Energy-Efficient Automatic Train Driving by Learning Driving Patterns

Jin Huang, Yue Gao*, Sha Lu, Xibin Zhao*, Yangdong Deng and Ming Gu
School of Software/KLISS/TNList
Tsinghua University, Beijing, 100084, China.
* indicates corresponding authors, emails: {gaoyue,zxb}@tsinghua.edu.cn

Abstract

Railway is regarded as the most sustainable means of modern transportation. With the fast-growing of fleet size and the railway mileage, the energy consumption of trains is becoming a serious concern globally. The nature of railway offers a unique opportunity to optimize the energy efficiency of locomotives by taking advantage of the undulating terrains along a route. The derivation of an energy-optimal train driving solution, however, proves to be a significant challenge due to the high dimension, nonlinearity, complex constraints, and time-varying characteristic of the problem. An optimized solution can only be attained by considering both the complex environmental conditions of a given route and the inherent characteristics of a locomotive. To tackle the problem, this paper employs a high-order correlation learning method for online generation of the energy optimized train driving solutions. Based on the driving data of experienced human drivers, a hypergraph model is used to learn the optimal embedding from the specified features for the decision of a driving operation. First, we design a feature set capturing the driving status. Next all the training data are formulated as a hypergraph and an inductive learning process is conducted to obtain the embedding matrix. The hypergraph model can be used for real-time generation of driving operation. We also proposed a reinforcement updating scheme, which offers the capability of sustainable enhancement on the hypergraph model in industrial applications. The learned model can be used to determine an optimized driving operation in real-time tested on the Hardware-in-Loop platform. Validation experiments proved that the energy consumption of the proposed solution is around 10% lower than that of average human drivers.

Introduction

Regarded as the most sustainable means of modern transportation, the railway system itself is facing the challenge of energy efficiency. In fact, the fast-growing train fleet size and railway mileage make the energy consumption of trains a global concern. According to the Railway Handbook 2016, the railway system was responsible for 2,200 PJ, or 0.6% of global energy consumption. Such an amount suggests that even a 1% energy-saving suffices to support the residential power of two major metropolitan cities like New York. The nature of railway offers a unique opportunity to optimize the energy efficiency of trains by taking advantage of the undulating terrains along a route. The Automatic Train Driving system, as shown in Figure 1, serves the purpose of driving the train with rather optimized and consistent solutions compare to human drivers. However, the derivation of an energy-optimal driving solution for a train is challenging because both the geographic environment of the route and the inherent conditions of a train have to be considered. The high dimension, nonlinearity, complex constraints, and time-varying characteristics make it hard to generate an optimized driving operation (Yang et al. 2016).

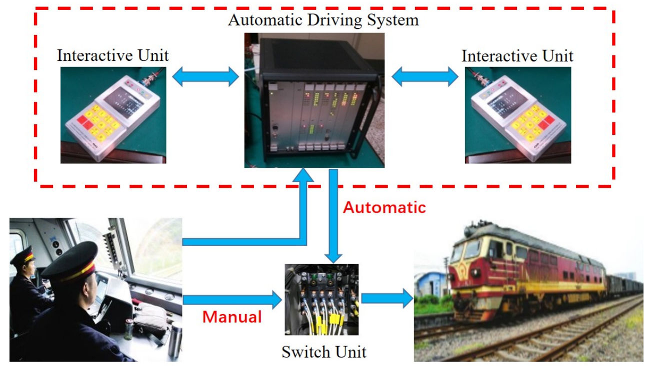

Figure 1: The illustration of the developed Automatic Train Driving system. It has two interactive units at both side of the locomotive. The Train driving mode can be switched between manually driving and automatic driving.

Significant research and engineering efforts have been dedicated to derive the optimized driving operation. As it was proved that a unique optimal solution can be derived under certain assumptions, many analytical solutions were proposed (Vu 2009; Albrecht et al. 2011; Howlett and Pudney 2012). However, the assumptions may not hold under realistic situations. In addition, complex driving constraints have to be respected in today's railway system. To handle the realistic conditions and complex constraints, advanced numerical searching techniques, like the Genetic Algorithm (Han et al. 1999; Li and Hou 2007), Evolutionary Algorithms (Chang and Kwan 2004), Dynamic Programming (Vašak et al. 2009; Miyatake and Ko 2010; Franke, Terwiesch, and Meyer 2000), and Nonlinear Programming (Betts 2010; Zhong, Xu, and Zhang 2016), *etc.* have been applied to compute the energy optimized driving operation. Such approaches offer superior solutions but tend to be compute intensive. Online techniques for real-time control were also widely studied(Siahvashi and Moaveni 2010; Gao et al. 2013), but these techniques suffer from a lack of guarantee of solution quality(Corman and Quaglietta 2015). More comprehensive surveys on energy-efficient train driving techniques can be referred to References (Lu et al. 2013; Huang et al. 2016). Although the problem is suitable for employing certain machine learning techniques, there is hardly any such works can be found.

The energy-efficient train driving problem can be viewed as a decision making problem to generate the driving operation according to the real-time correlation analysis with the environment. In this work, we leverage the correlation learning techniques, which are able to identify complex correlation in the studied objects(Michalski, Carbonell, and Mitchell 2013; Zhang et al. 2017; Liu et al. 2017a; Bansal, Blum, and Chawla 2004; Liu et al. 2017b), to resolve the problem. This paper proposes a high-order correlation learning based scheme for online generation of the energy-efficient train driving solutions. We carefully designed a feature set for such a learning purpose. Based on the records data from experienced human drivers, a hypergraph model was built to cluster the operations corresponding to the feature set for the decision of a driving operation. A reinforcement process was then employed to update the model to get better driving solutions by continuously mining the best-ranked driving records. The learned model is train-specific and can be used to determine an optimized driving operation in real-time. The proposed techniques are validated on a Hardware-in-Loop platform. Compared with average human drivers, our techniques enable an energy saving of around 10%. We believe that this work is the first to raise the train driving feature-set and is among the first to propose the machine learning techniques to derive solutions for automatic train driving problems. The proposed inductive hypergraph learning method also serves the novel contribution of this paper.

The Energy-Efficient Train Driving

Problem Formulation

The motion of a train can be formulated by treating the train position s as an independent variable as follows (Vu 2009):

$$\begin{aligned} m\rho\frac{\mathrm{d}v(s)}{\mathrm{d}s} &= f(s) - R_b(v(s)) - R_l(s), \\ \frac{\mathrm{d}t}{\mathrm{d}s} &= \frac{1}{v(s)}. \end{aligned} \tag{1}$$

where m is the mass of the train, ρ is a factor accounting for the rotating mass, v is the velocity of the train, s is the position (i.e., displacement) of the train, $f(s)$ is the traction or braking force along the position that is determined by the discrete or continuous level of gear on most modern locomotives, $R_b(v(s))$ is the basic resistance including both roll resistance and air resistance along s, and $R_l(s)$ is the corresponding line resistance caused by track grade, curves and tunnels. These resistance forces usually can be formulated as empirical equations with parameters determined with experiments (Mao 2008). The above equation is widely used in analytical and numerical optimization problems originated from train operations(Khmelnitsky 2000).

The optimization objective of energy-efficient train driving problem is to minimize the energy consumption as well as the time deviation from the time table under various operation constraints, which can be further formulated in the position depended form as:

$$\begin{aligned} J_E &= \int_{s_{start}}^{s_{end}} \phi(f(s)) \left(f(s) + \lambda \left| \frac{\mathrm{d}f(s)}{\mathrm{d}s} \right| \right) \mathrm{d}s, \\ J_T &= \left| T - \bar{T} \right|, \end{aligned} \tag{2}$$

subject to the following constraints and boundary conditions

$$\begin{aligned} & f_{min} \leq f(s) \leq f_{max}, \quad 0 \leq T(s) \leq T_{max}(s), \\ & v(s) \leq v_{limit}(s), \quad s(0) = s_{start}, \quad v(0) = v_{start}, \\ & s(T) = s_{end}, \quad v(T) = v_{end}. \end{aligned} \tag{3}$$

Here, J_E and J_T represent the optimization objective on energy consumption and the time deviation, respectively. $\phi(f)$ stands for the coefficient capturing the gear behavior. $\bar{T}$ is the scheduled time for a train trip and T is the actual time cost for the trip. The maximum allowable velocity $v_{limit}(s)$ is determined by the train model and the route condition. It is usually represented as a piecewise function of the train position s. s_{start} and s_{end} are the train position at the beginning and ending of a trip, while v_{start} and v_{end} be the initial and final speeds, respectively. v_{start} are the position and the velocity at the beginning of a trip, the duration of the trip $\bar{T}$ is usually given by the timetable. It must be noted that there is a potential trade-off between the energy consumption and the time deviation during the bi-objective optimization process. It is extremely hard to derive an optimized solution to the energy-efficient train driving problem due to its high dimension, inherent nonlinearity, complex constraints and potential variations of the elements in a sequence of driving operations(Howlett and Pudney 2012). On the other hand, human drivers are able to synthesize the multiple factors and derive driving solutions in a real-time fashion. As a result, we attack the problem by analyzing the human driving experience.

Analysis of Human Driving Records

Counterintuitively, a considerable portion of the railway routes are built on undulating terrains. The optimal train driving problem can be formulated as a general optimization problem with a discrete driving operation as the output, which can then be exerted to the master controller, by considering factors such as railway properties, train states, train attributes, marshaling information, and various external disturbances. A human driver usually makes decisions of accelerating or decelerating the train according to the lessons learned from their teacher drivers and their experiences accumulated so far. Every operation has an impact on the final energy consumption and punctuality. After reviewing a large number of human driving records, we find that there are common patterns that can be identified in the driving behaviors. Figure 2 shows a few representative patterns represented as velocity and driving operations. The hidden patterns of human drivers actually provide important clues to derive an optimized driving solution. We thus intend to put effort on learning the correlation from the experienced drivers.

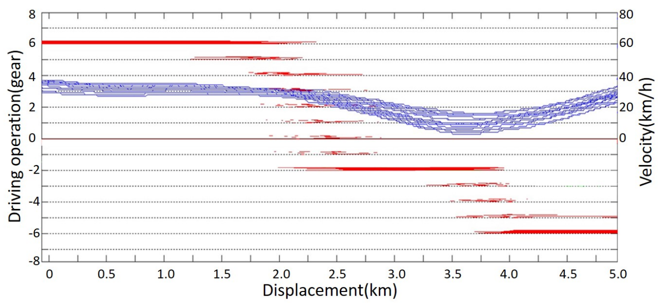

Figure 2: A snippet of human driving data. The blue and red lines are the velocity and the driving operations, respectively. Certain patterns can be easily recognized for most of the route.

High-Order Correlation Learning for Train Driving

Problem Definition

The energy-efficient train driving problem is a complex optimization problem with the driving operation as the output under various environmental and operational constraints. As a result, the driving data varies from one driver to another because not only different drivers may have distinguished driving habits but also different trips are subject to changing environments and operation requirements. This actually caused much difference in the energy consumption as well as the punctuality. However, there still exists certain patterns in the driving records, especially those collected from well-experienced drivers. We thus propose to perform learning on a selected group of “good” drivers such that above-average driving patterns can be expected. By saying “good” drivers, we mean the best ones with their performance in terms of energy-efficiency under the constraints of arrival time within 2 minutes of the timetable and having no over speed limit, frequent parking, and frequent gear changing.

Then an important problem has to be raised as follows. Given the current and historical driving data, how an optimized driving operation for a specific train at running can be accurately predicted? In other words, we need to build a model capturing the high-order correlation between the driving operation and the environmental status by mining the selected records data from the good drivers. The model can then be used to predict an optimized driving operation under realistic train operations. We will further discuss the problem in the remaining of this paper.

Feature Set Design

The driving data provides important information about train status. Nevertheless, to the best of the authors’ knowledge, it is still an open question to define a compact but complete feature set for the driving data. This work starts from the design of such a feature set. According to the analysis on energy-efficient driving, the factors that affect the energy consumption include the railway properties, the train attributes, the marshaling information, train running state, the human factors, and other disturbances(Howlett and Pudney 2012). The above factors are illustrated in Figure 3. For a certain train, here we design a feature set consisting of features that can be classified into three categories, the train attributes, the railway properties and the running information. The train attributes include 4 items as weight, length, cargo number, loaded cargo number. The railway properties include 36 items in total mainly on the gradient properties and the speed limit properties while looking forward and backward. The running information mainly include the dynamic information of the train’s displacement/speed, history of driving operations, and the short zone gradients. Figure 4 lists the details of the features that we designed. It must be noted that the selection of the features influence much on the final learning performance as the prediction of the driving operation mainly relies on such features.

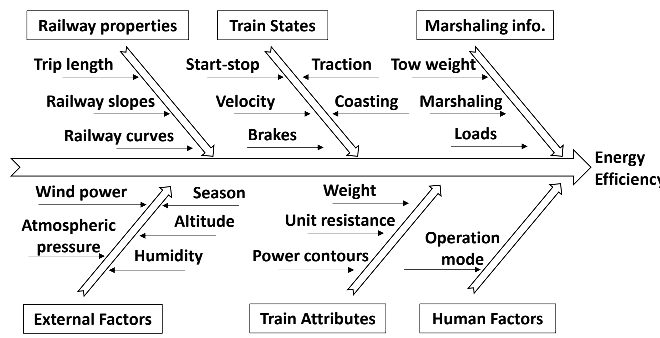

Figure 3: The effect factors of energy-efficient train driving.

A preprocessing of the driving data is performed according to the requirements of the feature extraction process. To avoid the situation where too many oddments in the uphill or downhill trajectories, we categorize and merge the fragmentary gradients into longer ones. Patterns are learned over the newly updated road sections. Table 1 shows the route section that we generated after the merging of neighboring segments.

Table 1: The route section categories

Section Type	Lable	Gradient Scope
Steep Downhill Section	-2	≤ -3
Gentle downhill Section	-1	-1 to -3
Gentle grade Section	0	-1 to 1
Gentle Uphill Section	1	1 to 3
Steep Uphill Section	2	≥ 3

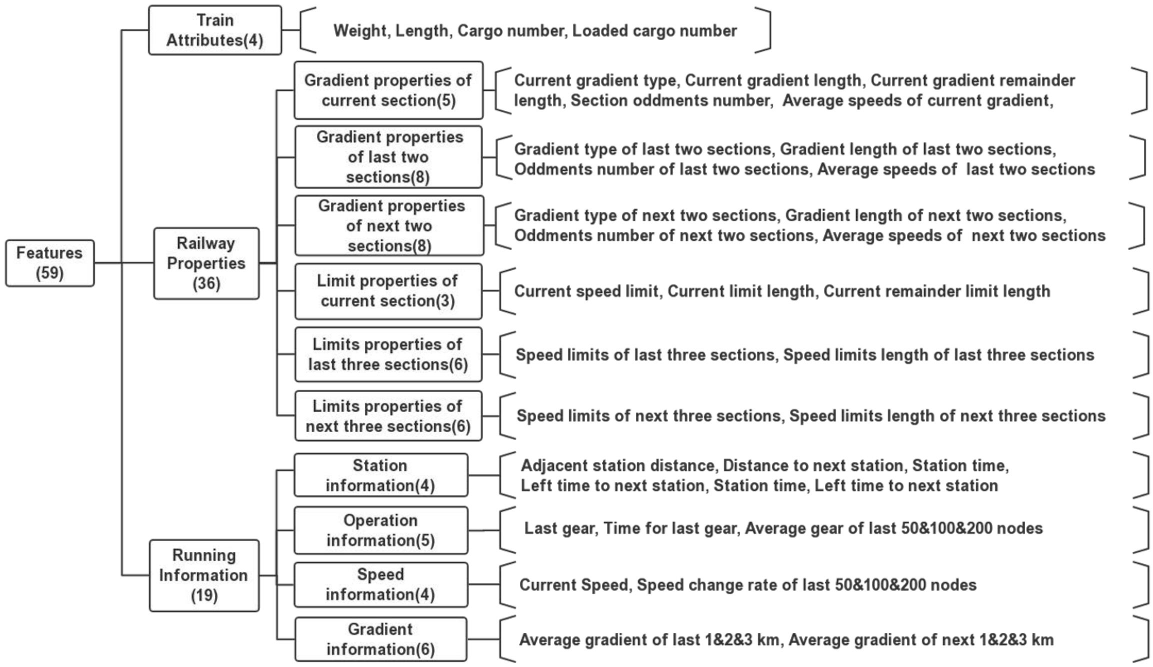

Figure 4: The designed 59-dimensional feature set for the driving records covering the train attributes, the railway properties and the running information.

Hypergraph Model for High-Order Correlation Learning

With the train features, the key task here is to identify the suitable driving operation by considering both driving experience and the current status. It can be regarded as a classification task. However, the correlation among these train features is complex and it is challenging to explore the high-order correlation among these data.

Here we employ the hypergraph model to capture the high-order correlation for driving operation classification. A hypergraph is a generalization of the graph in which an edge can connect any number of vertices(Pu and Faltings 2012; Ghio et al. 2017; Gao et al. 2012; 2014; Han et al. 2017; Zhang, Meng, and Han 2017). A hyperedge may have an arbitrary number of nodes. Here, assuming there are m existing driving experience with train features and corresponding driving operations, the objective here is to learn an optimal hypergraph embedding to project the train features to existing driving operations. We first record all training data in a hypergraph structure. We offer here a quite preliminary definition for hypergraph. A hypergraph $\mathcal{G} = (\mathcal{V}, \mathcal{E}, w)$ is composed of a vertex set $\mathcal{V}$, an edge set $\mathcal{E}$, and the weights of the edges w. Each edge e is assigned a weight $w(e)$. The hypergraph $\mathcal{G}$ can be denoted by a $|\mathcal{V}| \times |\mathcal{E}|$ incidence matrix $\mathbf{H}$, in which each entry is defined by:

$$h(v,e) = \begin{cases} 1 & \texttt{if } v \in e \\ 0 & \texttt{if } v \notin e \end{cases} \tag{4}$$

For a vertex $v \in \mathcal{V}$, its degree is defined by:

$$d\left(v\right) = \sum\nolimits_{e \in \mathcal{E}} \omega\left(e\right) h\left(v, e\right). \tag{5}$$

For an edge $e \in \mathcal{E}$, its degree is defined by:

$$\delta(e) = \sum\nolimits_{v \in \mathcal{V}} h(v,e). \tag{6}$$

Here we let $\mathbf{D}_v$ and $\mathbf{D}_e$ denote the diagonal matrices of the vertex degrees and the edge degrees, respectively, and $\mathbf{W}$ denote the diagonal matrix of the edge weights.

To learn the high-order correlation among driving data, the training data from the driving records of experienced drivers are employed to build a hypergraph. In the hypergraph, each vertex denotes one driving record node with a respective driving operation label. A clustering is performed to serve the purpose of constructing the hyperedges. During the clustering process, each time one vertex is selected as the centroid, and its nearest neighbors are selected to be connected by a corresponding hyperedge. In this work, the top 5 neighbors are employed. Figure 5 illustrates the hypergraph model construction process. Based on this hypergraph, the incidence matrix $\mathbf{H}$, the edge degree matrix $\mathbf{D}_e$ and the vertex degree matrix $\mathbf{D}_v$ can be generated accordingly.

To optimize the hypergraph embedding, an inductive hypergraph learning process is conducted targeting on a regularized projection to discriminate different categories. The cost function Ψ for learning the projection matrix $\mathbf{M}$ is composed of three parts: hypergraph Laplacian regularizer $\Omega\left(\mathbf{M}\right)$, empirical loss $\mathcal{R}_{emp}\left(\mathbf{M}\right)$, and the regularizer on the project matrix $\Phi\left(\mathbf{M}\right)$:

$$\Psi = \left\{\Omega\left(\mathbf{M}\right) + \lambda \mathcal{R}_{emp}\left(\mathbf{M}\right) + \mu \Phi\left(\mathbf{M}\right)\right\}. \tag{7}$$

The hypergraph Laplacian regularizer for $\mathbf{M}$ is under the assumption that strongly connected vertices should have similar labels. The hypergraph Laplacian regularizer can be

$$
\begin{aligned}
\Omega\left(\mathbf{M}\right) &= \frac{1}{2}\sum_{k=1}^{c}\sum_{e\in\mathcal{E}}\sum_{u,v\in\mathcal{V}}\frac{\mathbf{W}(e)\mathbf{H}(u,e)\mathbf{H}(v,e)}{\delta(e)}\left(\frac{(\mathbf{X}^{\mathrm{T}}\mathbf{M})(u,k)}{\sqrt{d(u)}}-\frac{(\mathbf{X}^{\mathrm{T}}\mathbf{M})(v,k)}{\sqrt{d(v)}}\right)^2 \\
&= \sum_{k=1}^{c}\sum_{e\in\mathcal{E}}\sum_{u,v\in\mathcal{V}}\frac{\mathbf{W}(e)\mathbf{H}(u,e)\mathbf{H}(v,e)}{\delta(e)}\left(\frac{\mathbf{F}(u,k)^2}{d(u)}-\frac{(\mathbf{X}^{\mathrm{T}}\mathbf{M})(u,k)(\mathbf{X}^{\mathrm{T}}\mathbf{M})(v,k)}{\sqrt{d(u)d(v)}}\right) \\
&= \sum_{k=1}^{c}\left\{\sum_{u\in\mathcal{V}}\mathbf{F}\left(u,k\right)^2\sum_{e\in\mathcal{E}}\frac{\mathbf{W}(e)\mathbf{H}(u,e)}{d(u)}\sum_{v\in\mathcal{V}}\frac{\mathbf{H}(v,e)}{\delta(e)}-\sum_{e\in\mathcal{E}}\sum_{u,v\in\mathcal{V}}\frac{(\mathbf{X}^{\mathrm{T}}\mathbf{M})(u,k)H(u,e)\mathbf{W}(e)\mathbf{H}(v,e)(\mathbf{X}^{\mathrm{T}}\mathbf{M})(v,k)}{\sqrt{d(u)d(v)}\delta(e)}\right\} \\
&= \sum_{k=1}^{c}(\mathbf{X}^{\mathrm{T}}\mathbf{M})(:,k)^{\mathrm{T}}\left(\mathbf{I}-\mathbf{\Theta}\right)(\mathbf{X}^{\mathrm{T}}\mathbf{M})(:,k) = \mathrm{tr}\left(\mathbf{M}^{\mathrm{T}}\mathbf{X}(\mathbf{I}-\mathbf{\Theta})\mathbf{X}^{\mathrm{T}}\mathbf{M}\right) \\
&= \mathrm{tr}\left(\mathbf{M}^{\mathrm{T}}\mathbf{X}\Delta\mathbf{X}^{\mathrm{T}}\mathbf{M}\right)
\end{aligned} \tag{8}
$$

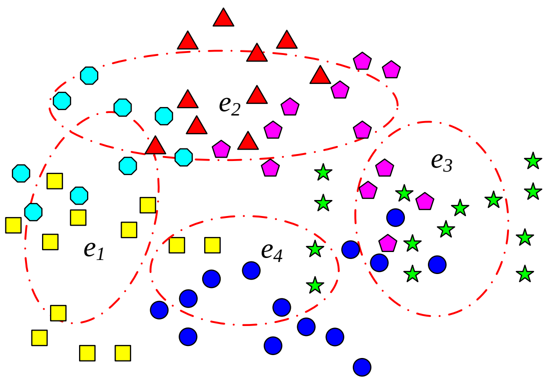

Figure 5: A schematic illustration of the hypergraph model construction with each set of a specific mark as the sectional slices of a driving trip from the selected experienced driver and the dot dash circle be the possible clustering in constructing the hypergraph edges e_i's (i=1,2,...).

written as Eq. (8), where $\mathbf{\Theta} = \mathbf{D}^{v-\frac{1}{2}}\mathbf{HWD}^{e-1}\mathbf{H}^{\mathrm{T}}\mathbf{D}^{v-\frac{1}{2}}$. Eq. (8) serves as the regularizer on hypergraph structure, which guarantees that the stronger connections of vertices on hypergraph lead to higher similarities of the corresponding labels. The empirical loss term on $\mathbf{M}$ is defined as

$$\mathcal{R}_{emp}\left(\mathbf{M}\right) = ||\mathbf{X}^{\mathrm{T}}\mathbf{M}-\mathbf{Y}||^2. \tag{9}$$

$\Phi(\mathbf{M})$ is a ℓ_2 norm regularizer to avoid overfiting for $\mathbf{M}$, which is defined as:

$$\Phi(\mathbf{M}) = ||\mathbf{M}||^2. \tag{10}$$

The inductive learning task on hypergraph can be written as:

$$\arg\min_{\mathbf{M}}\left\{\Omega\left(\mathbf{M}\right)+\lambda\mathcal{R}_{emp}\left(\mathbf{M}\right)+\mu\Phi\left(\mathbf{M}\right)\right\} \tag{11}$$

To solve the above learning task, we can derive it to $\mathbf{M}$ as:

$$
\begin{aligned}
&\frac{\partial}{\partial M}\left\{\mathrm{tr}\left(\mathbf{M}^{\mathrm{T}}\mathbf{X}\Delta\mathbf{X}^{\mathrm{T}}\mathbf{M}\right)+\lambda||\mathbf{X}^{\mathrm{T}}\mathbf{M}-\mathbf{Y}||^2+\mu||\mathbf{M}||^2\right\}=0 \\
&\Rightarrow 2\mathbf{X}\Delta\mathbf{X}^{\mathrm{T}}\mathbf{M}+2\lambda\mathbf{X}\mathbf{X}^{\mathrm{T}}\mathbf{M}+2\mu\mathbf{M}-2\lambda\mathbf{XY}=0
\end{aligned} \tag{12}
$$

We then can achieve

$$\mathbf{M} = \lambda\left(\mathbf{X}\Delta\mathbf{X}^{\mathrm{T}}+\lambda\mathbf{X}\mathbf{X}^{\mathrm{T}}+\mu\mathbf{I}\right)^{-1}\mathbf{XY}. \tag{13}$$

For the coming data, x, the prediction of x's operation can be achieved by

$$\arg\max_{k} x^{\mathrm{T}}\mathbf{M}. \tag{14}$$

It means that once a hypergraph model was trained, a real-time driving operation can be generated for real-time train driving.

We note that the intelligent train control problem is highly in nonlinear, complex in constraints, and the hypergraph model is superior on modeling of such high-order relationship.

Tuning and High-Order Correlation Reinforcement Updating

In training the hypergraph model, a normalization on the selected features is performed. To improve clustering performance, certain parameters like λ in Eq. (13) are fine-tuned for specific circumstances. Extra training and simulation are also needed to tune the weights on the key parameters, *e.g.,* the weight and length of the train, the slope of the railway, the speed limit. In addition, certain rules have to be applied for safety concerns. One such rule is to guarantee that the train will not exceed the speed limit. The above hypergraph model guarantees an above-average driving solution. To improve the solution quality, we perform a reinforcement updating on the high-order correlation model shown in Algorithm 1. It should be noted that the cost function for model updating considers both energy-efficiency and punctuality.

Experiments and Application

Experimental Platform

The experimental locomotive that we employed has a gear of 17 levels in which levels 1 to 8 is for traction, level 0 for neutral, and levels -1 to -8 for braking. The power characteristics of the locomotive for both traction and braking are shown in Figure 6. Typically, different gear of the train will represent different power and, of course, with different energy consumption rate, that is, the higher the gear is, the higher in power and energy consumption rate will be. The automatic driving controller can imitate the gear handling on the train to send control instructions according to the onboard calculation based on the trained hypergraph model. Evaluations on the consumption of time and energy can be done by the train motion model in the simulation platform.

We choose a typical railway route for the experimental use. The complex geographical features of the railway line is shown in Figure 7. The railway route feature mainly include gradient, curve and tunnel information along the mileage that will reflect the running resistance of a train, speed limit

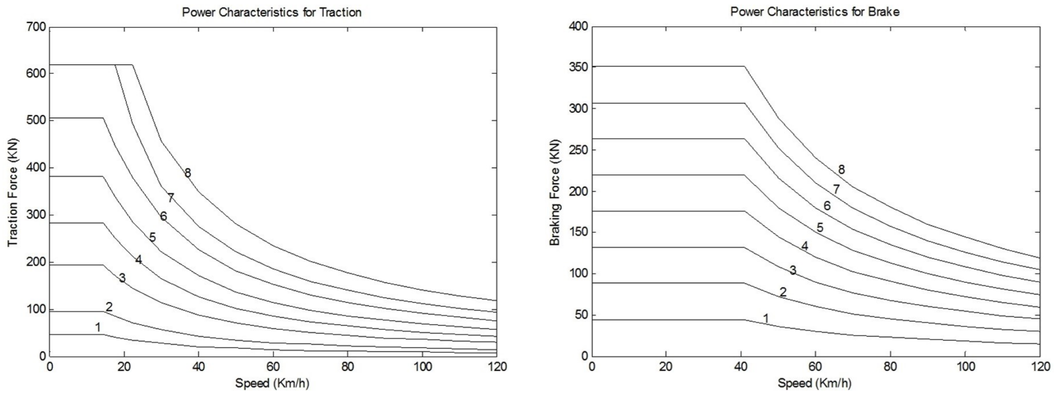

Figure 6: Power characteristics of the selected locomotive for traction and braking. Different gear will represent different power and of course with different energy consumption rate. Generally, the higher the gear, the higher in power and energy consumption rate.

Algorithm 1: The high-order correlation reinforcement updating process

Input: The n_s slices as training data from the top ranking n_r trips with the 59-dimensional feature set preprocessed
Output: The projection matrix M

while $True$ **do**
- High-order correlation learning for the projection matrix M;
- Get the simulation result with the chosen n_t testing trips;
- Evaluate the n_t trips with $J_i = \frac{(T_i - \overline{T})}{\overline{T}} + \alpha \frac{E_i}{W_i}$; ($T_i, E_i, W_i$ are the corresponding time, energy and train weight for trip i)
- Rank the total $n_r + n_t$ trips with J_i;
- **if** *Convergence or reach a maximum cycle number* **then**
  - $break$;
- Choose the top n_r trips to get n_s slices for the training data, preprocess.

return M

for safety concern, stations with distance and time requirements, *etc.* The Automatic Train driving problem is to plan certain train running profile under the chosen route environments. The loads inputs and the trip constraints in the simulations are given according to different actual driving data.

Experiments were carried out on the Hardware-in-Loop test platform shown in Figure 8. The platform is constructed with the onboard Automatic Train Driving system hardware device developed in this work, and other devices from the real freight train in operation. Such devices will send train running information to the Automatic Train Driving system by doing simulation with the longitudinal dynamics of the train. The Automatic Train Driving system will generate the optimized throttle sequence to drive the simulated train model. The outputs of the train model then serve as the feedbacks to the Automatic Train Driving system.

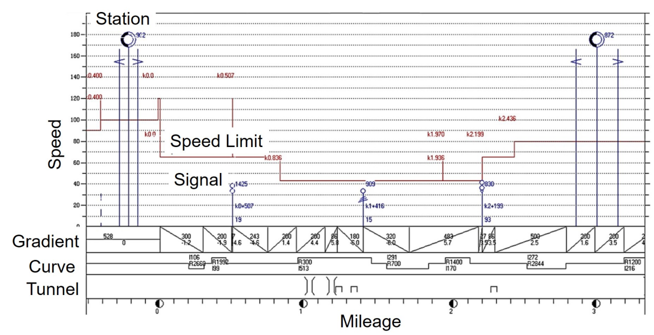

Figure 7: Illustration of part of the railway route using in the experiments. The variance of the gradient, the curve, and the tunnel will reflect as different resistance against the train driving. Strictly punctuality is required between the stations, and the train has to satisfy the speed limit constraints all along the route.

Experiments and Analysis

The hypergraph-based high-order correlation learning approach was implemented as an integrated software platform for off-line learning and an onboard execution. To enhance the accuracy of prediction, different hypergraph models can be trained for trains with according to the weight range of the trains. Such a weight range decentralization helps increase the flexibility and efficiency of the model.

In the experiment, we choose 400 best valued trips from thousands of records with train weights in the range of 3000 to 4000 tons. The evaluation on a trip is simply chosen in an ascending order by energy amount over weight as *E/W*. Then $23,499$ sectional slices were collected from the records as the base of the training data, with each slice having the feature set preprocessed. The parameter λ was tuned to 0.1 according to the tested results from a value set of [0.001,

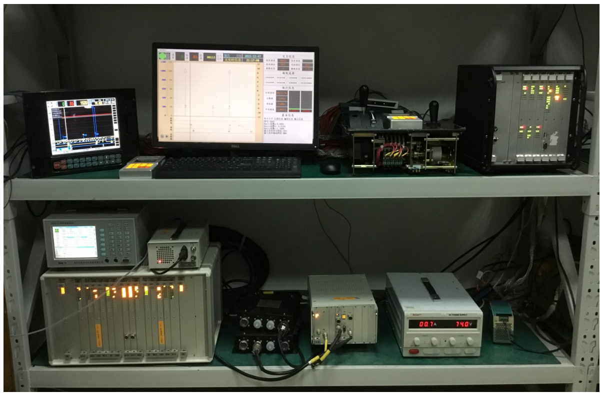

Figure 8: The Hardware-in-Loop testing platform for the Automatic Train Driving system.

0.01, 0.1, 1, 10, 100, 1000], and the Mapminmax method in Matlab were employed for normalization. The railway route is with a length of 15.85 Km between two stations. To evaluate the proposed techniques, we randomly chose other 400 real trips of operating trains of a commercial locomotive depot. The evaluation are directly addressed to the average amount of energy consumption for the testing trips compare to the original training data.

As a result, the statistical results show that, the driving trips derived by the first round learning model achieve a result of 8.16% lower than the average level of the 400 drivers, with the total energy consumption averagely at about 159.95 Kg by the proposed approach and 173.01 Kg by the human drivers. The reinforcement training process were carried on by select the new best 400 trips according to Algorithm 1. Further steps of a 10-round recursive reinforcement updating process enhance the energy-saving to 9.86%. Such an energy saving proved the effectiveness of the proposed techniques. The well-trained hypergraph model was tested on our Hardware-in-Loop platform with a timely decision be achieved.

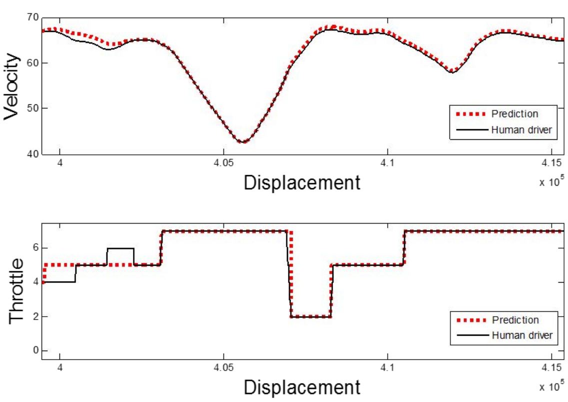

Figure 9: The illustration of an example driving trip by the proposed approach compare to a human driver record.

It shows that the proposed approach works generally even better than the training data in terms of energy-efficiency. An example driving trip by the trained model is illustrated in Figure 9 which shows the consistency with the experienced driver. A compared illustration data by using SVM is shown in Figure 10, which is relatively weak in the precision of gear prediction, and may cause gradually deviation of the velocity. We did not compare the proposed approach with analytical or numerical optimization approaches because: 1) analytical solutions are only feasible under a set of assumptions that hardly hold in realistic; 2) the automatic train driving problem is a sequential decision problem by nature and thus a numerical solution can be extremely expensive. The solution in this work does not depend on the assumptions and is also compute-efficient.

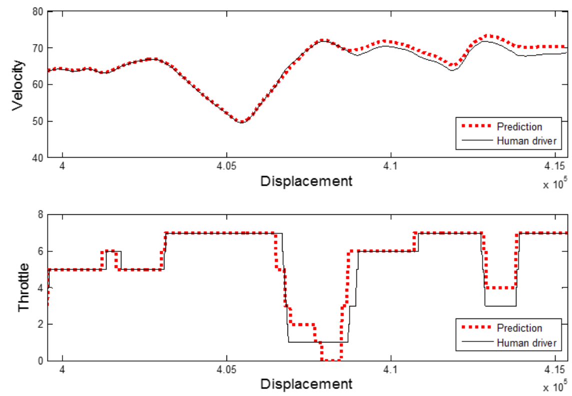

Figure 10: The illustration of an example driving trip by SVM compare to a human driver record.

Conclusion

The derivation of an energy-optimal train driving solution proves to be a significant challenge. This paper introduced a high-order correlation learning process for online generation of the energy optimized driving solution for railway trains. A novel feature set was proposed for the driving records of railway trains. Starting from the driving data of experienced human drivers, a hypergraph model is used to learn the optimal embedding from the specified feature set for the decision of a driving operation. The proposed techniques are validated on a Hardware-in-Loop platform. Experimental results proved that an energy saving of around 10% could be achieved when compared with the average level of the drivers. A large collection of directions worth investigating in the future, *e.g.*, refining the feature set, exploring more machine learning algorithms on such problem and the fleet train problems.

Acknowledgments

This research is sponsored in part by National Key R&D Program of China (Grant No. 2017YFC011300), National Natural Science Funds of China (61671267, 61527812), National Science and Technology Major Project (No. 2016ZX01038101), MIIT IT funds (Research and application of TCN key technologies) of China, and The National Key Technology R&D Program (No. 2015BAG14B01-02).

References

Albrecht, A.; Howlett, P.; Pudney, P.; and Vu, X. 2011. Optimal train control: analysis of a new local optimization principle. In *Proceedings of the 2011 American Control Conference*, 1928–1933. IEEE.

Bansal, N.; Blum, A.; and Chawla, S. 2004. Correlation clustering. *Machine Learning* 56(1-3):89–113.

Betts, J. T. 2010. *Practical methods for optimal control and estimation using nonlinear programming*. SIAM.

Chang, C. S., and Kwan, C. M. 2004. Evaluation of evolutionary algorithms for multi-objective train schedule optimization. In *Australasian Joint Conference on Artificial Intelligence*, 803–815. Springer.

Corman, F., and Quaglietta, E. 2015. Closing the loop in real-time railway control: framework design and impacts on operations. *Transportation Research Part C: Emerging Technologies* 54:15–39.

Franke, R.; Terwiesch, P.; and Meyer, M. 2000. An algorithm for the optimal control of the driving of trains. In *Decision and Control, 2000. Proceedings of the 39th IEEE Conference on*, volume 3, 2123–2128. IEEE.

Gao, Y.; Wang, M.; Tao, D.; Ji, R.; and Dai, Q. 2012. 3-d object retrieval and recognition with hypergraph analysis. *IEEE Transactions on Image Processing* 21(9):4290–4303.

Gao, S.; Dong, H.; Chen, Y.; Ning, B.; Chen, G.; and Liang, Q. 2013. Neuro-adaptive and robust automatic train control subject to unknown dynamics. In *Control Conference (CCC), 2013 32nd Chinese*, 8130–8134. IEEE.

Gao, Y.; Ji, R.; Cui, P.; Dai, Q.; and Hua, G. 2014. Hyperspectral image classification through bilayer graph-based learning. *IEEE Transactions on Image Processing* 23(7):2769–2778.

Ghio, M.; Malpetti, D.; Rossi, M.; Bruß, D.; and Macchiavello, C. 2017. Multipartite entanglement detection for hypergraph states. *arXiv preprint arXiv:1703.00429*.

Han, S. H.; Byen, Y. S.; Baek, J. H.; An, T. K.; Lee, S. G.; and Park, H. J. 1999. An optimal automatic train operation (ato) control using genetic algorithms (ga). In *TENCON 99. Proceedings of the IEEE Region 10 Conference*, volume 1, 360–362. IEEE.

Han, J.; Cheng, G.; Li, Z.; and Zhang, D. 2017. A unified metric learning-based framework for co-saliency detection. *IEEE Transactions on Circuits and Systems for Video Technology*.

Howlett, P. G., and Pudney, P. J. 2012. *Energy-efficient train control*. Springer Science & Business Media.

Huang, J.; Deng, Y.; Yang, Q.; and Sun, J. 2016. An energy-efficient train control framework for smart railway transportation. *IEEE Transactions on Computers* 65(5):1407–1417.

Khmelnitsky, E. 2000. On an optimal control problem of train operation. *IEEE Transactions on Automatic Control* 45(7):1257–1266.

Li, Y.-s., and Hou, Z.-s. 2007. Study on energy-saving control for train based on genetic algorithm [j]. *Journal of System Simulation* 2:038.

Liu, M.; Zhang, J.; Yap, P.-T.; and Shen, D. 2017a. View-aligned hypergraph learning for alzheimers disease diagnosis with incomplete multi-modality data. *Medical image analysis* 36:123–134.

Liu, Q.; Sun, Y.; Wang, C.; Liu, T.; and Tao, D. 2017b. Elastic net hypergraph learning for image clustering and semi-supervised classification. *IEEE Transactions on Image Processing* 26(1):452–463.

Lu, S.; Hillmansen, S.; Ho, T. K.; and Roberts, C. 2013. Single-train trajectory optimization. *IEEE Transactions on Intelligent Transportation Systems* 14(2):743–750.

Mao, B. 2008. The calculation and design of train operation.

Michalski, R. S.; Carbonell, J. G.; and Mitchell, T. M. 2013. *Machine learning: An artificial intelligence approach*. Springer Science & Business Media.

Miyatake, M., and Ko, H. 2010. Optimization of train speed profile for minimum energy consumption. *IEEJ Transactions on Electrical and Electronic Engineering* 5(3):263–269.

Pu, L., and Faltings, B. 2012. Hypergraph learning with hyperedge expansion. In *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*, 410–425. Springer.

Siahvashi, A., and Moaveni, B. 2010. Automatic train control based on the multi-agent control of cooperative systems. *The Journal of Mathematics and Computer Science* 1(4):247–257.

Vašak, M.; Baotić, M.; Perić, N.; and Bago, M. 2009. Optimal rail route energy management under constraints and fixed arrival time. In *Control Conference (ECC), 2009 European*, 2972–2977. IEEE.

Vu, X. 2009. *Analysis of Necessary Conditions for the Optimal Control of a Train: New Neccessary Conditions for Energy-efficient Train Control*. VDM Publishing.

Yang, X.; Li, X.; Ning, B.; and Tang, T. 2016. A survey on energy-efficient train operation for urban rail transit. *IEEE Transactions on Intelligent Transportation Systems* 17(1):2–13.

Zhang, Z.; Bai, L.; Liang, Y.; and Hancock, E. 2017. Joint hypergraph learning and sparse regression for feature selection. *Pattern Recognition* 63:291–309.

Zhang, D.; Meng, D.; and Han, J. 2017. Co-saliency detection via a self-paced multiple-instance learning framework. *IEEE transactions on pattern analysis and machine intelligence* 39(5):865–878.

Zhong, W.; Xu, H.; and Zhang, W. 2016. Energy-efficient train reference speed profile optimization using control parameterization method. In *Control Conference (CCC), 2016 35th Chinese*, 10128–10133. IEEE.