Intl. Trans. in Op. Res. 11 (2004) 381–394

INTERNATIONAL TRANSACTIONS IN OPERATIONAL RESEARCH

Analysing stability and investments in railway networks using advanced evolutionary algorithms

O. Engelhardt-Funke and M. Kolonko

Institut für Mathematik, Technical University Clausthal, Erzstr. 1, D-38678 Clausthal-Zellerfeld, Germany
Corresponding author e-mail: kolonko@math.tu-clausthal.de

Received 3 September 2002; received in revised form 5 May 2003; accepted 21 July 2003

Abstract

We consider a network of periodically running railway lines. Investments are possible to increase the speed and to improve the synchronisation of trains. The model also includes random delays of trains and the propagation of delays across the network. We derive a cost-benefit analysis of investments, where the benefit is measured in reduced waiting time for passengers changing lines. We also estimate the actual mean waiting time simulating the train delays. This allows us to analyse the impact that an increasing synchronisation of the timetable has on its stability. Simulation is based on an analytical model obtained from queueing theory. We use sophisticated adaptive evolutionary algorithms, which send off avant-garde solutions from time to time to speed up the optimisation process. As there is a high correlation between scheduled and estimated waiting times for badly synchronised timetables, we are even able to include the time consuming simulation into our optimisation runs.

Keywords: railway network; timetable optimisation; simulation of delays; evolutionary algorithms; multi-criteria optimisation; simulation and optimisation

1 Introduction

To analyse the performance of a railway network is a very complex task. In this paper we show how timetable optimisation can help to solve this problem.

We consider a network of railway lines that run periodically according to a timetable T. Timetables are optimised with respect to several cost criteria reflecting different aspects of their performance. The resulting optimal cost values and optimal solutions (timetables) then show the interdependence of the criteria, and may yield cost-benefit analyses.

The first cost criterion for a timetable T is the total scheduled waiting time $W(T)$. This is the total time all passengers have to wait according to T either at stations when changing lines, or in the train during stops. An exact definition follows below. We then extend the network model to a

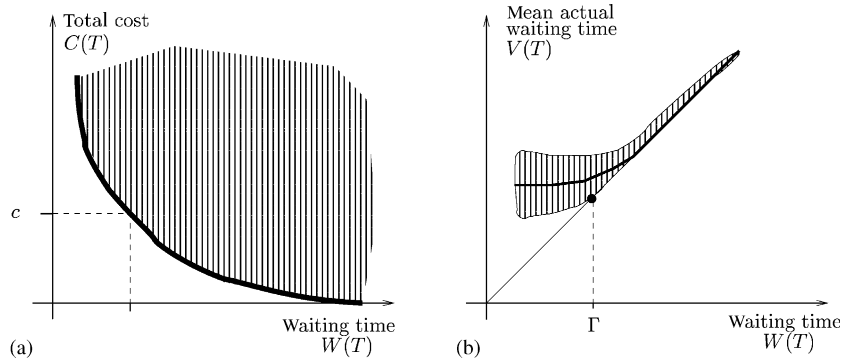

Fig. 1. (a) The Pareto-optimal solutions form a cost-benefit-curve. (b) Mean actual and scheduled waiting time are highly correlated for badly synchronised timetables.

model of strategic investment decisions: investments can be made in the tracks to speed up the trains. This will allow a better synchronisation of trains at stations reducing the waiting time $W(T)$. Let $C(T)$ denote the total investment required by a timetable T, see details below.

We may now try to find the Pareto-optimal timetables with respect to the two cost functions $W(\cdot)$, $C(\cdot)$. A timetable T is called *Pareto-optimal* (or non-dominated) if there is no timetable T' with

$$W(T') < W(T) \quad \text{and} \quad C(T') < C(T).$$

that is, there is no T' which has less waiting time *and* less investment costs than T. The cost function values of the Pareto-optimal solutions in the $(W(\cdot), C(\cdot))$-plane will then yield a cost-benefit-curve. See Fig. 1(a) for an example. Here, the hatched region denotes the cost function values of all timetables, the Pareto-optimal timetables belong to the bold curve. From this curve one may, for example, see the minimal scheduled waiting time that is enabled by an investment of c euro. The underlying Pareto-optimal timetable contains exact information on where the waiting time occurs and where the money has to be invested (see details in Section 2 below).

In the practical operation of a network, small delays to trains are probably inevitable, therefore another reasonable requirement for timetables is that they should be 'stable'. We measure the stability of a timetable T by its mean actual total waiting time $V(T)$ under random delays of trains, and include $V(\cdot)$ as a third cost function into our optimisation model.

How will the scheduled waiting time $W(T)$ and the mean actual waiting time $V(T)$ be related? A scheduled waiting time at a station is a waste of time from the perspective of the traveller, but it also serves as a time buffer for the delays of trains. Therefore, one should expect that for timetables T with larger values of $W(T)$, delays do not affect actual waiting times so that the mean actual waiting time $V(T)$ will be of similar size as the scheduled time $W(T)$. Timetables T with $W(T) \sim V(T)$ are regarded as stable, in Fig. 1(b) this is illustrated by the right side of the graph. During the optimisation process, timetables with reduced scheduled waiting time are found. Here, the time buffers at stations are becoming smaller and the chance to miss a connection when arriving in a delayed train increases. Therefore, one should expect that the mean actual waiting time also increases or is at least not reduced at the same rate as $W(T)$. This is indicated in the left side of Fig. 1(b) (see also Fig. 6 below for empirical evidence of this tendency).

When investments are made to increase the synchronisation of the trains and to reduce the scheduled waiting time $W(T)$, the point Γ in Fig. 1(b) is of particular interest. Beyond Γ, additional investments into the synchronisation of trains have decreasing benefit as they lead to unstable timetables. Further investments should then be made to improve the reliability of the operational times, for example, by additional tracks parallel to others. In the present paper however, we only consider investments into existing tracks that reduce the running times of trains. This will not affect the occurrence of delays but only their reduction by time buffers (see below). Hence, the question here is the balance of acceleration and synchronisation against stability at a *given* level of random disturbances.

Figure 1 shows ideal situations. In practice, the networks are much too complex to obtain exact Pareto-optimal solutions even in the case of the two simple cost functions W and C. At present, we are not able to calculate the mean actual delay $V(T)$ analytically for realistic delay distributions and networks of realistic size. Therefore we have to replace the exact value by an estimate (which we shall also call $V(T)$) obtained from computer simulation.

A cost function that can only be simulated excludes all classical mathematical optimisation strategies, therefore we used sophisticated evolutionary (genetic) algorithms instead. As a consequence, our solutions are only approximating the Pareto-optimal set and instead of the complete cloud of cost function values as in Fig. 1, we shall only see samples of it as they appear in the optimisation process.

We developed an adaptive multi-objective genetic algorithm which is enhanced by additional features. First we use ‘avant-garde’ breeding. This operation produces single offspring solutions that tend to run ahead of the rest of the population. The population then inherits the good properties from this solution in just a few generations. A further acceleration is obtained by exploiting the correlation between the two waiting-time cost functions $W(T)$ and $V(T)$. As simulation is very time-consuming (compared to the calculation of the scheduled waiting time $W(T)$), the cost function $V(T)$ is switched off during early optimisation stages. From time to time it is switched on to check the correlation between $W(T)$ and $V(T)$ in the present population. If this correlation is low, simulation is used more often. This turned out to be sufficient to obtain pictures like Fig. 1(b) and rough estimates of Γ.

Our model is based on earlier work of Nachtigall (1996, 1998) and Nachtigall and Voget (1997). Investigations into timetable optimisation are also reported in Zimmermann and Lindner (2003). Delays are considered in a slightly different set up in Goverde (1998, 2002), where the so-called max-plus approach is used to model the propagation of a single delay situation and to evaluate the efficiency of counter-measures.

The structure of the paper is as follows. In Section 2 we describe the railway network and its deterministic mathematical model in some more detail. Section 3 shows how random delays are incorporated into this model. The genetic algorithm and its refinements are presented in Section 4 together with some empirical results with real railway networks. The final Section contains some conclusions.

2 The railway network and its mathematical model

We consider a network with a common period of τ minutes. Let $S_1, \ldots, S_K$ denote the stations of the network and $L_1, \ldots, L_M$ its lines, where a line is defined by the sequence of consecutive stations it serves (every τ minutes). A timetable T consists of two lists: a list of departure times $\pi(S, L), 0 \leqslant \pi(S, L) < \tau$, for each pair (S, L) such that line L runs through station S, and a list of running times $\delta(S, S')$ such that stations S, S' are neighbouring stations in at least one line.

To simplify matters, we neglect problems of starting and closing daily operation and simply assume that trains are running continously every τ minutes. Thus, line L leaves station S every τ minutes with one train leaving at $\pi(S, L)$ minutes past midnight. It then needs $\delta(S, S')$ minutes to arrive at the next station S'. Note that the running time between two stations is prescribed by the timetable.

We model the network at a rather general level without details like safety distances and capacity restrictions. This reduces the complexity considerably and is justified at least in a strategic planning situation where for example, the track layout is of no concern. (See Engelhardt-Funke and Kolonko, 2001, for more details on the model.)

We may now formally define the scheduled total waiting time

$$W(T) := \sum_{(S,L,L')} w_T(S, L, L') \cdot g(S, L, L') \tag{1}$$

where the sum is over all triplets (S, L, L') that denote a possible change from line L to line L' at station S. $w_T(S, L, L')$ is the scheduled time a passenger has to wait for the next arrival of line L' after he has arrived with line L at station S. If $L = L'$ then $w_T(S, L, L)$ is the duration of the scheduled train stop at station S. In both cases this is the 'pure' waiting time, without transfer times like the time necessary to change platforms, and for boarding and alighting trains. $g(S, L, L')$ is a weighting function, for example, the estimated mean number of passengers that use connection (S, L, L'). It may also be used to put an additional penalty on the waiting time spent on a platform compared to the waiting time spent in a train during stops. Note that $W(T)$ is computed on the basis of the scheduled times given in timetable T.

We now further extend the model of train operation to a model of strategic investment planning. We assume that investments into the tracks of the network are possible for example, by rebuilding a crossing or improving a switch, and that these improvements shorten the running time of trains on that section. More precisely, we assume that for each section there is a catalogue of possible improvements including their costs and the acceleration obtainable on that section.

A timetable T fixes a certain running time δ (S, S') on each section (S, S') which may require some investment for its realisation. Let $c_{(S,S')}(t)$ denote the amount of money that has to be invested into section (S, S') according to the catalogue to allow a running time of t minutes on $(S, S') \cdot c_{(S,S')}(\cdot)$ will usually be a step function as in Fig. 2, where each step is connected to a particular investment. If there is no possibility to improve (S, S') then $c_{(S,S')}(t)$ will be zero for the present running time $t = t_1(S, S')$ and infinity for all $t < t_1(S, S')$.

The total investment cost required by timetable T (i.e. by the running times $\delta(S, S')$ contained in it) is defined as

$$C(T) := \sum_{(S,S')} c_{(S,S')}(\delta(S, S')) \tag{2}$$

where the sum runs over all pairs of neighbouring stations S, S'.

Note that in our model, timetable T contains a detailed investment plan of total amount $C(T)$. A decision to invest a certain sum c into the network allows us to run the network with the

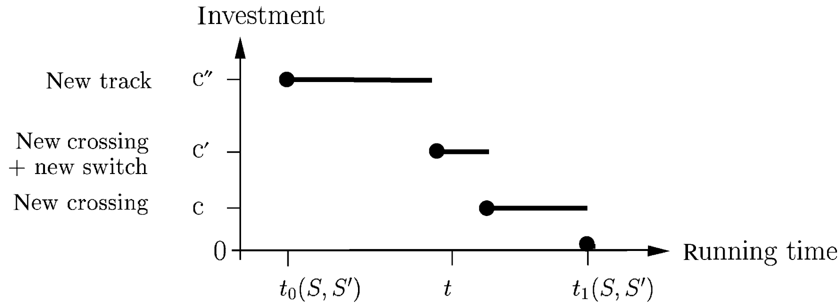

Fig. 2. A simple local cost function $c_{(S,S')}(\cdot)$.

Pareto-optimal timetable T that has $C(T) \leqslant c, C(T)$ maximal. $W(T)$ evaluates the impact that all investments required by T have on the waiting times across the whole network. This includes the (possibly negative) effects that an acceleration of trains on one section may have on connections at stations along the line.

During the optimisation process, equations (2) and (3) have to be computed very often. It is therefore reasonable to reduce the complexity of the network as far as possible. As waiting times can occur only at stations where at least two lines cross and passengers can change lines, we may drop all other 'non-changing' stations from the model. Let S, S' be two changing stations on a line with only non-changing stations $\tilde{S}_1, \ldots, \tilde{S}_n$ between them. Then all sections $(\tilde{S}_i, \tilde{S}_{i+1})$ between two neighbouring non-changing stations are aggregated to form one new 'super-section' (S, S'). The catalogue of possible investments for this super-section has to be built from the aggregated ones by carefully folding the cost functions $c_{(\tilde{S}_i,\tilde{S}_{i+1})}(\cdot)$ into a new function $c_{(S,S')}(\cdot)$. (See Engelhardt-Funke and Kolonko, 2001; Nachtigall and Voget 1997, for more details on this point.)

In the next Section we introduce random delays into our model.

3 Timetables under random delays

The real operation of trains will hardly ever be in complete accordance with the timetable due to unavoidable small disturbances like technical failures, construction sites along the tracks or delays at stations during boarding. Note that we do not consider large deviations from the timetables caused for example, by heavy accidents.

Delays may accumulate during a train journey and may be propagated to connecting trains of other lines. Modelling the propagation of delays is a most complex task for which we use computer simulation. For the accumulation of delays along a line, however, we can give an approximate analytical solution which may then be used to speed up the simulation.

3.1 The distribution of delays

We want to derive an analytical expression for the distribution of the resulting delay at the end of super-sections in the aggregated network (see Section 2 above). In particular, we have to model how delays may be reduced. Here we use a slightly simplified version of the mechanisms actually applied in the German railway system.

We assume that delays on sections are caused by stops at random places of random duration and that delays at stations are additional stopping times of random length.

There are two types of time buffers to reduce these delays. First, trains may increase their speed as soon as they are delayed to make up for the delay. In other words, the scheduled running time $\delta(S, S')$ contains a small time buffer that may be used to reduce delays before the next station is reached. In our simulation experiments we assumed that the length of this buffer is fixed for each super-section (S, S') independently of the timetable. Another time buffer may be contained in the stopping times at stations, as the departure time $\pi(S, L)$ scheduled in timetable T may be larger than the arrival time plus the necessary transfer time. This additional stopping time is denoted $w_T(S, L, L)$ (see equation 1). Both these types of buffers reduce the delay of a single train.

For a further simplification, we assume that along a super-section all delays are caused by stops of independent and identically distributed (iid.) random length and that the distances between these stops also form an iid. sequence. This is justified if the number of non-changing stations is large and if the delays at stations are more important than the delays acquired during driving (as is the case in practice).

Then the accumulation and reduction of delays along the super-section can be modelled as a queueing system: the occurrence of a single delay can be viewed as the arrival of a customer that brings its duration as workload to the server (train). The service consists of reducing the delay by increasing the speed of the train. Queueing theory gives expressions of the workload lying ahead of the server (the so-called virtual waiting time), which in our case is the remaining accumulated delay. Let $D(S, S')$ denote this resulting delay of a train at the end of a super-section (S, S') provided the train started as scheduled in S. See Fig. 3 for a typical picture.

If the distances between consecutive disturbances and the duration of the stops are iid. exponentially distributed with parameters λ and μ respectively, and if all random variables are

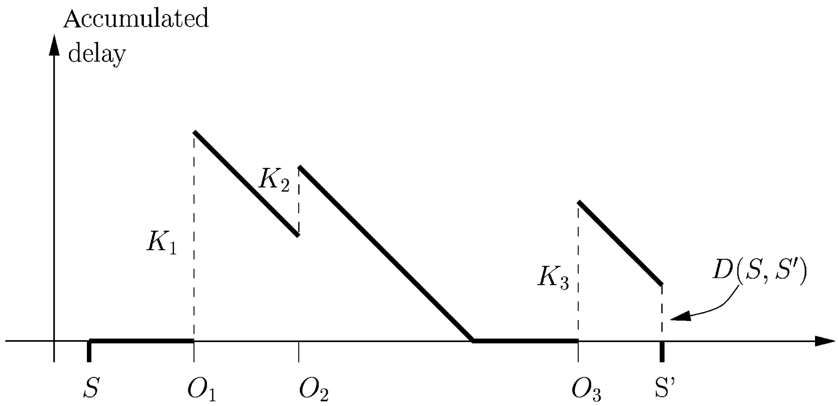

Fig. 3. On the journey from S to S', random stops of duration K_1, K_2, K_3 occur at O_1, O_2, O_3 resulting in a delay $D(S, S')$.

independent from each other, then $D(S, S')$ has approximately (for long super-sections (S, S')) the following distribution

$$\mathbf{P}(D(S,S') \leqslant t) = \begin{cases} 1 - \frac{\lambda}{c\mu} & \text{if } t = 0 \\ \frac{\lambda}{c\mu}(1 - \mathrm{e}^{-t(\mu - \lambda/c)}) & \text{if } t > 0 \end{cases}. \tag{3}$$

Here,

$$c := \frac{1}{\text{normal speed}} - \frac{1}{\text{maximal speed}} \tag{4}$$

is the rate at which delays can be reduced during driving. Note that $1 - \lambda/c\mu$ is the probability of arriving in time and $1 - e^{-t(\mu - \lambda/c)}$ is the conditional distribution function of the delay given the train is delayed. A similar result holds under more general assumptions on the process of disturbances (see Engelhardt-Funke and Kolonko, 2000; Engelhardt-Funke, 2002, for details). These analytical results are in good accordance with empirical data. (See Herrmann, 1996 and Mühlhans, 1990, for the German railway system.)

Next we want to include an initial delay $B(S, L)$ of a train of line L when leaving station S for the super-section (S, S'). In the queueing context, it can be reduced at rate c whenever the server is ‘idle’, i.e. there is no other delay. From queueing theory we know that asymptotically, this is the case for a fraction of $1 - \frac{\lambda}{c\mu}$ of the running time. Therefore, at the end of the super-section (S, S') the initial delay $B(S, L)$ has been reduced to

$$\left[B(S,L) - cs\left(1 - \frac{\lambda}{c\mu}\right)\right]^+ \tag{5}$$

where s is the length of (S, S') and $[\cdot]^+$ denotes the positive part. As a result we obtain the following approximate expression for the total delay of a train of line L at the end of super-section (S, S')

$$A(S', L) := D(S, S') + \left[B(S,L) - cs\left(1 - \frac{\lambda}{c\mu}\right)\right]^+. \tag{6}$$

Note that in our model, the super-section (S, S') is determined by its terminal S' and the line L. Equation (6) allows to treat the super-sections in the aggregated model as one object during simulation without having to keep track of all disturbances. The determination of the initial delay $B(S, L)$ at changing stations has to take into consideration all delays propagated from feeder trains.

In our simulation model we assumed that the intensity and frequency of the single delays emerging along super-sections and at stations are not affected by the timetable or investments. This means in particular, that even if the running times on (S, S') are shortened by an investment, we assume that the rate c given by (4) and the length of the buffer sc for that super-section remain constant. This may seem a little unrealistic but simplifies simulation as the underlying data do not have to be adapted to each new timetable. However, timetable and investments do have an impact on the way delays are reduced by the time buffers $w_T(S, L, L')$.

3.2 The actual waiting time

To reduce the propagation of delays from one train to another, time buffers $w_T(S, L, L')$ at changing connections (S, L, L') are needed. This is the (scheduled) time between the arrival of line L at station S and the departure of line L' from S minus the necessary transfer time. To further limit the absolute amount of propagated delay, there is a so-called 'connection control limit' κ. This is the maximal time a connecting train has to wait for its delayed feeder train, possibly depending on the type (importance) of the connections.

The time buffers $w_T(S, L, L')$ determine the cost function 'scheduled waiting time' $W(T)$ as given by (2.1). Minimisation of $W(T)$ will decrease these buffers and may lead to an unstable system as the actual waiting time of passengers increases due to missed connections. Therefore, optimisation of timetables should also take into account the resulting actual waiting time of passengers.

There are different ways to formally define such a cost function. In Goverde (2002) the exact waiting time of a passenger is taken, that is, the time from the arrival (on a possibly delayed train) until the (possibly delayed) departure. This may result in actual waiting times that are less than the scheduled ones, for example, if for a connection with a large scheduled waiting time the arriving train is late but just reaches the connecting train (which may be delayed itself). In some cases this may result in a reward for delays, although the total travel time is unaffected and passengers arrive at the station later than scheduled.

Therefore we chose a different approach that also enforces a reliable operation of the network close to the timetable. We define $v_T(S, L, L', a)$, the (local) waiting time for a passenger arriving at station S on line L waiting for the connecting line L' if L arrives a minutes late:

$$v_T(S, L, L', a) := \begin{cases} w_T(S, L, L') & \text{if } a < w_T(S, L, L') + \kappa \\ \tau & \text{if } a \geqslant w_T(S, L, L') + \kappa \end{cases}, \tag{7}$$

where κ is the connection control limit. This means that if the delayed passenger on line L still reaches the scheduled connection, the whole scheduled waiting time $w_T(S, L, L')$ is taken as his waiting time, instead of the smaller exact waiting time $[w_T(S, L, L') - a]^+$. If the delay a is larger than the scheduled time buffer $w_T(S, L, L')$ plus the connection control limit κ, we assume that the connecting train of line L' is missed and the waiting time is set to τ. Note that here we do not take into account that L' may be late too, so that the individual passenger may still reach the connection and that if he misses it, he may not have to wait for a complete period τ. The value τ serves as a penalty for not arriving on time. A possible delay in the departure of L' is not taken into account here but it may affect the waiting time of passengers arriving on L' at the next station.

The formal definition of the mean actual waiting time $V(T)$ over the whole network is a little involved. We define it as the sum over all connections of the expectation of the average local waiting times:

$$V(T) := \sum_{(S,L,L')} \mathrm{E}\left(\frac{1}{N} \sum_{k=1}^{N} v_T(S, L, L', A_k(S, L)) \right) \cdot g(S, L, L') \tag{8}$$

The expectation is taken with respect to the distribution of the random delay $A_k(S, L)$ of the k-th train of line L arriving at station S, see (6). Note that because of the random nature of the disturbances acting on each train individually, the strict periodicity of the operation is lost and we have to take the average of all trains of line L. As an approximation, N is the number of trains that run during a reasonable time interval, e.g. 24 hours. $g(S, L, L')$ are the same weights as in equation 1.

As was mentioned before, we are not able to compute the propagated delay at changing stations exactly. Hence, the distribution of $A_k(S, L)$ and the expectation in equation 8 cannot be determined analytically. Instead we estimate $V(T)$ by a computer simulation of the reduced model making, use of equations 3 and 4. For this estimation, a 24-hour period of operation is simulated repeatedly until the averages become stable. The result is taken as the cost function value 'mean actual waiting time'.

4 Optimisation by enhanced genetic algorithms

4.1 Multi-criteria genetic algorithms

Multi-criteria optimisation problems are well suited for an approximate solution by genetic algorithms (see e.g. Zitzler *et al.*, 2001, for a recent overview). We developed a new variant of adaptive control of the optimisation process, which incorporates and improves several of the classical multi-criteria approaches of genetic algorithms.

Genetic algorithms work with a 'population' of solutions, which are timetables in our application. The starting population is formed by timetables with random departure times and random running times. This random population evolves through a fixed number of generations. In each generation, the population is first enlarged by offspring, which are produced from the present population using genetic operators such as cross-over and mutation. From this enlarged set of solutions the next population is selected where solutions with low cost function values are preferred in some sense. The solutions belonging to the lower envelope of the cost function values of the populations are used as an approximation to the Pareto-optimal set of timetables.

In our case, the genetic operators are simple to design as any lists of departure times $0 \leqslant \pi(S, L) < \tau$ and running times $t_0(S, S') \leqslant \delta(S, S') \leqslant t_1(S, S')$ yield a feasible timetable (here $t_0(S, S')$ is the minimal running time for section (S, S')). Therefore mutation may just change a departure or running time randomly within these limits. For cross-over, we select two timetables from the present population at random, and then mix the two lists of departure times or the two lists of running times from the 'parents'.

The crucial point is the selection of the timetables for the next generation based on the cost function values. Our system adapts the selection to the state of the present population. As this part of the algorithm is described in detail in Kolonko and Voget (1998) we only give a short description here.

The random starting population P_0 is regarded as a random sample from the solution space. The cost functions are normed using mean and variation from this sample. The enclosing rectangle R_P of the cost function values of a population P is taken as the state of the population. We postulate that the ideal evolution should lead the centre of R_P along a line that connects the centre R_{P_0} of the starting population with the origin. This would develop solutions which have low investment costs *and* low waiting times.

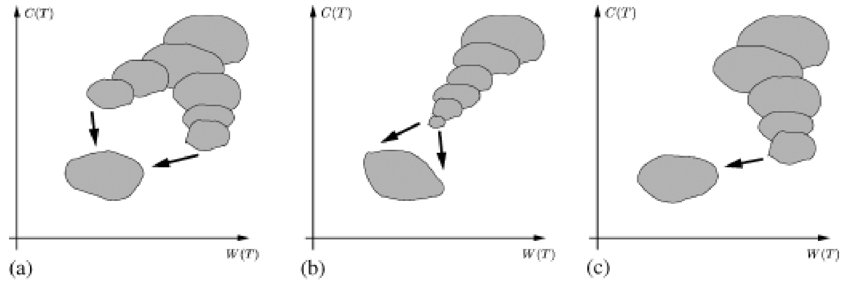

Fig. 4. The type of selection is adapted to the state of the optimisation process.

During the optimisation process the population (or rather its cost function value cloud) often has an undesirable shape as is indicated in Fig. 4 for the two cost functions $W(T)$, $C(T)$. The cloud may become disconnected as in Fig. 4(a), lose its variation as in (b) or lose the balance between the two cost criteria as in (c). We use the enclosing rectangle R_P of the present population as an indicator of such situations. Then the type of selection for the next generation is chosen in order to force the population into the 'right' direction. In case (a), a selection according to the so-called Pareto-rank of the solution (see Fonseca and Fleming, 1993) is used. It tends to contract the population as indicated by the black arrows. This is a disadvantage if it is applied constantly, the population will then develop as in Fig. 4(b). If situation (b) occurs in our algorithm, we apply the selection procedure 'VEGA' from Schaffer (1985) which if used solely leads to separated sub-populations as in (a). If one of the cost criteria is not properly minimised as in (c), we simply restrict selection to this cost function alone. Thus, the selection is adapted for each new generation leading to a rather uniform exploitation of the most important part of the Pareto front.

4.2 'Avant-garde breeding'

To speed up the optimisation process, we introduced a new concept in which the optimisation effort is not always spread evenly among the elements of the population. From time to time, a single solution is picked at random from the present population and an intensive local improvement procedure is applied to it. The result is a new solution which in most cases is much better than the rest of the population, runs ahead of it in cost function space. This avant-garde solution is put back into the population and offspring are produced for the next population by the standard process. Typically, within a few generations the population catches up with the avant-garde (see Fig. 5 below). The other solutions inherit the structure of a good solution very quickly once it is present in the population. This process of sending out an avant-garde solution and then catching up with it is repeated every K_{Avant} generations and the intensity of the local search (e.g. number of trials) is increased each time.

For the local improvement in the avant-garde step, simulated annealing turned out to be most efficient. In simulated annealing, a random neighbour T' of the present solution T is produced by applying the mutation operator to T. T' is accepted as the new solution if it is better than T and

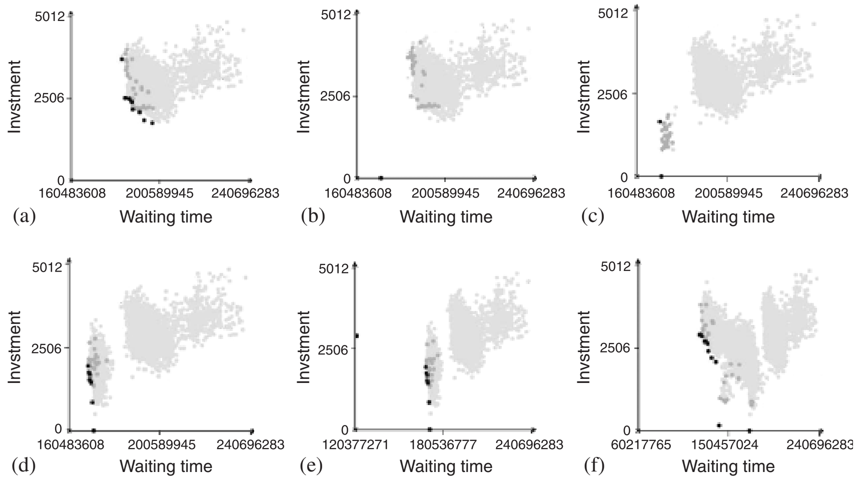

Fig. 5. The evolution with an avant-garde. Note that the x-axis is permanently rescaled.

with a certain probability even if it is worse than T (uphill climbing). To be able to compare the quality of any pair of timetables T, T' we need a one-dimensional criterion. Following an idea of Ishibuchi and Murata (1996), we use a linear combination of the original cost functions $W(T)$, $C(T)$ and $V(T)$ where the linear coefficients (and thus the direction of the selection pressure) are selected at random for each new simulated annealing run. The 'temperature' schedule for the annealing that controls the acceptance probability, is adapted to the course of the optimisation (see Kolonko, 1999; Kolonko and Tran, 1997, for more details on this type of simulated annealing).

Figure 5 shows the effect of the avant-garde in some screen-shots. It is from a network of a regional train service in the federal state of Sachsen-Anhalt in Germany, which has 467 stations and 295 lines. Here, the dark grey dots represents the present population (its cost function values), the black dots are the Pareto best timetables found so far, and the light grey dots are the old solutions which have not survived. Figure 5(a) shows the population after 200 generations with the adaptive genetic algorithm. Then the first avant-garde solution is formed which is the isolated black dot in Fig. 5(b). Only five generations later, the whole population has inherited good properties of the avant-garde as is shown in Fig. 5(c). Figure 5(d) is the population after another 100 generations, just before the second avant-garde is formed which appears in (e). Figure 5(f) shows the population 200 generations later. It clearly shows how the population has skipped large parts of the solution space towards better solutions.

4.3 Simulation with correlated cost functions

Genetic algorithms and simulated annealing produce a huge number of tentative solutions, which are thrown away after evaluation of the cost functions. Evaluating $V(T)$ requires the repeated simulation of the whole network and slows down the optimisation process considerably, for example, if $V(T)$ is used in the avant-garde forming process with simulated annealing.

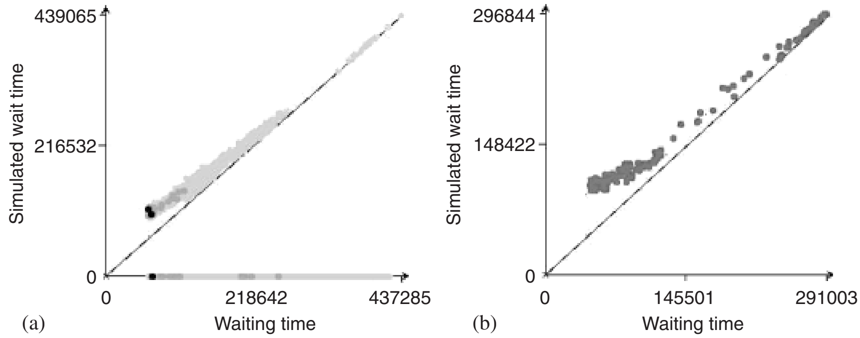

Fig. 6. The correlation between scheduled and mean actual waiting time degrades with increasing synchronisation of trains. (a) shows a single optimisation run, (b) shows solutions collected from several runs.

However, as was mentioned above, one could expect that the values of the scheduled and the estimated mean waiting time, $W(T)$ and $V(T)$, are very close as long as the time buffers at stations are large enough, that is, as long as $W(T)$ is large. But then $V(T)$ does not add much to the selection of timetables for future generations. We therefore switch off the expensive cost function $V(T)$ for the first generation and calculate $W(T)$ and $C(T)$ only. Then, after a number of generations we simulate $V(T)$ for all timetables in the population and calculate the empirical correlation of $W(T)$ and $V(T)$ in this sample. If the correlation is above a fixed threshold, simulation is switched off again for the next K_{Simul} generations. If it is below, then the value of K_{Simul} is decreased and simulation of the population takes place more often.

Figure 6 shows the result of this approach for (a slightly simplified version of) the German ICE network. In (a) the ($W(T)$, $V(T)$) pairs for 10,000 generations are shown. When simulation is switched off, we put $V(T) := 0$ producing points on the x-axis. The timetables with simulated $V(T)$ values (every $K_{Simul} = 200$ generations) produce dots along the diagonal, quite closely at the beginning, that is, for high scheduled waiting time $W(T)$ and less closely for better timetables. For this simulation we used the time buffers on super-sections that were fixed to 5% of the initial running time $t_0(S, S'), \lambda/(c\mu) \equiv 0.25$ and $\mu = 0.66$. This means that with probability 0.75 there will be no delays on a super-section. If the train is delayed, then its conditional mean delay is $(\mu - \lambda/c)^{-1} = 2$ minutes. Moreover, the additional delays at changing stations are exponentially distributed with mean 2.5 minutes. The 10,000 generations with the above described 'smart simulation' took about 4 minutes on a 600 MHz PC.

Figure 6(b) shows a set of solutions for the same network which were obtained from several longer optimisation runs. The $V(T)$ values were computed by simulation afterwards. As more solutions with small scheduled waiting time $W(T)$ are present, the change in the stability becomes more visible. Again the bad solutions (with respect to $W(T)$) have high correlation whereas the better ones show that there is in fact a point Γ, where the estimated mean waiting time decreases rather slowly and the variation along the y-axis increases. This means that from this point onwards we found more timetables that have (almost) identical scheduled waiting time $W(T)$ but differ in their $V(T)$ values. This is an empirical support of the conjecture about the point Γ as stated in the introduction.

5 Conclusions

Our railway optimisation has three main goals: (1) it produces timetables that are (approximately) Pareto optimal with respect to complex cost criteria, (2) it yields cost-benefit analyses between the cost functions, for example, between investments into the network and the scheduled waiting time, and (3) it allows the estimation of the 'turning point' for the synchronisation of the timetables from which on a mere reduction of waiting time decreases the stability of the timetables under random delays.

To obtain these results, special evolutionary algorithms had to be developed. First, we introduced the adaptive selection scheme, which guarantees a broad approximation of the Pareto front. The procedure is accelerated by allowing single avant-garde solutions which make the population jump through the solution space. A third concept allows inclusion of the simulated cost function $V(T)$ into the optimisation at a reduced expense by exploiting its correlation with the simpler function $W(T)$ in large portions of the solution space. The border of the regions of high correlation between $W(T)$ and $V(T)$ is of interest in itself as it describes a stability property of the network.

Future work will concentrate on the further improvement of the optimisation procedure and a closer examination of the point Γ for different network instances. We shall also try to include investments that not only accelerate the trains but that also allow reduction of the frequency or intensity of random disturbances making the running and stopping times of trains more stable.

Acknowledgement

The authors wish to thank the referee for a number of valuable suggestions.

References

Engelhardt-Funke, O., Kolonko, M., 2000. Optimal Time-Tables: Modelling Stochastic Pertubations. Proc. 7. International Workshop on Project Management and Scheduling (PMS 2000), Osnabrück, pp. 125–127.

Engelhardt-Funke, O., Kolonko, M., 2001. Cost-Benefit-Analysis of Investments into Railway Networks with Randomly Perturbed Operations. In: Voß, S., Daduna, J. R. (Eds.) *Computer-Aided Scheduling of Public Transport (CASPT 2000). Lecture Notes in Economics and Mathematical Systems 505*, Springer, pp. 442–459.

Engelhardt-Funke, O., 2002. Stochastische Modellierung und Simulation von Verspätungen in Verkehrsnetzen für die Anwendung bei der Fahrplanoptimierung, Doctoral Thesis, Technical University Clausthal, Germany.

Fonseca, C.M., Fleming, P.J., 1993. Genetic Algorithms for Multiobjective Optimization: Formulation, Discussion and Generalization, International Conference on Genetic Algorithms, pp. 416–423.

Goverde, R., 1998. Synchronization Control of Scheduled Train Services to Minimize Passenger Waiting Time. Proc. of the 4th TRAIL Annual Congress, Part 2, TRAIL Research School, Delft, pp. 1–35.

Goverde, R., 2002. Performance Evaluation of Network Timetables Using PETER. In: Allan, J., Anderson, E., Brebbia, C. A., Hill, R., Sciutoo, G. Sone, S. *Computers in Railways VIII*, WIT Press Southhampton.

Herrmann, U., 1996. Untersuchung zur Verspätungsentwicklung von Fernreisezügen auf der Datengrundlage der RZÜ Frankfurt am Main. Doctoral Thesis, TH Darmstadt.

Ishibuchi, H., Murata, T., 1996. Multi-objective genetic local search algorithm. Proceedings of 1996 IEEE International Conference on Evolutionary Computation (ICEC '96), May 20–22, Nagoya University, Japan, Piscataway, NJ IEEE Service Center, pp. 119–124.

Kolonko, M., 1999. Some new results on simulated annealing applied to the job shop scheduling problem. *European Journal of Operational Research*, 113, 123–136.

Kolonko, M., Tran, M.T., 1997. Convergence of Simulated Annealing with Feedback Temperature Schedules. Prob. in the Engineering and Informational Sciences, 11, 279–304.

Kolonko, M., Voget, S., 1998. Multidimensional optimization using fuzzy genetic alorithms. *Journal of Heuristics*, 4, pp. 221–244.

Mühlhans, E., 1990. Berechnung der-verspätungsentwicklung bei zugfahrten. *Eisenbahntechnische Rundschau*, 39, 465–468.

Nachtigall, K., 1996. Periodic network optimization with different arc frequencies. *Discrete Applied Mathematics*, 69, 1–17.

Nachtigall, K., 1998. Periodic Network Optimization and Fixed Interval Timetables. Habilitationsschrift, Universität Hildesheim

Nachtigall, K., Voget, S., 1997. Minimizing waiting times in integrated fixed intervall timetables by upgrading railway tracks. *European Journal of Operations Research*, 103, 610–627.

Schaffer, J.D., 1985. Multiple Objective Optimization with Vector Evaluated Genetic Algorithms, Genetic Algorithms and their Applications: Proceedings of the First International Conference Genetic Algorithms, pp. 93–100.

Weigand, W., 1981. Verspätupgsübertragung in fernverkehrsnetzen. *Eisenbahntech-nische Rundschau*, 30, 915–919.

Zimmermann, U., Lindner, T., 2000. *Train Schedule Optimization in Public Transportation*. Available from www.math.tubs.de/mo/research/preprints.html

Zitzler, E., Deb, K., Thiele, L., *et al.* 2001. *Evolutionary Multi-Criterion Optimization, Lecture Notes in Computer Science 1993*, Springer Verlag.