# Solving Trajectory Optimization Problems by Influence Diagrams

Jiří Vomlel[1,2](✉) and Václav Kratochvíl[1,2]

[1] Institute of Information Theory and Automation, Czech Academy of Sciences,
Pod Vodárenskou Věží 4, 182 08 Prague 8, Czechia
vomlel@utia.cas.cz

[2] Faculty of Management, University of Economics,
Prague Jarošovská 1117/II, 377 01 Jindřichův Hradec, Czechia

**Abstract.** Influence diagrams are decision-theoretic extensions of Bayesian networks. In this paper we show how influence diagrams can be used to solve trajectory optimization problems. These problems are traditionally solved by methods of optimal control theory but influence diagrams offer an alternative that brings benefits over the traditional approaches. We describe how a trajectory optimization problem can be represented as an influence diagram. We illustrate our approach on two well-known trajectory optimization problems – the Brachistochrone Problem and the Goddard Problem. We present results of numerical experiments on these two problems, compare influence diagrams with optimal control methods, and discuss the benefits of influence diagrams.

**Keywords:** Influence diagrams · Probabilistic graphical models · Optimal control theory · Brachistochrone problem · Goddard problem

## 1 Introduction

Influence diagrams (IDs) were originally proposed by Howard and Matheson (1981). They extend Bayesian network models (Pearl 1988) by utility and decision nodes. They can be used to solve optimal decision problems. For a detailed introduction to influence diagrams, see, for example, Jensen (2001). In the regular influence diagrams proposed in 80's it was required that (a) a total ordering of decision nodes that specifies the order in which the decisions are made must be specified and (b) the total utility is the sum of utility values of all utility nodes in the influence diagram. In this paper we will see that both requirements are naturally satisfied for many trajectory optimization problems.

IDs have been applied to diverse decision problems. Kratochvíl and Vomlel (2016) applied IDs to the speed profile optimization problem. The experiments performed on a real problem – the speed control of a Formula 1 race car – revealed that IDs can provide a good solution of the problem very quickly and that this

This work was supported by the Czech Science Foundation through projects 16-12010S (V. Kratochvíl) and 17-08182S (J. Vomlel).

A. Antonucci et al. (Eds.): ECSQARU 2017, LNAI 10369, pp. 146–155, 2017.
DOI: 10.1007/978-3-319-61581-3_14

solution can be used as an initial solution for the methods of the optimal control theory, which significantly improves the convergence of these methods.

In this paper we will generalize the approach presented in (Kratochvíl and Vomlel 2016). In Sect. 2 we will describe how IDs can be used to solve trajectory optimization problems. We consider not only problems where the goal is to find the trajectory but also problems where we want to optimize certain criteria (e.g., the total time, the fuel consumption) for a given trajectory. We use the suggested approach to solve two well-known optimal control problems: the Brachistochrone problem in Sect. 3 and the Goddard problem in Sect. 4. We conclude the paper by a discussion in Sect. 5.

## 2 Influence Diagrams for Trajectory Optimization

In this section we will describe how one can use IDs to solve a trajectory optimization problem. Next we describe general guidelines for the construction of an ID for the trajectory optimization. We will illustrate this construction on two problems in the next sections.

1. Specify the state variables, the control variables, and the utility function.
2. Describe the system dynamics using a system of ordinary differential equations (ODEs). Often, the system dynamics is described with respect to time. In the trajectory optimization it is often more convenient or even necessary to rewrite the ODEs with respect to the trajectory.
3. Discretize the trajectory to short segments.
4. Find the analytical formula for the state transitions as a function of previous values of the state and control variables for one segment. If the analytical solution is not available use approximations by an appropriate method – candidates are, for example, the Euler method, a Runge-Kutta method, or an implicit method as the Gauss-Legendre method.
5. If necessary, discretize the state and control variables.
6. If states are discretized and the state transitions lead to states that are not in the set of state values then use the stochastic approximation of the state transition by a mixture of two nearest states whose probability is proportional to their closeness to the computed state value, so that the expected value (conditioned on previous state and control values) is equal to the computed state value, see (Kratochvíl and Vomlel 2016, Sect. 5.2).
7. Construct the ID with state variables as chance nodes, the control variables as decision nodes, and a utility node for each segment.
8. Specify the state transitions using conditional probability tables (CPTs).
9. Find and store the optimal policy for each state and control configuration and for each segment of the trajectory by solving the ID.
10. During the application in a real control problem the optimal policy for the actual observed values of state variables at each point of the trajectory is used. In practice, the controlled object often deviates from the optimal solution. The reason can be a measuring and control imprecision, a bias, unexpected interventions, etc. Therefore, it is very useful to have the optimal policy stored for all configurations of the parents of all decision nodes.

The additive utility requirement means that the utility function decomposes additively along the segments of the trajectory. Such utility functions are common in practice. This condition is satisfied, for example, by total time or by the total fuel consumption. The requirement of the total ordering of the decisions is also natural for trajectory optimization problems. The decisions at coordinates closer to the origin are taken before those more distant ones. Another natural total ordering is the ordering by time elapsed from the beginning.

## 3 Brachistochrone Problem

Formulated by Johan Bernoulli in 1696, the Brachistochrone Problem is: given two points find a curve connecting them such that a mass point moving along the curve under the gravity reaches the second point in minimum time. We will consider this problem as an optimal control problem (Bertsekas 2000, Example 3.4.2). The state variable is the vertical coordinate $y$ and it is a function of the horizontal coordinate $x$. The variable $u$ controls the derivative of $y$:

$$\frac{dy(x)}{dx} = u(x). \tag{1}$$

It is assumed that the initial speed at the origin is zero. Speed $v$ is defined by the law of energy conservation – kinetic energy equals to the change of gravitational potential energy, which results in

$$v = \sqrt{-2 \cdot g \cdot y}. \tag{2}$$

For an infinitesimal segment of length $dx$ with an infinitesimal change $dy$ of the vertical position $y$ we can write for the speed, which is the derivative of the position $s$ with respect to time $t$:

$$v = \frac{ds}{dt} \quad = \quad \sqrt{\left(\frac{dy}{dt}^{2}\right) + \left(\frac{dx}{dt}\right)^{2}} \quad = \quad \left(\sqrt{1 + \left(\frac{dy}{dx}\right)^{2}}\right) \frac{dx}{dt}. \tag{3}$$

By substituting (1) and (2) to (3) we get

$$dt = \frac{ds}{v} \quad = \quad \left(\frac{1}{\sqrt{-2 \cdot g \cdot y}} \cdot \sqrt{1 + u^2}\right) dx \quad \overset{df}{=} \quad q\,(y, u)\, dx. \tag{4}$$

The solution of the Brachistochrone problem is a function $y = f(x)$ that minimizes the total time $T$ necessary to get from the point $(0, 0)$ to the point $(a, b)$, where $a > 0$ and $b < 0$.

### 3.1 Influence Diagram for the Brachistochrone Problem

Next, we will illustrate how an ID can be used to find an arbitrary precise solution of the problem. We will discretize the problem and use the following symbols:

the number of discrete intervals $n$, the distance discretization step $\Delta x = \frac{a}{n}$, the index of the interval $i = 0, 1, \ldots, n$, x-coordinate $x_i = i \cdot \Delta x$, y-coordinate $y_i$, the speed $v_i$, and time to get from $x_{i-1}$ to $x_i$ denoted as $t_i$. The control value $u_i$ defines the vertical shift: $y_{i+1} = y_i + u_i$. In each segment we will assume that the path is a line segment[1], i.e. for $x \in [x_i, x_{i+1}]$ and for $y \in [y_i, y_{i+1}]$ it holds

$$y = \frac{u_i}{\Delta x} \cdot x + y_i. \tag{5}$$

By substituting (5) to (4) and solving the definite integral

$$\int_{x_i}^{x_{i+1}} q\left(y, u\right) dx = \int_{x_i}^{x_{i+1}} q\left(\frac{u_i}{\Delta x} \cdot x + y_i, u_i\right) dx \tag{6}$$

we get the formula for the time spent at the segment $[x_i, x_{i+1}]$:

$$t_{i+1} = \begin{cases} \dfrac{\Delta x}{\sqrt{-2 \cdot g \cdot y_i}} & \text{if } u_i = 0 \\ -\sqrt{\dfrac{2}{g}} \cdot \left(\dfrac{(\Delta x)^2 + u_i^2}{u_i}\right) \cdot \left(\sqrt{-y_i} - \sqrt{-u_i - y_i}\right) & \text{otherwise.} \end{cases} \tag{7}$$

The goal is to find a control strategy $\boldsymbol{u} = (u_0, \ldots, u_{n-1})$, $u_i \in \mathbb{R}$, $i = 0, 1 \ldots, n-1$ so that we get from the initial point $(x_0, y_0)$ to the terminal point $(x_n, y_n)$ minimizing the total time$\sum_{i=1}^{n} t_i$ and satisfying the state constraints $y_i \leq y_0$ for $i = 1, \ldots, n$.

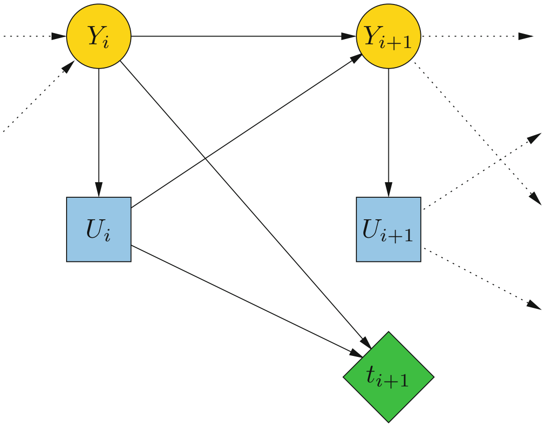

**Fig. 1.** One segment of the ID for the Brachistochrone Problem

[1] This is an approximation only, but the smaller distance discretization step the smaller the approximation error.

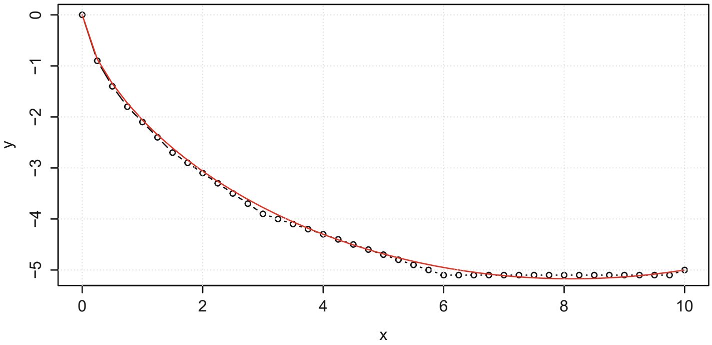

**Fig. 2.** Comparison of the optimal solution with the ID solution.

The structure of a segment of the ID for the discrete version of the Brachistochrone Problem is presented in Fig. 1. The state transition CPT is deterministic and defined as:

$$P(Y_{i+1} = y_{i+1}|U_i = u_i, Y_i = y_i) = \begin{cases} 1 & \text{if } y_{i+1} = y_i + u_i \\ 0 & \text{otherwise.} \end{cases} \tag{8}$$

The utility function for node $t_{i+1}$ is defined by formula (7).

### 3.2 Experimental Results

The solution of the Brachistochrone Problem is known – it is a part of a cycloid, which can be specified by two functions of a parameter $\varphi \in [0, M]$, $M \leq 2\pi$:

$$x = \frac{K}{2}(\varphi + \sin\varphi) + L, \;\; y = -\frac{K}{2}(1 - \cos\varphi). \tag{9}$$

$K, L, M$ are specified so that the cycloid goes through the points $(0, 0)$ and $(a, b)$.

In Fig. 2 we compare the optimal trajectory (full red line) with the solution found by the ID (circles connected by lines) for $\Delta x = 0.25$, $\Delta y = 0.1$ and $(a, b) = (10, -5)$. We can see that the solution found by the ID approximates well the optimal solution. The difference between the optimal trajectory and the ID solution can be reduced by decreasing the lengths of the discretization steps $\Delta x$ and $\Delta y$. More details about the experiments and the R code used for the experiments can be found in our research report (Vomlel 2017).

## 4 Goddard Problem

Formulated by Robert H. Goddard, see (Goddard 1919), the problem is to establish the optimal thrust profile for a rocket ascending vertically from the Earth's surface to achieve a given altitude with a given speed and pay load and with the minimum fuel expenditure. The aerodynamic drag and the gravitation vary with the altitude. We assume a bounded thrust. The problem has become a benchmark in the optimal control theory due to a characteristic singular arc behavior in connection with a relatively simple model structure.

In this paper we consider the normalized Goddard Problem. For the derivation of the normalized version and more details about the approximation methods see our research report (Vomlel and Kratochvíl 2017). We specify the Goddard Problem as an optimal control problem. The movement of the rocket is described by ordinary differential equations (ODEs). We describe the system dynamics with respect to the altitude $h$ measured as the distance from the Earth's center. The rocket's mass $m$ is composed from the pay load and the fuel, the latter is burnt during the rocket ascent. The speed is denoted by $v$.

The control variable $u$ controls the engine thrust, which is the derivative of the rocket's mass $m$ with respect to time $t$ multiplied by the jet speed $c$, i.e.,

$$u = c \cdot \frac{dm}{dt} \quad = \quad c \cdot \frac{dm}{dh} \cdot \frac{dh}{dt} \quad = \quad \frac{dm}{dh} \cdot c \cdot v. \tag{10}$$

The derivatives of mass $m$ and speed $v$ with respect to $h$ are defined using functions of $g$ and $f$ as it follows:

$$\frac{dm}{dh} = g(u,v) \overset{df}{=} \frac{u}{c \cdot v} \tag{11}$$

$$\frac{dv}{dh} = f(h,m,u,v) \overset{df}{=} -\frac{u}{m \cdot v} - \frac{v \cdot s \cdot c_D}{2 \cdot m} \cdot \rho_0 \cdot \exp\left(\beta \cdot (1-h)\right) - \frac{1}{v \cdot h^2}, \tag{12}$$

where $s$ is the cross-section area of the rocket, $c_D$ is the drag constant, $\rho_0$ is the density of the air at the Earth's surface, and $\beta$ is a dimensionless constant.

We will use model parameter values presented in (Tsiotras and Kelley 1991) and (Seywald and Cliff 1992). The aerodynamic data and the rocket's parameters originate from (Zlatskiy and Kiforenko 1983) and correspond roughly to the Soviet SA-2 surface-to-air missile, NATO code-named Guideline. The control will be restricted to $u \in [-3.5, 0]$. It is assumed that the rocket is initially at rest at the surface of the Earth and that its fuel mass is 40% of the rocket total mass. The nondimensionalized values of these constants and the initial and terminal values are:

$$\beta \; = \; 500, \; s \cdot \rho_0 \; = \; 12400, \; c_D \; = \; 0.05, \; c \; = \; 0.5,$$
$$h_0 \; = \; 1, \; h_T \; = \; 1.01, \; m_0 \; = \; 1, \; m_T \; \geq \; 0.6 \cdot m_0 \; = \; 0.6, \; v_0 \; = \; 0.$$

### 4.1 The Influence Diagram for the Goddard Problem

In each segment $i$ of the ID there are (a) two state variables – a speed variable $V_i$ and a mass variable $M_i$, (b) one decision variable $U_i$ controlling the thrust of the rocket engine, (c) one utility node $f_i$ representing the fuel consumption in the segment. The structure of one segment of the ID for the discrete version of the Goddard Problem is presented in Fig. 3.

We discretize the trajectory to segments of length $\Delta h$ with a constant control. In each segment a solution of the system of two ODEs (11) and (12) is found by an

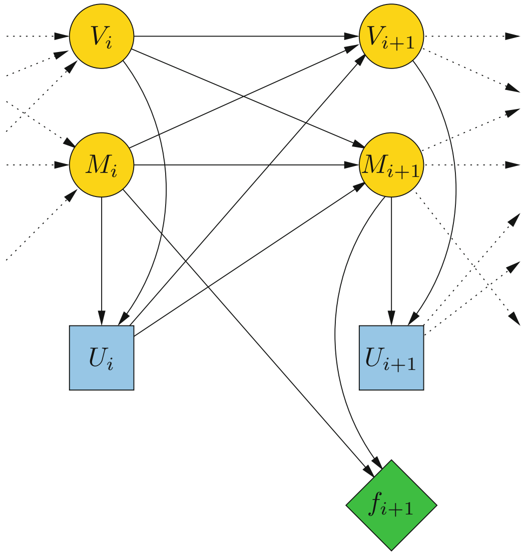

**Fig. 3.** One segment of the ID for the Goddard Problem

ODE approximation method. The solution provides values of the mass $m(h+\Delta h)$ and speed $v(h+\Delta h)$ at the end of the segment. ODE approximation methods can be used, e.g. the Euler, Runge–Kutta, and Gauss–Legendre methods. See (Vomlel and Kratochvíl 2017) for a derivation of these methods for the Goddard Problem. The computed mass and speed values will not lay in the discrete set of values of these variables. Therefore we will approximate the state transformations by non-deterministic CPTs $P(V_{i+1}|U_i, V_i, M_i)$ and $P(M_{i+1}|U_i, V_i, M_i)$ as it is described in (Kratochvíl and Vomlel 2016, Sect. 5.2).

### 4.2 Experimental Results

In Fig. 4 we compare the control, speed, and mass profiles of the optimal solution found by Bocop[2] (Team Commands, Inria Saclay 2016) with solutions found by IDs with different discretizations and different approximation methods. It is known (Miele 1963) that the optimal solution consists of three sub-arcs: (a) a maximum-thrust sub-arc, (b) a variable-thrust sub-arc, and (c) a coasting sub-arc, i.e., a sub-arc with the zero thrust.

For the solutions found by IDs we use the following name schema `v.u.m.M.h` composed from the parameters used in the experiments:

- `v` ... the number of states of the speed variables,

[2] Bocop package implements a local optimization method. The optimal control problem is approximated by a nonlinear programming (NLP) problem using a time discretization. The NLP problem is solved by Ipopt, using sparse exact derivatives computed by ADOL-C.

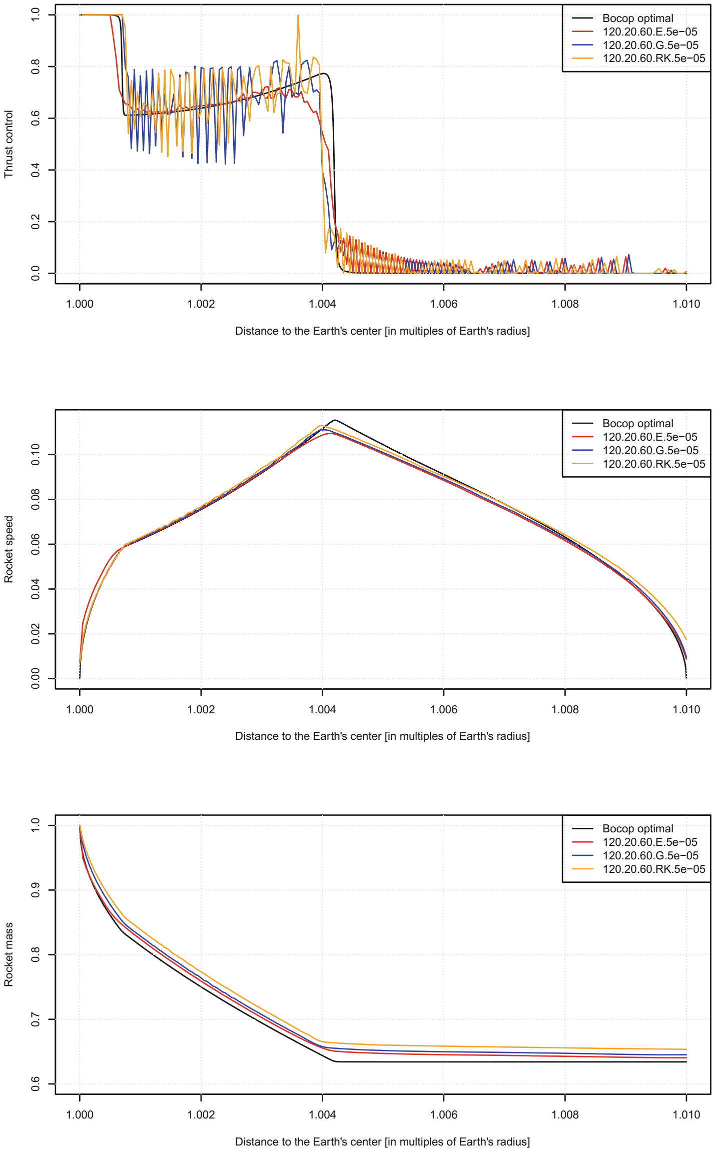

**Fig. 4.** Comparisons of the optimal solution with ID solutions.

– `u` ... the number of states of the control variables,
– `m` ... the number of states of the mass variables,
– `M` ... the discretization method for solving ODEs (`E` is the Euler method, `G` the Gauss–Legendre method, and `RK` the Runge–Kutta RK4 method), and
– `h` ... the length of the trajectory segment.

By looking at Fig. 4 we can conclude that the Euler method best approximates the optimal control and it suffers from smaller oscillations of the control than other methods. The control strategy found by Runge-Kutta and the Gauss-Legendre methods have larger oscillations. However, the speed and the mass profiles are similar for all methods and they are close to the optimal profiles found by BOCOP[3].

The quality of the solution is influenced by the number of states of speed, control, and mass variables. We had to find a proper balance between these parameters to avoid large oscillations. This issue deserves a further study to allow the application of IDs to problems where no optimal solution is known.

## 5 Conclusions and Future Work

We have described how IDs can be used to solve trajectory optimization problems. We applied the suggested approach to two trajectory optimization problems. The ID solution methods were tailored for these problems and can be considered a special case of dynamic programming (Bellman 1957). The numerical experiments reveal that the solutions found by IDs approximates well the optimal solution and the quality of the approximation improves with finer discretizations. It is important that IDs work well also in problems where the optimal strategy is more complex than a simple bang-bang strategy[4].

The trajectory optimization problems are traditionally solved by methods of the optimal control theory but IDs offer an alternative that can bring several benefits over the traditional approaches. IDs can incorporate uncertainty about the state transitions into the model. For each decision node in an ID the optimal decision is computed for all configurations of its parents. This is very handy in the situations where the controlled objects deviates for some reason from the optimal trajectory. The new optimal trajectory is thus available without any delay. In noisy environments or in environments with interactions with other objects the imposed deviations from the optimal trajectory can be quite common.

For many control problems it would be natural to use continuous IDs. Unfortunately, exact ID solution methods are available only for special cases that cannot be used in the problems studied in this paper. We leave a deeper study of applications of continuous IDs to trajectory optimization for future research.

[3] The initial mass of the rocket is the same for all methods but in the third plot of Fig. 4 we can see that the terminal mass slightly differs. The lowest fuel consumption is observed in case of RK method but we should conclude from that the RK method is optimal but rather that it has the largest approximation error.

[4] A bang-bang strategy is a strategy that consists of extreme values only, e.g. it consists of the full thrust and the zero thrust phases only. Bang-bang strategies are optimal solutions of a wide class of optimal control problems.

## References

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

Bertsekas, D.P.: Dynamic Programming and Optimal Control, 2nd edn. Athena Scientific, Belmont (2000)

Goddard, R.H.: A method for reaching extreme altitudes, volume 71(2). Smithsonian Miscellaneous Collections (1919)

Howard, R.A., Matheson, J.E.: Influence diagrams. In: Howard, R.A., Matheson, J.E. (eds.) Readings on the Principles and Applications of Decision Analysis, vol. II, pp. 721–762. Strategic Decisions Group (1981)

Jensen, F.: Bayesian Networks and Decision Graphs. Springer, New York (2001)

Kratochvíl, V., Vomlel, J.: Influence diagrams for speed profile optimization. Int. J. Approximate Reasoning. (2016, in press). http://dx.doi.org/10.1016/j.ijar.2016.11.018

Miele, A.: A survey of the problem of optimizing flight paths of aircraft and missiles. In: Bellman, R. (ed.) Mathematical Optimization Techniques, pp. 3–32. University of California Press (1963)

Pearl, J.: Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference. Morgan Kaufmann series in representation and reasoning. Morgan Kaufmann, Burlington (1988)

Seywald, H., Cliff, E.M.: Goddard problem in presence of a dynamic pressure limit. J. Guidance Control Dyn. **16**(4), 776–781 (1992)

Team Commands, Inria Saclay: BOCOP: an open source toolbox for optimal control (2016). http://bocop.org

Tsiotras, P., Kelley, H.J.: Drag-law effects in the goddard problem. Automatica **27**(3), 481–490 (1991)

Vomlel, J.: Solving the Brachistochrone Problem by an influence diagram. Technical report 1702.02032 (2017). http://arxiv.org/abs/1702.02032

Vomlel, J., Kratochvíl, V.: Solving the Goddard Problem by an influence diagram. Technical report 1703.06321 (2017). http://arxiv.org/abs/1703.06321

Zlatskiy, V.T., Kiforenko, B.N.: Computation of optimal trajectories with singular-control sections. Vychislitel'naia i Prikladnaia Matematika **49**, 101–108 (1983)