b corresponds to a driving situation just before reaching one of the speed limits of 50 km/h. Therefore more combinations of speed and control values are forbidden. However, the shape of the expected utility function is similar.

Remark 4. Note the scale of axis z in Figures 4a and 4b. Recall that, in every point, we are using a weighted mixture of normalized utility functions with values from interval $\langle 0, 1\rangle$. By maximization, we usually select combinations with values close to 1 and that is why the values of expected utility corresponds well to the number of the current algorithm iteration.

Our future goal is to move from discrete variables to the continuous ones. Therefore, it is interesting to see the shape of the expected utility function with respect to the control value and for a given speed. Let us take the utility function from Figure 4a and select five speed values. Respective slices are depicted in Figure 5. The gray solid lines show values from Figure 4a, the black lines show piecewise-linear approximations of each line. All approximations are composed from three lines. To find the best approximation of each curve, we used R package *segmented* (Muggeo, 2008). The package estimates linear and generalized linear models with one or more segmented relationships in the linear predictor. Estimates of the slopes and of the (possibly multiple) breakpoints are provided. In our experiments, we decided to fix the number of breakpoints to two and let the algorithm find their best positions.

In an influence diagram with continuous variables we would need to represent the optimal control policy at each step i by a function $u_i : \mathcal{V} \to \mathcal{U}$ (see Definition 2). The optimal control policy at point $i = 87$ of the path is depicted in Figure 6. We can see that piecewise linear functions may again represent good approximations.



(a) $i = 87$, ($1740m$, 15th iteration) (b) $i = 39$, ($780m$, 62nd iteration)

Figure 4: Expected utility as a function of values of variables V_i, U_i.

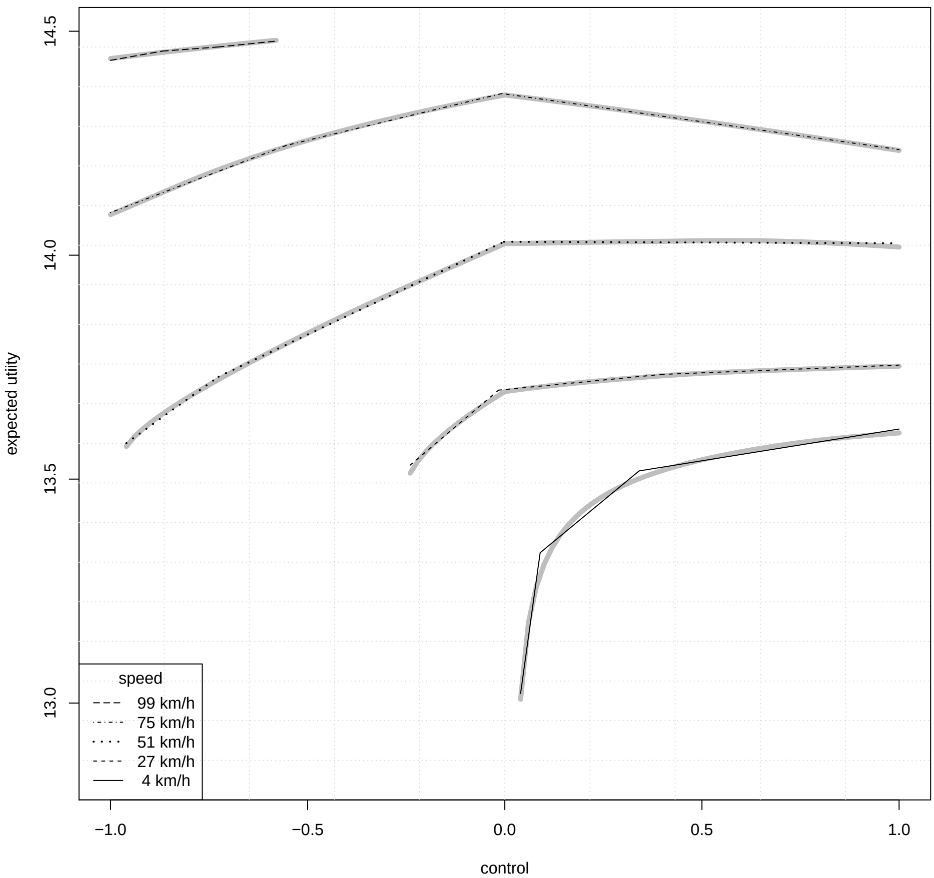


Figure 5: Expected utility as a function of the control values for several initial speed values, $i = 87$

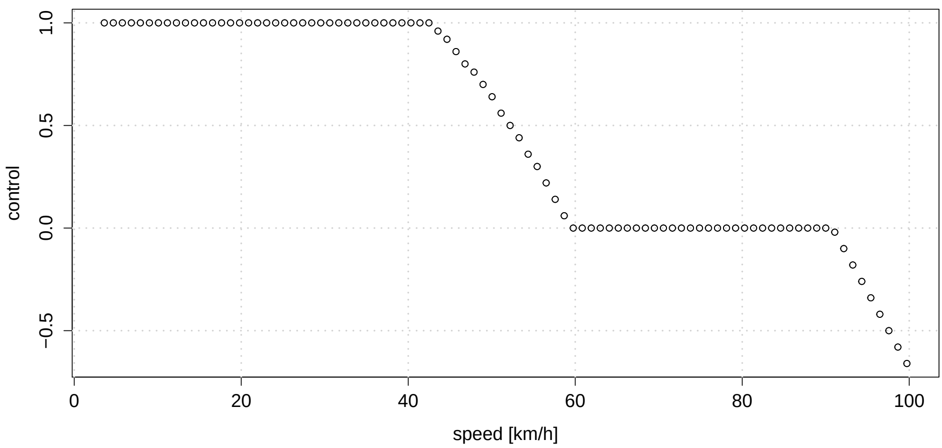


Figure 6: Optimal control policy, $i = 87$

7 Conclusions

We applied influence diagrams to optimization of a vehicle speed profile and performed numerical experiments on a 2-km-long path with few speed constraints. We considered optimality criteria based on a mixture of the fuel consumption and the total driving time. We derived the general inference algorithm for this type of influence diagrams and presented efficient modifications of this algorithm for specific cases. Finally, we used the numerical experiments to elicit the shape of expected utility and policy functions. In both cases piecewise linear functions seem to be good approximations that can be used in influence diagrams with continuous variables.

Acknowledgment

This work was supported by the Czech Science Foundation through Project 13-20012S.

References

Bellman, R. (1957). *Dynamic Programming.* Princeton University Press.

Chang, D. J. and Morlok, E. K. (2005). Vehicle speed profiles to minimize work and fuel consumption. *Journal of Transportation Engineering*, 131(3):173–182.

Howard, R. A. and Matheson, J. E. (1981). Influence diagrams. In Howard, R. A. and Matheson, J. E., editors, *Readings on The Principles and Applications of Decision Analysis*, volume II, pages 721–762. Strategic Decisions Group.

Jensen, F., Jensen, F. V., and Dittmer, S. L. (1994). From influence diagrams to junction trees. In *Proceedings of the Tenth Conference on Uncertainty in Artificial Intelligence*, pages 367–373. Morgan Kaufmann.

Kratochvíl, V. and Vomlel, J. (2015). Influence diagrams for the optimization of a vehicle speed profile. In *Proceedings of the Twelfth Annual Bayesian Modeling Applications Workshop.* Accepted for publication.

Muggeo, V. M. R. (2008). segmented: an R package to fit regression models with broken-line relationships. *R News*, 8(1):20–25.

Shachter, R. D. and Peot, M. A. (1992). Decision making using probabilistic inference methods. In *Proceedings of the Eighth Conference Annual Conference on Uncertainty in Artificial Intelligence (UAI-92)*, pages 276–283, San Mateo, CA. Morgan Kaufmann.

Shenoy, P. P. (1992). Valuation based systems for Bayesian decision analysis. *Operations Research*, 40:463–484.

A Proof of Theorem 1

Proof. For any $j = 1, \ldots, n$ we will denote the joint probability distribution as

$$P(U_1, \ldots, U_j, V_1, \ldots, V_j) = P(V_1) \cdot \prod_{i=2}^{j} P(V_i | U_{i-1}, V_{i-1}) \cdot \delta(U_{i-1} | V_{i-1})$$

and the total utility as

$$f(V_1, \ldots, V_j) = \sum_{i=1}^{j-1} f(V_i, V_{i+1}) \ .$$

For the maximal expected value it holds that

$$\begin{aligned} E_f^* &= \max_{U_1,\ldots,U_{n-1}} \mathcal{M}_{V_1,\ldots,V_n} \Big(P(U_1, \ldots, U_{n-1}, V_1, \ldots, V_n) \cdot f(V_1, \ldots, V_n) \Big) \\ &= \max_{U_1,\ldots,U_{n-1}} \mathcal{M}_{V_1,\ldots,V_n} \left(\begin{array}{l} P(U_1, \ldots, U_{n-1}, V_1, \ldots, V_n) \\ \cdot \Big(f(V_1, \ldots, V_n) + \psi(V_n) \Big) \end{array} \right) \qquad (23) \\ &= \max_{U_1,\ldots,U_{n-1}} \mathcal{M}_{V_1,\ldots,V_{n-1}} \left(\begin{array}{l} P(U_1, \ldots, U_{n-1}, V_1, \ldots, V_{n-1}) \\ \cdot \sum_{V_n} P(V_n | V_{n-1}, U_{n-1}) \cdot \left(\begin{array}{l} f(V_1, \ldots, V_{n-1}) \\ + f(V_{n-1}, V_n) \\ + \psi(V_n) \end{array} \right) \end{array} \right) . \end{aligned}$$

We can write

$$E_f^* = \max_{U_1,\ldots,U_{n-1}} \mathop{\mathcal{M}}_{V_1,\ldots,V_{n-1}} \left(\begin{array}{l} P(U_1,\ldots,U_{n-1},V_1,\ldots,V_{n-1}) \\ \cdot\Big(\xi(V_1,\ldots,V_{n-1}) + \psi(U_{n-1},V_{n-1})\Big) \end{array} \right) ,$$

where

$$\xi(V_1,\ldots,V_{n-1}) = \mathop{\mathcal{M}}_{V_n}\Big(P(V_n|V_{n-1},U_{n-1}) \cdot f(V_1,\ldots,V_{n-1})\Big) \quad (24)$$

$$\psi(U_{n-1},V_{n-1}) = \mathop{\mathcal{M}}_{V_n} P(V_n|V_{n-1},U_{n-1}) \cdot \Big(f(V_{n-1},V_n) + \psi(V_n)\Big) . \quad (25)$$

Equation (24) can be simplified to

$$\begin{aligned} \xi(V_1,\ldots,V_{n-1}) &= \Big(\mathop{\mathcal{M}}_{V_n} P(V_n|V_{n-1},U_{n-1})\Big) \cdot f(V_1,\ldots,V_{n-1}) && (26) \\ &= f(V_1,\ldots,V_{n-1}) , && (27) \end{aligned}$$

where the second transformation is due to $\mathop{\mathcal{M}}_{V_n} P(V_n|V_{n-1},U_{n-1}) = 1$. This implies

$$E_f^* = \max_{U_1,\ldots,U_{n-1}} \mathop{\mathcal{M}}_{V_1,\ldots,V_{n-1}} \left(\begin{array}{l} P(U_1,\ldots,U_{n-1},V_1,\ldots,V_{n-1}) \\ \cdot\Big(f(V_1,\ldots,V_{n-1}) + \psi(U_{n-1},V_{n-1})\Big) \end{array} \right) .$$

As the next step, we will for each $v_{n-1} \in \mathcal{V}$ find a value u_{n-1} of decision variable U_{n-1} that maximizes E_f over the terms containing U_{n-1}. Note that the value of U_{n-1} cannot influence the past since when deciding on U_{n-1} the value of V_{n-1} is already known. It means that the values of V_{n-1} effectively separate the influence diagram into two parts and maximization over U_{n-1} can be performed only in the part containing U_{n-1}:

$$E_f^* = \max_{U_1,\ldots,U_{n-2}} \mathop{\mathcal{M}}_{V_1,\ldots,V_{n-1}} \left(\begin{array}{l} P(U_1,\ldots,U_{n-2},V_1,\ldots,V_{n-1}) \\ \cdot \max\limits_{U_{n-1}} \delta(U_{n-1}|V_{n-1}) \cdot \left(\begin{array}{l} f(V_1,\ldots,V_{n-1}) \\ +\psi(U_{n-1},V_{n-1}) \end{array} \right) \end{array} \right) .$$

Since $f(V_1,\ldots,V_{n-1})$ does not depend on U_{n-1} we get

$$E_f^* = \max_{U_1,\ldots,U_{n-2}} \mathop{\mathcal{M}}_{V_1,\ldots,V_{n-1}} \left(\begin{array}{l} P(U_1,\ldots,U_{n-2},V_1,\ldots,V_{n-1}) \\ \cdot\,(f(V_1,\ldots,V_{n-1}) + \psi(V_{n-1})) \end{array} \right) . \quad (28)$$

where

$$\psi(V_{n-1}) = \max_{U_{n-1}} \psi(U_{n-1},V_{n-1}) .$$

From formula (23) we can get formula (28) by substituting $n-1$ for n. Therefore we can repeat the transformations again and again until $n = 2$. In case $n = 2$ formula (28) reduces to

$$E_f^* = \mathop{\mathcal{M}}_{V_1} P(V_1) \cdot \psi(V_1) ,$$

which is formula (17) of the theorem we want to prove. □