# Georg-August-Universität
# Göttingen

## Identifying dependencies among delays

**Carla Conte and Anita Schöbel**

## Nr. 2007-05

Preprint-Serie des
Instituts für Numerische und Angewandte Mathematik
Lotzestr. 16-18
D - 37083 Göttingen

# Identifying dependencies among delays

Carla Conte Anita Schöbel
Georg-August-Universität Göttingen,
Institut für Numerische und Angewandte Mathematik,
Lotzestrasse 16-18, D-37083 Göttingen, Deutschland,
e-mail: conte@math.uni-goettingen.de
e-mail: schoebel@math.uni-goettingen.de

**Abstract**
Nowadays, railway transportation needs to become more and more competitive, so new features are required to improve the planning process. Since delay propagation is often considered as one of the main reasons for the poor attractiveness of railway transport the goal in daily operation is to compensate perturbations to the scheduled timetable, in particular to meet the passengers' needs concerning transfers and changes in a better way. This leads to the delay management problem in which dispositioners have to decide about the new timetable. Detailed knowledge about the critical points of the system (in particular about where the source delays are and how they spread out into the system) may make a better re-scheduling possible. To identify such dependencies, we apply a stochastic approach called Tri-graph. This is a graphical modeling approach in which full conditional modeling is carried out in small subgraphs with only three vertices that will then be combined into the full model. The idea of our approach is to describe the delay propagation by using a small set of abstract constraints instead of the inventory or headway constraints that are usually used in re-scheduling problems.
Our approach has been applied and tested on real-world data of German railway, provided by Deutsche Bahn within the context of a larger project named DisKon.

**Keywords**
train delays, dependencies, graphical method, tri-graph, capacity constraints, re-scheduling

## 1 Introduction

Nowadays, railway transportation needs to become more and more competitive, so new features are required to improve the planning process. In daily operations the goal is to compensate perturbations to the scheduled timetable, in particular to meet in a better way the passengers' needs concerning transfers and changes. This leads to the delay management problem. Delay propagation is often considered as one of the main reasons for the poor attractiveness of railway transport. In fact, a better re-scheduling of the timetable in order to minimize the disadvantages for the passengers will be possible if the critical points of the system are known. This includes knowledge about dependencies among delays in order to be able to point out where the source delays are and how they spread out into the system. Delays and their behavior in railway systems have recently been investigated by [19] and [6]. As usual in the literature (Ref: [9]) we distinguish between two types of delay: source delays, i.e. delays that are caused from the outside and not from other trains

(see “Urverspätungen” in [15]), which usually spread out into the system inducing a second kind of delay, called ”forced delay” (see “Folgeverspätungen” in [15]).
We further distinguish between the following three types of delay propagation:

1. propagation along the same train. Delay is carried over along the path of each delayed train, i.e. if a train starts with a delay it is likely to reach its next station with a forced delay (propagation along a driving activity), and if it arrives at a station with a delay it will probably depart with a delay (propagation along a waiting activity);

2. propagation from one train to another due to connections. If a connecting train waits for a delayed feeder train, the delay of the feeder train may spread out to the connecting train (propagation along a transfer activity);

3. propagation from one train to another due to the limited capacity of infrastructure. If two trains share the same infrastructure (a part of a track or a platform) one of them has to wait until the other has left. This is a third possibility to obtain forced delays for the trains that have to wait (propagation along a ”virtual activity”).

The first two types of delay propagation are easy to handle from an analytical point of view, since the minimal duration of every activity is known and is hence explicit and given parameters. However, the third kind of delay propagation is more complicated to deal with. This is due to the fact that it requires a detailed knowledge of the track system on a microscopical level. An overview about approaches dealing with the third type of delay propagation is given by [14]. Note that also classical inventory constraints (Ref: [13]) or job-shop-scheduling (Ref: [2] and [3]) may be used. In contrast to these approaches, the goal of this paper is to propose a procedure which enables us to detect dependencies of delays of the third type without explicit knowledge of all details of the infrastructure.

## 2 Analytical Method

The scheduling of a timetable can be considered as a project in which a set of interacting tasks (journeys of the trains) require time (e.g. driving time, waiting time . . . ) and resources (e.g. tracks, platforms . . . ) to be completed.
Given two sets, $\mathcal{T}$ for the trains and $\mathcal{V}$ for the stations, that have to be studied, we represent the railway system by a network, the so called ***Public Transportation Network*** $PTN = (\mathcal{V}, \mathcal{B})$ in which every node represents a station and every edge is a set of (blocks of) tracks connecting two different stations.
The PTN is intuitive but the information it contains is not enough to study the problem from an analytical point of view. That’s why we will instead consider the so called ***Activity-on-arc Project Network*** $\mathcal{N} = (\mathcal{E}, \mathcal{A})$ (Ref: [8] and [9]).
We define a set of events $\mathcal{E}$ corresponding to the arrivals and departures of all trains in all stations of their journeys, and a set of activities $\mathcal{A}$ (driving along an edge, waiting in a station or connection between two trains) so that

$$\mathcal{E} = \mathcal{E}^{\text{dep}} \cup \mathcal{E}^{\text{arr}} \tag{1}$$

where

$$\begin{array}{rcl} \mathcal{E}^{\text{dep}} & = & \{(t, v, \text{dep}) : t \in \mathcal{T}\ v \in \mathcal{V}\ :\ t \text{ departs from } v\} \\ \mathcal{E}^{\text{arr}} & = & \{(t, v, \text{arr}) : t \in \mathcal{B}\ v \in \mathcal{V}\ :\ t \text{ arrives in } v\} \end{array} ;$$

and

$$\mathcal{A} = \mathcal{A}^{\text{drive}} \cup \mathcal{A}^{\text{wait}} \cup \mathcal{A}^{\text{change}} \tag{2}$$

where

$$\begin{aligned}
\mathcal{A}^{\text{drive}} &= \{((t,v,\text{dep}),(t,u,\text{arr})) \in \mathcal{E}^{\text{dep}} \times \mathcal{E}^{\text{arr}} \ : \ v,u \in \mathcal{V}\} \\
\mathcal{A}^{\text{wait}} &= \{((t,v,\text{arr}),(t,v,\text{dep})) \in \mathcal{E}^{\text{arr}} \times \mathcal{E}^{\text{dep}}\} \\
\mathcal{A}^{\text{change}} &\subseteq \{((t,v,\text{arr}),(t',v,\text{dep})) \in \mathcal{E}^{\text{arr}} \times \mathcal{E}^{\text{dep}} \ : \ t,t' \in \mathcal{T}\}
\end{aligned}$$

where the set of guaranteed connections contained in $\mathcal{A}^{\text{change}}$ should be defined according to the passengers' needs.

The graph in Figure 1 is a small example of how an Activity-on-arc Project Network looks like when two connections between two trains are possible:

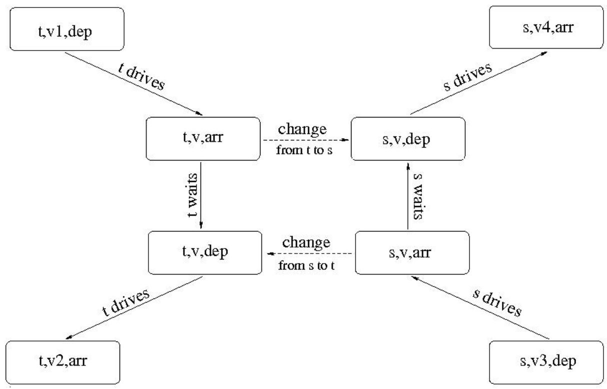


Figure 1: Example of Activity-on-arc Project Timetable Network

We define now the parameters

| | |
|---|---|
| $\pi_i$ | scheduled time of event $i \in \mathcal{E}$ |
| $d_i$ | source delay associated to event $i \in \mathcal{E},\ d_i \geq 0$ |
| $L_a$ | minimal duration of activity $a \in \mathcal{A}$ |

and the new variables

$x_i$ re-scheduled timetable of event $i \in \mathcal{E}$.

Consequently the delay of event $i$ is given by $x_i - \pi_i$. If we consider just delays of the first type (propagation along the same train) and of the second type (propagation from one train to another due to connections), we can write our Timetable Model [TM-1] as:

$$\min \quad \sum_{i \in \mathcal{E}} x_i \tag{3}$$

$$\text{s.t.} \quad x_j - x_i \geq L_a \quad \forall a = (i,j) \in \mathcal{A} \tag{4}$$

$$x_i \geq \pi_i + d_i \quad \forall i \in \mathcal{E} \tag{5}$$

$$x_i - \pi_i \leq T \quad \forall i \in \mathcal{E} \tag{6}$$

$$x_i \in \mathbb{Z}^+ \quad \forall i \in \mathcal{E} \tag{7}$$

The objective function $\sum_{i\in\mathcal{E}} x_i$ is equivalent to the delay function $\sum_{i\in\mathcal{E}}(x_i - \pi_i)$ since the scheduled timetable $\pi_i$ is a constant parameter of our model.
The constraints represent the time limits of our problem:

- the real duration of an activity must respect the (technically) minimal one, i.e. the real duration must be larger than the given lower bound;
- the real timetable must respect the scheduled one and the delays;
- the delay of an event must be smaller than the period $T$ of the model;
- the variables $x_i$ are in $\mathbb{Z}$ since minutes or seconds are the minimal time units of the system.

Constraint (6) is an ("implicit") condition in a periodic timetable. If the delay of a train at one station $y_i$ is greater than the period $T$ ($y_i > T$), it is preferable (in order to avoid delay propagation) to cancel the train and ask the passengers to get on the next scheduled train. This condition gives an upper bound for delays, that can be interpreted as a deadline for every activity of the system. Thus the Timetable Model can be read as a problem in which every activity has to be executed inside a time window (i.e. a time interval) defined by the scheduled timetable and the ("pre-defined") deadline:

$$x_i \geq \pi_i + d_i \text{ and } x_i - \pi_i \leq T$$

that is $x_i \in [\pi_i + d_i, \pi_i + T]$.

This is a broad interpretation of these "implicit" constraints since inside a periodic timetable, every set of trains traveling on the same route has a specific period, which is usually smaller than the general period $T$. For example Hannover and Göttingen are connected by a train every hour, but the trains between the main station in Hannover and the Hannover airport have a higher frequency. Therefore it makes sense to consider, instead of the constant period $T$ a specific period $T_i$ that depends on the route of train $t$ corresponding to event $i$. Hence we can rewrite the Timetable Model as [TM-2]

$$\begin{array}{lll} \min & \sum_{i\in\mathcal{E}} x_i & \\ \text{s.t.} & x_j - x_i \geq L_a & \forall a = (i,j) \in \mathcal{A} \\ & x_i \geq \pi_i + d_i & \forall i \in \mathcal{E} \\ & x_i - \pi_i \leq T_i & \forall i \in \mathcal{E} \\ & x_i \in \mathbb{Z}^+ & \forall i \in \mathcal{E} \end{array} \tag{8}$$

In case of its feasibility, [TM-2] can be solved by the critical path method (CPM) which looks for a longest path in the events-activity-network. In a connected network it always exists a longest path between two nodes of it, if and only if it does not contain any direct cycle with positive length. We can assume the absence of direct cycles with positive length since the event-activity-network is a time-expanded network such that a cycle would represent a sequence of meaningless precedences.
Now we want to introduce in the [TM-2] the third type of delay (propagation from one train

to another due to limited capacity of infrastructure), that is the one we mainly want to investigate. A possible way to proceed is to avoid any overlapping between two consecutive events. This can be interpreted as a capacity constraint since formally we forbid that two trains can use the same track/platform simultaneously.
To mathematically define these capacity constraints we could use the following Capacitated Timetable Model [CTM-1]

$$
\begin{aligned}
\min \quad & \sum_{i \in \mathcal{E}} x_i & & \\
\text{s.t.} \quad & x_j - x_i \geq L_a & & \forall a = (i, j) \in \mathcal{A} \\
& x_i \geq \pi_i + d_i & & \forall i \in \mathcal{E} \\
& x_i - \pi_i \leq T_i & & \forall i \in \mathcal{E} & (9) \\
& g_{ije}(x_j - x_i - L_a) \geq 0 & & \forall i, j \in S_e \text{ where } a = (i, k) \in \mathcal{A} \setminus \mathcal{A}^{change} & (10) \\
& (1 - g_{ije})(x_i - x_j - L_{a'}) \geq 0 & & \forall i, j \in S_e \text{ where } a' = (j, k') \in \mathcal{A} \setminus \mathcal{A}^{change} & (11) \\
& x_i \in \mathbb{Z}^+ & & \forall i \in \mathcal{E} \\
& g_{ij} \in \{0, 1\} & & \forall m \in \mathcal{M}\ \forall i, j \in \mathcal{E}
\end{aligned}
$$

Here $S_e$ refers to the set of (departure) events that use the same edge $e$ of the underlying physical network in their next (driving) activity and $g_{ije}$ is a binary variable equal to 1 if event $i \in S_e$ happens before event $j \in S_e$ on the edge $e$, 0 otherwise. The constraints (10) and (11) are the capacity constraints on every edge (before a new action starts on the edge, the previous scheduled one must have been completed). These constraints are more restrictive than what we need. In practice, two trains can simultaneously use the same edge. To guarantee the safety while doing so, each edge is separated into blocks as shown in Figure 2. We hence rewrite [CTM-1] in a more detailed formulation. Firstly we define the set $\mathcal{M}$ as the blocks (part of tracks or platforms) of the system. For every block $m \in \mathcal{M}$ we also define a set $S_m$ of events in $\mathcal{E}^{dep}$ that have to take place on it. Besides we introduce a new binary variable $g_{ijm}$ equal to 1 if event $i \in S_m$ happens before event $j \in S_m$, zero otherwise, and a new parameter $h_{ijm}$, called headway, corresponding to the security distance that the events $i$ and $j$ have to respect on block $m$.
Hence the Capacitated Timetable Model can be written as [CTM-2]

$$
\begin{aligned}
\min \quad & \sum_{i \in \mathcal{E}} x_i & & \\
\text{s.t.} \quad & x_j - x_i \geq L_a & & \forall a = (i, j) \in \mathcal{A} \\
& x_i \geq \pi_i + d_i & & \forall i \in \mathcal{E} \\
& x_i - \pi_i \leq T_i & & \forall i \in \mathcal{E} \\
& g_{ijm}(x_j - x_i - h_{ijm}) \geq 0 & & \forall m \in \mathcal{M}\ \forall i, j \in S_m & (12) \\
& (1 - g_{ijm})(x_i - x_j - h_{jim}) \geq 0 & & \forall m \in \mathcal{M}\ \forall i, j \in S_m & (13) \\
& x_i \in \mathbb{Z}^+ & & \forall i \in \mathcal{E} \\
& g_{ijm} \in \{0, 1\} & & \forall m \in \mathcal{M}\ \forall i, j \in \mathcal{E}
\end{aligned}
$$

Note that (12) and (13) can be replaced by

$$
\begin{aligned}
g_{ijm}(x_j - x_i - \overline{h}_{ijm}) & \geq 0 \quad \forall m \in \mathcal{M}\ \forall i, j \in S_m & (14) \\
(1 - g_{ijm})(x_i - x_j - \overline{h}_{jim}) & \geq 0 \quad \forall m \in \mathcal{M}\ \forall i, j \in S_m & (15)
\end{aligned}
$$

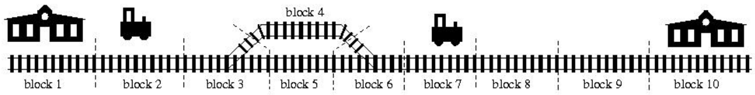


Figure 2: Blocks sections

where $\overline{h}_{ijm} = \max\limits_{\substack{m \in \mathcal{M} \\ i,j \in S_m}} h_{ijm}$. The model can be linearized or treated using disjunctive constraints, see the investigation in in [12].

Next we want to show that [CTM-2] is NP-complete. To this end we introduce a well-known NP-complete problem *Sequencing within Intervals Problem* [SIP] (Ref: [5]) . The [SIP] determines whether the tasks $c$ of a finite set $C$, with a given duration $l_c$, can be sequenced in order to obey temporal restrictions (the execution has to happen inside a pre-defined time interval), with at most one task ever being executed at a time. A formal description is the following: we define the temporal restriction of the [SIP] as a minimal starting time $p_c$ and a deadline $t_c$, and we are looking for a function $\sigma : C \to \mathbb{Z}^+$ such that for each $c \in C$ , $\sigma_c \geq p_c$, $\sigma_c + l_c \leq t_c$ and, if $c' \in C \backslash \{c\}$, then either $\sigma_{c'} + l_{c'} \leq \sigma_c$ or $\sigma_{c'} \geq \sigma_c + l_c$. The objective is to minimize the total tardiness of the problem, i.e. the sum of the completion times of the tasks in the set $C$. Accordingly to that we can write the [SIP] as

$$
\begin{array}{lrl}
\min & \sum\limits_{c \in C} \sigma_c & \\
\text{s.t.} & \sigma_c \geq p_c & \forall c, c' \in C \\
& \sigma_c + l_c \leq t_c & \forall c, c' \in C \\
& g_{cc'}(\sigma_c - \sigma_{c'} + l_c) \leq 0 & \forall c, c' \in C \\
& (1 - g_{cc'})(\sigma_{c'} - \sigma_c + l_{c'}) \leq 0 & \forall c, c' \in C \\
& \sigma_c \in \mathbb{Z}^+ & \forall c \in C \\
& g_{cc'} \in \{0, 1\} & \forall c, c' \in C
\end{array}
$$

where a variable $g_{cc'} = 1$ if task $c$ is executed before $c'$, 0 else. Consequently if task $c$ is started at time $\sigma_c$, it is completed at time $\sigma_c + l_c$, it cannot be started before time $p_c$, it must be completed by time $t_c$, and its execution cannot overlap the execution of any other task $c'$. We are now going to show that [CTM-2] can be rewritten as [SIP] and hence we will have that

**Proposition 2.1.** *Capacitated Timetable Model [CTM-2] is a NP-complete problem.*

*Proof.* Given an instance of [SIP] we interpret the tasks $c \in C$ as events $i \in \mathcal{E}$ of the [CTM-2] and obtain an instance of [CTM-2] by the following correspondences. We define $\mathcal{E} = \mathcal{E}^{dep} = C$ and also $S_m = C$ since we consider just one block, $\mathcal{M} = \{m\}$. Then we are able to define the parameters of [CTM-2] as follows.

$$
\begin{array}{rcll}
d_i & = & 0 & \forall i \in \mathcal{E} \\
\pi_i & = & p_i & \forall i \in \mathcal{E} \\
T_i & = & t_i - (p_i + l_i) & \forall i \in \mathcal{E} \\
h_{ijm} & = & l_i & \forall i, j \in \mathcal{E}
\end{array}
$$

where $T_i \geq 0$ otherwise the problem would be infeasible, since it would be required to complete a task in less than its minimal execution time. We also define $L_a = -\infty$ so that constraint (4) can be neglect from the model.
The result of [CTM-2] then is a timetable $x$, from which we obtain $\sigma : C \to \mathbb{Z}^+$ by the identity $\sigma_i = x_i$. It holds that $x$ is a feasible timetable if and only if $\sigma$ feasible for [SIP]. Furthermore, the objective values of $x$ and $\sigma$ are equal. Hence [CTM-2] is NP-complete. □

Discovering that our problem is NP-complete, we look for other ways to find a practicable model. We suggest a new approach based on the concept of "virtual activities", that will be described in Section 3.

## 3 Identifying dependencies through a stochastic approach

Instead of using all headway constraints for each block as in [CTM-2] we want to apply a stochastic procedure which points out the critical points of the system, in particular where the source delays are and how they spread out into the system. Thus we can reduce the [CTM-2] formulation by restricting the set of capacity constraints to a smaller set of abstract constraints which contains just these critical points.
Using the delays of each event in this model as random variables we use a stochastic approach to analyze the dependencies among these variables. This reveals the dependencies among the events of the system, i.e. among the arrival and departure events of the trains in the stations. These dependencies represent information about all three types of delay propagation. While delay propagation of type 1 and 2 belongs to driving, waiting and changing activities, delay propagation of type 3 does not correspond to any $a \in \mathcal{A}$. Hence we introduce a set of "virtual activities" $\mathcal{A}^{virtual}$ describing the dependencies of type 3 which have to be identified by the stochastic approach. These activities ensure that an event can not happen before another event has taken place: that means, for example, that a train can not enter a station before the platform assigned to it, is free. Hence a "virtual activity" does not belong to the set of activities defined in $\mathcal{A}$ but it can be considered as a precedence constraint in the railway problem, that has to be satisfied to avoid infrastructure conflicts (using the same track or the same platform) due to the limited capacity of the track system and to operational rules of the security system. The resulting model is similar to [CTM-2]:

$$\begin{array}{lll}
\min & \sum_{i \in \mathcal{E}} x_i & \\
\text{s.t.} & x_j - x_i \geq L_a & \forall a = (i, j) \in \mathcal{A} \\
& x_i \geq \pi_i + d_i & \forall i \in \mathcal{E} \\
& x_i - \pi_i \leq T_i & \forall i \in \mathcal{E} \\
& x_j \geq v^1_{ij} x_i + v^2_{ij} & \forall i, j \in \mathcal{E} \text{ such that } a = (i, j) \in \mathcal{A}^{virtual} \\
& x_i \in \mathbb{Z}^+ & \forall i \in \mathcal{E}
\end{array} \qquad (16)$$

where we replace constraints (10) and (11) with linear dependency constraints (16) between pairs of events connected with "virtual" edges in the set $\mathcal{A}^{virtual}$. We call these constraints *virtual constraints*. The parameters $v^1_{ij}$ and $v^2_{ij}$ will be estimated by the stochastic approach (see Section 7). Consequently we restrict the huge set of capacity constraints of [CTM-2] just to the crucial ones corresponding to critical points of the system.

## 4 Graphical Methods

Graphical methods have their origin in several scientific areas and they can be considered as a marriage between probability theory and graph theory. They provide a natural tool for dealing with uncertainty and complexity. The basic idea is the notion of modularity, so that a complex system can be built by combining simpler parts. Probability theory provides the glue whereby the parts are combined, ensuring that the system as a whole is consistent and providing ways to interface models and data. Graph theory provides both an intuitively appealing interface by which humans can model highly-interacting sets of variables as well as data structures that lend themselves naturally to design general-purpose algorithms. Therefore we can define them as a sort of multivariate analysis that uses graphs to represents models.

Probabilistic graphical models are graphs in which the nodes represent random variables and the (lack of) arcs represents conditional independence. Hence they are a compact representation of a multi-variate probability distributions.

We first review some methods based on the multivariate normal distribution. Two of them (*Full Conditional Independent Graph* and *Covariance Graph*) are classical methods, the third one (*Tri-graph*) has been suggested in 2004 by Wille and Bühlmann and it is the method that will be mainly applied in this work.

**Gaussian Assumption**:
We suppose we have a p-dimensional random variable $X = \{X_1 \dots X_p\}$ with a multivariate normal distribution:

- mean $\mu = (\mu_1 \dots \mu_p)$
- covariance matrix $\Sigma = (\sigma_{ij})$ where $i, j \in \{1 \dots p\}$ and $\sigma_{ij} = cov(X_i\ X_j)$
- precision matrix $\Omega = \Sigma^{-1} = (\omega_{ij})$ where $i, j \in \{1 \dots p\}$
- density $f_X = \dfrac{1}{(2\pi)^{\frac{p}{2}}} \dfrac{1}{|\Sigma|^{\frac{1}{2}}} exp(-\frac{1}{2}(x-\mu)^T \Sigma^{-1}(x-\mu))$

**Full Conditional Independence Graph**

In the Full Conditional Independence Graph (FCIG), an edge between two variables $X_i$ and $X_j$ is drawn if and only if the two variables are conditionally dependent given all the other variables $X_k\ \forall k \in \{1 \dots p\} \setminus \{i, j\}$ of the system. Due to the Gaussian assumption, we can rewrite that condition as:

$$\text{we draw an edge } X_i \to X_j \iff \overline{\omega}_{ij} = \frac{\omega_{ij}}{\sqrt{\omega_{ii}\omega_{jj}}} \neq 0 \tag{17}$$

where $\omega_{ij}$ is the element in the $i^{\text{th}}$ row and $j^{\text{th}}$ column of the precision matrix $\Omega$ and $\overline{\omega}_{ij}$ is the **partial correlation coefficient**.

The method has the advantage of including all the variables in the evaluation of the dependencies. Its weak point is the numerical implementation. To calculate the inverse of the covariance matrix it is necessary to have a large sample of data for an accurate estimation. Moreover, to determine which elements of the precision matrix are equal to zero, it is necessary to do a super exponentially number of tests (likelihood tests), and for a large number of variables this is hardly tractable.

**Covariance Graph**

A natural choice to avoid the problem arising form the FCIG would be to work just with the covariance matrix, and not with the precision matrix. Then it would be possible to work with a small number of data and a large set of variables. Based on this idea the Covariance Graph (CG) draws an edge between two variables $X_i$ and $X_j$ if and only if the correlation coefficient of the two variables is different from zero:

$$\text{we draw an edge } X_i \to X_j \iff \rho_{ij} = \frac{\sigma_{ij}}{\sqrt{\sigma_{ii}\sigma_{jj}}} \neq 0 \tag{18}$$

where $\sigma_{ij}$ is the element in the $i^{\text{th}}$ row and $j^{\text{th}}$ column of the covariance matrix $\Sigma$ and $\rho_{ij}$ is the **correlation coefficient**.

The disadvantage of this method is the neglect-ion of possible interactions of all other variables in the explanation of the dependency of a particular pair, i.e. this method is not able to capture complex patterns.

**Tri-graph**

The Tri-graph (TG) arises from the combination of the two previous methods (Ref: [16] and [17]). This is a simplified graphical modeling in which full conditional modeling is carried out in small subgraphs with only three vertices that will be then combined into the full model. The vertices considered in the re-scheduling problem are the events of the Activity-on-arc Project Network defined in the Timetable Model.

To evaluate the conditional dependency of a pair of variables $X_i$ and $X_j$ proceed as follows. Instead of considering all the other variables simultaneously as in the FCIG, each other variable $X_k$, $k \in \{1 \dots p\} \setminus \{i, j\}$, is considered separately. The **pairwise partial correlation coefficients** are defined as

$$\omega_{ij|k} = \frac{\rho_{ij} - \rho_{ik}\rho_{kj}}{\sqrt{(1-\rho_{ik}^2)(1-\rho_{kj}^2)}} \tag{19}$$

The definition of the TG can be formalized as:

**Definition 4.1.** (see [16]) We draw an edge $X_i \to X_j$ if and only if $\rho_{ij} \neq 0$ and $\omega_{ij|k} \neq 0$ for all $k \in \{1 \dots p\} \setminus \{i, j\}$ .

The name Tri-graph refers to the fact that triples of variables are considered to calculate the pairwise partial coefficients. Its definition can be re-read as: an edge between two variable, $X_i$ and $X_j$, is drawn if and only if it does not exist any other variable $X_k$ that can explain their dependency.

Through the definition of the set $T_{ij} = \{\rho_{ij},\ \omega_{ij|k}\ \ \forall k \in \{1 \dots p\} \setminus \{i, j\}\}$ we can rewrite Definition 4.1 as

**Definition 4.2.** (see [16]) We draw an edge $X_i \to X_j$ if and only if $\tau = \arg\min_{\tau \in T_{ij}} |\tau| > 0$ .

The method is based on the covariance matrix like the Covariance Graph, so it can be used even if the sample of data is small compared to the number of variables. Moreover it has the advantage to look for more complex structures than the correlation, so it can capture the Full Conditional Independence Graph, but just in some cases it coincides with it.

## 4.1 Examples

To see how the three methods work, two small examples will be presented.

**Example 4.1.** A railroad yard is located along a single track route between the stations $v$ and $u$. Two trains $t$ and $s$ travel in different directions along this track. Figure 3 shows the Activity-on-arc Project Network of the problem. We suppose that the train $t$ leaves the

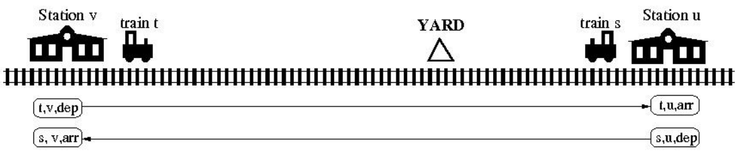


Figure 3: construction area on a single track line

station $v$ on time but at the yard it will get a source delay. Train $s$ has to wait at station $u$ until the arrival of train $t$ to receive the green light to proceed. Therefore it will have a forced delay, that might spread out along its journey.
We consider four variables:

- $X_1$ the departure of train $t$ from station $v$;
- $X_2$ the arrival of train $t$ at station $u$;
- $X_3$ the departure of train $s$ from station $u$;
- $X_4$ the arrival of train $s$ at station $v$.

We generated some normal distributed data corresponding to the following covariance matrix:

$$\Sigma = \begin{pmatrix} 0.8 & 0.6 & 0.4 & 0.2 \\ 0.6 & 1.2 & 0.8 & 0.4 \\ 0.4 & 0.8 & 1.2 & 0.6 \\ 0.2 & 0.4 & 0.6 & 0.8 \end{pmatrix}$$

Thus the precision matrix is:

$$\Omega = \begin{pmatrix} 2 & -1 & 0 & 0 \\ -1 & 2 & -1 & 0 \\ 0 & -1 & 2 & -1 \\ 0 & 0 & -1 & 2 \end{pmatrix}$$

To see how the Tri-graph works, we recall the definitions (19) and (4.1) and we compute the value of the partial correlation coefficients:

| $(X_i,X_j)$ | $X_k$ | $\omega_{ij\|k}$ | Edge |
|---|---|---|---|
| (1,2) | $\forall k$ | $\neq 0$ | Yes |
| (1,3) | 2 | $= 0$ | NO |
| (1,4) | 3 | $= 0$ | NO |

| $(X_i,X_j)$ | $X_k$ | $\omega_{ij\|k}$ | Edge |
|---|---|---|---|
| (2,3) | $\forall k$ | $\neq 0$ | Yes |
| (2,4) | 3 | $= 0$ | NO |
| (3,4) | $\forall k$ | $\neq 0$ | Yes |

The results of the three methods are shown in Figure 4. In our example the Tri-graph

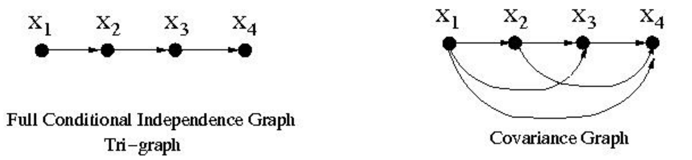


Figure 4: Tri-graph coincides with FCIG

method coincides with the FCIG and furthermore it's able to identify the "virtual connection" between trains $t$ and $s$. This is a really good result, but it is not always true.

**Example 4.2.** We consider three stations $v$, $u$, $w$ among which three trains $r$, $s$ and $t$ are traveling such that the corresponding Activity-on-arc Network is the one shown in Figure 5.

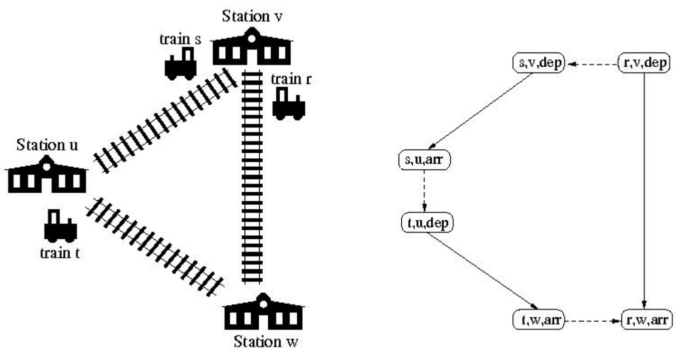


Figure 5: triangular connection

The three trains are pairwise connected in the three stations:

- $r$ and $s$ at station $v$;
- $s$ and $t$ at station $u$;
- $t$ and $r$ at station $w$.

Therefore the passengers have two possibilities to travel from station $v$ to $w$: directly with train $r$, or using the connection between $s$ and $t$ at station $u$. In reality such a situation often occurs: for example among Göttingen, Hannover und Wolfsburg.
Six variables are considered:

- $X_1$ the departure of train $r$ from station $v$;
- $X_2$ the departure of train $s$ from station $v$;
- $X_3$ the arrival of train $s$ at station $u$;
- $X_4$ the departure of train $t$ from station $u$;
- $X_5$ the arrival of train $t$ at station $w$;
- $X_6$ the arrival of train $r$ at station $w$.

We suppose that the precision matrix $\Omega$ obtained from the delay data of the three trains is

$$\Omega = \begin{pmatrix} 1 & 0.5 & 0 & 0 & 0 & 1 \\ 0.5 & 4 & 1 & 0 & 0 & 0 \\ 0 & 1 & 3 & 0.5 & 0 & 0 \\ 0 & 0 & 0.5 & 2 & 1 & 0 \\ 0 & 0 & 0 & 1 & 1 & 0.3 \\ 1 & 0 & 0 & 0 & 0.3 & 2 \end{pmatrix}$$

The corresponding covariance matrix is dense, thus the CG will give the complete set of possibles edges among the six variables as result, while the FCIG contains just the edges of a cycle, as shown in Figure 6.

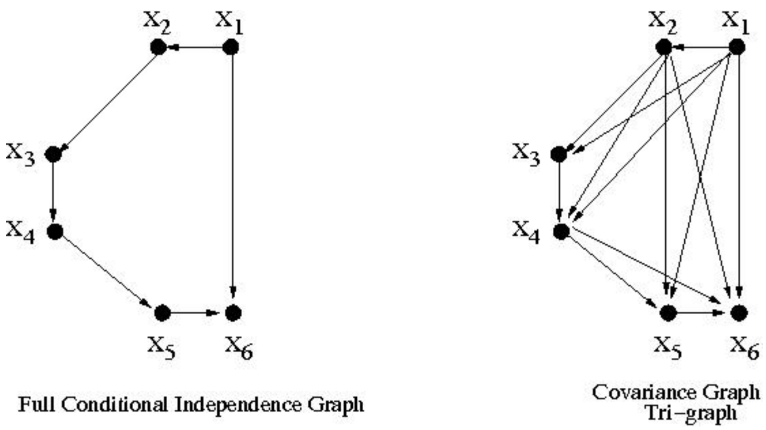


Figure 6: Tri-graph coincides with CG

Evaluating the partial correlation coefficients for the Tri-graph we get:

| $(X_i,X_j)$ | $X_k$ | $\omega_{ij\|k}$ | Edge |
|---|---|---|---|
| (1,2) | $\forall k$ | $\neq 0$ | Yes |
| (1,3) | $\forall k$ | $\neq 0$ | Yes |
| (1,4) | $\forall k$ | $\neq 0$ | Yes |
| (1,5) | $\forall k$ | $\neq 0$ | Yes |
| (1,6) | $\forall k$ | $\neq 0$ | Yes |
| (2,3) | $\forall k$ | $\neq 0$ | Yes |
| (2,4) | $\forall k$ | $\neq 0$ | Yes |
| (2,5) | $\forall k$ | $\neq 0$ | Yes |

| $(X_i,X_j)$ | $X_k$ | $\omega_{ij\|k}$ | Edge |
|---|---|---|---|
| (2,6) | $\forall k$ | $\neq 0$ | Yes |
| (3,4) | $\forall k$ | $\neq 0$ | Yes |
| (3,5) | $\forall k$ | $\neq 0$ | Yes |
| (3,6) | $\forall k$ | $\neq 0$ | Yes |
| (4,5) | $\forall k$ | $\neq 0$ | Yes |
| (4,6) | $\forall k$ | $\neq 0$ | Yes |
| (5,6) | $\forall k$ | $\neq 0$ | Yes |

Hence the Tri-graph method coincides with the CG and is not able to identify only the dependencies pointed out by the FCIG.

In general, the following theorem holds.

**Theorem 4.1.** *(see [16]) If the Full Conditional Independence Graph does not contain any cycle, then the Tri-graph coincides with the Full Conditional Independence Graph.*

The proof of the theorem (that can be found in [16]) is based on the global Markov property.

## 5 Implementation

The implementation of the three graphical methods has been done in R (Ref: [10]). The input are the delay data and a quantile (in our case of 5%) for the likelihood test to control if the covariance and the partial correlation coefficients are different from zero.
In our application we want to reduce the possibility to neglect existing edges, i.e. to make an error of the second kind.
Special features for the Tri-graph have been already implemented and included in public libraries (Ref: [10]).

## 6 Numerical Experiments

The Tri-graph approach has been applied and tested on real-world data of German railway, corresponding to the Harz region. The data has been provided by Deutsche Bahn within the context of a larger project called DisKon (Ref: [1]).

The data files consist of delay of regional trains over a period of nine months (between January and September 2005) at eight stations in the Harz area: Bad Harzburg, Goslar, Herzberg, Oker, Salz-Ringelheim, Seesen, Vieneburg and Wolfenbüttel (see Figure 7). Apart from the delay we also have data about the timetable and the infrastructure in this region such that we are able to analyze the stochastic results.
We considered 928 events corresponding to 177 regional trains traveling inside the Harz area in 30 working days. The choice of the (number of) days has been done in order to maximize the size of the set of events $\mathcal{E}$ that could be defined from the list of events in the data file. We decided not to generate any missing data, therefore we do not always have complete chains of events for the train journeys.

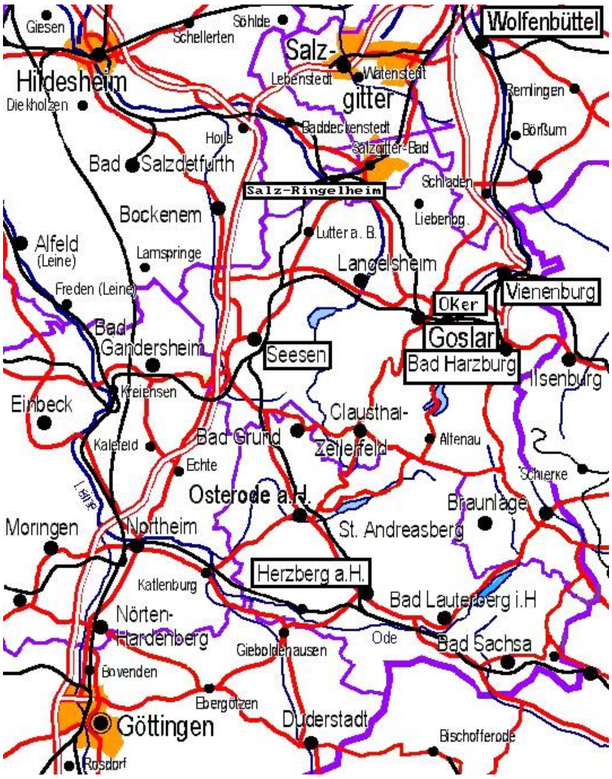


Figure 7: The Harz area

The Full Conditional Independence Graph could not be applied since the delays corresponding to some pairs of events were identical. This is due to the fact that the measurement precision in the data is up to the minute, so if an activity has a small slack time there will be no significant change in the measured delay. As a consequence there are linear dependencies between some columns of the covariance matrix, and hence the inverse does not exist.

The Covariance Graph pointed out around 18.000 edges. Most of them were due to the transitivity property of the covariance: if the correlation coefficient between the couples of events $(i,j)$ and $(j,k)$ are different from zero, $\rho_{ij} \neq 0$ and $\rho_{jk} \neq 0$, then it follows automatically that $\rho_{ik} \neq 0$. Hence if the CG point out the edges $(i,j)$ and $(j,k)$, it will automatically point out also the edge $(i,k)$.

The Tri-graph identified 182 edges, that can be subdivided into four classes:

- 132 (72.5%) edges correspond to waiting activities;
- 28 (15.4%) edges correspond to driving activities;
- 11 (6.06%) edges correspond to " virtual activities" and describe dependencies due to capacity constraints;
- 11 (6.06%) are "not clear";

The method identified more waiting activities than driving activities since the waiting activities usually have a smaller slack time, hence with a high probability delays will spread out more often along them than along driving activities. We have classified events as "not clear" if the corresponding edge connects events occurring in stations too far away from each other or events with a too large time distance to each other.

## 7 Including "virtual connections" in the analytical model

) In order to replace the classical inventory constraints (10) and (11) in the [CTM-2] we want to include the "virtual connections" pointed out with the Tri-graph, gathered in the set $\mathcal{A}^{virtual}$ as described in Section (3). Furthermore we want to use the information about the mean and variance of the delays arising form the stochastic analysis. Our idea is to suppose linear dependence among the delays, thus if events $i$ and $j$ are connected by an edge, the corresponding normal distributed delay variables can be correlated as

$$y_j = \alpha_{ij} y_i + \beta_{ij} \tag{20}$$

To estimate the parameters $\alpha_{ij}$ and $\beta_{ij}$ we use some basic properties of the variance and the mean of a random variable $Y$:

- adding a constant $\beta$ to $Y$ does not affect the variance, i.e. $Var(Y + \beta) = Var(Y)$, and increases the expected value by that constant, i.e $E(Y + \beta) = E(Y) + \beta$;

- multiplying $Y$ by a constant $\alpha$ increases the variance by the square of the constant, i.e. $Var(\alpha Y) = \alpha^2 Var(Y)$, and multiplies the expected value by that constant, i.e. $E(\alpha Y) = \alpha E(Y)$.

Consequently in our case we have

$$\begin{aligned} Var(y_j) &= \alpha_{ij}^2 Var(y_i) &(21)\\ E(y_j) &= \alpha_{ij} E(y_i) + \beta_{ij} &(22) \end{aligned}$$

From (21) we obtain

$$\alpha_{ij} = \pm \sqrt{\frac{Var(y_j)}{Var(y_i)}}$$

We consider $Var(y_j) \neq 0$ since we neglect the case of a constant delay (otherwise we should speak about ill-posedness of the timetable). Moreover we restrict the parameter $\alpha_{ij}$ to be positive in order to represent positive dependencies between delays, therefore

$$\overline{\alpha}_{ij} = \sqrt{\frac{Var(y_j)}{Var(y_i)}}$$

Inserting the value of $\overline{\alpha}_{ij}$ in (22) we get:

$$\overline{\beta}_{ij} = E(y_j) - \overline{\alpha}_{ij} E(y_i) = E(y_j) - \sqrt{\frac{Var(y_j)}{Var(y_i)}} E(y_i) \ .$$

Recalling the definition of delay as $x_i - \pi_i$, we can rewrite (20) using the re-scheduled timetable variable of [TM-2]

$$x_j - \pi_j \leq \overline{\alpha}_{ij}(x_i - \pi_i) + \overline{\beta}_{ij} \tag{23}$$

where we prefer to introduce an inequality to better represent the stochastic nature of the delays. As a result we can write the following Virtual Capacity Timetable Model [VCTM]:

$$\begin{array}{lrl}
\min & \sum_{i \in \mathcal{E}} x_i & \\
\text{s.t.} & x_j - x_i \geq L_a & \forall a = (i,j) \in \mathcal{A} \\
& x_i \geq \pi_i + d_i & \forall i \in \mathcal{E} \\
& x_i - \pi_i \leq T_i & \forall i \in \mathcal{E} \\
& x_j \geq \pi_j + \overline{\alpha}_{ij}(x_i - \pi_i) + \overline{\beta}_{ij} & \forall i, j \in \mathcal{E} \ : \ a = (i,j) \in \mathcal{A}^{virtual} \\
& x_i \in \mathbb{Z}^+ & \forall i \in \mathcal{E}
\end{array}$$

In the next future we will implement like to solve this model with some appropriate classical optimization techniques (as the CPM method) and compare the results.

## 8 Conclusion and future research

We presented the application of a graphical method to identify dependencies among train delays. The algorithm completes its task in a relatively short time (e.g. for the numerical experiment presented in Section 6 it needs less than 15 minutes) and it does not require detailed knowledge about the tracks and platforms of the system. Comparisons of its performance have been carried out against the Covariance Graph and the Full Conditional Independence Graph. FGIC can be used only for small sets of data, because of the high probability to have linear dependencies between columns of the covariance matrix. In all larger numerical experiments the number of edges computed by the Tri-graph is significantly smaller than the one obtained with the Covariance Graph. The ability to generate a manageable set of new constraints by the Tri-graph method partially justifies the larger amount of calculations to be performed compared to direct covariance evaluation. The outcome of several numerical experiments shows that Tri-graph results can be expected to lie somewhere in between the results of the other two classical graphic methods. However Tri-graph has the advantages of being based on the covariance matrix (and not on its inverse precision matrix as FCIG) and of being able to detect more complex relations than CG.
We currently work on implementing a solution algorithm for [VCTM] by classic optimization techniques and on comparing the results with those of a plain analytical model.

**Acknowledgment** This work was partially supported by the Future and Emerging Technologies Unit of EC (IST priority - 6th FP), under contract no. FP6-021235-2 (project ARRIVAL). The authors are grateful to Harald Börner at Deutsch Bahn for his helpful comments, and for providing the data for the numerical study. We also want to thank Dr. Juliane Schäfer at ETH Zürich for the fruitful discussions about Tri-graph.

## References


[1] Bissantz N., Güttler S., Jacobs J., Kurby S., Schaer T., Schöbel A. and Scholl S., *"DisKon ¢ Disposition und Konfliktlösungsmanagement für die beste Bahn"*, Eisenbahntechnische Rundschau (ETR),12/05 pp.809-821, 2005.

[2] Brucker P., Heitmann S. and Knust S., *Scheduling railway traffic at a construction site*, OR Spectrum (2002) 24: 19-30.

[3] Brucker P. and Knust S., *Lower bounds for resource-constrained project scheduling problems*, European Journal odf Operational Research (2003) 149: 302-313.

[4] Edwards D., *"Introduction to Graphical Modelling"*, Springer $2^{nd}$ edition, New York, 2000.

[5] Garey M. and Johnson D., *Computer and Intractability. A guide to the theory of NP-completeness*, W.H.Freeman and Company, New York, 1979.

[6] Güttler S., *Statistical Modeling of Railway Data*, Diplom thesis at Georg-August-Universität, Göttingen, 2006.

[7] Jacobs J., *Rechnerunterstützte Konfliktermittlung und Entscheidungsunterstïzung bei der Disposition des Zuglaufs*, Dissertation, Veröffentlichungen des Verkehrswissenschaftlichen Institutes der RWTH Aachen, Heft 61 (2003).

[8] Nachtigall K., *"Periodic Network Optimization and fixed interval timetables"*, Habilitation, Deutsche Zentrum für Luft- und Raumfahrt, Institut für Flugführung, Braunschweig 1998.

[9] Pachl J., *"Systemtechnik des Schienenverkers. Bahnbetrieb planen, steuern und sichern"*, Teubner, 2004.

[10] The program R can be for free downloaded at: http://www.r-project.org. The necessary packages to implement the graphical methods can be found at: http://cran.r-project.org.

[11] Schöbel A., *"Customer-Oriented Optimization in Public Transportation"*, Springer, 2006.

[12] Schöbel A., *Capacity constraints in delay management*, Georg-August-Universität, Göttingen, Germany, CASPT 2006, Leeds, UK, June 21-23, 2006.

[13] Schwindt C., *"Resource Allocation in Project Management"*, Springer, 2005.

[14] Törnquist J., *Railway traffic disturbance management*, Doctoral thesis at Blekinge Institute of Technology, Karlshamm, Sweden, 2006.

[15] UIC Code 406, DB 405.0102 Anhang 3 Tab. 301

[16] Wille A. and Bühlmann P., *"Tri-graph: a Novel Graphical Model with Application to Genetic Regulatory Network"*, ETH Zürich, Switzerland, 2004.

[17] Wille A. and Bühlmann P., *"Low-Order Conditional Independence Graphs for Inferring Genetic Networks"*, ETH Zürich, Switzerland, 2006.

[18] Wegele S. and Schnieder E., *Dispatching of train operations using genetic algorithms*, Technical University Braunschweig, Germany, submitted to CASPT 2004, San Diego, California, August 9-11, 2004

[19] Yuang J., *Stochastic modeling of train delays and delay propagation in stations*, Doctoral thesis at TU Delf, The Nederland, Eburon Academic Publisher, 2006.

Institut für Numerische und Angewandte Mathematik
Universität Göttingen
Lotzestr. 16-18
D - 37083 Göttingen

Telefon: 0551/394512
Telefax: 0551/393944

Email: trapp@math.uni-goettingen.de URL: http://www.num.math.uni-goettingen.de

## Verzeichnis der erschienenen Preprints 2007:

| | | |
|---|---|---|
| 2007-01 | P. Serranho | A hybrid method for inverse scattering for sound-soft obstacles in 3D |
| 2007-02 | G. Matthies, G. Lube | On streamline-diffusion methods for inf-sup stable discretisations of the generalised Oseen Problem |
| 2007-03 | A. Schöbel | Capacity constraints in delay management |
| 2007-04 | J. Brimberg, H. Juel, A. Schöbel | Locating a minisum circle on the plane |
| 2007-05 | C. Conte, A. Schöbel | Identifying dependencies among delays |