18-osios jaunųjų mokslininkų konferencijos „Mokslas – Lietuvos ateitis" teminės konferencijos
TRANSPORTO INŽINERIJA IR VADYBA,
vykusios 2015 m. gegužės 6 d. Vilniuje, straipsnių rinkinys

Proceedings of the 18th Conference for Junior Researchers 'Science – Future of Lithuania'
TRANSPORT ENGINEERING AND MANAGEMENT, 6 May 2015, Vilnius, Lithuania

Сборник статей 18-й конференции молодых ученых «Наука – будущее Литвы»
ИНЖЕНЕРИЯ ТРАНСПОРТА И ОРГАНИЗАЦИЯ ПЕРЕВОЗОК, 6 мая 2015 г., Вильнюс, Литва

OPTIMISATION OF TRAIN CONTROL USING CONSTRAINT PROGRAMMING ALHORITHM

Inmars Didaitis

Department of Railway Transport, Vilnius Gediminas Technical University
E-mail: inmars.didaitis@gmail.com

Abstract. In this paper, a developed optimisation approach for train control optimisation problems is discussed. A mathematical model of train is developed and described. The model is used in optimisation algorithm, which is based on constraint programming paradigm. Sub-optimal control strategies in discrete time are found by algorithm and a set of sub-optimal solutions depending on time available for operation is returned. Results in such form can be later used for optimisation of schedule and a time limit allocation for each individual operation. An example of optimisation results is shown. It is concluded that this kind of algorithm can be applied practically, however it has limitations. The best application could be optimisation of simple operations e.g. shunting operations in station.

Keywords: optimal train control, control optimisation, optimisation algorithm, energy minimisation, energy efficiency, mathematical model of train, numeric integration of train's motion.

Introduction

In this paper, a different approach to optimisation of train control is presented.

An usual approach to optimisation of train control is using Bellman's principle (Jastremskas 2012) or Hamilton's method (Howlett 2009). Presented algorithm however is not using any of before-mentioned methods. It is based on the constraint programming paradigm. It finds a set of sub-optimal solutions for a simple (well-known) case of train operation based on limitations applied to the motion of a train.

If a general control strategy is known, search for the optimal control is a problem of finding optimal switching points between regimes of the control. Given time limitation $t_{\lim}$ there must exist a single optimal control of train for the minimal energy consumption. It must be noted, that this algorithm is used in discretised time, therefore solutions are not optimal. They are however sufficiently close to optimal solutions, and will be referred as sub-optimal solutions.

Model used for this calculation is not using typical simplifications, since numeric the integration method and the optimisation algorithm can deal with any arbitrary input. A train is not treated as a mass point; reduction of track profile is excluded, etc.

This relatively simple algorithm while effective for finding solutions to simple problems, requires high computational effort, therefore its application is limited. Best application is simple problems, like shunting operations, or an analysis of track elements for local energy consumption optimisation in a global control optimisation strategy.

Previous studies

An optimal train control is researched by number of researchers. Many different approaches to this problem have been developed and used. It has been extensively used in the railway transport, and recently same approaches have been implemented in other kinds of vehicles, e.g. solar powered vehicles.

Most of recent research papers present optimisation algorithms that are based either on Bellman's principle, e.g. Jastremskas (2012), or Hamilton's method, e.g. Howlett *et al.* (2009).

Optimisation algorithms differ between optimisation of all journey of a train that solve control of a train from start of movement until stopping (Jastremskas 2012), or localised optimisation of control, that is implemented in a global control strategy (Howlett *et al.* 2009) latter is widely used for on board driver assistance equipment.

A typical approach in mathematical modelling of a train is to discretise time (Jastremskas 2012; Lukaszewicz 2001). However some analytical optimisation methods for simple cases, of localised control are developed in a continuous time (Howlett *et al.* 2009). Some models use position x as independent variable of differential equations that describe motion of a train. It is typical to discretise position as well.

Most of models presented in research papers and described in other literature are simplified in order to reduce calculation complexity. However if calculation is performed using a computer, most of simplifications are not needed. They reduce accuracy while computation time is sometimes even increased.

Mathematical model of train

Equation of motion can be formulated in a following form:

$$\frac{dv}{dt}=\frac{F(v,u)-W(v,x,u)-B(v,u)}{M}, \tag{1}$$

where: $\frac{dv}{dt}$ – acceleration of train, F – force of traction, depending on velocity v and control u. W – resistance to motion, depending on velocity v, position x and control u. B – force of braking, depending on velocity v and control u. M – mass of a train, including rotational mass of wheelsets.

Usually resistance depends only on velocity and position. It is taken into account however, that locomotive's rolling resistance increases without power applied to traction motors. Resistance caused by gravitational acceleration due to track gradient and resistance in curves of track are added to total rolling resistance W therefore; it also depends on a position of vehicle x.

Although it is generally accepted that a train can be modelled as a point mass, Jastremskas (2012) showed, that error caused by such assumption is significant. This simplification decreases accuracy more when the train is longer, so it is especially important not to use it while calculating motion or heavy haul freight trains. A force caused by gravitational acceleration due to track gradient in this model is calculated for each individual wagon of the train. An option to simplify calculation by treating train as a point mass is however possible and implemented in algorithm.

Time needed for a train to reach its destination can be calculated by:

$$t(X)\int_0^X \frac{dt}{dx}dx=\int_0^X \frac{1}{v}dx . \tag{2}$$

Euler's method of numeric integration is used to calculate motion of train. Change of position of train and its velocity is found by iteratively solving following equations in discrete time:

$$\begin{cases} v_i = v_{i-1} + a\cdot\Delta t, \\ x_i = x_{i-1} + v\cdot\Delta t + \dfrac{a\cdot\Delta t^2}{2}, \end{cases} \tag{3}$$

where: Δt – size of time step, i – number of time step, a – acceleration calculated using equation 1.

Train control executed by the driver is represented by variable u. It is common to include only control of traction into this variable with two levels of control $u(t)=\{0,1\}$ in the developed model control of braking was also defined in a same variable. Therefore control variable u has three levels $u(t)=\{-1,0,1\}$, where levels mean: braking, coasting, and traction respectively. Such approach simplifies implementation of algorithm.

Energy consumed for the traction is calculated by integrating force of traction over distance travelled.

$$\int_0^X F_{tr}dx . \tag{4}$$

Such approach doesn't take into account efficiency of traction rolling stock, and energy consumption during coasting of train if diesel locomotive is used. However if needed an algorithm can be easily modified to include these parameters in model.

Optimisation algorithm

The algorithm is based on the constraint programming paradigm and is derived from the brute force approach. With a known strategy of control, initial and end conditions and limitations to motion of train the algorithm is searching for an optimal switching points between control regimes. The algorithm finds a set of control strategies that are optimal for different time limits of train operation. Results can be further used for scheduling of operations or implementing result in global control strategy. An example of algorithm usage is used to explain how it works. Single movement of train, using *power-coast-braking* strategy is optimised. Typical characteristics found in literature (Postol and Kuzmichev, 2011) are used to model single locomotive and 20 freight wagons. Train has to move a distance $x = 1000\ldots1020$ m on a level track. It starts stationary, and has to stop at the end between defined limiting points of position.

First step – full power run

First step of algorithm is finding a switching point for *power-brake* control strategy. Algorithm finds the switching point that minimises the travel time: $t = t_{\min}$.

This control strategy is fully defined by two variables: total travel time t, time that is used for traction t_{tr}, rest of the time is used for braking t_{br}. Based on this control strategy a set of control strategies is found.

Second step – iterative search for control strategies

During a second step of algorithm switching points for *power-coast-break* control strategy are found. The time used for traction t_{tr} is iteratively decreased by time steps, and then the rest of journey is calculated, by trial and error attempts, to find the shortest time to reach destination. A coasting time t_{cst} is increased until end conditions are satisfied. Such approach can be classified between constraint programming and brute-force method but since a number of constraining limitations are applied to search algorithm, it is closer to constraint programming paradigm.

For each t_{tr} one control strategy is found and, as soon as algorithm finds control strategy that satisfies constraints, iteration is ended, then new one for different traction time t_{tr} is started. Such control strategy is fully defined by three variables: total travel time t, time that is used for traction t_{tr}, time that is used for coasting t_{cst}, rest of the time is used for braking t_{br}.

For each control strategy obtained by second step energy usage is calculated by equation 5. This algorithm gives set of control strategies with different travel times and energy consumption. The dependence is shown in Figure 1.

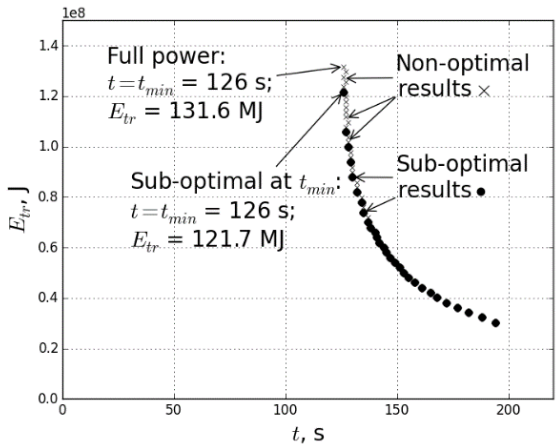

Fig. 1. Dependence between run time and energy consumption of train obtained by second step of optimisation algorithm

Discretisation of a train control during optimisation introduces some error in results. First of all such approach does not let us get optimal result, only sub-optimal that is sufficiently near optimal. Second, as clearly visible in figure 1, some of solutions (marked by ×) while feasible, are definitely not optimal for given run time t. It is worth noticing that the most of non-optimal strategies are obtained near shortest time runs, and that the full power run obtained in first step is non-optimal solution for this problem. The error caused by discretisation makes it necessary to post-process results, in order to eliminate non-optimal values form the set. This is performed by third step of algorithm.

Third step – post-processing of results obtained in second step

Algorithm compares control strategies that have same run time, and eliminates the ones that use more energy to perform operation in same time.

After removing values, which are known not to be optimal we get following dependence shown in Figure 2.

It is now visible that for each unique run time there is only one control strategy that uses lowest (sub-optimal) amount of energy. Graph follows generally known dependence between run time and energy consumption and gives sufficiently accurate values for this particular case.

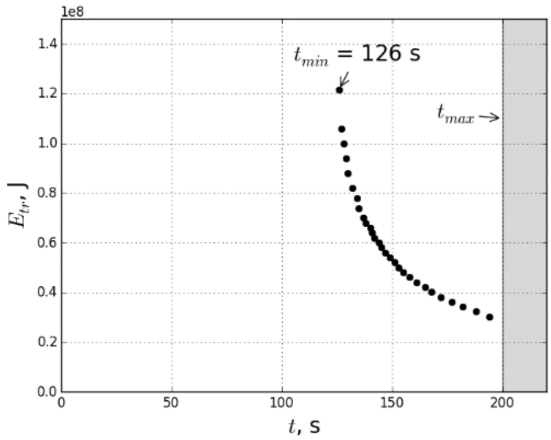

Fig. 2. Dependence between run time and energy consumption of train obtained by third step of optimisation algorithm

To further explain the obtained result, a set of sub-optimal control strategies is shown by plotting their velocity profiles. Results are shown in Figure 3.

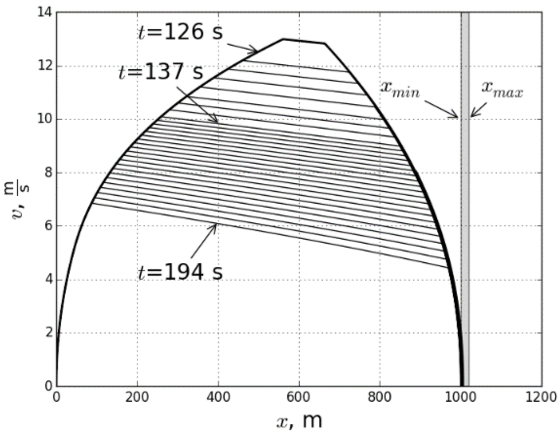

Fig. 3. Speed profiles of motion of a train for different sub-optimal control strategies

Here each line represents different control strategy. They differ in velocity, run time, and energy consumption. It is also visible here that *power-brake* (full power) strategy for this problem is no longer in set of sub-optimal control strategies as only the *power-coast-brake* strategies remained in the set of solutions.

The obtained results can later be implemented in optimisation of operation scheduling. When multiple operations have to be performed in limited time, an algorithm can be developed to assign a time limit and control strategy for each operation, in order to minimise total energy consumption of all operations. For example such approach can be used in planning of shunting operations in a station.

Conclusions and research continuity

1. Presented optimisation method can be practically applied, as demonstrated by an example.

2. An algorithm is relatively simple and can be easily modified to implement new functionalities or change its application.

3. A discrete time optimisation cannot guarantee optimality. While solutions are sufficiently close to optimal, they must not be referred as optimal, they are sub-optimal.

4. Presented algorithm is limited, general strategy of control has to be known in advance, complicated cases while can be solved, requite substantial computational effort, and can better be solved using different algorithms.

5. In order to expand usability of train model and optimisation algorithm code of programme has to be improved. Efficiency needs to be increased to expand its field of application.

6. Functionality of programme has to be expanded for better user experience and to allow its usage in practical applications.

As mentioned, efficiency of the code has to be improved this can be achieved by optimising the code of a programme.

Additional features need to be implemented, to expand its usability and field of application. Code has to be applicable for a more general case of optimisation.

To improve accuracy of model different method of numeric integration should be implemented e.g. second order Euler's method, or Runge-Kutta method.

Algorithm for operation schedule optimisation needs to be created, and interoperability of these two algorithms has to be guaranteed.

References

Howlett, P. G.; Pudney, P. J. 1995. *Energy-efficient train control*. In Advances in industrial control. London: Springer.

Howlett, P. G.; Pudney, P. J., Vu Xuan. 2009. Local energy minimization in optimal train control, *Automatica* 45: 2692–2698.

Jastremskas, V. 2012. *Lokomotyvų efektyvaus naudojimo tyrimas vertinant vežimų apimčių dinamiką.* Daktaro disertacija. Vilnius: Technika. (In Lithuanian).

Lukaszewicz, P. 2001. *Energy Consumption and Running Time for Trains*. Doctoral Thesis. Stockholm.

Postol, B. G.; Kuzmichev, E. N. 2011. *Tyaga poyezdov*. Khabarovsk: Izdatel'stvo DVGUPS. (in Russian).