Expected Passenger Travel Time for Train Schedule Evaluation and Optimization

Peter Sels [1,2,3], Thijs Dewilde[1], Dirk Cattrysse[1], Pieter Vansteenwegen[1]
[1]Katholieke Universiteit Leuven,
Centre for Industrial Management/Traffic & Infrastructure,
Celestijnenlaan 300a, 3001 Heverlee, Belgium

[2]Logically Yours BVBA,
Plankenbergstraat 112 bus L7, 2100 Antwerp, Belgium
e-mail: sels.peter@gmail.com, corresponding author

[3]Infrabel, Department of Rail Access,
Fonsnylaan 13, 1060 Brussels, Belgium

Abstract
When evaluating and optimizing the national railway timetable for all passenger trains in Belgium, we use the criterium of expected total passenger travel time in practice. During evaluation, we measure this, during optimization, we try to minimize this. The focus in this paper is to analytically derive the total passenger time as a stochastic function of the schedule. In a previous paper [9], we already derived the necessary local passenger flows from ticket sales data. These flows will serve as fixed weights in this new objective function.

We start from measured delays in the current schedule and assume that they can be described well by a negative exponential distribution. In the case of constructing an optimized schedule, we assume delays will also occur in the new schedule according to the same distribution. We believe this is a conservative assumption, since the optimized schedule will be more robust against delays.

We then decompose the global schedule of actions into local actions or pairs of subsequent actions, and derive probabilities and objective functions on the corresponding one or two local actions only. We sum these local functions to obtain a global objective function. This then becomes the schedules' global quality criterium. We linearize this and use the result as the objective function of a linear programming based optimizer. With this objective function, we optimized the train schedule for the Belgian railways, which proves the method is scalable to large and complex real-life problems.

Keywords
Railway Timetabling, Expected Passenger Time, Objective Function, Stochastic Optimization

1 Introduction

We consider total passenger travel time in practice as the most important criterium for judging the quality of a train schedule [1]. In such a objective function, we need to use passenger numbers [9] instead of trains as weight factors. In evaluating an existing schedule we should check that this objective function stays low and that few or no planned transfers are missed frequently, especially if many people are transferring. In optimizing a schedule, we should guarantee these properties. Not only transfers, but also ride and dwell actions should be planned with supplements that are neither too small, such that they are unable to absorb any practical delay, nor too big, such that they create unnecessary idle time in practice. In this paper, we analytically derive the stochastic expected passenger time on each location throughout the network, so that it can be used for evaluating if a specific supplement is well chosen or not. The global function of total expected passenger time is derived from this, which directly gives a practical quality criterium for the whole schedule. This global function (or some convex, linear approximation of it) will also be used as the objective function for a (linear) solver when optimizing.

2 Related Research Overview and Comparison

Concerning linear modeling of a train service, some theoretical research has been carried out already [3, 4, 5, 6, 8, 10, 11]. The first schedule that has been mathematically calculated, be it only for feasibility and not for optimality, was the one for the Netherlands, and is due to Kroon et al. [5]. Whether one of these schedules was actually put in practice is unclear. Christian Liebchen states [6] to have produced the first schedule that was mathematically calculated and then also put into practice. This included both schedule feasibility as well as optimization of some selected transfers and concerned the Berlin Underground, containing 37 train relations. We go further here, by handling a much larger network - all periodic passenger trains in Belgium - and more importantly, using an all-embracing optimization objective function: passenger time in practice, which automatically considers all *potential transfers* [9]. Since we minimize the objective function, the resulting timetable will be robust, and this, in a natural, consistent and balanced way, across all ride, dwell and transfer actions. The idea of using expected passenger time as a objective function to minimize is based on research from Vansteenwegen et al.[12, 13] and Dewilde et al. [1]. The great power of this objective function is that it gives an automatic and sensible trade-off between a speedy yet robust service, where buffers and supplements are neither too large nor too small. It also weighs and balances all minutes spent in the network fairly across all passengers. For optimization of a schedule, apart from a objective function, of course, a large number of hard constraints is necessary. We implemented and use all of these in practice, but they are not discussed in this paper. The sole focus of this paper is the objective function.

3 Decomposing the Train Service Network into Actions with Local Passenger Flows

We model each train trip as consisting of five types of actions. Figure 1 shows that each action can be represented by a rectangle with a width, indicating the planned duration of this activity and the height, indicating the number of people participating in that action. A train trip can be seen as a sequence of mainly blue *train ride* and yellow *train dwell actions*. Additionally, there are three types of *passenger actions*: green colored *source actions*, which represent people getting on the train, orange colored *transfer actions* representing passengers walking from one train to another, and red *sink actions*, representing people arriving at their destination.

Like Meng [7], Goverde [2], and Vansteenwegen et al. [12, 13], we suppose a general negative exponential delay distribution for the duration of each of the actions: ride, dwell and transfer. Source and sink actions can be considered as having a zero expected delay. The duration of source and sink actions does not influence the planning. They are only present in Figure 1 to indicate per train, the number of passengers entering or exiting the station at each of its stops. The ride, dwell and transfer actions all have a given *minimal duration*. After each of these three actions, a time supplement is added. Figure 1 shows these in a lighter color shade of blue, yellow and orange for the respective actions. These supplements contain the time to buffer possible delays of the previous action(s), but can also be much bigger than that in case the subsequent action has to start much later to allow departing or transferring passengers to enter the train considered. The durations of these buffers or filler actions are called the *supplement times*, and are the only relevant time parameters present in our objective function: the total expected passenger travel time. The optimization problem we consider here, in the so called *retiming phase* [9], is to determine a supplement time for each action, so that the total *expected* passenger travel time is minimal. Note that Figure 1 shows the *planned* schedule and that actual delays on actions can be smaller than their shown supplements, in which case there is slack time, but that when actual delays are bigger than the supplements, delay will propagate, yet hopefully be absorbed by supplements later.

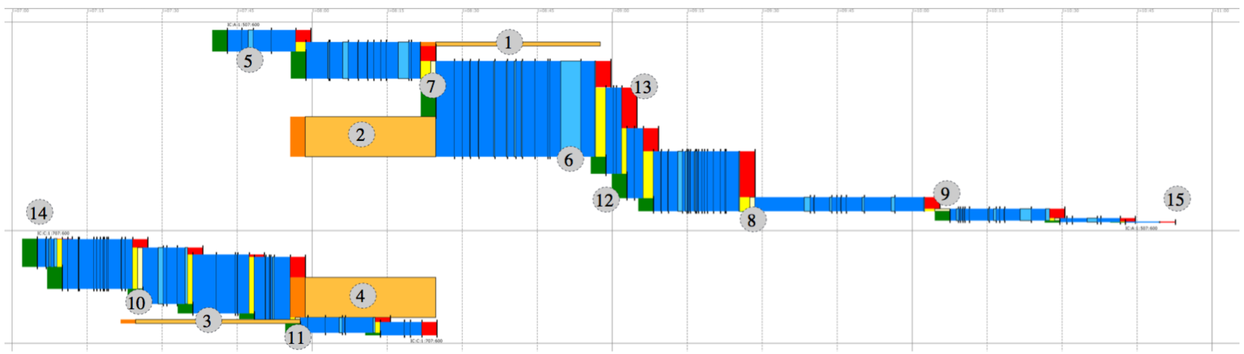
The top couple of trains in Figure 1 are taken from the current Belgian planning, and the bottom couple of trains show the time optimized version according to the objective function we will derive here. Some corresponding actions between both schedules are indicated with the same numbers in grey circles. Note that there are two passenger transfers between the two trains (Numbers 1=3 and 2=4) and that this influences the relative time alignment between the two trains. We also suppose that each of the trains occurs every hour. It must be noted that the current expected delay distributions on each action currently are realistic, but fictive. In the end, these will be replaced by distributions based on actual measurements.

Whether the bottom couple of trains takes more or less *planned* passenger time is not so relevant. However, in operation, taking all statistical occurrences of these train trips together, for all the actual numbers of passengers, the *expected* passenger time in practice will be lower than for the original top train couple. This is the case, because the bottom couple of trains will be more robust against operational delays on its actions. So, this global objective function allows us to evaluate as well as optimize the timing for trains. To derive this function, we will now start with introducing the level of action pair types.

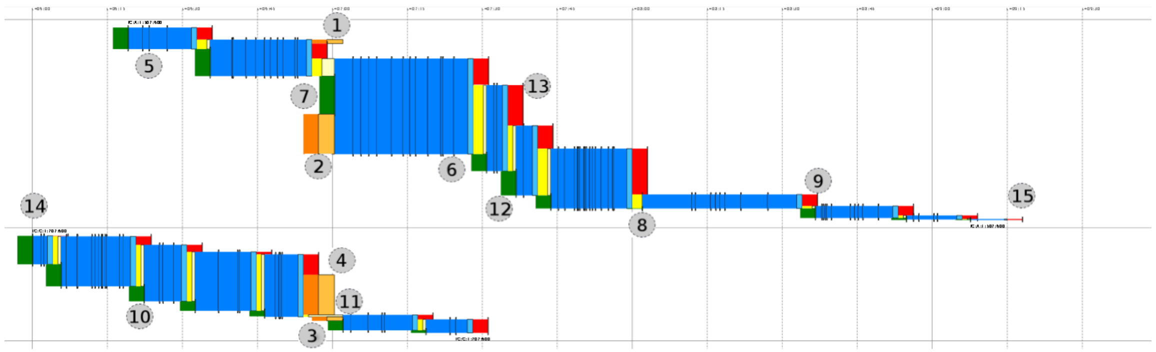
4 Regrouping per Action Pair Type

Using a bottom up approach, we now cut up a train schedule at ride departure times. We are then left with separated parts, consisting of pairs of subsequent actions.

Similar to Vansteenwegen et al. [12, 13] and Dewilde et al. [1], we call passengers who experience the successive actions: ride and dwell, *through passengers*. *Transfer passengers* are the ones taking a ride and then a transfer action to the next ride action. *Departing passengers* follow a source action to a ride action. *Arriving passengers* come from a ride action and go to a sink action. We call these four kinds of passenger types, the *passenger flow*



(a) Original Schedule



(b) Optimized Version

Figure 1: Two timetables displaying *planned* time for a pair of trains with transfers between them: Original and Optimized. Horizontal dimensions are action durations. Vertical dimensions are passenger numbers participating in these actions. Ride actions are blue, dwell actions yellow, transfer actions orange, departing actions green and arriving actions red. Supplements or slack time are indicated in the lighter shade of the preceding action they belong to. Circled numbers correspond between schedules. In the original schedule (a) , the supplement of transfer 1 (08:25 to 8:57) = transfer 3 (7:25 to 7u57) takes 33 minutes, while in the optimized schedule (b), the supplement of transfer 1 (06:59 to 7:02) = transfer 3 (06:59 to 7:02) takes only 3 minutes which is more efficient and still robust. For the opposite transfers, in the original schedule (a), transfer 2 (7:58 to 8u25) = transfer 4 (7:58 to 8u25) has 27 minutes supplement, while in the optimzed schedule (b) transfer 2 (6:58 to 7:01) = transfer 4 (6:58 to 7:01) only takes 3 minutes supplement. Again , this is more efficient and still robust. Note that since all trains are repeated periodically every hour, also the transfers between them occur every hour. For blocks of consecutive ride-dwell action pairs that have no interaction with other trains, nor entering, transferring or exiting passengers, the light blue colored supplement occurs at the end, which is indeed the best place to buffer against delays.

types. We summarize these and their corresponding action types in table 1. In conclusion, there are 5 types of actions which only occur in 4 types of pairs of subsequent actions.

Table 1: Four Passenger Flow Types and Corresponding Action Sequences

Passenger Flow Type	Action a	Action b
Departing	(Source)	Ride
Through	Ride	Dwell
Transfer	Ride	Transfer
Arriving	Ride	(Sink)

It must be clear that during their journey, a single departing passenger, becomes a through passenger, can become a transfer passenger, a through passenger again and finally always becomes an arriving passenger. Each passengers' flow type is only locally valid, for each two successive actions.

So we decomposed the total passenger flows on all trains into actions, to then recompose them into pairs of subsequent actions, corresponding to a particular passenger flow type. Each type of passenger is interested in having a minimal *expected* travel time for all the action pairs he undergoes.

For *through* and *transfer passengers*, the exact trade-off will be dictated by minimization of the expected passenger travel time. This depends on two actions and their delays, so for this, convolution of the delay distributions of both actions will be necessary.

For *arriving passengers*, only the last ride action matters for the delay on this last part of their journey.

For *departing passengers*, we will reasonably suppose their travel time starts counting at the planned departure time. Indeed, their choice to potentially arrive earlier to avoid missing their departure train, is independent of any particular time we will assign to this departure event. So their relevant departure delay is the delay of the previous ride and dwell action pair of their train, when such a pair exists. If such a pair does not exist, the departing passengers are catching a train at its very first ride action, and we suppose there is no possible delay nor associated passenger time cost.

For each of the four passenger flow type cases, a corresponding pair of action types, makes up the time in between two subsequent ride departure times. They correspond to the times we want to evaluate or optimize, weighted with the net corresponding passenger flow number on this action pair.

We know that, in reality, delays can be accumulated over a full train journey from beginning to end. This is often the case when at places where delays often occur, no time supplements are inserted to absorb these delays. In this research, we try to do exactly this: calculate supplements that absorb enough delay locally, not globally, yet not too much, since we still want to maintain a speedy service. In this way, we avoid that delays propagate in most cases. On top of this, for each train, we will accumulate sequences of *undisturbed blocks* of ride-dwell-pairs up to the point where either the passenger numbers change, or another train enters or exits the same track. For each such block, the measured delay is accumulated and the (accumulated) supplements are then correctly placed at the end of the block only, as can be seen in Figure 1(b) by the position of the light blue rectangles in the optimized schedule.

5 Local Objective Function Derivation

We now analytically derive the expected passenger time in practice of one action and of an action pair, based on their delay distributions.

5.1 One Single Action Only

Fitting Model Delay Distribution to Delays
We first consider an action a or process with a minimum time m and a probabilistic delay variable x as described by Vansteenwegen et al. [12, 13]. We can measure the occurrences of delays for this process and *fit* a normalized negative exponential distribution on these measurements. The probability $p_a(x)$ on a delay x can then be expressed as

$$p_a(x) = ae^{-ax}. \tag{1}$$

Note that (1) is *normalized* which is required for a probability distribution. Indeed

$$\int_0^\infty p_a(x)dx = \int_0^\infty ae^{-ax} = -e^{-ax}\big|_0^1 = (-0) - (-1) = 1. \tag{2}$$

The fitting is done by choosing the single negative exponential distribution that has the same expected value $\overline{d_a}$ as the originally measured discrete delay sample histogram. For action a, for the measured discrete histogram, considering samples collected in N groups, per value x_i at index i and respective number of occurrences h_i, this gives

$$\overline{d_a} = \frac{\sum_0^{N-1} h_i x_i}{\sum_0^{N-1} h_i}. \tag{3}$$

For the continuous distribution the equivalent expected value is [1]

$$\overline{c_a} = \int_0^\infty xae^{-ax} = \frac{1}{a}, \tag{4}$$

Our fitting criterium is that the expected value of the fitted distribution must equal the expected value of the original distribution. This corresponds to

$$\overline{c_a} = \overline{d_a}, \tag{5}$$

which, considering (3) and (4) gives

$$a = \frac{\sum_0^{N-1} h_i}{\sum_0^{N-1} h_i x_i}. \tag{6}$$

In Time Probability
For this section, to ease the derivation, we ignore the part of the time cost that is related to the minimum time m for an activity. This can be done because this part does not depend on any supplement choice and as such will not influence comparative evaluation nor optimization.

For now, we suppose our activity a is to be followed by another activity b that, according to the planning, starts at time D_0 later than the end of the current activity a. So we suppose

[1] Whenever a calculus step is not trivial, the PDF version of this paper allows the reader to click right of the respective equal sign, which will send the formula in Mathematica notation to www.wolframalpha.com and his invoked browser will then display the result.

that D_0 is the time supplement we will decide to add between activities a and b. Activities a and b are the pairs of activities discussed above, D_0 is the supplement between these.

We have a possibility that we end our current activity within time supplement D_0. We calculate the *in time probability* that this happens as

$$p_{a,x\leq D_0}(D_0) = \int_0^{D_0} p_a(x)dx = \int_0^{D_0} ae^{-ax}dx = 1 - e^{-aD_0}. \tag{7}$$

In Time Cost
Since all time units are supposed to be equally weighted, the corresponding unit time cost of this supplement D_0 is

$$UC_{a,x\leq D_0}(D_0) = \int_0^{D_0} D_0 p_a(x)dx = p_{a,x\leq D_0}(D_0) * D_0 = (1 - e^{-aD_0})D_0. \tag{8}$$

We call $UC_{a,x\leq D_0}$in (8) the *in time cost* of a supplement D_0. This cost is called *unit cost* because this is the cost for one passenger and for each time unit experienced with a unity annoyance weight or level.

Over Time Probability
Now we suppose that we missed the start of activity b because we were delayed in activity a by more than the introduced supplement D_0. We now also suppose that there is a second chance (for instance the next train) to catch b in time D_1 i.o. D_0, where $D_1 > D_0$. The *over time probability* that we miss b at D_0 but can catch b at D_1 is

$$p_{a,x\geq D_0}(D_0) = \int_{D_0}^{D_1} p_a(x)dx = \int_{D_0}^{D_1} ae^{-ax}dx = -e^{-ax}\Big|_{D_0}^{D_1} = e^{-aD_0} - e^{-aD_1}. \tag{9}$$

D_1 could be taken to be ∞ if no further b alternatives are considered.

Over Time Cost
When the delay of the activity a somehow takes longer than D_0 we cannot catch the next activity b that starts at that time. We suppose we must then catch the alternative activity, which we suppose starts at time $D_1 > D_0$. When trying to catch the ride of another train after a transfer for example, this will be the ride action of the next train in initially the same direction. We will often find times like about one hour or half hour or so in a practical, regular train service. This can be enforced by requiring that n trains per hour leave at regular time intervals of about $1/n$-th of an hour. The parameter D_1 is supposed to be known, while D_0, in the optimization context, is the variable we want to find an optimum for. We can again, similarly, calculate the probability that we can catch the activity at time D_1 and the corresponding time cost as

$$\left\{\begin{array}{lcl} p_{a,D_0\leq x\leq D_1}(D_0, D1) & = & \int_{D_0}^{D_1} p_a(x)dx = \int_{D_0}^{D_1} ae^{-ax}dx = e^{-aD_0} - e^{-aD_1} \\ UC_{a,D_0\leq x<D_1}(D_0, D1) & = & \int_{D_0}^{D_1} p_a(x)D_1dx = p_{D_0\leq x\leq D_1}D_1 \\ & = & (e^{-aD_0} - e^{-aD_1})D_1. \end{array}\right. \tag{10}$$

Note that probabilities and unit costs are positive.

Going on one more step, from D_1 to D_2 we have similar formulas, by simply substituting D_2 for D_1 and D_1 for D_0 in (10) or in general

$$\begin{cases} p_{a,D_{n-1}\leq x\leq D_n}(D_{n-1},D_n) & = \int_{D_{n-1}}^{D_n} p_a(x)dx = \int_{D_{n-1}}^{D_n} ae^{-ax}dx \\ & = e^{-aD_{n-1}} - e^{-aD_n} \\ UC_{a,D_{n-1}\leq x<D_n}(D_{n-1},D_n) & = p_{D_{n-1}\leq x\leq D_n} D_n \\ & = (e^{-aD_{n-1}} - e^{-aD_n})D_n. \end{cases} \tag{11}$$

All Time Units are Weighted the Same

If all time units are considered to have the same subjective annoyance, the total probability and unit time cost over all subsequent time intervals is

$$\begin{cases} p_{a,D_0\leq x<D_n}(D_0,D_1,D_2,...,D_N) & = \sum_{n=0}^{N-1} p_{a,D_{n-1}\leq x\leq D_n}(D_{n-1},D_n) \\ UC_{a,D_0\leq x<D_n}(D_0,D_1,D_2,...,D_N) & = \sum_{n=0}^{N-1} UC_{a,D_{n-1}\leq x\leq D_n}(D_{n-1},D_n), \end{cases} \tag{12}$$

where N is the number of terms we want to consider and $D_{-1} = 0$ to simplify notation. The higher n gets, the lower the contribution to the result becomes, due to the decreasing probabilities of the integration interval between D_{n-1} and D_n. Figure 2(c), shows a plot of these probability functions. We call this the one dimensional case because only a delay on one action plays a role here. $D1 = 39$ and $D_2 = 54$, so 15 time units after the first repetition, a new repetition of the next action creates a possibility to catch it again.

Figure 2(e) shows an example of the first three terms of the unit cost in (12) for chosen values 39 and 54 for D_1 and D_2 respectively. The expected delay duration of action a is $d_a = 6$. Clearly the total time cost gets minimal when $D_0 = 10$ time units. Note that the third term, $UC_{D_1\leq x<D_2}$ is constant for all values of D_0 and as such does not influence the value of the optimal D_0 supplement. Also, this third term is much less important in absolute value, compared to the first two terms.

In practice, for this case with constant annoyance level for all types of time units, we will only use the first two terms to calculate the total cost.

Time Units Can Have Different Weights

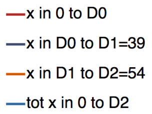
For passengers, some time units are sometimes considered more annoying than others. This depends on the activity. For example, riding feels more useful and pleasant than dwelling. Also, when train connections are missed, annoyance grows.

Like [12, 13], we model these annoyances by multiplying each time unit by a weight. We introduce the weight $w_{a,i}$ for the delay x of an activity a where i stands for the number of next activity instances that has been missed already. This holds for a passenger on a specific switching from one activity to the next. Of course, annoyance grows when missing more next activity instances, so it will hold that

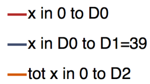
$$\forall a \in A : \forall i \in \mathbb{N} : w_{a,i} \leq w_{a,i+1}. \tag{13}$$

We further introduce $w_{c,i}$ as the weight for the left over idle time $D_0 - x$. The probability calculation remains the same as in (8) and (10) in the previous paragraph, but the corresponding unit cost integrals become somewhat more involved. Indeed,

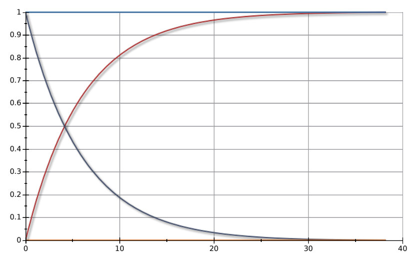
$$\begin{aligned} wUC_{a,x\leq D_0}(D_0) &= \int_0^{D_0}(w_{a,0}x + w_{c,0}(D_0 - x))p_a(x)dx \\ &= w_{c,0}D_0\int_0^{D_0} p_a(x)dx + (w_{a,0} - w_{c,0})\int_0^{D_0} xp_a(x)dx \\ &= w_{c,0}D_0 p_{a,x\leq D_0}(D_0) + (w_{a,0} - w_{c,0})UC_{a,x\leq D_0}(D_0). \end{aligned} \tag{14}$$



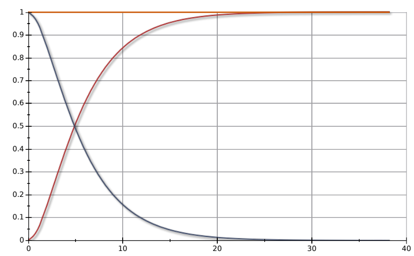
(a)



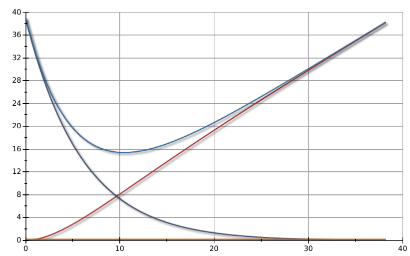
(b)



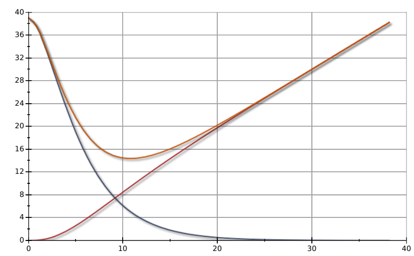
(c) Delay Probabilities for 1 Action, $\overline{d_a} = 6.0$



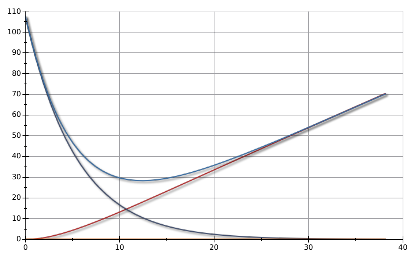
(d) Delay-Prob. for 2 Actions, $\overline{d_a} = 4.0$, $\overline{d_b} = 2.0$



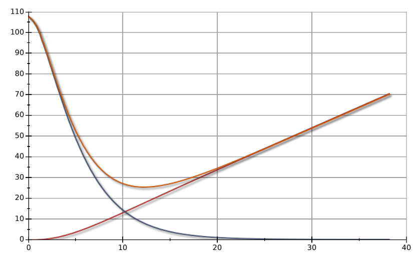
(e) Objective Unit Costs for 1 Action, $\overline{d_a} = 6.0$



(f) Obj. Unit Costs for 2 Actions, $\overline{d_a} = 4.0$, $\overline{d_b} = 2.0$



(g) Subjective Unit Costs for 1 Action, $\overline{d_a} = 6.0$



(h) Subj. Unit Costs for 2 Actions, $\overline{d_a} = 4.0$, $\overline{d_b} = 2.0$

Figure 2: Probablities (p), Expected Passenger Time (UC) and Weighted Expected Passenger Time (wUC) as functions of the (sum of) the chosen supplements (x). Different curves for supplements being less than D0 (in-time), between D0 and D1 (over-time), and their sum (total-time), on the vertical axis. x on the horizontal axis. All units are in minutes. (a) is the legend for (c), (e) and (g). (b) is the legend for (d), (f) and (h).

Note that $UC_{a,x\leq D_0}(D_0)$, in this weighted case, has been redefined as $\int_0^{D_0} xp_a(x)dx$ instead of $\int_0^{D_0} D_0 p_a(x)dx$, in the unweighted case. From the context it will be clear what we mean with it.

We derive the needed expressions for probabilities and unit costs as

$$\begin{array}{lcl} p_{a,0\leq x\leq D_0}(D_0) & = & \int_0^{D_0} ae^{-ax}dx = 1 - e^{-aD_0} \\ UC_{a,0\leq x\leq D_0}(D_0) & = & \int_0^{D_0} xae^{-ax}dx = \frac{1-(aD_0+1)e^{-aD_0}}{a}. \end{array} \tag{15}$$

Note that some terms in (14) may become negative, for example when $w_{a,0} - w_{c,0} < 0$, but the total value of $wUC_{a,x\leq D_0}(D_0)$ will always be positive. The same holds for the next interval, (D_0, D_1) where,

$$\begin{array}{ll} & wUC_{a,D_0\leq x<D_1}(D_0, D1) \\ = & \int_{D_0}^{D_1}(w_{a,0}D_0 + w_{a,1}(x - D_0) + w_{c,1}(D_1 - x))p_a(x)dx \\ = & ((w_{a,0} - w_{a,1})D_0 + w_{c,1}D_1)\int_{D_0}^{D_1} p_a(x)dx + (w_{a,1} - w_{c,1})\int_{D_0}^{D_1} xp_a(x)dx \\ = & ((w_{a,0} - w_{a,1})D_0 + w_{c,1}D_1)p_{a,D_0\leq x\leq D_1}(D_0, D_1) + (w_{a,1} - w_{c,1})UC_{a,D_0\leq x\leq D_1}(D_0, D_1), \end{array} \tag{16}$$

where we derive the expressions for probabilities and unit costs as

$$\begin{array}{lcl} p_{a,D_0\leq x\leq D_1}(D_0, D_1) & = & \int_{D_0}^{D_1} ae^{-ax}dx = \ = e^{-aD_0} - e^{-aD_1} \\ UC_{a,D_0\leq x\leq D_1}(D_0, D_1) & = & \int_{D_0}^{D_1} xae^{-ax}dx = \frac{(aD_0+1)e^{-aD_0} - (aD_1+1)e^{-aD_1}}{a}. \end{array} \tag{17}$$

Based on

$$\begin{array}{ll} & wUC_{a,D_{n-1}\leq x<D_n}(D_0, D_1, ..., D_{n-1}, D_n) \\ = & (\sum_{i=0}^{n-1}(w_{a,i} - w_{a,i+1})D_i + w_{c,n}D_n)p_{a,D_{n-1}\leq x\leq D_n}(D_{n-1}, D_n) \\ + & (w_{a,n} - w_{c,n})UC_{a,D_{n-1}\leq x\leq D_n}(D_{n-1}, D_n), \end{array} \tag{18}$$

one can prove that the formula (16), generalized for N subsequent chances on the next action, becomes

$$\begin{array}{ll} & wUC_{a,0\leq x<D_N}(D_0, D_1, D_2, ..., D_N) \\ = & \sum_{n=0}^{N}[(\sum_{i=0}^{n-1}(w_{a,i} - w_{a,i+1})D_i + w_{c,n}D_n)p_{a,D_{n-1}\leq x\leq D_n} \\ + & (w_{a,n} - w_{c,n})UC_{a,D_{n-1}\leq x\leq D_n}] \\ = & \sum_{n=0}^{N-1}((w_{a,n} - w_{a,n+1})D_n p_{a,D_n\leq x\leq D_N}) \\ + & \sum_{n=0}^{N}(w_{c,n}D_n p_{a,D_{n-1}\leq x\leq D_n} + (w_{a,n} - w_{c,n})UC_{a,D_{n-1}\leq x\leq D_n}), \end{array} \tag{19}$$

where the probabilities p and unit costs UC take the form as in equation (17), with the respective indices.

As in the previous case without the weights, we demonstrate that this weighted unit objective function, considered as a function of D_0, with fixed $D_i, \forall i > 0$, has a minimum for a certain D_0. Figure 2(g) shows this for the subjective time unit weights $w_{a,0} = 1.0, w_{a,1} = 1.5, w_{a,2} = 2.0$ and $w_{c,0} = 2.0, w_{c,1} = 3.0$ and $w_{c,2} = 4.0$. If the weights would all be 1 we get the same cost curves as in Figure 2(e).

The total cost has now increased since for every time unit a weight factor equal or bigger than 1.0 is used. We also see that the minimum cost is now shifted to about $D_0 = 12$. That the optimal supplement is bigger here stems from the fact that the higher annoyance $w_{c,i}$ of the time units after having missed the consecutive next action instances leads to a higher cost for supplements D_0 that are so low that they often cause such a miss. So the subjective appreciation of each type of time unit can have an effect on the optimum supplement value.

Note that the formula (18) for wUC_{D_{n-1},D_n} with subjective weights, simplifies to the formula for UC_{D_{n-1},D_n} for the case where all weights are equal to 1 case in (11). Also, the formulas (18) and (19) remain even valid if another delay probability distribution than the negative exponential is chosen. Of course the corresponding $p_{a,D_{n-1}\leq x\leq D_n}$ and $UC_{a,D_{n-1}\leq x\leq D_n}$ have to be recalculated then.

By simply taking the minimum of the objective function, we are now able to calculate the optimal supplement that should be inserted in the case where only one probabilistic delay plays a role. Sometimes however, two consecutive activities with their own delay distribution will occur. We treat this case in the next section.

5.2 Pair of Two Subsequent Actions

Sum of Delays

We consider two subsequent actions a and b. Their probabilistic delay variables are called x and y respectively. Again, we can measure the occurrences of delays for both processes and fit a normalized negative exponential through these measurements. This gives

$$\begin{cases} p_a(x) & = & ae^{-ax} \\ p_b(y) & = & be^{-by}. \end{cases} \tag{20}$$

The fitting is done similarly to the one dimensional case and gives for actions a and b, for their respective expected delays, also called a and b:

$$\begin{cases} a & = & 1/\overline{d_a} \\ b & = & 1/\overline{d_b}. \end{cases} \tag{21}$$

We suppose the variables x and y and their distributions are independent, which could be a simplification of reality.

In train planning, the most important times are the ones where a train leaves a station after a stop. This is the latest time a passenger can catch the train, are the times to be published and respecting them in real time is what matters most to passengers. This is the promise to the customer that has to be kept as good as possible. Traditionally arrival times are also published. This is practical for a passenger to be able to plan the activity after his train arrival. But consider that it is generally worse to miss your train at departure than being a few time units later at arrival.

We consider a to be a ride action. b can then be a subsequent dwell, transfer or sink action. Dwell means waiting in a station inside the train. Transfer means a transfer to another train in the same station. Sink means the passenger exits the station and the train system. Dwell and transfer actions have endings that can be delayed. For a sink, a delay is not useful to model since the time for sinking is not part of the train service model.

To mathematically formulate expected passenger time, we are first interested in the probability that the summed delay $x+y$ of both actions together is smaller or bigger than a certain supplement D, that is supposed to statistically, mostly absorb this summed delay. We also will calculate the associated time costs of introducing this supplement. Balancing probabilities and respective costs, we will obtain closed formulas for the total cost of introducing a supplement D on a subsequent pair of actions. Weighting these costs with the number of passengers on the second action of each pair, we arrive at a formula for the total cost of each *fork* structure. A fork is the union of a ride edge and its directly subsequent, unique dwell edge, unique sink edge and potentially multiple transfer edges to different other trains. In

Figure 1, a fork can be identified as a collection of a blue (ride) rectangle and its right direct neighbours: the yellow (dwell), orange (transfer) and red (exiting) rectangles.

Our graph, inherently present in the representation in Figure 1, is set up so that a dwell edge combines all passenger flows dwelling in one edge, irrespective of their origin and destination. Similarly a source edge combines all passengers in one edge, irrespective of their destination and a sink edge all passengers in one edge, irrespective of their origin.

In Time Cost

Contrary to the one dimensional case in the previous paragraph, we will immediately consider the most general case of weighted, subjective time units. If all time units are considered to have the same weight, the w values all equal 1.

The cost of suffering a delay x on action a (e.g.: ride) as well as a delay y on action b (e.g.: dwell) while we are still in time for the next ride start is

$$w_{a,0}x + w_{b,0}y, \tag{22}$$

where $w_{a,0}$ is the possibly subjective weight for an action a in these conditions. Similarly for $w_{b,0}$.

If we are still in time, meaning the supplement D_0 against delay, is not exceeded yet, some time until D_0 is left over. Passengers will wait after actions a and b have completed, during this time $D_0 - x - y$. This time also has to be accounted for and has an annoyance weight of $w_{c,0}$. So this cost is

$$w_{c,0}(D_0 - x - y). \tag{23}$$

Together, the total in time cost of experiencing a delay $x + y \leq D$ on these two actions is the sum of (22) and (23), so

$$w_{a,0}x + w_{b,0}y + w_{c,0}(D_0 - x - y). \tag{24}$$

Note that for $w_{a,0} = w_{b,0} = w_{c,0} = 1$, the total in time cost is D_0.

For every combination of delays x and y, we now have to see how often they occur. This is given by the combined probabilities

$$p_a(x)p_b(y). \tag{25}$$

Integrating over $x+y \leq D_0$ which forms a rectangular triangle gives us the in time cost

$$\begin{array}{ll} & wUC_{0\leq x\leq D_0} \\ = & \int_0^{D_0}\int_0^{D_0-x}\left(w_{a,0}x + w_{b,0}y + w_{c,0}(D_0-x-y)\right)p_a(x)p_b(y)dydx \\ = & w_{c,0}D_0\int_0^{D_0}\int_0^{D_0-x}p_a(x)p_b(y)dydx \\ + & (w_{a,0}-w_{c,0})\int_0^{D_0-x}xp_a(x)p_b(y)dydx + (w_{b,0}-w_{c,0})\int_0^{D_0-x}yp_a(x)p_b(y)dydx \\ = & w_{c,0}D_0p_{a\backsim b,0\leq x+y\leq D_0} \\ + & (w_{a,0}-w_{c,0})UC_{a\backsim b,x,0\leq x+y\leq D_0} + (w_{b,0}-w_{c,0})UC_{a\backsim b,y,0\leq x+y\leq D_0}. \end{array} \tag{26}$$

For $a \neq b$ we get

$$
\begin{array}{rl}
& p_{a\neq b,0\leq x+y\leq D_0} = \int_0^{D_0}\int_0^{D_0-x}(ae^{-ax}be^{-by})dydx = \frac{a(1-e^{-bD_0})-b(1-e^{-aD_0})}{a-b} \\
& UC_{a\neq b,x,0\leq x+y\leq D_0} = \int_0^{D_0}\int_0^{D_0-x} x(ae^{-ax}be^{-by})dydx \\
= & \frac{1-(aD_0+1)e^{-aD_0}}{a} - [e^{-aD_0}((b-a)D_0-1)+e^{-bD_0}]\frac{a}{(a-b)^2} \\
& UC_{a\neq b,y,0\leq x+y\leq D_0} = \int_0^{D_0}\int_0^{D_0-y} y(ae^{-ax}be^{-by})dxdy \\
= & \frac{1-(bD_0+1)e^{-bD_0}}{b} - [e^{-bD_0}((a-b)D_0-1)+e^{-aD_0}]\frac{b}{(a-b)^2}.
\end{array}
\tag{27}
$$

Care is taken that the last version of each expression in (27) only has negative exponents, so that numerical stability is guaranteed and overflow is avoided.

When $a = b$ we would get a division by zero for each of the expressions in (27), so we recalculate for this special case. This gives

$$
\begin{array}{l}
p_{a=b,0\leq x+y\leq D_0} = \int_0^{D_0}\int_0^{D_0-x}(ae^{-ax}ae^{-ay})dydx = 1-(aD_0+1)e^{-aD_0} \\
UC_{a=b,x,0\leq x+y\leq D_0} = \int_0^{D_0}\int_0^{D_0-x} x(ae^{-ax}ae^{-ay})dydx = \frac{e^{-aD_0}(-aD_0(aD_0+2)-2)+2}{2a} \\
UC_{a=b,y,0\leq x+y\leq D_0} = \int_0^{D_0}\int_0^{D_0-y} y(ae^{-ax}ae^{-ay})dxdy = \frac{e^{-aD_0}(-aD_0(aD_0+2)-2)+2}{2a}.
\end{array}
\tag{28}
$$

Over Time Cost

The integral in time delay cost $wUC_{0\leq x\leq D_0}$ for delays below D_0 in (26) gives us a cost supposing $x + y \leq D_0$. Since $x + y > D_0$ can also hold, we calculate this corresponding *over time cost* next.

When $x + y = D_0$, the ride action after a and b takes place immediately and can still just be caught by the passengers. If $x + y > D_0$, they will miss this next action. For action b being a transfer, this means they will have to catch the next train in the same direction.

Suppose the next leaving train in the desired direction leaves at time distance $D_1 > D_0$. Then we get for the cost part for $x+y$ between D_0 and D_1, which forms a trapezium which fits snuggly against the triangle area, from (26) as

$$
\begin{array}{rl}
& wUC_{D_0\leq x\leq D_1} \\
= & \int_0^{D_0}\int_{D_0-x}^{D_1-x}[(w_{a,0}+w_{b,0})/2\cdot D_0+(w_{a,1}+w_{b,1})/2\cdot(x+y-D_0) \\
+ & w_{c,1}(D_1-x-y)]\,p_a(x)p_b(y)dydx \\
+ & \int_{D_0}^{D_1}\int_0^{D_1-x}[(w_{a,0}+w_{b,0})/2\cdot D_0+(w_{a,1}+w_{b,1})/2\cdot(x+y-D_0) \\
+ & w_{c,1}(D_1-x-y)]\,p_a(x)p_b(y)dydx \\
= & ((w_{a,0}+w_{b,0})/2-(w_{a,1}+w_{b,1})/2)\cdot D_0+w_{c,1}D_1) \\
& (\int_0^{D_0}\int_{D_0-x}^{D_1-x} p_a(x)p_b(y)dydx+\int_{D_0}^{D_1}\int_0^{D_1-x} p_a(x)p_b(y)dydx) \\
+ & ((w_{a,1}+w_{b,1})/2-w_{c,1}) \\
& (\int_0^{D_0}\int_{D_0-x}^{D_1-x} xp_a(x)p_b(y)dydx+\int_{D_0}^{D_1}\int_0^{D_1-x} xp_a(x)p_b(y)dydx) \\
+ & ((w_{a,1}+w_{b,1})/2-w_{c,1}) \\
& (\int_0^{D_0}\int_{D_0-y}^{D_1-y} yp_a(x)yp_b(y)dydx+\int_{D_0}^{D_1}\int_0^{D_1-y} yp_a(x)p_b(y)dxdy) \\
= & ((w_{a,0}+w_{b,0})/2-(w_{a,1}+w_{b,1})/2)\cdot D_0+w_{c,1}D_1) \\
& (p_{a\backsim b,D_0\leq x+y\leq D_1,p1}+p_{a\backsim b,D_0\leq x+y\leq D_1,p_2}) \\
+ & ((w_{a,1}+w_{b,1})/2-w_{c,1})(UC_{a\backsim b,x,D_0\leq x+y\leq D_1,p_1}+UC_{a\backsim b,x,D_0\leq x+y\leq D_1,p_2}) \\
+ & ((w_{a,1}+w_{b,1})/2-w_{c,1})(UC_{a\backsim b,y,D_0\leq x+y\leq D_1,p_1}+UC_{a\backsim b,y,D_0\leq x+y\leq D_1,p_2}),
\end{array}
\tag{29}
$$

where $a \backsim b$ stands for the relation between a and b: equality or non-equality and p_1 stands for part 1, the parallellogram and p_2 for part 2, the triangle, that together make up the trapezium over which we integrate here. Note that the swapping of integration order of x and y is only done to make the temporary resulting formulas as simple as possible. For each line in (29), the sum of two double integrals over the full trapezium is the same when integrated in both integration orders.

The averaging of weights in $(w_{a,i} + w_{b,i})/2$ appears because for a time $x + y$ where $D_0 \leq x + y \leq D_1$, it is not known which part of the time was spent for activity a versus b.

The formulas for the unit probabilities and unit cost in this D_0 to D_1 case for $a \neq b$ are

$$\begin{aligned}
& p_{a\neq b,D_0\leq x+y\leq D_1,p1} = \int_0^{D_0} \int_{D_0-x}^{D_1-x} ae^{-ax}be^{-by}dydx \\
= \quad & (e^{-bD_0} - e^{-aD_0} - e^{-bD_1} + e^{-aD_0-b(D_1-D_0)})\tfrac{a}{a-b} \\
\\
& p_{a\neq b,D_0\leq x+y\leq D_1,p2} = \int_{D_0}^{D_1} \int_0^{D_1-x} ae^{-ax}be^{-by}dydx \\
= \quad & -e^{-b(D_1-D_0)-aD_0}\tfrac{a}{a-b} + e^{-aD_1}\tfrac{b}{a-b} + e^{-aD_0} \\
\\
& UC_{a\neq b,x,D_0\leq x+y\leq D_1,p1} = \int_0^{D_0} \int_{D_0-x}^{D_1-x} xae^{-ax}be^{-by}dydx \\
= \quad & ((-aD_0 + bD_0 - 1)e^{-aD_0} + e^{-bD_0})(1 - e^{-b(D_1-D_0)})\tfrac{a}{(a-b)^2} \\
\\
& UC_{a\neq b,x,D_0\leq x+y\leq D_1,p2} = \int_{D_0}^{D_1} \int_0^{D_1-x} xae^{-ax}be^{-by}dydx \\
= \quad & [(D_0(b-a) - 1)e^{-aD_0-b(D_1-D_0)} + (D_1(a-b) + 1)e^{-aD_1}]\tfrac{a}{(a-b)^2} \\
+ \quad & [(aD_0 + 1)e^{-aD_0} - (aD_1 + 1)e^{-aD_1}]/a \\
\\
& UC_{a\neq b,y,D_0\leq x+y\leq D_1,p1} = \int_0^{D_0} \int_{D_0-y}^{D_1-y} yae^{-ax}be^{-by}dxdy \\
= \quad & [(-bD_0 + aD_0 - 1)e^{-bD_0} + e^{-aD_0})(1 - e^{-a(D_1-D_0)}]\tfrac{b}{(a-b)^2} \\
\\
& UC_{a\neq b,y,D_0\leq x+y\leq D_1,p2} = \int_{D_0}^{D_1} \int_0^{D_1-y} yae^{-ax}be^{-by}dxdy \\
= \quad & [(D_0(a-b) - 1)e^{-bD_0-a(D_1-D_0)} + (D_1(b-a) + 1)e^{-bD_1}]\tfrac{b}{(a-b)^2} \\
+ \quad & [(bD_0 + 1)e^{-bD_0} - (bD_1 + 1)e^{-bD_1}]/b.
\end{aligned} \tag{30}$$

For the $a = b$ case the equivalent expressions are

$$\begin{aligned}
& p_{a=b,D_0\leq x+y\leq D_1,p1} = \int_0^{D_0} \int_{D_0-x}^{D_1-x} ae^{-ax}be^{-by}dydx = aD_0(e^{-aD_0} - e^{-aD_1}) \\
& p_{a=b,D_0\leq x+y\leq D_1,p2} = \int_{D_0}^{D_1} \int_0^{D_1-x} ae^{-ax}be^{-by}dydx = e^{-aD_1}(a(D_0 - D_1) - 1) + e^{-aD_0} \\
& UC_{a=b,x,D_0\leq x+y\leq D_1,p1} = \int_0^{D_0} \int_{D_0-x}^{D_1-x} xae^{-ax}be^{-by}dydx = aD_0{}^2(e^{-aD_0} - e^{-aD_1})/2 \\
& UC_{a=b,x,D_0\leq x+y\leq D_1,p2} = \int_{D_0}^{D_1} \int_0^{D_1-x} xae^{-ax}be^{-by}dydx \\
= \quad & [e^{-aD_1}(a^2(D_0{}^2 - D_1{}^2) - 2aD_1 - 2) + 2(aD_0 + 1)e^{-aD_0}]/(2a) \\
& UC_{a=b,y,D_0\leq x+y\leq D_1,p1} = \int_0^{D_0} \int_{D_0-y}^{D_1-y} yae^{-ax}be^{-by}dxdy \\
= \quad & aD_0{}^2(e^{-aD_0} - e^{-aD_1})/2 \\
& UC_{a=b,y,D_0\leq x+y\leq D_1,p2} = \int_{D_0}^{D_1} \int_0^{D_1-y} yae^{-ax}be^{-by}dxdy \\
= \quad & [e^{-aD_1}(a^2(D_0{}^2 - D_1{}^2) - 2aD_1 - 2) + 2(aD_0 + 1)e^{-aD_0}]/(2a).
\end{aligned} \tag{31}$$

As in the previous paragraph, we could calculate a next cost term for $wUC_{D_1 \leq x \leq D_2}$ but from Figures 2(e) and 2(g) we know that this will be a low cost and we can safely ignore it. So the total local cost we consider is $wUC_{loc} = wUC_{0 \leq x+y \leq D_1} = wUC_{0 \leq x+y \leq D_0} + wUC_{D_0 \leq x+y \leq D_1}$ which can be calculated from (30) or (31).

Figure 2(d) shows a plot of the probability functions for the two subsequent action cases. It differs from Figure 2(c) in that now both actions have delay distributions. Figure 2(d) shows probability curves of which the second derivative changes its sign. This is due to the convolution of these two distributions.

Figure 2(f) represents the total costs and its component costs from 0 to D_0 and from D_0 to D_1 for objective time unit costs, all equal to 1. The minimum cost is obtained for about $D_0 = 11$ time units.

Figure 2(h) represents the same cost for the subjective time unit costs, $w_{a,0} = 1.0$, $w_{a,1} = 1.5$, $w_{b,0} = 1.0$, $w_{b,1} = 1.5$, $w_{c,0} = 2.0$ and $w_{c,1} = 3.0$. We see that the minimum in Figure 2(h) is not the same as in Figure 2(f). Now the minimum cost is obtained for $D_0 = 12$ time units. Similarly to the one dimensional case higher annoyance for time units spent after having missed a next action results in introduction of a bigger supplement D_0 to help avoid this.

5.3 Optimal Sum of a Range of Supplements for (Ride,Transfer)-Action Pair

We now have derived the objective function for the (ride, transfer) action pair. We show in Figure 3 that we are able to calculate the predicted optimum for the sum of two supplements associated with two consecutive actions a and b. We show the results obtained for different combinations of values of expected delays for action a $(0, 1, 4, 10, 20)$ and b $(0, 1, 2, 4)$. All units can be seen as minutes. D_1 is supposed to always be 60 here. The case $(0, 0)$, meaning no expected delay on either action, results in no supplements at all. One can also see that an increase in any of the two expected delays results in an increased optimal supplement sum, which is only logical.

Note that these optima are only valid for one action pair. A single supplement can occur in multiple action pairs and their respective local objective functions. Each can have a different optimum value. The global optimization will find a trade-off between these local optima. Section 6 describes how this is done.

5.4 Specialization to All Flow Types: Different Combinations of In Time and/or Over Time Costs

We have derived the objective function for a specific single action and for the (ride, transfer) action pair. These are the most general cases. The other three cases, as mentioned in Table 1 can be derived from it. Departing passengers only have an over-time cost and no in-time cost. Through and arriving passengers only have an in-time cost and no over-time cost. The resulting functions are given in Figures 4(a) to 4(d).

5.5 Linearization

To be able to use the derived objective function in an optimization for a train service network for an entire country, we want to use linear programming solvers. This means that our objective function has to be linearized.

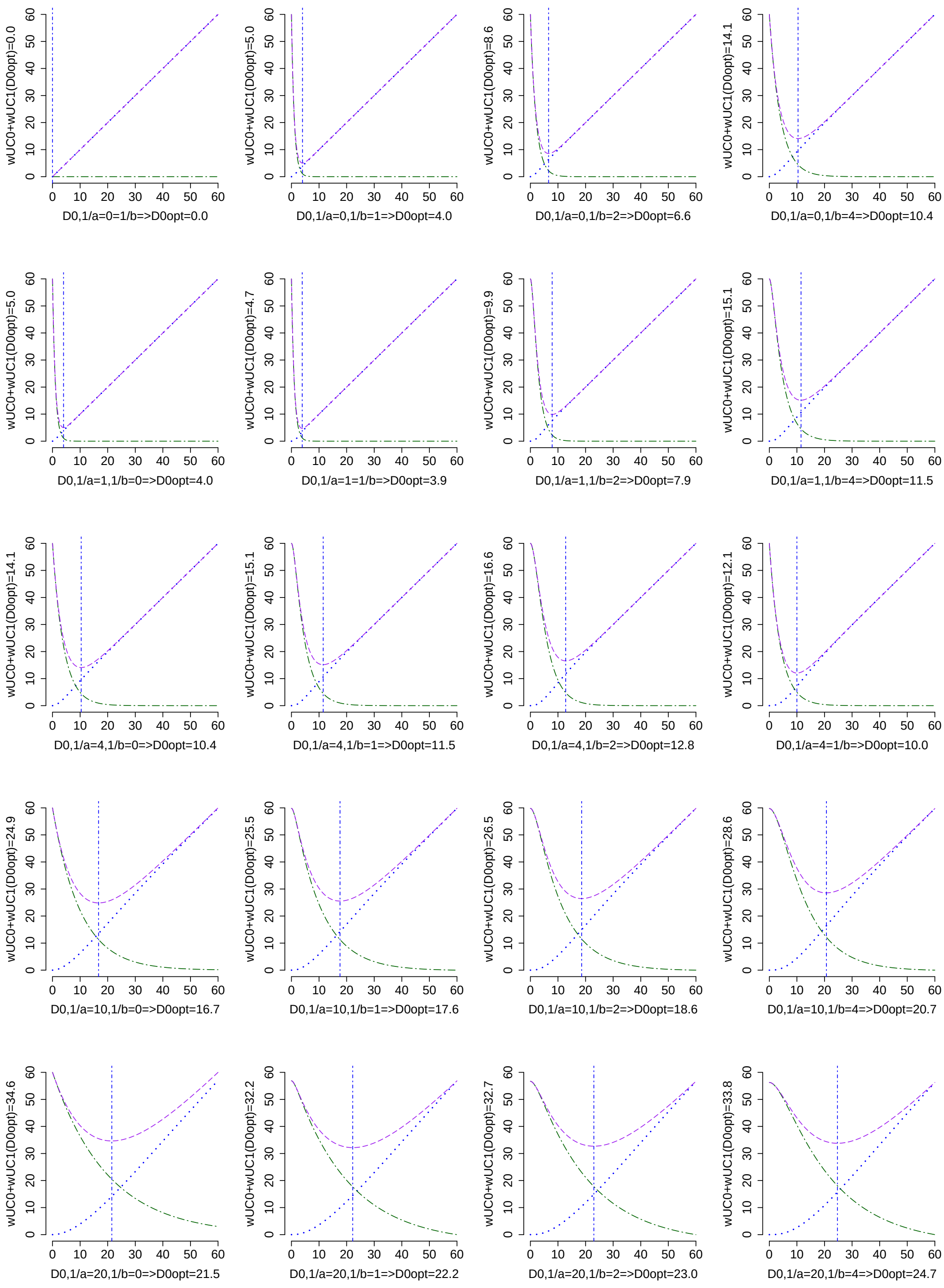
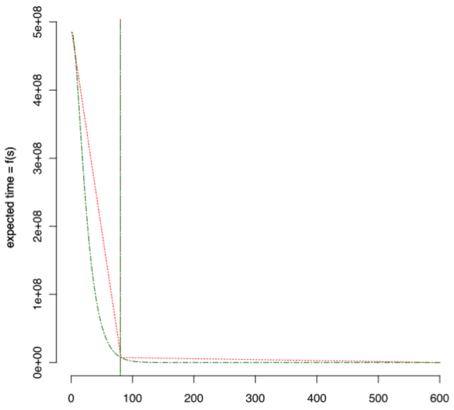
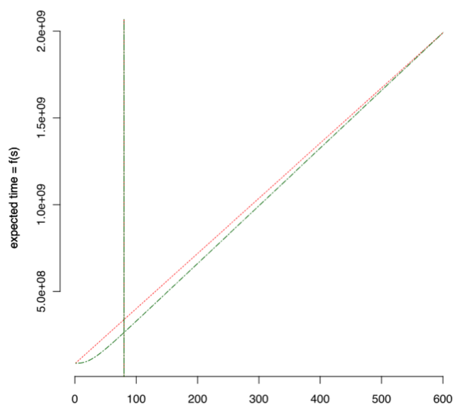


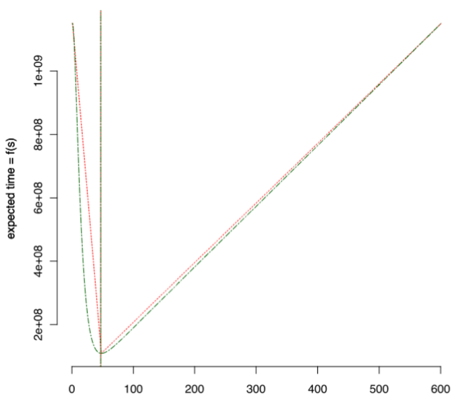
Figure 3: Supplement Sum with Minimal Cost for a Range of Action Pair Expected Delays



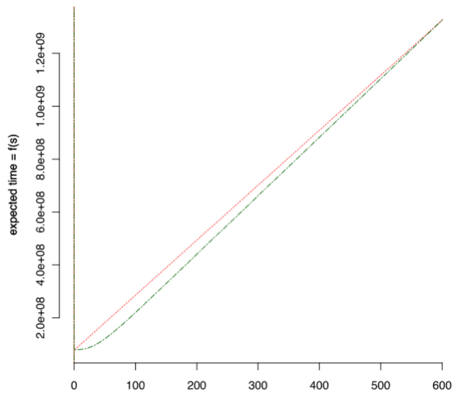
(a) Departing Passenger Objective Function



(b) Through Passenger Objective Function



(c) Transfer Passenger Objective Function



(d) Arriving Passenger Objective Function

Figure 4: Expected Passenger Time as Objective Functions of sum of supplements chosen for the four Passenger Flow Types.

There are two options here to linearize the curves of the calculated objective functions in Figures 2(e), 2(f), 2(g) and 2(h). One is to discretize the independent variable to some fixed resolution, like a minute or a tenth of a minute. The other is to approximate the curves with a number of line segments, which allows for the independent variables to remain continuous. When inspecting these objective functions, one can see that they are quite smooth. There are no sudden peaks or dips. This pleads for the second option. With only two line segments the minimum resulting from the intersection of the two line segments is already very close to the minimum of the original curves. For the 2-segment-approximation we use the least squares method. We also tried the first option with a fixed resolution of a tenth of a minute, but this resulted in many variables and an optimal schedule calculation times of a few hours for just 20 trains. This also convinced us that the segment approximation method should result in a faster yet accurate tool. The result of the linearizations are shown in the Figures 4(a) to 4(d) by the approximation of the green curves by the red segments.

A black vertical line indicates the supplement that was chosen. In the cases 4(c) and 4(d) this results in the real local minimum. In the cases 4(a) and 4(b) the results are close to the local minimum. This can happen when constraints or local objective functions in other locations of the schedule play a more important role. We now explain how the local objective functions are composed to form the global objective function.

6 Recomposing the Global Objective Function: Simply Summing

To evaluate the expected passenger time of a schedule, we need to look at all action pairs present in the planned schedule and see how these contribute to the total passenger time.

In the previous paragraphs, we convoluted distributions of pairs of subsequent actions, of which the first is always a ride action. In so doing, we formed forks of actions for which we obtained a (weighted) unit time objective function. By multiplication with the correct number of passengers present on these actions, we obtain the total cost of each action pair.

The weighted global objective function for the whole schedule is just the sum of all local weighted objective functions. This is the case because, (1) we consider all passenger minutes equal and (2) we don't convolute the delay distributions across actions of different (subsequent) forks. The second reason implies that we will actually determine a schedule that is more robust than the real minimal expected passenger time would require. Our schedule will have the advantage though that ride to next ride departure times are also more dependable.

So our *expected passenger time*, which is the objective function to be minimized for schedule optimization, becomes

$$\begin{aligned} wGC(D_0(E)) &= \textstyle\sum_{e\in E_{src}} f_e \cdot wUC_{predpredride(e),predwell(e)}(D_{0,e,succride(e)}) \\ &+ \textstyle\sum_{e\in E_d\cup E_{tr}\cup E_{snk}} f_e \cdot wUC_{predride(e),e}(D_{0,predride(e),e}), \end{aligned} \tag{32}$$

where wGC is the weighted global cost, E is the graph of all actions in the train service network, $D_0(E)$ is a shorthand notation for all D_0 that are present in the graph E, E_d is the set of all dwell edges, E_{tr} is the set of all transfer edges, E_{snk} is the set of all sink edges, $succride(e)$ is the ride successor edge associated with edge $e \in E_{src}$, which is also the successor edge of edge e. $preddwell(e)$ is the ride predecessor edge associated with edge $e \in E_{snk}$, which is also the predecessor dwell edge of the successor ride edge $succride(e)$ of e. $predpredride(e)$ is the ride predecessor edge associated with edge $preddwell(e)$ of $e \in E_{snk}$. $predride(e)$ is the ride edge associated with edge e, which is also always the predecessor edge of edge e. f_e is the local passenger flow over that action or edge e. Note that only flows over non ride edges are directly present. The ride flow is simply the sum of all flows on actions leaving from that ride action, and since flows for all these next actions are present, the ride flows are present implicitly in the sum. D_{0,e_a,e_b} is the D_0 parameter for that specific edge sequence pair (e_a, e_b) and wUC_{e_a,e_b} its weighted local objective function.

The last three columns in Table 2 structurally summarize what quantities are used in (32) for each of the four passenger flow types. The column *Flow of Action* indicates the action that determines the number of passengers on each action pair. Note that the number of ride passengers is not present directly in any of the four cases, but as Figure 1 shows, it equals the sum of dwell, transfer and sink passenger numbers subsequent to the ride action considered. If we consider all terms in this sum, we account for every passenger correctly.

Table 2: Four Passenger Flow Types, their Corresponding Action Sequences, the Action Edges Used and their Time Cost Types

Passenger Flow Type	Action a	Action b	Time Cost	Flow of Action	Edge a	Edge b
Departing	(Source)	Ride	in	Source	PredPredRide	PredDwell
Through	Ride	Dwell	over	Dwell	Ride	Dwell
Transfer	Ride	Transfer	in+over	Transfer	Ride	Transfer
Arriving	Ride	(Sink)	over	Sink	Ride	

The quality of a given schedule with given $D_{0,e}, \forall e$ can now be directly calculated from (32). For the sink edges, the two action formulas may be used with an expected delay of $\overline{d_b} = 0$ or very low. This essentially results in using the one action formula on the associated ride edges, which can of course be used directly as well.

7 Knock-On Delays

In a similar way as the derivations made previously, we also derived a *knock-on objective function* for each couple of trains using the same piece of track. This is a function of the heading time between these two trains. Since in our schedule optimization, the order of trains on a track unit is unknown in advance, a knock-on cost between each of the $N*(N-1)$ couples of the N trains is added. When train a is scheduled very close before train b two factors increase the total knocked-on passenger delay on b. One is a high probability of delay on train a. The second is a high number of passengers on train b, since they are the passengers experiencing the knock-on delay. These factors are present in our knock-on objective function. We have used these knock-on costs in train schedule optimizations of all periodic passenger trains in Belgium of which many pairs have train tracks in common and even though there are many of them, they don't significantly increase the computation time.

8 Results

We have constructed a model with all necessary constraints and with the objective function derived here, for an increasing number of trains. The starting timetable is the one for all trains departing between 7am and 8am on 13 March 2013. We used Gurobi 5.1.0 as the MILP solver and ran it, set to 8 solver threads, on an Apple iMac with a 3.4 GHz Intel i7 processor and with 16GB 1333 MHz DDR3 Memory. The gap desired was set to what was obtained as the gap of the first returned solution in earlier trials. Our results are given in table 3.

The numbers of model variables (#columns) and constraints (#rows) both grow rapidly as trains are added, but the pre-solver is usually able to reduce both numbers by about 50%. For the last case, IC IR L P, we have rescheduled all 186 periodic (IC IR L) trains, as well as the 17 non-periodic peak (P) trains departing between 7am and 8am. For each ride and dwell action we assumed an expected primary delay of 3% on top of its minimum time. All periodic trains are repeated every hour. The P trains occur just once in practice, but due to our cyclic timetable, time slots are automatically reserved for them in every non-peak hour too. Outside the peak period, these empty P slots can possibly be used for freight trains if desirable.

Table 3: Scalability of our Integer Linear Programming Model with necessary Constraints and the Derived Objective Function

train types	trains (#)	model rows (#)	model columns (#)	solver time (s)	MILP gap (%)	passenger time reduction (%)	missed transfer probability (%)
IC	43	18747	13361	50	24.30	10.73	2.95
IC IR	82	48267	33035	449	32.35	12.38	3.11
IC IR L	186	102652	68504	2426	32.80	10.03	2.31
IC IR L P	203	225132	158860	3706	38.27	7.12	2.43

Compared to the IC IR L P-timetable currently in operation, our optimized timetable has some advantages compared to the current one. First, it respects all minimum ride- and dwell-times without exception. Second, it respects all headway time buffers of 3 minutes between all train pairs on the same track section. Third, our calculations show that the average chance of missing a transfer in the current timetable is 34.68% while in our optimized timetable it is only 2.43%. The expected passenger time of our optimized schedule is 7.12% lower than the original schedule. This result is obtained with supplements on ride and dwell actions up to 15 minutes are allowed. Fourth, generating our schedule only takes 3706 seconds, while it takes many human planners many months to generate the current timetable.

9 Conclusions

This paper has four main contributions. Firstly, by making a graphical representation as in Figure 1 it becomes clear that we can decompose a general train network into actions of five different types: ride and dwell train actions and the source, transfer and sink passenger actions and then recombine pairs of subsequent actions into four local passenger flow types: departing, through, transfer and arriving.

Thanks to this, secondly, supposing a general negative exponential delay distribution for each of these actions, we analytically derived the probability and objective functions for each of these local passenger flows, according to what matters for them at that location. For transfers, this includes the probabilistic trade-off for catching or missing a next action. These local functions of one or two supplement variables, indicate the cost tradeoff and can be used to evaluate a given schedule on quality or even optimality of these supplements.

Thirdly, summing these local functions for the whole schedule delivers a global objective function that can be used by an optimizing solver directly. We also show how, for linear solvers, optionally, convex linear approximate functions can be set up.

Lastly, together with the necessary train schedule constraints, we show in Figure 1 that our method produces a schedule for two trains that, at face value, has apparent merits: efficiency as well as robustness. We also used the derived objective function in producing a passenger time optimal timetable for all 203 passenger trains in Belgium. This timetable is quickly generated and is conflictless, efficient and robust by construction.

References

[1] Dewilde, T., Sels, P., Cattrysse, D., Vansteenwegen, P., "Defining Robustness of a Railway Timetable", *Proceedings of 4th International Seminar on Railway Operations Modelling and Analysis (IAROR): RailRome2011*,Rome, Italy, 2011.

[2] Goverde, R.M.P., "Optimal Scheduling of Connections in Railway Systems", *Paper presented at the 8th WTCR*, Antwerp, Belgium, 1998.

[3] Goverde, R.M.P., "Synchronization Control of Scheduled Train Services to minimize Passenger Waiting Times", *Proceedings of 4th TRAIL Annual Congress*, part 2, TRAIL Research School, Delft, 1998.

[4] Goverde, R.M.P., "Improving Punctuality and Transfer Reliability by Railway Timetable Optimization", *Proceedings of the 5th TRAIL Annual Congress*, TRAIL Research School, Delft, 1999.

[5] Kroon, L., Dekker, R., Vromans, M.J.C.M., "Cyclic Railway Timetabling: A Stochastic Optimization Approach", Geraets, F., Kroon, L., Schöbel, A., Wagner, D., Zaroliagis, C.D. (eds), *Algorithmic Methods for Railway Optimization. Lecture Notes in Computer Science*, pp. 41-66, 2007.

[6] Liebchen, C., "Periodic Timetable Optimization in Public Transport", *Operations Research Proceedings*, 2007, Volume 2006, Part II, 29-36.

[7] Meng, Y., "Bemessung von Pufferzeiten in Anschlüssen von Reisezügen", W. Schwanhäußer, P. Wolf (Hrsg.) *Veröffentlichungen des Verkehrswissenschaflichen Institutes de Rheinischen-Wesfälischen Technischen Hochschule Aachen*, Heft. 46, Aachen, Deutschland. 1991.

[8] Nachtigall, K., "Periodic network optimization with different arc frequencies", *Discrete Applied Mathematics*, Vol. 69, pp. 1-17, 1996.

[9] Sels, P., Dewilde, T., Cattrysse, D., Vansteenwegen, P., "Deriving all Passenger Flows in a Railway Network from Ticket Sales Data", *Proceedings of 4th International Seminar on Railway Operations Modelling and Analysis (IAROR): RailRome2011*, Rome, Italy, 2011.

[10] Schrijver, A., Steenbeek, A., "Spoorwegdienstregelingontwikkeling (Timetable Construction)" *Technical Report, CWI Center for Mathematics and Computer Science, Amsterdam*, (in Dutch), 1993.

[11] Serafini, P., Ukovich, W., "A Mathematical Model for Periodic Scheduling Problems", *SIAM Journal on Discrete Mathematics*, Vol. 2, pp. 550-581, 1989.

[12] Vansteenwegen, P., Van Oudheusden, D., "Developing railway timetables which guarantee a better service", *European Journal of Operational Research"*, Vol. 173, pp. 337-350, 2006.

[13] Vansteenwegen, P., Van Oudheusden, D., "Decreasing the passenger waiting time for an intercity rail network", *Transportation Research Part B: Methodological*, Vol. 41, pp. 478-492, 2007.