Distributed Bayesian Decision-Making for Urban Traffic Control

Vaclav Šmídl[1], Jan Přikryl[1,2]

[1]Institute of Information Theory and Automation, Czech Academy of Sciences, Prague, Czech Republic
[2]Department of Applied Mathematics, Faculty of Transport Sciences, Czech Technical University, Prague, Czech Republic
{smidl,prikryl}@utia.cas.cz, http://as.utia.cz

Abstract— **Bayesian approach to decision-making is successfully applied in control theory for design of control strategy. The approach is based on the assumption that only one decision-maker is an active part of the system. This assumption was recently relaxed yielding distributed Bayesian decision-making theory. This extension is useful for large distributed systems which intrinsically contains a lot of uncertainty. In this paper, we apply the theory to distributed control of one example of such a system, the surface traffic network in dense urban areas.**

Index Terms— **Bayesian decision-making, multi-agent systems, fully probabilistic design, urban traffic control.**

I. Introduction

In recent years, it becomes obvious that the traditional centralized approach to control of large systems has reached its limits. Decentralization of control and decision-making is seen as the future direction of research in both academia and industry [6], [2]. Many successful applications of distributed control in classical sense, or modern multi-agent systems [15] have been published. However, the available paradigms of design of distributed control very often neglect uncertainty in the controlled system, so that it is difficult to design controllers for systems where uncertainty is intrinsic and so significant that it can not be ignored. The rigorous Bayesian theory of decision-making under uncertainty [3] could help us to overcome this trouble. However, it is heavily based on the assumption that only a single decision-maker exists in the system, which corresponds to the centralized approach in control theory. Recently, a series of papers attempts to offer an extension of the Bayesian theory to the case many decision-makers [1], resulting in decentralized stochastic control.

Surface traffic in dense urban areas is a typical example of a system with both strong presence of uncertainty, and the need for decentralized controllers. At present, most of the traffic control systems works in isolation, processing only local measurements to estimate traffic intensity and derive a strategy of switching particular signals. These systems do not take into account what is happening in their neighbourhood. One reason of this situation is the inherent uncertainty how these observations would augment our knowledge about the local behaviour of traffic. In this paper, we show how Bayesian distributed control can be used to address this problem using an existing model of traffic flow.

In Section II, we present a brief summary of distributed Bayesian decision-making followed by a list of probabilistic operations that are needed for the implementation of the theory. In Section III, we present the domain of traffic control and define the models and operations that are required in order to apply the theory. Preliminary results based on simulated data are presented in Section IV.

II. Distributed Bayesian Decision-Making

Bayesian decision-making (DM) is based on the following principle [3]: *Incomplete knowledge and randomness have the same operational consequences for decision-making*. Therefore, all unknown quantities are treated as random variables and formulation of the problem and its solution are firmly based within the framework of probability calculus.

Traditionally, the decision-maker is assumed to be the *only* entity that intentionally influences the environment [3]. It consists of a model of its environment, its individual aims, and a pre-determined strategy of decision-making. The Bayesian decision-maker is designed by the means of the Bayesian theory, which results in probabilistic representation of all the components, i.e. its model of the environment, aims and strategy.

In distributed systems, the partial controller (decision-maker) directly influences only a part of the environment, i.e. its *neighbourhood*. Preoccupation of a controller with its own aims and achieving them on its neighbourhood may lead to undesired behaviour of the overall system. This can be avoided only by communication between the decision-makers. A decision-maker that is able to communicate with its neibours will be called an *agent*, for its close relationship to an intelligent agent in multi-agent systems [15]. Since communication is not a part of the Bayesian decision-making theory, the theory has to be extended for communication and negotiation as operations on probability distributions. It was shown that the technique of *fully probabilistic design (FPD)* [8] reduces the task of agent cooperation into *reporting and merging of probability density functions* [1]. In what follows, we summarize (i) the tasks of an autonomous Bayesian agent, (ii) the mechanism of cooperation between the Bayesian agents, and (iii) the formal definition of probabilistic operations that are needed for the implementation.

A. Standard Bayesian decision-making

An autonomous agent is not aware of presence of other agents, hence, it is equivalent to the Bayesian decision-maker

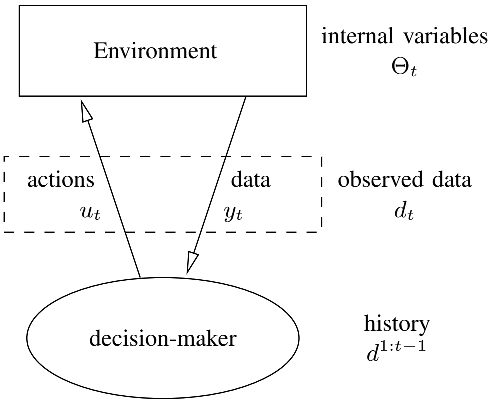


Fig. 1. Basic decision-making scenario with single decision-maker (autonomous agent).

in standard theory [3], which is now shortly reviewed:

1) Model parametrization: Each agent must have its own model of its neighbourhood. This model is necessary for description of the dependence of the observed data, y_t, on the decisions, u_t. The intrinsic uncertainty of the system is modelled by internal variables Θ_t, see the illustration in Fig. 1. In the Bayesian paradigm, all models have the form of probability density functions (*pdf*). Specifically, $f\left(y^{1:t}, \Theta^{1:t}, u^{1:t}\right)$ is the most complete model of the system. Here, $f(\cdot)$ denotes a pdf of its argument, and the notation $y^{1:t}$ is used to denote the whole trajectory of observations $y^{1:t} = [y_1, \ldots, y_t]$.

2) Learning: Improvement of agent's knowledge of the uncertain internal variables Θ_t is achieved by an operation of probability calculus, which uses the observed data to improve its belief in different values of the internal variables. This operation is formalized as evaluation of the *posterior* distribution of internal variables $f\left(\Theta_t|d^{1:t}\right)$.

3) Prediction: Since parametrization of the system via the internal variables Θ_t is an artificial step, the prediction of the future outputs of the system should not depend on the chosen model structure. Hence, the predictor is evaluated in the form of pdf relating the uncertain future outputs to the history of observation $f\left(y_{t+h}|d^{1:t}\right)$, where h denotes the prediction *horizon*.

4) Aim of decision-making: An agent's strategy is designed to achieve some desired performance. There are many ways how to formalize a performance criterion. One possibility is to use fully probabilistic criteria of performance, which uses an *ideal pdf*, ${}^I f\left(y^{1:t}, u^{1:t}\right)$ to evaluate desirability of the overall performance. Intuitively, the maximum of this pdf is located at the best possible performance, and each deviations from the maximum has assigned a probability which indicate how much is such a deviation acceptable. Zero probability is assigned to a totally unacceptable (or physically impossible) behaviour.

5) Design of DM strategy: The final aim is to design the decision-making (DM) strategy, i.e. a rule how to choose actions u_t based on the history of observations $d^{1:t-1}$ and the current observation y_t. In classical control theory, this rule is deterministic, however, the Bayesian approach yields probabilistic form of the controller, i.e. pdf $f\left(u_t|y_t, d^{1:t-1}\right)$. In practical applications, the actual action is chosen deterministically, e.g. as the expected value of this pdf.

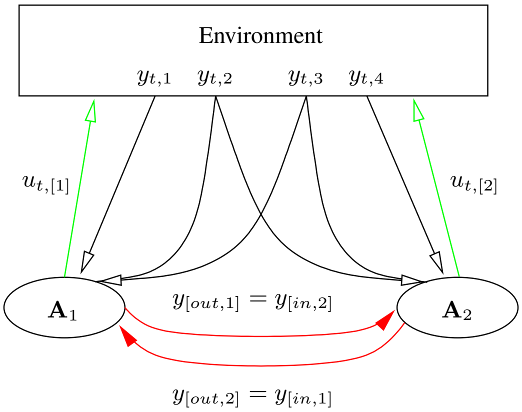


Fig. 2. Illustrative scheme of communication of two Bayesian agents, $\mathbf{A}_1$ and $\mathbf{A}_2$. Agents can exchange information only about objects that are observed by both of them, i.e. $y_{t,2}$and $y_{t,3}$ in this case. By design, we assume that $\mathbf{A}_2$ models $y_{t,2}$ and gives up modelling $y_{t,3}$, while $\mathbf{A}_1$ models $y_{t,3}$ and does not model $y_{t,2}$. The missing models will be replaced by communication.

This is only a brief summary of objects and operations that will appear in the text. More details can be found in Section II-D.

B. *Communication and cooperation of agents*

Distributed Bayesian decision-making extends the traditional Bayesian decision-maker—which performs the basic operations of learning and design of DM strategy—by two extra operations: (i) exchange of its models and aims with its neighbours, and (ii) modifications of its own models or aims in the light of the information received from its neighbours. From many possible scenarios of cooperation of Bayesian agents [1], we restrict our attention to the simplest scenario that is relevant for the traffic control, i.e. a scenario where all the strategies of cooperation and the level of trust between agents is prescribed by the designer.

Decomposition of the overall system inbetween agents is illustrated in Figure 2. Namely, we distinguish three main categories of the observed data y_t: (i) data that are modelled by the agent—i.e. the agent is able to make predictions of their future values—will be denoted $y_{[\text{out}]}$, (ii) data that are not modelled by the agent—i.e. their predictions must be supplied by the neighbouring agents—will be denoted $y_{[\text{in}]}$, and finally (iii) data on which the predictions are not needed, $y_{[\text{no}]}$. In this paper, we suppose that these sets of data are disjoint, i.e. $y_t = \left[y_{[\text{in}],t}, y_{[\text{out}],t}, y_{[\text{no}],t}\right]$. This assumption is not necessary in general cases [1]. Naturally, communication between agents $\mathbf{A}_1$ and $\mathbf{A}_2$ can be established only if the modelled data of agent $\mathbf{A}_1$, $y_{[\text{out},1]}$, contains elements that are also present in the non-modelled data of agent $\mathbf{A}_2$, $y_{[\text{in},2]}$, i.e. $y_{[\text{out},1]} \cup y_{[\text{in},2]} \neq \{0\}$.

C. *On-line algorithm of a Bayesian agent*

Each Bayesian agent iteratively performs the following steps:

1) Read: The observed data are read from the environment. Moreover, the following information is read from the neighbouring agents: (i) the predictive pdf of the non-modelled data on horizon h, $\mathbf{f}(y_{[\text{in}]}^{t+1:t+h}|d^{1:t})$, (ii) the ideal pdf of the modelled data on the same horizon, ${}^{I}\mathbf{f}(y_{[\text{out}]}^{t+1:t+h})$. Note that these pdfs are created by composition of many lower-dimensional pdfs from many agents. The notation $\mathbf{f}(\cdot)$ is used exclusively for the pdfs obtained from the neighbours.

2) Merge: The pdfs from the neighbours are used to modify the model and aims of the agent. Since we assume that each agent builds predictors of different variables, the obtained predictions are directly used, i.e. $f(y_{[\text{in}]}^{t+1:t+h}|\cdot) \equiv \mathbf{f}(y_{[\text{in}]}^{t+1:t+h}|\cdot)$. However, the obtained ideal distributions represents the aims of the neighbours which may contradict the individual aims of the agent. Therefore, a compromise must be achieved by the merging operation

$$ {}^{I}\tilde{f}\left(y_{[\text{out}]}^{t+1:t+h}\right) \overset{\text{merge}}{\longleftarrow} {}^{I}f\left(y_{[\text{out}]}^{t+1:t+h}\right), {}^{I}\mathbf{f}\left(y_{[\text{out}]}^{t+1:t+h}\right). \quad (1)$$

This operation is defined in Section II-D.4.

3) Learn: The observed data are used to increase the knowledge about the environment. This step is identical to the autonomous case, Section II-D.2.

4) Design of the DM strategy: FPD is performed in the same fashion as for the autonomous case, however, the aim of decision-making is defined by the *merged ideals,* ${}^{I}\tilde{f}(\cdot)$.

5) Write: The designed DM strategy is used to select a decision u_t which will be written to the environment. Moreover, communication to the neighbours is written into the allocated space from where they can read it. Specifically, predictions of the modelled data $f(y_{[\text{out}]}^{t+1:t+h+1}|d^{1:t})$, and the desired ideal pdfs of the input data ${}^{I}f(y_{[\text{in}]}^{t+1:t+h+1})$ are written there.

D. Necessary probability calculus for Bayesian agents

In this Section, we review mathematical operations that are needed in order to implement the approach.

1) Model Parametrization: The most general probabilistic model in Section II-A, is often simplified as follows:

$$ f\left(y^{1:t}, \Theta^{1:t}, u^{1:t}\right) \equiv \prod_{\tau=1}^{t} f\left(y_\tau|u_\tau, \Theta_\tau, d^{1:\tau-1}\right) \times \\ f\left(\Theta_\tau|u_\tau, \Theta_{\tau-1}, d^{1:\tau-1}\right) f\left(u_\tau|d^{1:\tau-1}\right). \quad (2)$$

This simplification is known as the state-space model, since it is assumed that future behaviour of the system is fully determined (up to random effects) by the input u_t and the state Θ_t, i.e. conditional independence between the observations in different times is assumed. These three parts of the model have specific roles in the whole approach and therefore specific names: (i) *the observation model*, $f\left(y_t|u_t, \Theta_t, d^{1:\tau-1}\right)$, (ii) *the internal model*, $f\left(\Theta_t|u_t, \Theta_{t-1}, d^{1:\tau-1}\right)$, and (iii) *the DM strategy*, $f\left(u_t|d^{1:t-1}\right)$. Models (i) and (ii) are given by the designer, however the DM strategy (iii) is a result of optimization, Section II-D.3.

2) Learning via Bayesian filtering: The task of learning is to evaluate the posterior distribution of the state Θ_t given the observed data, $f\left(\Theta_t|d^{1:t}\right)$. This pdf can be computed recursively as follows:

$$ f(\Theta_t|u_t, d^{1:t-1}) = \\ \int f(\Theta_t|u_t, \Theta_{t-1}, d^{1:t-1}) f(\Theta_{t-1}|d^{1:t-1})\, d\Theta_{t-1}, \quad (3)$$

$$ f(\Theta_t|d^{1:t}) \propto \frac{f\left(y_t|u_t, \Theta_t, d^{1:t-1}\right) f\left(\Theta_t|u_t, d^{1:t-1}\right)}{f\left(y_t|u_t, d^{1:t-1}\right)}, \quad (4)$$

$$ f\left(y_t|u_t, d^{1:t-1}\right) = \\ \int f\left(y_t|u_t, \Theta_t, d^{1:t-1}\right) f\left(\Theta_t|u_t, d^{1:t-1}\right) d\Theta_t. \quad (5)$$

In general, evaluation of the above pdfs is a complicated task, which is often intractable and many approximate techniques must be used [5]. In this text, we are concerned with conceptual issues and we assume that all operations (3)–(5) are tractable.

Equation (5) defines the one-step-ahead predictor. Prediction of the whole trajectory is

$$ f\left(y^{t+1:t+h}|d^{1:t}\right) = \\ \int f\left(y^{t+1:t+h}, \Theta^{t:t+h}, u^{t:t+h}|d^{1:t}\right) d\Theta^{t:t+h} du^{t:t+h}. \quad (6)$$

3) Design of the DM strategy: The fully probabilistic design of the DM strategy [8] is an alternative to the standard stochastic control design, which is formulated as the minimization of an expected loss function with respect to all possible decision-making strategies [4]. The key difference is in specification of the decision-making aim in the form of an *ideal pdf* of the closed loop. Then, the loss function is defined as the Kullback-Leibler divergence between the actual and the desired behaviour on the DM horizon. Formally,

$$ \mathsf{L}\left(f\left(u_t|d^{1:t-1}\right), t+h\right) = \\ \mathsf{D}\left(f\left(d^{t:t+h}, \Theta^{t:t+h}\right) || \, {}^{I}f\left(d^{t:t+h}, \Theta^{t:t+h}\right)\right), \quad (7)$$

where $\mathsf{D}(\cdot||\cdot)$ denotes the Kullback-Leibler divergence.

This approach has two major advantages: (i) the optimal DM strategy is found in a *closed form*, and (ii) the aims of individual agents can be communicated in the *same form as predictions*, i.e. in the form of pdfs. The minimum of the loss function (7) is evaluated recursively as follows:

$$ f\left(u_t|d^{1:t-1}\right) = {}^{I}f(u_t|d^{1:t-1}) \frac{\exp[-\omega(u_t, d^{1:t-1})]}{\gamma(d^{1:t-1})}. \quad (8)$$

Here, $\omega(\cdot)$ and $\gamma(\cdot)$ are integral functions of all involved pdfs (these are not presented here for brevity, see [8] for details). The decisions are then generated using a simplified version of stochastic dynamic programming [4].

4) Merging of pdfs: Merging is a probabilistic operation for the fusion of knowledge from many pdfs into one, while preserving as much information as possible [1]. In general, this operation can be defined as an optimization problem with Kullback-Leibler divergence as the criterion of optimality [9]. In this paper, we assume that $y_{[\text{in},1]} = y_{[\text{out},2]}$ (Fig. 2), and thus the merged pdfs are defined on the same variable $\tilde{y} \equiv y_{[\text{out}]}^{t+1:t+h}$

(1). The merging operation is a specific variant of the general scheme [9], with the criterion of optimality chosen as follows:

$$\mathsf{L}_M(f(\tilde{y})) = \alpha \mathsf{D}\left(f_2(\tilde{y})||f(\tilde{y})\right) + (1-\alpha)\,\mathsf{D}\left(f_1(\tilde{y})||f(\tilde{y})\right). \quad (9)$$

Here, the weight $\alpha \in \langle 0, 1 \rangle$ is the level of importance of each source (agent), and $f(\tilde{y})$ is the optimized pdf. The merged pdf, $\tilde{f}(\tilde{y})$, is found by functional minimization or minimization over a set of chosen parametric distributions. We will use the latter approach for its computational simplicity. The family of the target distribution is chosen as a specific pdf $f(\tilde{y}|\Phi)$ parameterized by a multivariate parameter Φ. Then, the the merged pdf is fully determined by the parameter $\tilde{\Phi}$, which is the result of optimization,

$$\tilde{\Phi} = \arg\min_{\Phi} \mathsf{L}_M(f(\tilde{y}|\Phi)). \quad (10)$$

III. Application in Traffic Control

In many cities, street networks cannot easily accommodate the vast volume of traffic, which results in regular traffic congestions. Efficient urban traffic control (UTC) mechanisms can, in many cases, improve the situation and achieve higher throughput of an urban transport network without changing its topology. We will now just briefly outline some terms and concepts of UTC. Interested readers can refer to any of the existing monographs on this topic, e.g. [12].

In most cases, UTC is targeted on *signalled intersections*, where the traffic is controlled by *traffic signals*. The sequence of the traffic signal settings for an intersection is called a *signal plan*. A signal plan cycle typically consist of several *stages*, where one of the conflicting traffic flows has the green light on and the others have to wait. The *intersection controller* is an industrial micro-controller that selects the order of stages and the stage lengths in attempt to maximize the possible throughput of the intersection. It is usually connected to *traffic detectors* (inductive loops, infra-red or video sensors) providing measurements of some traffic quantities such as the intensity of the traffic flow or the detector occupancy. In areas with high traffic intensity, the intersection controllers may be mutually interconnected in an attempt to optimize the throughput of the whole traffic network.

The above mentioned approaches often use heuristics and/or long-time statistics to derive the optimal control strategy. Our proposal is to build the strategy adaptively in a collaborative environment of "intelligent" traffic agents, where each agent is responsible for one intersection. The task of our agents is to agree on the overall traffic signal settings that would minimize the time spent by vehicles inside the controlled region and thus maximize the throughput of the network. The traffic agents are designed using the theory of distributed Bayesian decision-making, Section II. The general algorithm of Section II-C can be directly used if we choose the following constituents: the model pdfs (Section II-D.1), and the ideal pdfs (Section II-D.3).

A. Probabilistic model of the traffic flow

Each agent receives measurements from detectors on its intersection, specifically, the vector of intensities of traffic flow, I_t, and the vector of detector occupancies, O_t. The measurements related to the incoming flow will be denoted $y_{[\text{in}],t} = \left[I_{[\text{in}],t}, O_{[\text{in}],t}\right]$, and similarly for the outgoing flow, $y_{[\text{out}],t} = \left[I_{[\text{out}],t}, O_{[\text{out}],t}\right]$. A single detector can supply input and output measurements for neighbouring agents, see Fig. 3. However, as we said in Section II-B, only $y_{[\text{out},1]}$and $y_{[\text{out},2]}$are modelled. Since the outgoing flow of one agent is the incoming flow of another agent, the communication between the agents is in terms of intensities and occupancies. The traffic flow is modelled using a particle flow model [7]. This is a special case of state-space model [10], where the state Θ_t consists of the queue lengths ξ_t and some extra entries (see below).

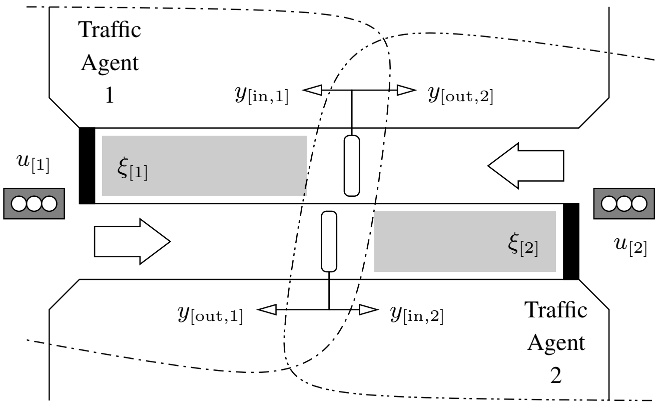


Fig. 3. Illustration of information-distribution between the neighbouring traffic agents. Both agents measure their $y_{[\text{in}]}$ and $y_{[\text{out}]}$. Agent 1 is responsible for prediction of $y_{[\text{out},1]}$ and Agent 2 predicts development of $y_{[\text{out},2]}$.

The model is formalized by the following probabilistic relations:

$$f(\Theta_t|\Theta_{t-1}, u_{t-1}, y_{[\text{in}],t-1}) = t\mathcal{N}\left(A_t\Theta_{t-1} + B_t u_{t-1}, Q_t, \langle 0, \Theta_{\max}\rangle\right), \quad (11)$$

$$f(y_{[\text{out}],t}|\Theta_t, u_t, y_{[\text{in}],t-1}) = t\mathcal{N}\left(C_t\Theta_t + D_t u_t, R_t, \langle 0, y_{\max}\rangle\right). \quad (12)$$

Here, $t\mathcal{N}\left(\mu, \Sigma, \langle x_{\min}, x_{\max}\rangle\right)$ denotes a Gaussian distribution with mean μ, variance Σ restricted on support interval $\langle x_{\min}, x_{\max}\rangle$, and u_t are the current signal plan settings. The 'mean value' of the relations (11)–(12) is a linear combination of the state Θ_t and the signal plan settings u_t. The coefficients of the combination (matrices A_t, B_t, C_t, D_t) and the 'variances' of the pdfs (matrices Q_t, R_t) are composed of the construction parameters of the intersections and are functions of the observations of the incoming flow, e.g. $A_t = A_t(y_{[\text{in}],t-1})$ [7]. Moreover, since many entries of matrices $A_t, \ldots, D_t, Q_t, R_t$ are uncertain, these can be considered as extra entries of the state Θ_t, yielding non-linear model of the dynamics. Such non-linear model has better modelling capabilities, however, it will cause major computational problems for design of the DM strategy.

B. The aims in the form of ideal distributions

The global aim of our UTC approach is to minimize the total time spend by cars in the network. It was shown that it is equivalent to minimization of the waiting queues ξ_t while maximizing the output intensities $I_{[\mathrm{out}],t}$ [11]. This aim can be formalized in the form of pdfs as follows:

$$ {}^{I}f(\xi_t) = t\mathcal{N}\left(0, V_{\xi}, \langle 0, \xi_{\max} \rangle\right), \tag{13} $$

$$ {}^{I}f(I_{[\mathrm{out}],t}|\xi_t) = t\mathcal{N}\left(I_{[\mathrm{out}],\max}, V_{I[\mathrm{out}]}, \left\langle 0, I_{[\mathrm{out}],\max} \right\rangle\right). \tag{14} $$

Here $\xi_{\max}$ denotes the maximum queue length allowed by intersection construction. The ideal on the queue length, (13), favours short queue lengths, since an estimate $f(\xi_t)$ with lower mean value is closer to the ideal. The 'strength' of the request is governed by the variance V_{ξ}; higher V_{ξ} allows higher deviation from the ideal value. Similarly for (14).

Since the maximization of outgoing intensities may be counterproductive if the neighbouring agent is facing a congestion, each agent also formulates its ideal pdf on its incoming intensities, $I_{[\mathrm{in}],t}$, and negotiates it with its neighbours. At present, we experiment with the following choice:

$$ {}^{I}f(I_{[\mathrm{in}],t}|\xi_t) = t\mathcal{N}\left(I_{\mathrm{w}}(\xi_t), V_{I[\mathrm{in}]}, \left\langle 0, I_{[\mathrm{in}],\max} \right\rangle\right). \tag{15} $$

The requested mean value $I_{\mathrm{w}}(\xi_t)$ decreases with increasing ξ_t, and $I_{[\mathrm{in}],\max}$ is the maximum possible (saturated) intensity. In order to communicate the ideal pdf (15) to the neighbour, it must be conditioned only on the mutually known variables. This can be easily achieved using the results of learning and the operation of marginalization,

$$ {}^{I}f(I_{[\mathrm{in}],t}^{t+1:t+h}) = \int {}^{I}f(I_{[\mathrm{in}],t}^{t+1:t+h}|\xi_t^{t+1:t+h}) f(\xi_t^{t+1:t+h}|d^{1:t}) d\xi_t. $$

C. Evaluating the involved probabilistic operations

General forms of integral equations listed in Section II-D are analytically tractable only for a limited range of models, such as linear Gaussian (LG). However, neither the model (11)–(12) nor the ideal distributions (13)–(15) of our application have such form. The approximations that are necessary for evaluation of the required distributions are outlined next.

1) Learning: In case of known coefficients $A_t, \dots, R_t$ in (11) and (12), the Bayesian learning is equivalent to the well known Kalman filtering [5], with the resulting posterior distributions in the Gaussian form,

$$ f\left(\Theta_t|d^{1:t}\right) = \mathcal{N}\left(\mu_t, \Sigma_{\Theta,t}\right), \tag{16} $$

where μ_t and $\Sigma_{\Theta,t}$ are functions of data. However, the presence of uncertain entries of these coefficients as well as the restrictions on the pdf support violates the assumptions of Kalman filter. Hence, we employ approximate methods for solving non-linear filtering, such as piece-wise linearization, divided difference and unscented filters [5]. In order to achieve tractability of the subsequent operations, the posterior distributions obtained by these non-linear techniques are projected back into the Gaussian form (16).

2) Design of the DM strategy: Since both the model and the ideals are represented as truncated Gaussian pdfs, we expect that the results of FPD for non-truncated Gaussian [8] can be used as a first approximation. In the LG case, FPD is equivalent to the classical technique of linear quadratic control. Specifically, this means that logarithm of the cost-to-go function $\gamma\left(d^{1:t}\right)$ is quadratic, and the resulting control law is a linear combination of the expected value of the state variable Θ_t, i.e.

$$ f\left(u_t|d^{1:t}\right) = \mathcal{N}\left(W\mu_t, \Sigma_{u_t}\right), \tag{17} $$

where μ_t is from (16), and W and Σ_{u_t} are computed from the coefficients of the model and the ideal pdfs. In the LG case, the control action is chosen as the expected value of (17), i.e. $u_t = W\mu_t$.

In our application, we face two problems: (i) non-stationary coefficients $A_t, \dots, R_t$ with (possibly) unknown entries, and (ii) distributions of disturbances in the form of truncated Gaussians. The first problem is addressed by fixing the coefficients at their expected values, using the communicated predictions and the latest estimates of unknown parameters. The second problem is addressed by approximating the truncated Gaussian distribution by a non-truncated variant for most operations, except for the final evaluation of the expected value of (17). The expected value of (17) with restricted support is $u_t = W\mu_t + \varphi(\mu_t, \Sigma_{u_t})$, where $\varphi(\cdot,\cdot)$ is a complex function involving the error functions [14].

3) Merging: In this application, merging of the ideal pdfs is of critical importance. For two Gaussian sources, $f_1 = \mathcal{N}(\mu_1, \Sigma_1)$ and $f_2 = \mathcal{N}(\mu_2, \Sigma_2)$, the minimum of (9) can be found using (10) under assignments $\Phi_1 = [\mu_1, \Sigma_1]$ and $\Phi_2 = [\mu_2, \Sigma_2]$. The merged pdf is $\tilde{f} = \mathcal{N}\left(\tilde{\mu}, \tilde{\Sigma}\right)$, with

$$ \begin{aligned} \tilde{\mu} &= \alpha_1\mu_1 + \alpha_2\mu_2, \\ \tilde{\Sigma} &= \alpha_1\Sigma_1 + \alpha_2\Sigma_2 + \alpha_1\alpha_2\left(\mu_1 - \mu_2\right)\left(\mu_1 - \mu_2\right)'. \end{aligned} \tag{18} $$

In the case of truncated Gaussian pdfs (14)–(15), direct minimization of (9) is too complex. We use the following approximation: (i) ignoring the restricted support of the original pdfs, we apply (18), (ii) we impose the supports of the sources on the merged pdf. The error of this approximation is small if the truncation occurs far from the mean-value-parameter, μ, of both sources, but may get quite large if this parameter is significantly outside of the support.

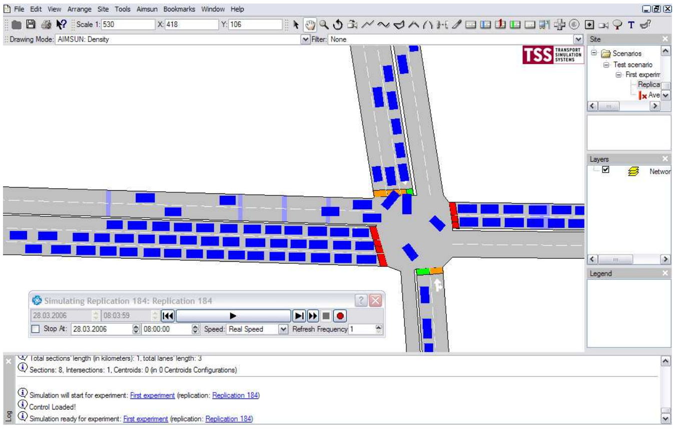
IV. Implementation

A. AIMSUN

An experimental system as the one described above can not be directly deployed on the real intersection controllers. As an intermediate step, we simulate some typical real-world scenarios using the AIMSUN traffic micro-simulator [13]. This simulator is able to conduct a true probabilistic simulation of every single vehicle driving through some modelled road network, and is commonly used to asses the consequences of changes in the road network structure. Using advanced features of AIMSUN programming interface, our prototype controller (programmed in Matlab) is able to receive traffic measurements from AIMSUN simulation and send the optimized signal plan settings back. A snapshot of AIMSUN user interface in the course of simulation is displayed in Figure 4.

B. Simulation results

Typical result of the *learning* phase of the on-line algorithm is displayed in Figure 5. The estimated queue length ξ_t


File Edit View Arrange Site Tools Aimsun Bookmarks Window Help
TSS TRANSPORT SIMULATION SYSTEMS
Site
Scenarios
Test scenario
First experim
Replica
Ave
Layers
Networ
Legend
Simulating Replication 184: Replication 184
Log

Scale 1: 530 X: 418 Y: 106

Drawing Mode: AIMSUN: Density Filter: None

28.03.2006 08:03:59

□ Stop At: 28.03.2006 08:00:00 Speed: Real Speed Refresh Frequency 1

```
Total sections' length (in kilometers): 1, total lanes' length: 3
Sections: 8, Intersections: 1, Centroids: 0 (in 0 Centroids Configurations)

Simulation will start for experiment: First experiment (replication: Replication 184)
Control Loaded!
Simulation ready for experiment: First experiment (replication: Replication 184)
```

Fig. 4. User interface of the AIMSUN environment in the course of a simulation run.

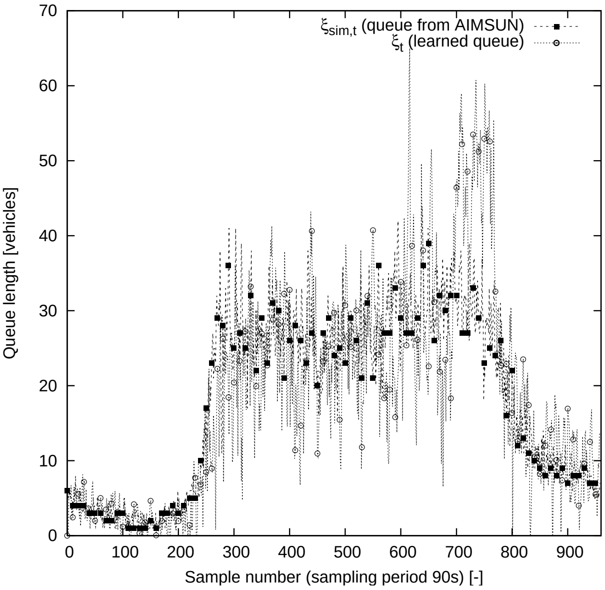


Fig. 5. Values of $\xi_{\mathrm{sim},t}$ and ξ_t for intersection of streets Lidickï¿$\frac{1}{2}$and V botanice in Prague. Traffic data are real measurements from the year 2004 with sampling period 90 sec. The simulation covers a single weekday and starts at midnight.

(evaluated as the expected value of (16)) is compared with the simulated queue length $\xi_{\mathrm{sim},t}$ provided by the AIMSUN. Note that ξ_t follows $\xi_{\mathrm{sim},t}$ with two interesting anomalies: At the beginning of the rush hours, the model tends to underestimate the queues, while it severely overestimates the queues in the evening. The reason of this behaviour may be twofold: (i) the linear coefficients and the variance of (11)–(12) are chosen from empirical rules [7], which may not be accurate, and (ii) the learning of the non-linear model of the internal variables involves approximations—such as projections onto (16)—which are not globally optimal, i.e. the error of approximation may be accumulated. At present, we investigate both possibilities.

V. Conclusion

The work presented in this paper is one of the first applications of distributed Bayesian decision-making theory. We have summarized all key objects and operations of Bayesian agents in their general form. In order to apply the general algorithm, specific choices of model families and parametrizations must be made. The choice of the model structure is very important since it determines if (i) the operations of Bayesian decision-making are computationally tractable, and (ii) if they represent a sufficiently rich description of reality. For the application in traffic control, we have chosen linear models with truncated Gaussian distribution of disturbances. This choice represents a trade-off between tractability and accuracy, since the truncated Gaussians can be approximated by non-truncated Gaussians for which the probabilistic operations are analytically tractable. Thus, the probabilistic operations needed in the traffic control application can be approximated by modifications of the classical algorithms. In the near future, we plan to compare the described approach with some available alternatives.

Acknowledgements

This work was supported by grants MŠMT 1M0572 (DAR) and AVČR 1ET 100 750 401 (BADDYR).

References

[1] J. Andrýsek, M. Kárný, and J. Kracík, editors. *Multiple Participant Decision Making*, Adelaide, May 2004. Advanced Knowledge International.

[2] Ana L. C. Bazzan. A distributed approach for coordination of traffic signal agents. *Autonomous Agents and Multi-Agent Systems*, 10(1):131–164, January 2005.

[3] J.O. Berger. *Statistical Decision Theory and Bayesian Analysis*. Springer-Verlag, New York, 1985.

[4] D.P. Bertsekas. *Dynamic Programming and Optimal Control*. Athena Scientific, Nashua, US, 2001. 2nd edition.

[5] Z. Chen. Bayesian filtering: From Kalman filters to particle filters, and beyond. Technical report, Adaptive Syst. Lab., McMaster University, Hamilton, ON, Canada, 2003.

[6] K. H. Hall, R. J. Staron, and P. Vrba. Holonic and agent-based control. In *Proceedings of the 16th IFAC Congress*, 2005.

[7] J. Homolová and I. Nagy. Traffic model of a microregion. In P. Horáček, M. Šimandl, and P. Zítek, editors, *Preprints of the 16th World Congress of the International Federation of Automatic Control*, pages 1–6, Prague, July 2005. IFAC.

[8] M. Kárný. Towards fully probabilistic control design. *Automatica*, 32(12):1719–1722, 1996.

[9] J. Kracík. On composition of probability density functions. In J. Andrýsek, M. Kárný, and J. Kracík, editors, *Multiple Participant Decision Making*, pages 113–121, Adelaide, May 2004. Advanced Knowledge International.

[10] L.Ljung. *System Identification-Theory for the User*. D. van Nostrand Company Inc., Prentice-hall. Englewood Cliffs, N.J, 1987.

[11] M. Papageorgiou. *Applications of Automatic Control Concepts to Traffic Flow Modeling and Control*, volume 50 of *Lecture Notes in Control and Information Sciences*. Springer-Verlag, Berlin, 1983.

[12] Roger P. Roess, Elena S. Prassas, and William R. McShane. *Traffic Engineering*. Prentice Hall, 3 edition, October 2003.

[13] Transport Simulation Systems. Aimsun: Advanced interactive microscopic simulator for urban and non-urban networks. Url: `http://www.aimsun.com/`, 2006.

[14] V. Šmídl and A. Quinn. *The Variational Bayes Method in Signal Processing*. Springer, 2005.

[15] Gerhard Weiss, editor. *Multiagent Systems: A Modern Approach to Distributed Artificial Intelligence*. The MIT Press, 2000.