ESANN'2003 proceedings - European Symposium on Artificial Neural Networks
Bruges (Belgium), 23-25 April 2003, d-side publi., ISBN 2-930307-03-X, pp. 401-406

Comparison of traditional and neural systems for train speed estimation

V. Colla*, M. Vannucci*, B. Allotta†, M. Malvezzi†

* Scuola Superiore Sant'Anna, Polo Sant'Anna Valdera
Viale Rinaldo Piaggio 34, 56025 Pontedera (PI), ITALY
† Dipartimento di Energetica "S. Stecco," Università di Firenze
Via S. Marta 3, Firenze, ITALY

Abstract. The paper presents and compares different methods for estimating the speed of a train from the measurement of the velocity of two axles in any wheel/rail adhesion conditions. Neural structures outperform traditional crisp reasoning, designed according to the indication of expert personnel, and revealed to be also simpler to build and tune, thus showing their enormous capability of extracting knowledge from data.

1 Introduction

An Automatic Train Protection (ATP) system [1] is generally composed of two subsystems: the *ground subsystem*, that updates the train position and line gradient information by exploiting some sort of absolute data source, such as fixed balises located along the line itself; the *on–board subsystem*, which estimates the actual train speed and position. The ground subsystem communicates to the on–board one the train distance from fixed reference positions, the speed limit that cannot be overcome by the train on that portion of the line and the line gradient. On the basis of such information, the on–board subsystems evaluates the minimum distance d_{min} that allows compliance of the speed limit at the next objective point and the distance to the next information points. d_{min} is evaluated by exploiting the actual train speed, the braking parameters and the objective speed as the distance at which the target speed is reached by applying the maximum deceleration. If the difference between d_{min} and the distance of one or more of the next objective points is smaller than a fixed value, the on–board subsystem intervenes, for instance by activating the emergency braking. It is evident that a correct estimation of the actual train velocity is crucial to evaluate the residual braking resources [2], so as to meet the speed requirements at the target points.

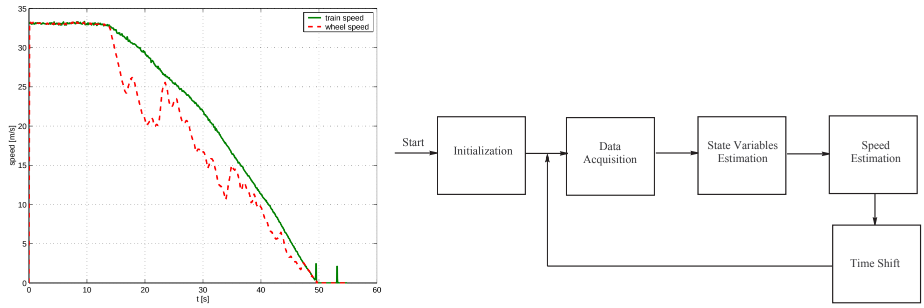
In the ATP system named SCMT [3], which has been developed by Trenitalia

ESANN'2003 proceedings - European Symposium on Artificial Neural Networks
Bruges (Belgium), 23-25 April 2003, d-side publi., ISBN 2-930307-03-X, pp. 401-406

SpA and the University of Florence for equipping most of the Italian trains, the actual train speed and distance from the next objective point is evaluated by processing the measurements provided by two incremental encoders positioned on two independent axles. This paper describes and compares the different algorithms that have been developed for the train speed evaluation.

The paper is organised as follows: the problem of the train speed estimate is presented in Sec. 2, while the estimator developed according to the Trenitalia indications is described in Sec. 3; some alternative neural estimators are described in Sec. 4, the estimator performances are compared in Sec. 5, while Sec. 6 presents some concluding remarks and the guidelines for future work.

2 Problem description



a) b)

Figure 1: a) Train (plain line) and wheel (dashed line) speed in a braking test conducted with a single vehicle in degraded adhesion conditions. b) Scheme of the Trenitalia algorithm

Odometry techniques based on sensors located on one or more axles of the train may be used for dead reckoning between two subsequent exact position measurements: when the train wheels adhere to the rails, the train speed v satisfies the equation $v = R\omega_k$, where ω_k is the angular speed of the k–th wheel and R is the wheel radius. On the contrary, when the adhesion conditions are degraded, which occurrs quite frequently when the train is accelerating or braking and rain, fog, ice, leaves and other similar external factors are present, the pure rolling conditions no longer hold and one ore more axles carrying the odometry sensors can slide. Consequently the train actual velocity must be estimated differently than in the previous case, as the quantity $\delta_{v_k} = v - R\omega_k$ is positive in a braking phase (see Fig.1.a) and negative in a traction phase. The SCMT system contains a module that provide an estimate $\hat{v}_i$ of the train velocity at the time i on the basis of the values $c_1(i)$ and $c_2(i)$ stored in the encoders impulse counters and of the line gradient $d(i)$. The paper describes different solutions for the design of such module.

ESANN'2003 proceedings - European Symposium on Artificial Neural Networks
Bruges (Belgium), 23-25 April 2003, d-side publi., ISBN 2-930307-03-X, pp. 401-406

3 Trenitalia–Unifi crisp algorithm

The first algorithm consists of a set of *if–then* rules, devised on the basis of the Trenitalia personnel's knowledge and of a series of experimental tests. It runs with a sample frequency of 10 Hz, a value representing a compromise between estimation accuracy and substainable computational burden. The estimation procedure (whose details cannot be provided due to confidentiality constraints) is composed of 4 phases (see Fig. 1.b):

1) Variable initialisation This phase is performed when the system is turned on and consists in defining and initialising some variables and parameter values (i.e. thresholds and filters cut–off frequencies) that are used in the procedure.

2) Data acquisition and conditioning The wheel peripheral speeds are evaluated from $c_1(i)$ and $c_2(i)$ and the wheel peripheral accelerations $a_1(i)$ and $a_2(i)$ are obtained by first–order finite differences followed by a first–order low–pass filter, that reduce the noise. The estimate of the train acceleration $\hat{a}(i)$ is obtained by the average of $a_1(i)$ and $a_2(i)$, followed by another first–order low–pass filter and a saturation stage, whose lower and an upper limits depend on the expected deceleration due to the line gradient.

3) State variables evaluation A number of variables is evaluated, whose function is to distinguish the state of the sensorial system (for instance if both the encoders are correctly working), the different kinds of train movements, (namely if a brakig or an accelerating phase occurrs) and the adhesion conditions. Such variables are evaluated by means of a very complex set of *if–then* rules, in which some quantities are compared with fixed thresholds. For instance, the speed of the k^{th} wheel is considered "stable" if its acceleration and variation of acceleration is lower than a predetermined threshold for a fixed time interval. The acceleration variation of an axle is given by $\Delta_{a_k}(i) = \max_{(i-J \leq j, m \leq i)} |a_k(j) - a_k(m)|$ where J is the number of samples which corresponds to a time window of 2 s.

4) Train speed evaluation In each situation, the estimate $\hat{v}(i)$ of the train velocity is obtained with a different elementary procedure (such as a mean value, an integration step, etc...) exploiting the velocity and eventually the acceleration values.

The described algorithm depends on several parameters (in total 27), used either for data processing or for reasoning with crisp *if-then* rules; the values of such parameters were initially set by exploiting the experts' knowledge. In a subcessive phase of the work, the parameter values have been optimised by means of Genetic Algorithms (GAs), by exploiting as *fitness function* f the normalised square root mean square error (NSRMSE) [4] on speed estimation:

$$f(\mathbf{p}) = \frac{1}{\sigma_v} \sqrt{\frac{1}{M} \sum_{i=1}^{M} [\hat{v}(i) - v(i)]^2} \tag{1}$$

where $\mathbf{p}$ is a vector containing the values of the parameters to be optimized, M is the number of samples in the training data set and σ_v is the standard deviation of the train speed.

ESANN'2003 proceedings - European Symposium on Artificial Neural Networks
Bruges (Belgium), 23-25 April 2003, d-side publi., ISBN 2-930307-03-X, pp. 401-406

4 Alternative neural algorithms

The previously described approach is complex, thus its computational time is not negligible and, moreover, its performance is very sensitive to the choice of the parameter values; the resulting relation among the system output $\hat{v}(i)$ and the measured velocities of the two axles is highly non–linear. As a huge amount of experimental data was available, some attempts have been made to simplify the estimator design by adopting a neural structure.

The first straightforward choice for the estimator consisted in a two–layered feedforward neural network. In a first design of the estimator, 4 inputs (wheel velocities, v_1 and v_2, and accelerations a_1 and a_2) have been considered, while in a second one 2 further inputs have been added, namely the acceleration variations Δ_{a_1} and Δ_{a_2}. A preliminary normalisation stage is needed in order to reduce the importance of the velocity values, which differs for one order of magnitude with respect to the other 4 entries of the input vector.

In order to improve the performance, two hierarchical neural structures have been tested; the general idea under these attempts was to be able to distinguish the different situations that should reflect in different kind of input patterns (for instance, in good adhesion conditions while braking, the wheel velocities and accelerations are similar and the aceleration values are negative and not negligible) *before* the estimation, similarly to what is made by the Trenitalia crisp algorithm. Thus, a Self Organising Map (SOM) [5] has been used in order to classify the six-dimensional patterns. In a first attempt, the result of the classification is directly input in a single 7-input neural network together with the six–dimensional input pattern, as depicted in Fig.2.a. In a second attempt, for each class a neural speed estimator has been designed: the results of the classification is used for fedding each six–dimensional pattern to one of the estimating networks according to the class it belongs to, as shown in Fig.2.b.

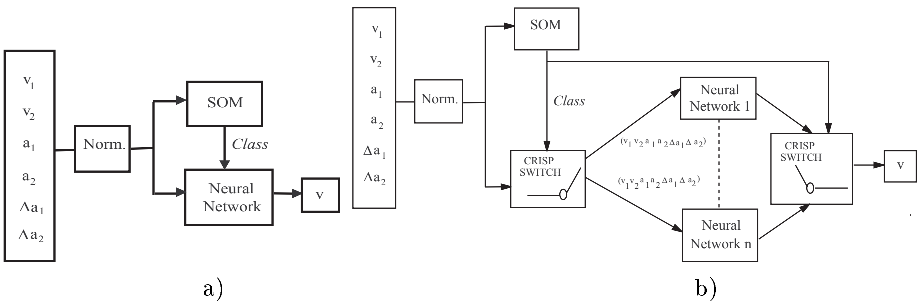


Figure 2: Scheme of the two hyerarchical neural systems for train speed estimation.

ESANN'2003 proceedings - European Symposium on Artificial Neural Networks
Bruges (Belgium), 23-25 April 2003, d-side publi., ISBN 2-930307-03-X, pp. 401-406

5 Numerical results discussion and comparison

Tab. 1 shows the best results obtained with the different train speed estimators: a comparison is possible in terms not only of performance, but also of time required for design and eventual training. As the table refers to the best performing estimator of each class, the design time should include not only the time for elaborating each estimation strategy but also the time for selecting the best particular solution: for instance, the time for elaborating the estimator consisting of a single NN was negligible but some days are required to select both the topology and the dimension of the NN that fits the problem at best. The performances are evaluated through the NSRMSE on speed error (see eq. 1).

Algorithm type	**Design time (days)**	**Training time (min)**	**Evaluation time (ms)**	**NSRMSE**
Trenitalia	60	-	7.9	0.047
Trenitalia Optimized by GAs	60+5	900	7.9	0.043
MLP 4 inp., 25 hidd.	3	50	2.7	0.050
MLP 6 inp., 20 hidd.	3	50	2.7	0.041
SOM + 7 inp. MLP	5	80	2.9	0.040
SOM + 6 inp. MLPs	7	55	2.8	0.037

Table 1: Comparison among the algorithms in terms of times required for design, training, on–line data processing and of performance.

From Tab. 1 it is evident that the design of the crisp algorithm was very complex and time consuming. As far as the optimisation through GAs is concerned, the idea is simple and most of the time was spent for the training: as the crisp algoritm is quite heavy from the computational point of view, also the GA training, which requires many evaluations of the crisp algorithm, has a considerable duration, but allowed a performance improvement of about 8.5%.

For the design of the single–network estimator, different networks with an increasing number m of neurons in the hidden layer have been tested. From the performed attempts, the multilayer perceptron (MLP) revealed to be the most suitable structure and the optimal performance was achieved with 25 neurons for the 4 inputs network, 20 neurons for the 6 inputs network. All the MLP networks were trained with the *Levemberg–Marquardt* method [6]. The network with 6 inputs outperforms the normal and optimised versions of the crisp algorithm with an estimate improvement of 12.8% and 4.7% respectively.

In the two hierarchical neural structures, the choice of the SOM topology for the pattern classifier was straightforward, but different topologies have been tried for the second stages. The two subsequent stages were trained separately: firstly the SOM has been designed for different number of classes (from 3 to 20); afterwards several attempts have been performend in order to chose the topology and the dimension of the following feedforward network. Also in these cases, the simple two-layered MLP structure outperforms the other ones. The

ESANN'2003 proceedings - European Symposium on Artificial Neural Networks
Bruges (Belgium), 23-25 April 2003, d-side publi., ISBN 2-930307-03-X, pp. 401-406

optimal solution have been found by trying many different combinations. To sum up, for the structure depicted in Fig.2.a, the optimal solution consists in a SOM with 10 classes and a MLP with 20 hidden neurons, while for the structure depicted in Fig. 2.b, 3 classes are sufficient and the following 3 MLPs have 15 hidden neurons each. The small networks dimensions justify the smaller time required for the training and for the speed evalution in the second hierarchical neural structure, that is also the best performing estimator, as the NSRMSE is reduced of 9,1% with respect to the sigle NN with 6 inputs and of 13,95% with respect to the optimised crisp algorithm.

6 Conclusions and future work

The problem of train speed estimation from the measurement of the velocity of two axles in all the adhesion conditions has been faced with different techniques, from very traditional crisp reasoning to soft–computing techniques. In particular neural systems showed their efficiency by providing a better estimate in any condition; moreover the design of the neural systems was far simpler and the possibility of training with experimental data make such systems capable of self-improvement in a straightforward manner. Current work is focused on the possibility of joint training of the neural subsystems in the hierarchical structures, as well as toward different solutions, such as fuzzy systems. As the system is devoted to enhance the trains security, a great amount of work is still to be done in order to make a neuro-fuzzy system safe and reliable in all the situations, so as to overcome the natural suspicion of the technical personnel, but the obtained results are encouraging.

References

[1] N. Bin: "Analysis of train braking accuracy and safe protection distance in automatic train protection (ATP) systems," *Proc. 5-th Int. Conf. on Computer Aided Design, Manufacture and Operation in the Railway and other Advanced Mass Transit Systems COMPRAIL 1996* pp. 111-118, 1996.

[2] P. Presciani, M. Malvezzi, G.L. Bonacci, M. Balli: "Development of a braking model for speed supervision systems," *Proc. 2001 World Cong. Railways Research WCRR 2001*, Koln, Germany, November 25-29, 2001.

[3] "SCMT Progetto dell'algoritmo per il calcolo della velocitá istantanea del treno e dello spazio percorso," UTMR/DT.S.PS, Trenitalia SpA, Unitá Tecnologie Materiale Rotabile, October 31–th 2000.

[4] Q. Zhang, A. Benveniste: "Wavelet Networks," *IEEE Transactions on Neural Networks*, Vol. 3, No. 6, pp. 889–898, November 1992.

[5] T. Kohonen: "The self-organizing map," *Proceedings of the IEEE*, Vol. 78, No. 9, pp. 1464-1480, 1990.

[6] D. Marquardt: "An algorithm for least–square estimation of nonlinear parameters," *SIAM Journal of Applied Mathematics*, No. 11, pp. 164–168.