

Project information

Optimization of realistic complex systems

Project 2: Optimization of electric motor usage in a PHEV with dynamic programming

1 Information

The task in project 2 is to optimize the usage of the electric motor in a PHEV (Plug-in Hybrid Electric vehicle). The method to be used is dynamical programming. You will write your own code in Matlab (Octave) or Python for making the computations.

2 Preparations

1. Study description of the PHEV.
2. Study information about Matlab/Octave or Python.
3. Formulate the problem so that it can be solved by dynamical programming. Specify all necessary definitions (states, control, transfer function etc.).
4. Implement the method in Matlab/Octave or Python.
5. Solve the problem.
6. Demonstrate your results by writing a report which can be read both by a person who is not familiar with optimization and a person who knows optimization and interested in how you solved the problem. Include the definitions from 3 and a description of your method and its implementation.
7. Optionally, prepare a presentation of the result/report.

3 Problem Description

A general description of a PHEV is given separately.

In this project we consider a simplified version of the problem. We have four settings for the engine:

1. Only electricity.
2. Mostly electricity, some gasoline.

3. Mostly gasoline, some electricity.
4. Only gasoline.

The roads are divided into five categories:

1. Motorway.
2. Country road.
3. Smaller road, hilly and/or curvy.
4. Road with less traffic (in small towns or suburbs).
5. Inner city road (traffic lights, many starts and stops).

We denote engine setting with index k , category of road with t and road section with j . We assume that the path to take is already decided. The whole path is divided into n pieces of road sections of various categories and lengths. Road section j has length l_j (m) and is of type t_j .

Based on the expected average speed it is estimated how much the battery level drops if one travels one meter on the road of category t with engine setting k , and this is denoted by b_{kt} . Moreover, a cost is estimated (including everything, i.e. gasoline, wear, environmental impact, etc.) of traveling one meter on a road of category t with engine setting k , and it is denoted by d_{kt} .

Since road section j has length l_j , the battery level drops by $a_{jk} = l_j b_{kt_j}$ if one drives with setting k , and the cost is calculated as $c_{jk} = l_j d_{kt_j}$.

We use variables $x_{jk} = 1$ if we drive on road section j with setting k , and 0 otherwise. Since one of the settings should be used for every road section, we get the following constraints:

$$\sum_k x_{jk} = 1 \text{ for all } j$$

If the battery is charged to level L_0 from the start, we can not let the battery level drop more than that, therefore we get the following constraints:

$$\sum_j a_{jk} x_{jk} \leq L_0$$

We wish to minimize the cost for the whole trip, which gives the following objective function:

$$\min \sum_j \sum_k c_{jk} x_{jk}$$

We also need

$$x_{jk} \geq 0 \text{ for all } j \text{ and } k$$

This model may not seem so complicated, but one should consider the size. For the optimization to give a good result, it is necessary to have a very fine division into road sections, i.e. many road sections are required.

3.1 Input data

Data to this problem consists of the coefficients b_{kt} and d_{kt} . Matrices b and d are rather small, the number of settings times the number of road categories in our simplified model is 4x5. These data is stored in the files *linkbatt.txt* and *linkcost.txt*. Furthermore, we need length l_j and category t_j for every road section, which could be many. These data are stored in the files *path0.txt*, *path1.txt* etc. These files also contain starting charge level of battery, L_0 .

One needs to read the data and compute coefficients a and c . (This is done by the available code, see below.)

3.2 Instances

Data is constructed from several various instances, i.e. paths.

Instance	Number of road sections	L_0	Type
path1	3	10	Small test
path2	3	4000	Medium test
path3	5	4000	Larger test
path4	20	1000	From a random map
path6	74	200000	Linköping - Jönköping
path7	74	200000	Linköping - Jönköping
path8	94	100000	Östergötland, east to west
path9	246	100000	Östergötland, north to south
path12	200	70000	Random
path13	300	100000	Random
path14	500	100000	Random
path15	700	200000	Random
path16	1000	300000	Random

Notice that the number of sections determines how many steps to be done in dynamical programming, while L_0 determines how large the table is at every iteration. The difference between the two first problems is only the size of L_0 .

3.3 Code

There is a suggested start and end of the matlab/python code in the file *initeldynp.m/initeldynppy.py*. You need to write the name of the data file in line 2, and a scaling factor for discretization in line 3. If it is set to 1, then every integer value of the battery charge level is considered, and if it set, e.g. to 10, then every 10th value. It is better to set the scaling factor to 1, but it is faster to solve the problem with large L_0 (approximately) for larger values of the scaling factor.

Then the input data files are read. You should not change this part. Sections of length zero are eliminated. If the battery charge level is sufficient to drive the whole path using

only electricity (i.e. if L_0 is too large for the optimization problem to be of interest), this is considered to be the solution (and the problem is considered to be solved).

Then the solution is computed for a “stupid driver” that drives only on electricity (setting 1) until the battery is finished, and then only on gasoline (setting 4). Compare later the cost of this solution to the cost of your best solution.

Then write your optimization algorithm. The solution should be represented as the vector `xopt` that is set to zero initially, but should contain the engine setting for every section. Furthermore, `z` contains the total cost and `s` contains the battery charge level in the end.

Several tips (matlab commands which could be good to use):

```
ix=ix-a;
```

Decreases the index vector `ix` by `a` units.

```
nix=1-ix>0;
```

`nix` is 0 in those positions where `ix` is larger than zero and 1 in those positions where `ix` is less or equal to zero.

```
ft0=circshift(ft1,[0 a]);
```

The vector `ft0` is the vector `nft1` circularly shifted by `a` positions to the right (for row-vectors).

```
ft3(nix)=inf;
```

The elements of the vector `ft3` with indices such that `nix=1` are set to a large value. (Therefore they will never give a minimum.)

Afterwards the code sums up how many sections were driven for each setting, and also how long path that was driven for each setting. These parts should not be changed.

4 Assignments

1. Solve the small test problems *path1.txt* and *path2.txt* to check that everything works correctly. The optimal cost for *path1.txt* is 97 with the solution 3-4-4. The optimal cost for *path2.txt* is 2144 with the solution 1-1-3.
2. Solve the larger problems. Note objective function value and solution, as well as the number of sections and length of path driven for each setting. Compare with the cost for a dumb driver. State also the scaling factor (if is not equal to one).
3. Solve *path7.txt* (Linköping - Jönköping) with $L_0 = 250000, 200000$ (already done), 150000, 100000, 50000 and 0, and study how the cost is increasing when the initial battery charge level is decreasing. Note also how the time for solving the problem changes.
4. In our examples the file *linkbatt.txt* does not contain negative coefficients, i.e., we do not consider the possibility for the battery charge level to increase. However, in practice the pure gasoline setting may charge the battery under certain conditions. How would that change the model and the method?
5. Sometimes changes in the conditions (weather, traffic) lead to unexpected drops of the battery charge level. If this happens during the journey, one would have to solve the optimization problem in the car. Discuss possibilities for using your software in such a case.