Mixture-model identification in traffic control problems

Ivan Nagy, Miroslav Kárný,
Jana Novovičová, Markéta Valečková

Institute of Information Theory and Automation, AVČR, P.O. Box 18, 182 08
Prague 8, Czech Republic, school@utia.cas.cz

November 23, 1999

Contents

This work has been supported by grants GAČR 102/97/0118 and 102/99/1564. Their aid is gratefully acknowledged.

1 Introduction

In applications one can often meet situations when the identified and controlled system is changing during the control. If the changes are "slow enough" and such, that they do not influence the structure of the identified model (e.g. do not increase the model order), we can cope with them by introducing some kind of more or less sophisticated forgetting (see e.g. [1]). But if the changes of the controlled system are too quick or (and) they even alter the model structure we have to look for an alternative solution.

Such cases occur typically in traffic-control area. Unless we need to go very deep into the analysis of traffic modeling we know, that traffic situations on roads of big towns are extremely different in the morning, at noon, in the evening and at night. Sometimes one is convinced that the only possibility for traveling in big towns by car is only at night. Fortunately, as it has been indicated, very often, instead of continuously changing single model it is sufficient to consider a finite set of basic components and to construct the model as their mixture (see [2]). The models are identified in parallel. The information carried by newly measured data is distributed to each component with a weight given by the degree of membership of the component to the data. The outcome of the algorithm is not only the estimate of parameters of all model components, but also an estimate of actual "state" – the most suitable component for the present moment can be selected. In this way the "states" of traffic can be modelled and the actual one estimated.

2 Problem formulation

General formulation

There are many different approaches to traffic control problems [3], [4], [5], [6], [7]. The control is necessary on highways or in towns [8]. It consists in making free some preferred directions through specified traffic areas or it prefers public traffic (busses) to private cars [9], [10], [11]. The means of control can be signal lights on crossroads or some more drastic actions as, for example, temporarily restricting traffic in chosen roads for private cars.

The paper focuses on identification intended for an automatic traffic control in big cities by means of traffic lights in crossroads. We suppose, the data (*traffic density* - number of cars per unit length and *traffic volume* - number of cars per unit of time) are measured on a road before crossroads. For description of the behaviour of cars in the crossroads, we shall consider a mixture of regression-type models with a finite number of components. Each component will reflect one of the states of the traffic. Structures of the components are supposed to be a priori defined up to a finite set of unknown parameters. They are "known", but we do not know which of them is "active" for the present time instant.

Our goal is to identify parameters of the mixture-model and, at each time instant, to select the single model (component) witch suits best to the newly measured data.

Mathematical formulation

A sequence of data $d(t), t \in t^*$ is measured on a system which can switch between a finite amount of states $f_t, t \in t^*$. Each of the states has its own *"data-model"* (component of a mixture), and it is represented by its conditional probability density function (pdf)

$$p(d_t|d(t-1), f_t, \Theta) = p(d_t|d(t-1), f_t, \Theta_{f_t}) \tag{1}$$

where the following denotations have been introduced:

d_t	currently measured data
$d(t-1)$	data measured up to the time $t-1$
$f_t \in f^*$	current state, $\quad f^* = \{1, 2, ..., \bar{f}\}$
Θ	vector of unknown model parameters for all states
Θ_{f_t}	subvector of Θ, corresponding to the f_tth state.

The states of the system f are supposed to be mutually independent and to have unknown but time invariant probabilities. This assumption gives a *"class-model"* describing probabilities of particular states (components of the mixture) in the form

$$p(f_t|d(t-1), \alpha) = p(f_t|\alpha) = \alpha_{f_t} \tag{2}$$

with

α denoting probabilities of the corresponding states (vector of parameters of the class-model).

Our goal is to predict the future data and to estimate the actual state using measured data, i.e. we are looking for the conditional pdfs

$$p(d_t|d(t-1)) \text{ and } p(f_t|d(t)). \tag{3}$$

The predictions may serve either as an advice for the operator who controls the traffic or they can be used directly for automatic control. State estimates may be used for detecting "faulty" states of the traffic.

3 Solution for pdfs

In this section, we give a formal solution to the estimation problem, i.e. solution in terms of conditional pdfs without specifying the real structure of the data-model components (1) containing the unknown parameter Θ. The class-model is given by (2), with the unknown parameter α. Our goal is to write an algorithm for recomputing prior pdf (describing unknown variables and using information up to time instant $t-1$) to posterior pdf (describing the same unknown variables but using information up to time instant t). The algorithm is well known Bayes rule, incorporating the actual information through the system model. Here, it is modified (and later on approximated) so that it can be practically used for mixture-models.

Structure of the prior joint pdf

The prior pdf at time instant t is a joint pdf $p(\alpha, \Theta, f_t|d(t-1))$ of unknown variables at the time instant t conditioned by all measured data $d(t-1)$ up to and including time instant $t-1$. It is posterior pdf from the previous time instant $t-1$. It can be decomposed according to the chain rule (see [12]). After decomposition and omission superfluous variables from the conditions, we get

$$p(\alpha, \Theta, f_t|d(t-1)) = p(f_t|\alpha)\ p(\alpha|d(t-1))\ p(\Theta|d(t-1)). \tag{4}$$

Bayes rule

The posterior pdf $p(\alpha, \Theta, f_t|d(t))$ has the same form as the prior one, but it is conditioned by data up to and including time instant t. This pdf can be recursively updated according to the Bayes rule

$$\begin{aligned} p(\alpha, \Theta, f_t|d(t)) \propto p(\alpha, \Theta, f_t, d_t|d(t-1)) = \\ = p(d_t|d(t-1), f_t, \Theta_{f_t})\ p(\alpha, \Theta, f_t|d(t-1)) \end{aligned} \tag{5}$$

using data-model $p(d_t|d(t-1), f_t, \Theta_{f_t})$ for the current data item d_t and the prior pdf (posterior one from the past step) $p(\alpha, \Theta, f_t|d(t-1))$. The variable Θ_{f_t} in the condition of the data-model denotes parameters corresponding to the f_tth state (i.e. the f_tth model component).

Results

From the joint posterior pdf (5), all other needed pdfs can be derived:

Parameter estimates

Estimates of data and class-model parameters can be computed as arguments, maximizing their posterior joint pdf

$$p(\alpha, \Theta|d(t)) = \sum_{f_t} p(\alpha, \Theta, f_t|d(t)) \tag{6}$$

with respect to α and Θ or as expected values of α, Θ.

State estimates

The state estimating pdf $p(f_t|d(t))$ can be constructed by integration of the pdf (5) over the parameters

$$p(f_t|d(t)) = \int\int p(\alpha, \Theta, f_t|d(t))\mathrm{d}\Theta\mathrm{d}\alpha \tag{7}$$

The maximally probable states (components) at the current time can be accepted as their estimates.

Data prediction
The data predictive pdf $p(d_t|d(t-1))$ (if necessary for control) is proportional to the sum over al states of the same integral as for state estimation but now from the right hand side of (5)

$$p(d_t|d(t-1)) \propto \sum_{f_t}\int\int p(\alpha, \Theta, d_t, f_t|d(t-1))\mathrm{d}\alpha\mathrm{d}\Theta \tag{8}$$

4 Solution for statistics

Product form of the pdfs

From computational viewpoint it is advantageous to express a component of the mixture in the following product form

$$p(d_t|d(t-1), f_t, \Theta_{f_t}) = \prod_{f\in f^*} p(d_t|d(t-1), f_t, \Theta_f)^{\delta(f_t,f)} \tag{9}$$

In the same way using (2), we can express the class-model

$$p(f_t|\alpha_{f_t}) = \prod_{f\in f^*} \alpha_f^{\delta(f_t,f)} \tag{10}$$

To ensure recursivity of the Bayes rule, we write the prior pdfs in the form analogous to the models, i.e.

$$p(\Theta|d(t-1)) = \prod_{f\in f^*} p(\Theta_f|d(t-1)) \tag{11}$$

for the prior pdf on Θ and

$$p(\alpha|d(t-1)) \propto \prod_{f\in f^*} \alpha_f^{\kappa_f(t-1)-1} \tag{12}$$

for the prior pdf on α. The variable κ_f for $f \in f^*$ is a statistic related to the class-model identification.

Regression data-model

To be able to specify the Bayes recursion for statistics of the pdfs we suppose a particular case of the data-model components in the form of $\bar{y}$-dimensional regression models with noise whose distribution belongs to the exponential class. For each state $f_t = f$, we have

$$d_t = \theta'_f \varphi_t + e_t \tag{13}$$

where
φ_t $=$ $[d_{t-1}, ..., d_{t-n}]$ vector of old data (regression)
θ_f vector of coefficients of the fth component regression and
e_t innovation (noise) with the distribution from the exponential class.

With this assumption on the noise, the data-model pdf gets the form

$$p(d_t|d(t-1), f_t, \Theta_f) \propto \exp\{q_f(\Theta)\ s(d_t, \varphi_t) + r_f(\Theta)\} \tag{14}$$

where q_f, s and r_f are specified functions of their arguments and φ_t is the model regressor from (13) and Θ vector of all unknown parameters concerning mixture-components (regression coefficients, characteristics of the noise).

Self reproducing prior pdf for data-model

Posterior pdf is proportional to a product of data-model pdfs multiplied by the prior one. If posterior pdf is to be structurally of the same form as the prior one then both pdfs must have the structure of a product of data-models. The product of the model components from an exponential class has the form

$$\prod_{\tau=1}^{\bar{\tau}} \exp\{q_f(\Theta)\ s(d_\tau, \varphi_\tau) + r_f(\Theta)\} = \exp\{q_f(\Theta) \sum_{\tau=1}^{\bar{\tau}} s(d_\tau, \varphi_\tau) + \bar{\tau}.r_f(\Theta)\}$$

It can be seen that the proper structure for the data prior pdf is

$$p(\Theta|d(0)) \propto \prod_{f \in f^*} \exp\{q_f(\Theta) S_f(0) + \vartheta_f(0)\ r_f(\Theta)\} \tag{15}$$

with some $S_f(0)$ and $\vartheta_f(0)$.
It can be viewed as a product of data-model components corresponding to the prior "history" i.e. the data $d(0)$, measured before the beginning of the identification. The variables $S_f(0)$ and $\vartheta_f(0)$ are prior statistics for data-model estimation. The indices f express the possibility to introduce different knowledge for each class separately.

The recursion for statistics

We aim to derive the recursion for updating the pdf $p(\alpha, \Theta, f_t|d(t))$ which is the basis for computing results (6), – (8) in terms of its statistics. Taking into account the structure of joint prior pdf (4) and exponential class data-model and prior pdfs (14) and (15), the Bayes rule (5) can be given the following structurally reproducing form

$$p(\alpha, \Theta, f_t | d(t)) \propto \tag{16}$$

$$= \prod_{f \in f^*} \exp\{q_f(\Theta)\ s(d_t, \varphi_t) + r_f(\Theta)\}^{\delta(f,f_t)}\ \alpha_f^{\delta(f,f_t)}.$$

$$. \exp\{q_f(\Theta) S_f(t-1) + \vartheta_f(t-1)\ r_f(\Theta)\}\ \alpha_f^{\kappa_f(t-1)-1} =$$

$$= \prod_{f \in f^*} \exp\{q_f(\Theta) S_f(t) + \vartheta_f(t)\ r_f(\Theta)\}\ \alpha_f^{\kappa_f(t)}$$

In (16) we denoted:

$$\begin{aligned} \vartheta_f(t) &= \vartheta_f(t-1) + \delta(f_t, f) \\ \kappa_f(t) &= \kappa_f(t-1) + \delta(f_t, f) \\ S_f(t) &= S_f(t-1) + \delta(f_t, f) s(d_t, \varphi_t). \end{aligned} \tag{17}$$

This is the searched for recursion. Unfortunately, it cannot be used directly as it is, because it contains unknown variable f_t. According to the Bayesian approach we could estimate it but the exact estimation brakes feasibility of the above recursive algorithm. The exact solution needs essentially to store values of the statistics for all possible sequences of $f(t) = (f_1, f_2, ..., f_t)$ what soon becomes prohibitive.

APPROXIMATION

Instead of the exact Bayesian estimation we will approximate the delta-function by its mean value

$$\delta(f_t, f) \approx \mathrm{E}[\delta(f_t, f) | d(t)] = p(f_t = f | d(t)) = w_f(t) \tag{18}$$

The needed conditional pdf $p(f_t | d(t))$ is given by (7) which, taking into account (4) and (5) can be written

$$p(f_t|d(t)) \propto \int p(d_t|d(t-1), f_t, \Theta_{f_t}) p(\Theta|d(t-1)) \mathrm{d}\Theta \int p(f_t|\alpha) p(\alpha|d(t-1)) \mathrm{d}\alpha =$$

$$= p(d_t|d(t-1), f_t).p(f_t|d(t-1)) \tag{19}$$

The last expression represents data and state predictive pdfs.

REMARK: Note, that the data-predictive pdf can be written

$$p(d_t|d(t-1), f_t) = \int p(d_t|d(t-1), f_t, \Theta_{f_t}) p(\Theta_{f_t}|d(t-1)) \mathrm{d}\Theta_{f_t} \tag{20}$$

because parameters $\Theta_f, f \neq f_t$ can be integrated leaving marginal density. This is an ordinary data prediction corresponding to the state f_t (see [12]).

General algorithm for statistic

Now solution to the whole task for the "general" regression model (13) can be outlined:
The Bayes rule (5) if expressed in terms of statistics gives the recursion (17) – with unknown $\delta(f_t, f)$ at this moment

1. Construct the joint pdf $p(\alpha, \Theta, f_t, d_t|d(t-1))$ according to (5) with (4) (i.e. compute the recursion (17) with $\delta(f_t, f) = 1$ but do not re-write values of the statistics)

2. Compute state prediction $w_f(t) = p(f_t = f|d(t))$ according to (7) for each state $f \in f^*$ using values of statistics from the previous step (i.e. for $\delta = 1$).

3. Substitute $\delta(f_t, f) = w_f(t)$ for $f \in f^*$ and update the statistics (i.e. recompute the recursion (17) for statistics with $\delta(f_t, f) = w_f$ and re-write the old values of statistics).

4. Construct data prediction and state estimation (8) and (7).

5. Compute estimates of parameters (if need be) usually as arguments of maxima of posterior pdfs (6)

Remark: The only specific evaluations that are needed are integrations over the joint pdf from (7) or (8). The same procedure is needed twice: first time with $\delta = 1$ for computing the state predictive pdf in Step 2 of the algorithm and second time for computing approximating weights $w(t)$ used for parameter estimation and data prediction in the rest of the algorithm.

5 Traffic-state identification

For presentation of the above algorithm for the mixture-model identification we use a simple but still practical example of estimation the level of service in a single traffic lane. This example should be a building block for construction of a more sophisticated method of identification of traffic networks in big cities.

The data measured for identification are supposed to be "intensity" and "density" of traffic flow which are currently measured by means of detectors placed under surface of the roadway. The intensity (we denote it $d_{2,t}$ at time instant t) is defined as a number of vehicles passing a specified point per time unit. The density (denoted by $d_{1,t}$) is a number of vehicles per distance unit. This characteristic is computed from so called occupancy of detectors, which is a ratio of total period for which the detector was occupied by vehicles and period when it was free. The two variables are considered to be sufficient for full description of the state of traffic in a specified point. It is necessary to have both of them (as, for example, from zero intensity it is not clear whether the road is empty or it is completely stuffed with standing cars). Nevertheless, they are bound by a well known relationship, which, considered as a relationship of intensity in dependence on density, can be approximately depicted as a concave parabola. Its shape corresponds with the major types of traffic states:

- *linear growth:* The road is free, the speed of cars is determined by the upper allowed speed limit in cities. With increasing density (the speed being still constant) the intensity grows linearly, up to a certain limit, when the distances between cars become so small, that the drivers are forced to decrease speed, because of safety reasons.

- *stagnancy:* The further growth of density results in continued deceleration of speed. The result is almost of a constant intensity. We are on the top of the reversed parabola.

- *decay:* If the distance between cars is shorter then certain critical value, the drivers are afraid to go on and they stop. The cars stack up and become a jumping tail-back. This stage represents collapse.

The parabolic shape of the function is common but its precise course varies in dependence of the road monitored and other factors like the kind of a weekday, season etc. In any case, the measured data pairs $[d_{1,t}, d_{2,t}] =$ $[\text{"density"}, \text{"intensity"}]_t$ represent points in the "intensity" - "density" plane which cluster and thus form regions. Each detected region reflects a specific state of the traffic area. Description of all such regions can be considered a model of the controlled system.

TRAFFIC MODEL

As we said above, the variable, which contains the whole necessary information about the identified state of traffic at time instant t, is the data pair $d_t = [d_{1,t}, d_{2,t}]$. For its modeling in a state f, we use two dimensional static regression models

$$\begin{bmatrix} d_{1,t} \\ d_{2,t} \end{bmatrix} = \begin{bmatrix} \theta_{1,f} \\ \theta_{2,f} \end{bmatrix} + \begin{bmatrix} e_{1,t} \\ e_{2,t} \end{bmatrix} \tag{21}$$

with zero mean Gaussian noise with covariance matrices $R_f, \quad f = 1, 2, 3.$

$$p(e_t) = N(0, R_f).$$

In the component-models, in accordance with (13), we denoted

$$d_t = \begin{bmatrix} d_{1,t} \\ d_{2,t} \end{bmatrix} \quad \text{measured data}$$

$$\theta'_f = \begin{bmatrix} \theta_{1,f} \\ \theta_{2,f} \end{bmatrix} \quad \text{coefficients of the } f\text{th model component}$$

$$\varphi_t = 1 \quad \text{regressor (of a static model)}$$

$$e_t = \begin{bmatrix} e_{1,t} \\ e_{2,t} \end{bmatrix} \quad \text{noise}$$

Index f indicates that the model belongs to the fth region (fth model component). Such model determines the probability of occurrence of the data point belonging to the region. The center of the region is specified by the mean value θ_f and its shape by the covariance matrix R_f. Through this matrix we can ensure the prior approximate shape of particular regions (increase, stagnancy and decay).

As for the future application of the model for control purposes it can be used either in the presented form like a classificator or the joint pdf can be recomputed to conditional one $p(d_1, d_2) \Rightarrow p(d_2|d_1)$ and used for minimization of some traffic criterion like e.g. maximization of intensity of the traffic flow in a controlled area.

Example algorithm

Using this simple example, we will show the structure of the identification algorithm in a more detailed way, following the general algorithm "steps 1.-5." from the end of Section 5.

Needed pdfs:
– *data-model components:*

$$p(d_t|f, \Theta_f) = N(\theta_f, R_f) = \frac{1}{2\pi\sqrt{|R_f|}} \exp\left\{-\frac{1}{2}[d_t - \theta_f]' R_f^{-1} [d_t - \theta_f]\right\} =$$
$$= \frac{1}{2\pi\sqrt{|R_f|}} \exp\left\{-\frac{1}{2} tr\ R_f^{-1}[-I, \theta_f] v_t [-I, \theta_f]'\right\} \quad (22)$$
$$\text{for} \quad f = 1, 2, 3$$

with

$$v_t = \begin{bmatrix} d_t \\ 1 \end{bmatrix} [d'_t, 1]$$

which belongs to the exponential class according to (14).

– *class-model components:*

$$p(f_t = 1|\alpha) = \alpha_1, \quad p(f_t = 2|\alpha) = \alpha_2, \quad p(f_t = 3|\alpha) = \alpha_3. \tag{23}$$

cf. (2) or (10)

– *prior pdf for* Θ*:*

$$p(\Theta|d(t-1)) \propto \tag{24}$$
$$\propto \prod_{f=1}^{3} \frac{1}{\sqrt{|R_f|^{\vartheta_f(t-1)}}} \exp\{-\frac{1}{2} tr\ |R_f|^{-1}[-I, \theta_f]V_f(t-1)[-I, \theta_f]'\}$$

cf. (11).

– *prior pdf for* α*:*

$$p(\alpha|d(t-1)) \propto \alpha_1^{\kappa_1(t-1)-1}.\alpha_2^{\kappa_2(t-1)-1}.\alpha_3^{\kappa_3(t-1)-1} \tag{25}$$

cf. (12)

Estimation:
Step 1 [construction of joint pdf]

$$p(\alpha, \Theta, f_t, d_t|d(t-1)) \propto \tag{26}$$
$$\propto \prod_{f=1}^{3} \frac{\exp\{-\frac{1}{2} tr\ R_f^{-1}.[-I, \theta_f]\{V_f(t-1) + \delta(f_t, f)v_t\}[-I, \theta_f]'\}}{\sqrt{|R_f|^{\vartheta_f(t-1)+\delta(f_t,f)}}}.$$
$$. \prod_{f=1}^{3} \alpha_f^{\kappa_f(t-1)+\delta(f_t,f)-1}$$

cf. (16).

Step 2 [state estimation]

$$w_f(t) = p(f_t|d(t)) \propto p(f_t, d_t|d(t-1)) = \int\int p(\alpha, \Theta, f_t, d_t|d(t-1)) \mathrm{d}\alpha \mathrm{d}\Theta \propto \tag{27}$$

$$\propto \int \prod_{f=1}^{3} \frac{\exp\{-\frac{1}{2} tr\ R_f^{-1}.[-I,\theta_f]\{V_f(t-1)+\delta(f_t,f)v_t\}[-I,\theta_f]'\}}{\sqrt{|R_f|^{\vartheta_f(t-1)+\delta(f_t,f)}}} \mathrm{d}\Theta.$$

$$.\int \prod_{f=1}^{3} \alpha_f^{\kappa_f(t-1)+\delta(f_t,f)-1} \mathrm{d}\alpha$$

$$\text{for } f = 1,2,3$$

cf. (7) with the integrand (26) from the previous step.

STEP 3 [approximate update of statistics]

Computation of approximated pdf $p(f_t = f|d(t)) \propto p(f_t = f, d_t|d(t-1))$. The results of integrations form the previous step remain the same, only with $\delta(f_t, f) = w_f$ for all $f \in f^*$.

STEP 4 [data prediction]

Computing data prediction can be done simply by standardization of pdf obtained in the Step 3. The predictive pdf for the state can be obtained in the same way. Nevertheless, here the problem is even more simple because of the finiteness of number of states. The most probable and thus predicted state is that which has the largest weight w_f.

STEP 5 [parameter estimation]

Point estimation of parameters is performed in a standard way. The good estimates are arguments maximizing the joint pdf form the Step 3.

Practically, the only specific computations that are necessary belong to Step 2 of the above algorithm. Now, we will briefly indicate the technique of this computation and show the results.

COMPUTATION OF INTEGRALS

For brevity we omit state indexes and time arguments whenever it is possible.

– *integral over* α

First, we express integral before updating statistics by a unit

$$I_\alpha = \int\int\int_\Omega \alpha_1^{\kappa_1-1}.\alpha_2^{\kappa_2-1}.\alpha_3^{\kappa_3-1}\ \mathrm{d}\alpha_1\mathrm{d}\alpha_2\mathrm{d}\alpha_3, \qquad \Omega = \{\alpha_i \geq 0,\ \sum_{i=1}^{3}\alpha_i = 1\}.$$

Introducing the conditions on Ω into the integral we get

$$I_\alpha = \int_0^1 \int_0^{1-\alpha_1} \alpha_1^{\kappa_1-1}.\alpha_2^{\kappa_2-1}.(1-\alpha_1-\alpha_2)^{\kappa_3-1}\mathrm{d}\alpha_1\mathrm{d}\alpha_2$$

The integrals can be solved by substitution e.g. $\alpha = \alpha_1$ and $\beta = \frac{\alpha_2}{1-\alpha_1}$. The result of the integration is

$$I_\alpha = B(\kappa_1, \kappa_2 + \kappa_3) B(\kappa_2, \kappa_3) = \frac{\Gamma(\kappa_1)\Gamma(\kappa_2)\Gamma(\kappa_3)}{\Gamma(\kappa_1 + \kappa_2 + \kappa_3)}$$

where we used the definition of beta function: $\quad B(m, n) = \frac{\Gamma(m)\Gamma(n)}{\Gamma(m+n)}$

For active state at time t say $f = 1$ the integral will be

$$I_\alpha(1) = \frac{\Gamma(\kappa_1 + 1)\Gamma(\kappa_2)\Gamma(\kappa_3)}{\Gamma(\kappa_1 + 1 + \kappa_2 + \kappa_3)} = \frac{\kappa_1}{\kappa_1 + \kappa_2 + \kappa_3} I_\alpha.$$

Similarly for states $f = 2, 3$.

So, we can write

$$I_\alpha(1) \propto \kappa_1; \quad I_\alpha(2) \propto \kappa_2; \quad I_\alpha(3) \propto \kappa_3 \tag{28}$$

– *integral over* Θ

As we mentioned (see Remark to (20)) the integral represents an ordinary data prediction for particular state. The methods for solving this task are well known [12]. The crucial step of the procedure is a completition of the integrand to square in the variable θ. If we denote in (27)

$$V_f(t) = V_f(t-1) + \delta(f_t, f).v_t = \begin{bmatrix} V_d & V'_{d\varphi} \\ V_{d\varphi} & V_\varphi \end{bmatrix}_f (t)$$

the completition reads

$$[-I \quad \theta] \begin{bmatrix} V_d & V'_{d\varphi} \\ V_{d\varphi} & V_\varphi \end{bmatrix} \begin{bmatrix} -I \\ \theta' \end{bmatrix} = \theta' V_\varphi \theta - 2\theta' V_{d\varphi} + V_d =$$

$$= [\theta - V_\varphi^{-1} V_{d\varphi}]' V_\varphi [\theta - V_\varphi^{-1} V_{d\varphi}] + V_d - V'_{d\varphi} V_\varphi^{-1} V_{d\varphi} =$$

$$= [\theta - \hat{\theta}]' V_\varphi [\theta - \hat{\theta}] + \Lambda$$

Integration over the completed part gives just normalization constant and the rest is dependent only on covariance R. This integral is can be computed using a definition of Gamma function. The result can be found e.g. in [12]. For our example it is

$$I_\Theta(f) = \frac{I_t(f)}{I_{t-1}(f)} \tag{29}$$

where

$$I_t(f) = \frac{2^{\vartheta_f(t)}\Gamma(\frac{\vartheta_f(t)+2}{2})\Gamma(\frac{\vartheta_f(t)+1}{2})}{|\Lambda_f(t)|^{\frac{\vartheta_f(t)+2}{2}} V_{\varphi,f}(t)} \tag{30}$$

with

$$\Lambda_f(t) = V_{d,f}(t) - V'_{d\varphi,f}(t) V^{-1}_{\varphi,f}(t) V_{d\varphi,f}(t) \tag{31}$$

For the weights w_f according to (27) it holds

$$w_f \propto \frac{I_t}{I_{t-1}} \kappa_f, \quad \text{for} f = 1,2,3 \tag{32}$$

Summary of the algorithm

Construction of initial statistics

for f=1,2,3

– set:
$\kappa_f(0)$ – state weights (defining initial probabilities of the states)
$\vartheta_f(0)$ – data weights (importance of initial data for prior pdf)
$V_f(0)$ – initial characteristics of unknown Θ_f and R_f

end of f

Real time identification

for t=1,2,...,N

– measure:
$d_t = [d_{1,t}, d_{2,t}]'$ – data pair (density and intensity)

– construct:

$$v_t = \begin{bmatrix} d_t \\ 1 \end{bmatrix} [d_t' \quad 1]$$ – data matrix

for f=1,2,3

– make first (trial) update
$\delta(f_t, f) = 1$
$\vartheta_f(t) = \vartheta_f(t-1) + \delta(f_t, f)$
$\kappa_f(t) = \kappa_f(t-1) + \delta(f_t, f)$
$V_f(t) = V_f(t-1) + \delta(f_t, f) v_t$
do not rewrite statistics

end of f

for f=1,2,3

– compute weights w

$w_f(t) = \frac{I_f(t)}{I_f(t-1)} \kappa_f(t), \qquad$ with $I_f(t)$ is defined by (30)

– normalize, so that $\sum w_f = 1$

end of f

for f=1,2,3

– make second (real) update
$\delta(f_t, f) = w_f(t)$
$\vartheta_f(t) = \vartheta_f(t-1) + \delta(f_t, f)$
$\kappa_f(t) = \kappa_f(t-1) + \delta(f_t, f)$
$V_f(t) = V_f(t-1) + \delta(f_t, f) v_t$
rewrite statistics

– find estimate of actual state (index of maximum of $w(t)$)

– make estimates of all parameters

$\hat{\alpha}_f(t) = \frac{\kappa_f(t)}{\sum_{i \in i^*} \kappa_i(t)}$

$\hat{\theta}_f(t) = V_{d\varphi,f}(t) / V_{\varphi,f}(t)$

$\hat{R}_f(t) = (V_{d,f}(t) - V_{d\varphi,f}(t) \hat{\theta}_f(t)) / \vartheta_f(t)$

end of f

end of t

6 Experiments

Layout of experiments

The purpose of experimenting with such a simple model is to verify "practically" (at least on simulation) its intended applicability. As it has been mentioned, the model should be a most simple building block for construction of an identification net which would estimate the state of large traffic systems (roads of big cities). The main task of such a cell of the intended network is to recognize the most probable state (formed by traffic intensity and density) on a single road. The states (points in the intensity–density plane) are known to lie around a parabolic-like curve, whose precise shape is identified. In addition, we suppose that the states we are going to identify are based on a real traffic situation and will be reflected by clustering of points creating several centers along the parabolic curve. In the ideal case, which we are starting from, one cluster should be on the arising part another on the top and the last one at the decaying part of the parabola. We want to identify their simulated positions and to decide currently, which of them is now actual.

Indeed, another possible view on the problem is to suppose, that not only one model component is valid at a time instant, but a real mixture of several of them – each one with a proper weight. It means, to suppose, that the areas, representing the states can overlap. In this case, from the point of view of identification, it is sufficient to detect this situation and at most to determine probabilities of components involved. From the point of view of control, besides taking the most probable state and the model connected with it for optimization, there is still another possibility – to perform optimization together for all the components with their weights. But the control is for now another story.

So, the main goal of the experiments is to verify ability of the proposed algorithm to detect more or less detectable clusters in a configuration as close as possible to the traffic problem, even if simulated.

Choice of initial conditions

Very important aspect is to explore all connected factors influencing the results of the identification. The most important of them is undoubtedly the start of the algorithm – the problem of initial conditions.

As we aim at a specific problem which is well known to us (we mean the possible traffic states at a specific road – not how to control them) and as the used Bayesian approach allows it, it will be very important to use and correctly imbed all prior information which can be gained. First of all, we should start the identification from initial centers which are as close as possible to real ones and to set covariances of components so that the directions of their shapes approximate the course of expected parabolic curve. The prior information can be gained either from some preliminary analysis or simply from the previous identification, in case it was successful.

The important point, common to all mixture-model identifications, is that the initial centers are to be set as best as possible. If some center is initially set too far from the real component position, the identification algorithm can never find it. That is why, at the very beginning, it is necessary on one hand let the estimates to move to find their positions but on the other hand not to allow them to to be misled by wrong data and move too far. The used algorithm gives a possibility for it.

Tracking of changes

Another aspect of the problem are changes of the basic model scheme represented by parabolic curve. There are two possibilities: Firstly, the changes are quick and large. Then it is well possible to extend our mixture-model approach to clusters along a single parabola to a model consisting of a mixture of parabolic functions. Then the algorithm would search not only the true cluster along a single parabola, but in addition it would choose the parabola itself, at the same time. Secondly, if the changes are small enough we can respect them simply through forgetting. The most suitable and used in this case we consider the alternative forgetting, which does not forget absolutely, but replaces forgotten information by prior one.

Results

The following pictures illustrate the discussed aspects of the identification.

Fig. 1 shows identification of the model, which has the same structure like the simulated one. The data are simulated from three different normal pdfs. The first picture shows generated data (marked by 'x') and contour-lines of the corresponding pdfs from which the data have been generated. The second picture displays the same data and contour-lines of identified (posterior) data-pdf. The task of identification is to detect clusters, i.e. to bound geometrical groups of data. It has been done successfully even if the data groups are not separated and they differ slightly only in varying density.

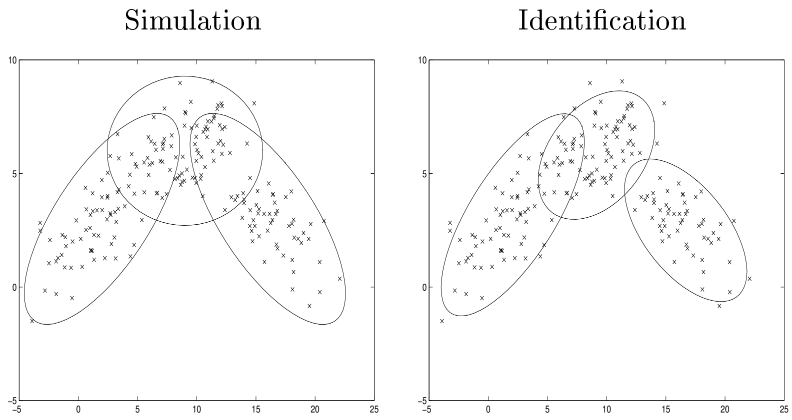

Fig. 1: Equal structures of simulation and identification

Fig. 2 presents an example where the structure of the simulation is not the same as that which is identified. The data are simulated simply around a parabolic shape with slight clustering at three points. The elliptic shaped pdfs have to detect the clusters and to bound data groups again. The data in the first picture are without outliers, the other two pictures contain data disturbed by some of them. The results are satisfactory. On the second picture, it can be seen that the identification is a bit confused at the beginning by "extra" data (it does not known exactly whether they are regular or not) but after some other data (picture 3) it recognized true clusters well and let outliers aside.

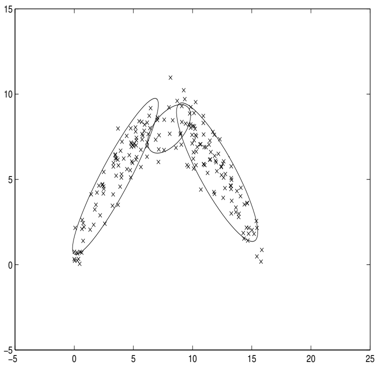

Fig 2.1

Data simply scattered around the parabola. The identification has to cover them by elliptic pdfs. Each ellipse represents one of the supposed state of the modelled traffic.

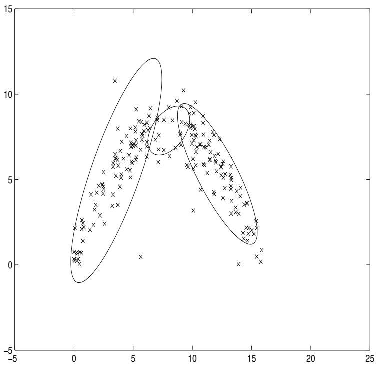

Fig 2.2

Data with outliers. The identification has to classify them, to decide whether include them into some group or to ignore them. At the beginning it is careful.

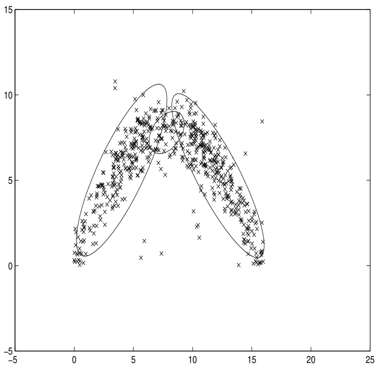

Fig 2.3

At the end (after simulating sufficient amount of data) it decides correctly. The groups are satisfactorily bounded and the outliers are outside.

Fig. 2: Different structure of simulation and identification

7 Conclusions

Identification algorithm for detecting clusters in two-dimensional traffic data has been presented. The data are intensity and density of the traffic flow. The configuration of these two variables can be taken as an evidence of the actual traffic state. Its shape is parabolic but it can vary between roads and even in longer time periods. The identification has to identify the true shape of parabola and to detect the area (denoting the traffic state) from which the data came.

The performed experiments show that the main purpose, detection of data clusters, is fulfilled satisfactorily. The algorithm detects not only clusters generated by the same structure which is identified (see Fig. 1) but also those which are substantially different (see Fig. 2).

The initial conditions during the start of the algorithm can considerably influence the course of the identification and even the result. They can be set, so that we are able to start from total ignorance (almost uniform prior pdfs) but with result that are especially at the beginning not so good. In our case, the prior information is abundant and can be fully utilized.

This paper represents only the first step in the indicated direction. It deals only with identification of a single traffic flow (it detects only the traffic state in a specific position of a single road). The further development will aim to the identification of some more complex situation (crossroads and area of crossroads) and finally to the control of traffic areas.

8 Appendix I.
Selection of initial conditions

For a good work of the algorithm, it is necessary to set carefully all initial values of variables which require it. The initial values are induced by prior information, which especially in this practical example is rather easily accessible. It must only be carefully handled and fully utilized.

The variables of the algorithm which need initialization are

ϑ_f, κ_f	weighted numbers of data items from the f^{th} state
V_f	extended information matrix of the identification

The effect of initial setting of these variables is following:
The weighted counters $\vartheta_f(0)$ and $\kappa_f(0)$ represent the number of fictitious data, which would bring the same information about the unknown parameters as that, incorporated into the prior information matrix. Briefly, the bigger their values are the more important is the prior information in comparison with that coming from data (at least at the beginning of the identification). The statistics ϑ concerns covariance of the unknown parameters θ, R, the statistics κ is connected with the probabilities of states α. The magnitude of items of κ set the credit of the initial estimates of state probabilities and in addition their ratios define their values. The initial data-model parameter estimates (means and covariances) are introduced by the information matrix V. For its initial importance it holds the same rule as for the counters: the biggest are its items the largest credit is given to it. To validate this fact we can again recall the trick of fictitious data. The initialization can be performed in the following procedure

- construct model with the desired parameters
- generate data from it
- accumulate the data into the information matrix

The resulting information matrix reflects the information about the parameters you have used in your fictitious model. The more data you incorporate the biggest will be items of the information matrix and the more important will be the incorporated information in comparison with the newly come data. This attitude is convenient mainly for introducing parameters of deterministic model when introducing of many fictitious data (for big credit of initial estimates) can be substituted by introducing only several data items multiplied by big weight (and affected by a negligible noise – for regularity reasons). In our case, when not only prior means of components (i.e. deterministic parameters) but also prior covariances of components have to be set, we will specify the initial information matrix directly.

First of all we recall the partitioning of information matrix and the meaning of its particular parts. For the present moment, let us consider the following case: we have started with no prior information and have incorporated n data items. Then

$$V = \sum_{i=1}^{n} \begin{bmatrix} d_i \\ 1 \end{bmatrix} . \begin{bmatrix} d_i' & 1 \end{bmatrix} = \begin{bmatrix} V_d & V_{d\varphi}' \\ V_{d\varphi} & V_\varphi \end{bmatrix}$$

where (denoting $\bar{x}$ an average of x)

$$V_d = \sum_{i=1}^{n} \begin{bmatrix} d_{1,i} \\ d_{2,i} \end{bmatrix} . \begin{bmatrix} d_{1,i} & d_{2,i} \end{bmatrix} =$$

$$= \begin{bmatrix} \sum_{i=1}^{n} d_{1,i}^2 & \sum_{i=1}^{n} d_{1,i} d_{2,i} \\ \sum_{i=1}^{n} d_{1,i} d_{2,i} & \sum_{i=1}^{n} d_{2,i}^2 \end{bmatrix} = n \begin{bmatrix} \overline{d_1^2} & \overline{d_1 d_2} \\ \overline{d_1 d_2} & \overline{d_2^2} \end{bmatrix},$$

$$V_{d\varphi}' = \sum_{i=1}^{n} \begin{bmatrix} d_{1,i} \\ d_{2,i} \end{bmatrix} = n \begin{bmatrix} \overline{d_1} \\ \overline{d_2} \end{bmatrix}$$

and

$$V_\varphi = n$$

The estimates are

$$\hat{\theta} = V_{d\varphi}' / V_\varphi = \bar{d}$$

$$\hat{R} = (V_d - V_{d\varphi}' V_\varphi V_{d\varphi})/n =$$

$$= \begin{bmatrix} \overline{d_1^2} & \overline{d_1 d_2} \\ \overline{d_1 d_2} & \overline{d_2^2} \end{bmatrix} - \begin{bmatrix} \overline{d_1} \\ \overline{d_2} \end{bmatrix} . \begin{bmatrix} \overline{d1} & \overline{d_2} \end{bmatrix} =$$

$$= \begin{bmatrix} \overline{(d_1 - \varphi_1)^2} & \overline{(d_1 - \varphi_1).(d_2 - \varphi_2)} \\ \overline{(d_1 - \varphi_1).(d_2 - \varphi_2)} & \overline{(d_2 - \varphi_2)^2} \end{bmatrix}$$

From the above structural decomposition of information matrix V and its connection with the estimates of parameters we can derive its proper initial form. To obtain initial pdf with circular contours we choose diagonal matrix V_d with items approximately equal to squares of average deviations of data from their means. If we want to built in the prior covariance its elliptical shape, we can use semi-axes and angle of rotation according to Appendix II. The biggest items are introduced the smallest is our belief in our information and the more easily will be the prior estimates changed by newly coming data.

The proper form of the prior information matrix seems to be

$$V(0) = \sigma. \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix} + \varepsilon. \begin{bmatrix} m_1 \\ m_2 \\ 1 \end{bmatrix} . \begin{bmatrix} m_1, & m_2, & 1 \end{bmatrix},$$

where

m is an estimate of (two dimensional) mean value
(coordinates of the center of the circular contour)

σ is a big number $\approx$ the square of average deviation of data from their mean
(the biggest the more easily is covariance changed)

ε is a small number $\approx$ 0,1 or less
(the small ϵ is the more easily are parameters changed)

9 Appendix II. Data visualization and simulation

The basic data-model component we consider here is a pdf function of two dimensional data $d = [d_1, d_2]$. It is a hillock determined by its mean "position" (a point in plane) and covariance "shape", which can be plot like a contour-line on some level. For Gaussian noise, the contour-line will be an ellipse. So, the model component can be visualized by this elliptical cut with the center representing the mean and shape determined by the maximal and minimal dispersion (semiaxes) and the rotation (elevation of semiaxes).

Let us suppose two dimensional normally distributed pdf $d = [d_1, d_2]'$ with

$$\text{mean } \theta = [\theta_1, \theta_2]' \qquad \text{and} \qquad \text{covariance } R = \begin{bmatrix} \sigma_1^2 & \sigma_{12} \\ \sigma_{12} & \sigma_2^2 \end{bmatrix}$$

i.e.

$$p(d|\Theta) \propto \mathrm{e}^{-\frac{1}{2}(d-\theta)' R^{-1}(d-\theta)}$$

The precision matrix $\Omega = R^{-1} = \frac{1}{1-\rho^2} \begin{bmatrix} \frac{1}{\sigma_1^2} & -\frac{\rho}{\sigma_1 \sigma_2} \\ -\frac{\rho}{\sigma_1 \sigma_2} & \frac{1}{\sigma_2^2} \end{bmatrix}$

where $\rho = \frac{\sigma_{12}}{\sigma_1 \sigma_2}$

For visualization we choose curve for which the exponent of the pdf is constant which is an elliptical contour defined e.g. by

$$(d-\theta)' R^{-1}(d-\theta) = (d-\theta)'\Omega(d-\theta) = 1$$

Now, the basic position when the main axis is horizontal (i.e. without rotation) corresponds to diagonal covariance and precision matrix

$$R = \begin{bmatrix} \sigma_1^2 & 0 \\ 0 & \sigma_2^2 \end{bmatrix} \qquad \Omega = \begin{bmatrix} \frac{1}{\sigma_1^2} & 0 \\ 0 & \frac{1}{\sigma_2^2} \end{bmatrix}$$

The rotation can be introduced using polar coordinates: in our "normal" plane with horizontal x and vertical y axes we introduce rotated plane with identical origin and axes u and v, rotated by an angle ϕ_0. For x, y plane it is

$$\begin{aligned} x &= \omega \cos(\phi) \\ y &= \omega \sin(\phi) \end{aligned}$$

and for rotated plane u, v

$$\begin{aligned} u = \omega . \cos(\phi - \phi_0) &= \cos(\phi_0)\omega \cos(\phi) + \sin(\phi_0)\omega \sin(\phi) \\ v = \omega . \sin(\phi - \phi_0) &= -\sin(\phi_0)\omega \cos(\phi) + \cos(\phi_0)\omega \sin(\phi) \end{aligned}$$

From this the transformation of rotation follows

$$\begin{bmatrix} u \\ v \end{bmatrix} = \begin{bmatrix} \cos(\phi_0) & \sin(\phi_0) \\ -\sin(\phi_0) & \cos(\phi_0) \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} \tag{33}$$

The rotated ellipse can be can be introduced like basic one, but in rotated coordinates u, v - i.e.

$$\frac{u^2}{a^2} + \frac{v^2}{b^2} = [u \quad v] \begin{bmatrix} \frac{1}{a^2} & 0 \\ 0 & \frac{1}{b^2} \end{bmatrix} \begin{bmatrix} u \\ v \end{bmatrix} = 1$$

Introducing the transformation of rotation and denoting $\cos(\phi_0) = C$ and $\sin(\phi_0) = S$ we get in the original coordinates x, y

$$[x \quad y] \begin{bmatrix} C & -S \\ S & C \end{bmatrix} \begin{bmatrix} \frac{1}{a^2} & 0 \\ 0 & \frac{1}{b^2} \end{bmatrix} \begin{bmatrix} C & S \\ -S & C \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} = [x \quad y] \; \Omega \begin{bmatrix} x \\ y \end{bmatrix} = 1$$

where the kernel of the rotated quadratic form is

$$\Omega = \begin{bmatrix} \frac{C^2}{a^2} + \frac{S^2}{b^2} & \frac{CS}{a^2} - \frac{CS}{b^2} \\ \frac{CS}{a^2} - \frac{CS}{b^2} & \frac{S^2}{a^2} + \frac{C^2}{b^2} \end{bmatrix} \tag{34}$$

The relations between semiaxes a, b and the elevation ϕ_0 and items of the matrix Ω are

$$\phi_0 = \frac{1}{2} \mathrm{arctg} \left(\frac{2\sigma_{12}}{\sigma_1^2 - \sigma_2^2} \right) \tag{35}$$

$$a^2 = \frac{C^2 - S^2}{\sigma_1^2 C^2 - \sigma_2^2 S^2}, \qquad b^2 = \frac{S^2 - C^2}{\sigma_1^2 S^2 - \sigma_2^2 C^2} \tag{36}$$

The derived relations can be used either for visualization or for simulation

Visualization of pdf

The task is to plot contour of pdf of two-dimensional normally distributed random variable given by mean and covariance matrix.

Given:

2-dimensional covariance matrix R and vector of mean values θ

$$R = \begin{bmatrix} \sigma_1^2 & \sigma_{12} \\ \sigma_{12} & \sigma_2^2 \end{bmatrix} \qquad \text{and} \qquad \theta = \begin{bmatrix} \theta_1 \\ \theta_2 \end{bmatrix}$$

Compute:
Elevation ϕ_0 and semiaxes a and b according to (35) and (36).

Construct:
Ellipsis with computed semiaxes in basic position (without elevation).

$$\begin{array}{c} u = a.\cos(t) \\ v = b.\sin(t) \end{array}, \quad t \in (0, 2\pi)$$

Elevate and shift:
Transform computed x, y to rotated u, v multiplying by the transformation matrix according to (33) and shift to the desired position

$$\begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} \cos(\phi_0) & -\sin(\phi_0) \\ \sin(\phi_0) & \cos(\phi_0) \end{bmatrix} \begin{bmatrix} u \\ v \end{bmatrix} + \begin{bmatrix} \theta_1 \\ \theta_2 \end{bmatrix}$$

Plot:
Plot y against x (e.g. in MATLAB): PLOT(x(0:2π),y(0:2π))

DATA SIMULATION

The task is to simulate data from two dimensional normally distributed pdf contours of which are ellipses with semi-axes a, b and rotation ϕ_0. The following procedure can be used:

Compute covariance matrix:
Using the desired semi-axes and angle of rotation construct the covariance matrix R as an inversion of the corresponding precision matrix according to (34)

$$R = \Omega^{-1}$$

Factorize covariance matrix:
Factorize it, i.e. construct upper triangular matrix Ψ such that

$$R = \Psi'.\Psi$$

This factorization can be easily computed by multiplying the matrices on the right-hand side and comparing corresponding elements of both sides. The standard MATLAB subroutine CHOL can be directly used, as well.

Filter by factor of covariance matrix:
Samples of zero mean random variable $\tilde{d} = [\tilde{d}_1, \tilde{d}_2]'$ can be obtained by filtering samples of two dimensional standard normal random variable $z = [z_1, z_2]'$ through the lower factor of the covariance matrix Ψ'

$$\tilde{d} = \Psi'.z$$

Justification of this procedure can be indicated as follows. Consider two-dimensional random variables $z \sim N(0, I)$ and $\tilde{d} \sim N(0, R)$ with know covariance matrix R. Exponents of their pdfs are

$$\frac{1}{2} z'.z \qquad \text{and} \qquad \frac{1}{2} \tilde{d}'.R^{-1}.\tilde{d}$$

Factorizing $R^{-1} = (\Psi'.\Psi)^{-1} = \Psi^{-1}.\Psi'^{-1}$ and comparing the exponents we get $z = \Psi'^{-1}.\tilde{d}$ which gives $\tilde{d} = \Psi'.z$

Shift: by mean value
Adding vector of mean values simply shifts the generated data so that they have the proper mean

$$d = \theta + \tilde{d}$$

10 Appendix III. Generation of samples from discrete pf

In the simulation, we deal with both continuous and discrete random variables. For generation of continuous samples with some frequently used distribution, we can utilize standard subroutines (e.g. in MATLAB they are RAND for generation of samples of uniformly distributed random variable of RANDN for normally distributed one).

For generation of samples from discrete probability function (pf), we can use geometrical view on probability. We consider a segment with unit length and a point which is falling uniformly to this segment. Then the probability that the point falls within any sub-segment of the length p is equal to its length p.

Now, our task is to generate samples from the discrete pf given by the following table

x	x_1	x_2	...	x_n
$p(x)$	p_1	p_2	...	p_n

i.e. to simulate only numbers $x_1, x_2 ... x_n$ and each of them with its probability $p_1, p_2, ... p_n$. To this end, we divide the virtual probability segment into intervals with lengths equal to probabilities $p_1, p_2, ... p_n$ from the pf and generate uniformly distributed random variable (instead of uniform falling of the point). After each generation we choose the value x_i corresponding to the interval into which the generated sample has fallen. It is a permitted value (one of that from pf) and its probability is p_i because the length of the interval is also p_i. The numbers x_i are thus samples from the pf given by the table.

References

[1] R. Kulhavý and M. B. Zarrop, "On general concept of forgetting", *International Journal of Control*, vol. 58, no. 4, pp. 905–924, 1993.

[2] M. Kárný, J. Kadlec, and E. L. Sutanto, "Quasi-Bayes estimation applied to normal mixture", in *Preprints of the 3rd European IEEE Workshop on Computer-Intensive Methods in Control and Data Processing*, J. Rojíček, M. Valečková, M. Kárný, and K. Warwick, Eds., Praha, September 1998, pp. 77–82, ÚTIA AV ČR.

[3] M Papageorgiou, *Applications of Automatic Control Concepts to Traffic Flow Modelling and Control.*, vol. 50, 1983.

[4] G. Confessore Bianco, L. and P. Reverberi, "Optimal location of traffic counting points for transport network control.", in *IFAC Transportation Systems, Chania*, pp. 890–895. Grece, 1997.

[5] G.T. Bowen and R.D. Bretherton, "Latest developments in scoot - version 3.1.", in *International Conference on Road Traffic Monitoring and Control. Inst. of Electrical Engineers.* London, 1996.

[6] D. Helbing, "Traffic data and their implications for consistent traffic flow modelling.", in *Transportation Systems, Chania*, pp. 809–814. Greece, 1997.

[7] W. Kronjäger and P. Konhäuser, "Applied traffic flow simulation.", in *IFAC Transportation Systems, Chania*, pp. 805–808. Greece, 1997.

[8] Saeed Ishtiaq Hounsell, N. and F. McLeod, "Short term forcasting of urban traffic congestion.", in *6th World Conference on Transport Research, Lyon.* France, 1992.

[9] F. McLeod Hounsell, N. and Saeed Ishtiaq, "Journey time prediction for bus priority at traffic signals.", in *IFAC Transportation Systems, Chania*, pp. 911–916. Greece, 1997.

[10] N. Hounsell and J.R. Landles, "Bus priority at traffic signals in london. progress, performance and opportunities.", in *Second World Congress on Intelligent Transport Systems '95.* Yokohama, 1994.

[11] J. Viegas and L. Baichuan, "Traffic control system with intermittent bus lines.", in *IFAC Transportation Systems, Chania*, pp. 905–910. Greece, 1997.

[12] V. Peterka, "Bayesian approach to system identification", in *Trends and Progress in System Identification*, P. Eykhoff, Ed., pp. 239–304. Pergamon Press, Oxford, 1981.