Lazy Fully Probabilistic Design: Application Potential

Tatiana V. Guy, Siavash Fakhimi Derakhshan, and Jakub Štěch

Adaptive System Department
Institute of Information Theory and Automation
of the Czech Academy of Sciences
P.O. Box 18, 182 08 Prague 8
{guy,fakhimi,stech}@utia.cas.cz

Abstract. The article addresses a lazy learning approach to fully probabilistic decision making when a decision maker (human or artificial) uses incomplete knowledge of environment and faces high computational limitations. The resulting lazy Fully Probabilistic Design (FPD) selects a decision strategy that moves a probabilistic description of the closed decision loop to a pre-specified ideal description. The lazy FPD uses currently observed data to find past closed-loop similar to the actual ideal model. The optimal decision rule of the closest model is then used in the current step. The effectiveness and capability of the proposed approach are manifested through example.

Keywords: lazy learning, fully probabilistic design, decision making, linear quadratic Gaussian control

1 Introduction

A closed decision-making (DM) loop consisting of *agent-environment* pair is described by the agent's actions and environment states (possibly partially observable). DM problem is to influence the environment behavior in a desired way by choosing and applying a tailored DM policy generating optional actions with respect to the environment. The DM formulation covers stochastic and adaptive control, estimation, filtering, prediction, classification, and others [1]. It has been shown that DM problem can be better treatable in a probabilistic way [2] such as Bayesian DM theory, [3], that provides well-justified solution of DM tasks. The applicability of Bayesian DM theory is limited by the curse of dimensionality, [4], therefore approximate non-linear estimation, [5], and approximate dynamic programming, [6], are mostly inevitable. Practically successful techniques rely on local approximations around the current realisation of the closed-loop behaviour.

This paper is a part of the project trying to lay a ground for *lazy* Fully Probabilistic Design. Lazy Learning (LL) is an approach that searches and uses relevant information from the past data. Inspired by human reasoning it decreases deliberation effort by employing early-developed solutions. A simple fact, that similar DM tasks tend to have similar solution, has caused the approach

has evolved in many areas under different names. Lazy-learning philosophy [7] has been presented as case-based reasoning, memory-based learning, analogical modelling, memory-based prediction, just-in-time modelling, transfer learning, see for instance [8–10]. All of these experience-based methods are problem solving processes in which an actual problem, defined on the same domain as the past problems, is solved by searching for a similar situation and using its solution. These methods are used for transfer learning aiming at improving performance and learning on a new domain by learning from the past [11].

FPD, an extension of the Bayesian DM, solves a DM problem by considering probabilistic description of both environment behaviour and DM preferences [2]. The main aim is then to find an optimal policy minimising the divergence the probabilistic description of *actual* closed-loop behaviour from that of *ideal* closed-loop behaviour, which expresses DM preferences.

In this paper, a combination of LL and FPD is employed to utilize the competence of both techniques in opting tailored action at each time step when the knowledge of environment is incomplete. As a result the proposed solution not only provides the desired decrease of computation demands, but also its overall performance is comparable to the performance of the standard FPD. The proposed approach focuses on single-agent DM aiming at creating efficient and scalable solution that can easily be extended to multi-agent settings.

The layout of the paper is as follows. Section 2 introduces formal notations and necessary preliminaries together with a formal description of FPD. Section 3 formulates the lazy FPD problem and outlines its solution. Experimental section demonstrates the effectiveness of our approach on attitude control of the hovering helicopter. Finally, Section 5 summarises the main results and outlines the open problems remained.

2 Underlying theory

This section introduces necessary conventions and notions.

2.1 Preliminaries

The sequence $(x_t, x_{t+1}, \ldots, x_{t+h})$ is shortened as $x(t, t+h)$. Discrete time instances are labelled by $\tau = 1, 2, \ldots, t$, $t \in \mathbb{N}$. Bold capital $\mathbf{X}$ represents a set of x values. An abbreviation *pdf* denotes probability density function. The Kullback-Leibler divergence (KLD), [12], measuring the proximity of two pdfs f and g, acting on a set $\mathbf{X}$, reads

$$\mathcal{D}(f||g) = \int_{\mathbf{X}} f(x) \ln \frac{f(x)}{g(x)} \mathrm{d}x, \tag{1}$$

with $\mathcal{D}(f||g) \geq 0$, $\mathcal{D}(f||g) = 0$ iff $f = g$ almost everywhere on $\mathbf{X}$.

Let us consider an interacting agent-environment pair, see Fig.1. The agent observes a new environment state $s_t \in \mathbf{S}$ at time t and chooses action $a_t \in \mathbf{A}$ to learn or influence the environment in accordance with the agent's DM preferences. Having action selected, the environment moves to the next state and the

agent receives one-step reward. The aim of the agent is to find optimal policy maximizing the future reward.

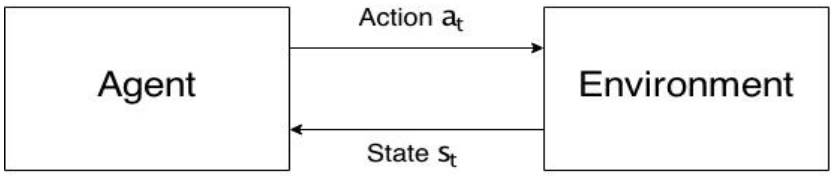

Fig. 1. *The closed decision loop.*

The closed-loop model of the *environment-agent* pair, is fully described by joint pdf $p(s_{t+h}, a_{t+h-1}, s_{t+h-1}, \ldots, s_{t+1}, a_t), s_\tau \in \mathbf{S}, a_\tau \in \mathbf{A}, t \leq \tau \leq t+h, t, \tau, h \in \mathbb{N}$, that can be factorised using the chain rule for pdfs, [13], as follows:

$$p_{(t,h)} = \prod_{\tau=t+1}^{t+h} p(s_\tau|s(t,\tau-1), a(t,\tau-1))p(a_{\tau-1}|s(t,\tau-1), a(t,\tau-2))p(s_t). \quad (2)$$

The first factor, $p(s_\tau|s(t,\tau-1), a(t,\tau-1))$, is environment model, the second factor, $p(a_\tau|s(t,\tau), a(t,\tau-1))$, is a randomised DM rule and $p(s_t)$ is a prior pdf of state. A sequence of DM rules, $\{p(a_\tau|s(t,\tau), a(t,\tau-1))\}_\tau$, up to time $t+h$, forms *DM policy* $\pi_\tau : (\mathbf{S}^\tau \times \mathbf{A}^{\tau-1}) \mapsto \mathbf{A}^\tau$.

2.2 Fully probabilistic design

Any systematic DM design selects a DM policy that makes the resulting closed-loop model (2) close to the desired one. FPD [2] considers the desired probabilistic closed-loop model as *ideal model* that expresses the agent's preferences. An advantage of FPD is an ability to explicitly describe multiple aims and constraints. The resulting optimal DM policy is randomised, unlike in the standard Bayesian DM. Let us consider the following simplified Markov version of (2):

$$p_{(t,h)} = \prod_{\tau=t+1}^{t+h} p(s_\tau|a_{\tau-1}, s_{\tau-1})p(a_{\tau-1}|s_{\tau-1})p(s_t). \quad (3)$$

In (3), t is a starting step and $h \in \mathbb{N}$ is a finite horizon. The corresponding ideal model reflecting agent's DM preferences reads:

$${}^I p_{(t,h)} = \prod_{\tau=t+1}^{t+h} {}^I p(s_\tau|a_{\tau-1}, s_{\tau-1}) {}^I p(a_{\tau-1}|s_{\tau-1})p(s_t). \quad (4)$$

FPD, provides a DM policy yielding minimum of the KLD, (1), from the *current* closed-loop description, (3), to the *ideal* one, (4). Thus optimal DM policy π^{opt} coming from the minimisation is

$$\pi^{opt} = \arg \min_{\{p(a_\tau|s(\tau))\}_{t\leq\tau\leq t+h-1}} \mathcal{D}(p_{(t,h)}||{}^{I}p_{(t,h)}), \quad \sum_{a_\tau\in\mathbf{A}} p(a_\tau|s(\tau)) = 1. \qquad (5)$$

3 Lazy fully probabilistic decision making

Lazy learning is an approach, which at the actual time step goes through the stored data and searches the relevant data to deal with a current DM problem, see Fig. 2. In this figure, the red points indicate the similar situations in the past and different closed-loop sequences $(s_{\tau+1}, a_\tau, s_\tau)$. We are intending to find an optimal DM policy that respects our current ideal, based on the past optimal actions.

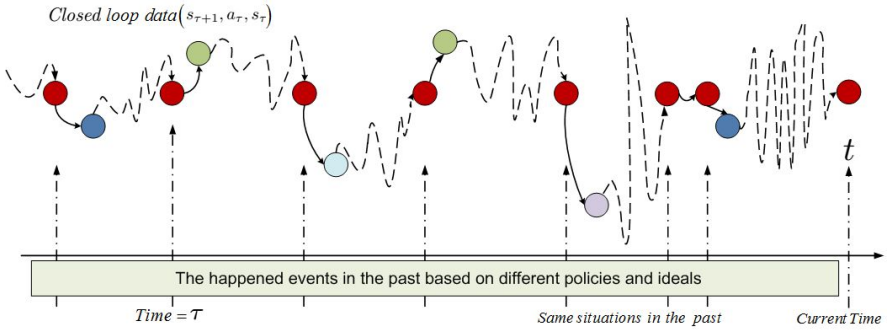

Fig. 2. *Lazy-learning fully probabilistic decision making strategy.*

This section describes the general idea of the proposed solution. Let us consider a DM task $Q_{(t,h)} = (p_{(t,h)}, {}^{I}p_{(t,h)})$ where $p_{(t,h)}$ and ${}^{I}p_{(t,h)}$ is given by (3) and (4), respectively. The collected historic data contain environment states $s_\tau \in \mathbf{S}$ and actions $a_\tau \in \mathbf{A}$, $\tau < t$, observed up to actual time t. The data describe past (solved) DM tasks $Q_{(\tau,h)}$, $\tau = 1, \ldots, t-h$. The following assumption is considered.

Assumption 1. *Actions $a_\tau, \tau < t$ applied in the past DM tasks sufficiently well approximate the optimal solution with respect to the past ideal models ${}^{I}p_{(\tau,h)}$.*

This assumption justifies considering the past actions and employing them to find current optimal action even without explicit knowledge of past ideal closed-loop models ${}^{I}p_{(\tau,h)}$, $\tau \leq t-h$. Next, we need to find a_t^{opt} which makes closed-loop $p_{(t,h)}$ close to its ideal counterpart ${}^{I}p_{(t,h)}$. The proposed solution requires the following assumptions reflecting real-life DM tasks.

Assumption 2. *There exists at least one past ideal ${}^{I}p_{(\tau,h)}$ that is sufficiently close to the current ideal closed-loop model ${}^{I}p_{(t,h)}$.*

Assumptions 2 ensures that past experience is sufficiently rich to cover the current DM task. It also allows to search for the similar task in the whole past history.

Assumption 3. *The environment behaviour does not significantly change over time period considered.*

Technically Assumption 3 means that probabilities in (3) do not change with time. Note that Assumption 3 is not so restrictive. Once its violation is suspected, different forgetting-like techniques [14] can be applied.
The proposed solution of the lazy FPD is given by the following proposition.

Proposition 1. *Consider a set of past DM tasks $Q_{(\tau,h)} = (p_{(\tau,h)}, {}^{I}p_{(\tau,h)}), \tau = 1, \ldots, t-h$ respecting Assumptions 1-3. Then optimal action a_t^{opt} for the current DM task can be found as follows:*

$$\begin{aligned} \tau^{opt} &= \arg \max_{\tau \in (0,t-h)} {}^{I}p(\tau, h) \\ a_t^{opt} &= a_{\tau^{opt}}. \end{aligned} \tag{6}$$

The maximisation in (6) runs over past sequence of states and actions

$$(s_{\tau+h}, a_{\tau+h-1}, s_{\tau+h-1}, \ldots, s_{\tau+1}, a_\tau), \quad \tau \in \mathbb{N}$$

such that states observed at times τ and t are virtually equal. An optimal action is then taken from a sequence maximising the current ideal closed-loop model.

4 Experiment

This section aims to verify the effectiveness of the proposed single-agent strategy. A linear model of the helicopter in hovering is considered as an example. The DM strategy designed by the presented approach is compared with a Linear Quadratic Gaussian (LQG) control strategy.

4.1 Lazy-learning fully probabilistic LQG

The helicopter has six degrees of freedom in its motion, [15]. There are four control inputs concerning its flight in addition to throttle control. By coordinating these inputs the helicopter can make forward and backward flight, sideward flight, hovering, hovering turn, vertical climb and descent, etc.

Assuming the main rotor is composed of two blades without dragging motion, the vehicle mass center is located under the rotor shaft, rotor angular velocity is constant in hovering, and the tail rotor is composed of two blades and its hub center is located on the fuselage longitudinal axis, the model of helicopter can

be separated into two parts. The first part represents main rotor dynamic and the second one models dynamics behaviour of the tail rotor.

In the hovering mode, only main rotor dynamics describes the roll and pitch movement of the craft. The aim is to move roll and pitch angle to zero values.

We consider the following linear model of helicopter, details see [16]:

$$\begin{bmatrix} \phi_{t+1} \\ \dot{\phi}_{t+1} \\ \theta_{t+1} \\ \dot{\theta}_{t+1} \end{bmatrix} = \begin{bmatrix} 1 & 0.021 & 0 & 0.0002 \\ 0 & 0.99 & 0 & 0.025 \\ 0 & -0.0013 & 1 & 0.02 \\ 0 & -0.1820 & 0 & 0.848 \end{bmatrix} \begin{bmatrix} \phi_t \\ \dot{\phi}_t \\ \theta_t \\ \dot{\theta}_t \end{bmatrix} + \begin{bmatrix} 0.06 & 0.0032 \\ 4.75 & 0.45 \\ -0.0098 & 0.313 \\ -1.18 & 27.356 \end{bmatrix} \begin{bmatrix} \theta_t^s \\ \theta_t^c \end{bmatrix} \tag{7}$$

In (7) t denotes discrete time step, $\dot{\theta}$ and θ are pitch angular velocity and pitch angle, $\dot{\phi}$ and ϕ are roll angular velocity and roll angle, and θ^s and θ^c are roll control (laterally cyclic) and pitch control (longitudinally cyclic), respectively. Under

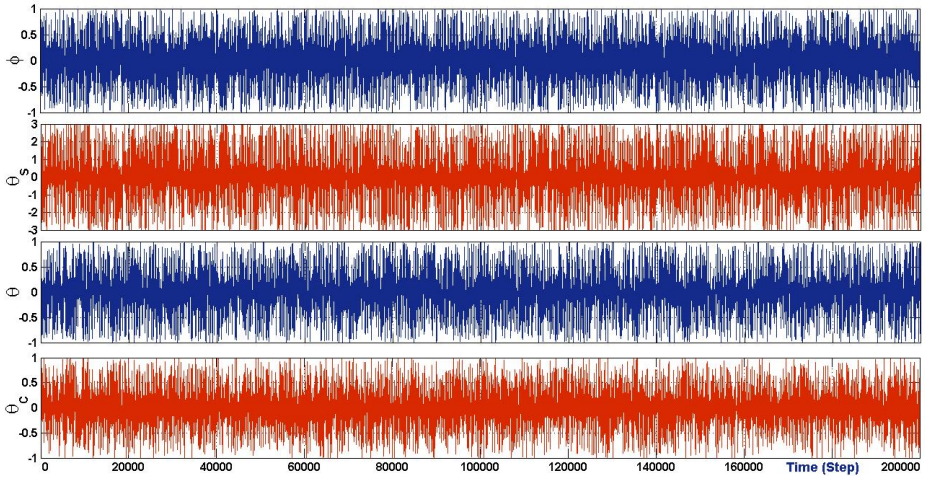

Fig. 3. *The closed loop behaviour under different strategies.*

different strategies implied by various unknown ideals, roll and pitch movements are shown in Fig. 3. In order to gather the closed-loop data, the decentralised Proportional-Derivative (PD) controllers, [17], with different parameters are employed. Since system outputs are continuous, finding similar past data requires an infinite database. To solve this problem, control actions and system outputs are discretised in values, see Fig. 4. As it can be seen in Fig. 4, roll and pitch movements respond correctly and the helicopter moves to the hovering position from the different initial states. Fig. 5 depicts a histogram of control actions when the current value of $\phi \in (0.798, 0.8)$ while the previous value of $\phi \in (0.998, 1.0)$. The diversity of actions guarantees that finding tailored set of control actions based on a given ideal FPD and past data is highly plausible.

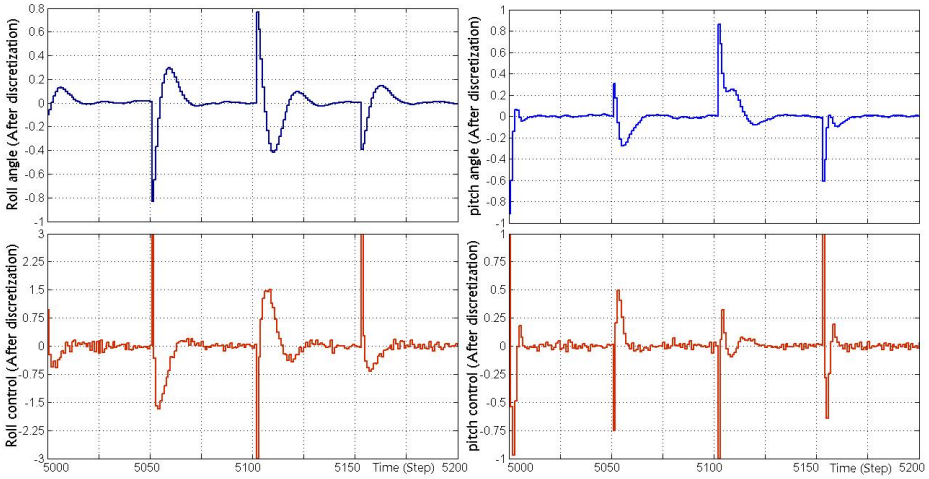

Fig. 4. *The discrete-valued system outputs and control actions.*

In order to formulate lazy-learning fully probabilistic LQG in hovering mode, the ideal state distribution and ideal controller strategy are assumed to be Gaussian with zero mean value and covariance matrices $\Sigma > 0$ and $R > 0$:

$$^{I}p(s_{\tau+1}|a_\tau, s_\tau) = \mathcal{N}_{s_{\tau+1}}(0, \Sigma) \tag{8}$$

$$^{I}p(a_\tau|s_\tau) = \mathcal{N}_{a_\tau}(0, R), \tag{9}$$

For the linear Gaussian state-space model, the controller found by FPD approach can be interpreted as a standard LQG with a state penalization matrix Σ^{-1} and input penalization matrix R^{-1}, details see [2].

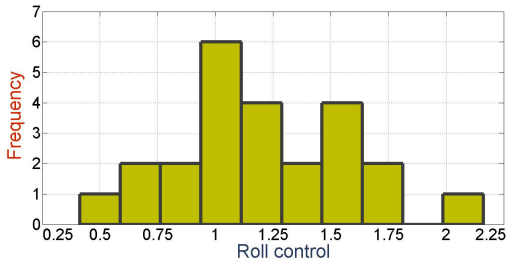

Fig. 5. *The histogram of control actions.*

By considering $\Sigma^{-1} = diag(1, 1, 0, 0)$ and $R^{-1} = \mathcal{I}_2$, where $\mathcal{I}$ is the identity matrix, and substituting (8) and (9) into (4), the ideal close-loop behaviour is

defined as follows:

$$ {}^{I}p_{(t,h)} \propto \prod_{\tau=t}^{t+h-1} e^{-(\hat{\theta}^2_{\tau+1}+\hat{\phi}^2_{\tau+1}+(\theta^s_\tau)^2+(\theta^c_\tau)^2)}. \tag{10} $$

Roll and pitch trajectories for the initial condition $s(0) = \left[0.75,\, 0,\, -0.5,\, 1.0\right]^T$ obtained by lazy-learning FPD $h = \{1, 10\}$, LQG approach and PD regulator can be seen at Fig. 6. Under the same initial condition, Fig. 7 illustrates the evolution of control signal. Fig. 6 and Fig. 7 indicate that the roll and pitch movements respond correctly and the helicopter is conduced to the hovering position. Fig. 6 and Fig. 7 clearly demonstrate closeness of the proposed approach

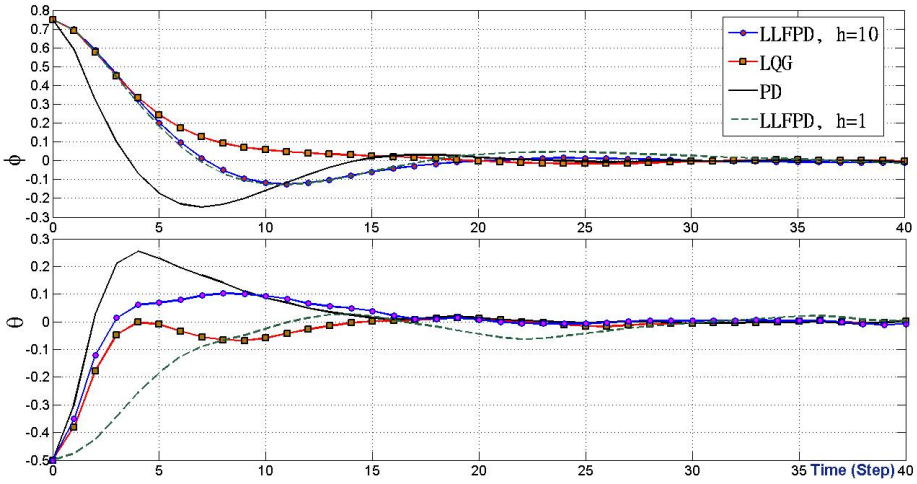

Fig. 6. *Trajectories of the roll and pitch angle.*

to the LQG control, and show that even when the decision maker uses incomplete knowledge of environment, the proposed approach is very effective. By other words, the proposed approach obviously alleviates the computational load needed and decreases the dependency on accurate knowledge of environment in the FPD approach (Proposition 2 in [2]).

A detailed comparison of the approaches (see Table 1) is based on performance indices are calculated for different strategies. In particular we considered:

- *Transient cost*: closed-loop performance index in the first 20 steps under the influence of initial state.
- *Persistent cost*: value of closed-loop cost function in the last 100 steps under the influence of the process noise and the measurement noise.
- *Total cost*: value of the performance indices under the influence of initial state and Gaussian noise.

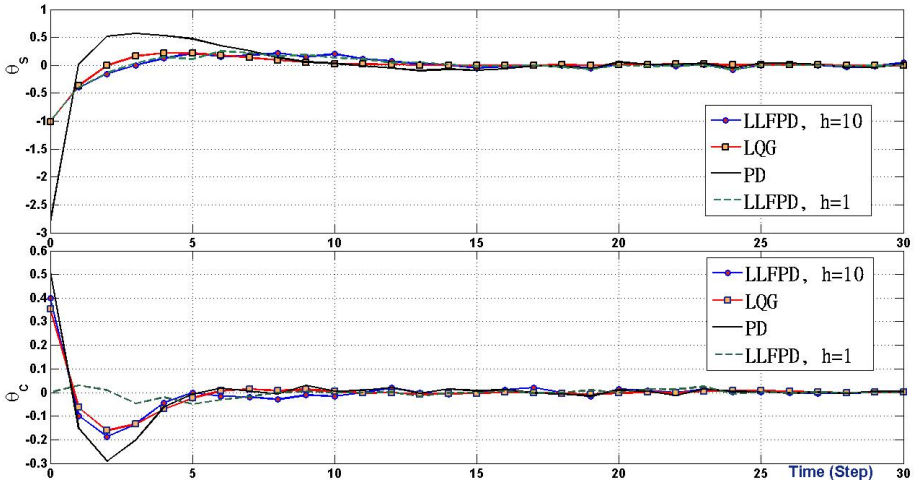

Fig. 7. *Evolution of the control signals.*

From Table 1, it can be seen that proposed approach (LLFPD LQG) with horizon $h = 10$ chooses the optimal control action. Compare to the standard LQG, LLFPD with $h = 10$ had competitive responses in the transient and persistent state. Moreover under incomplete knowledge, LLFPD is a good alternative to FPD approach. The proposed approach provides results that highly outperform PD controller. Moreover it can reach visibly high control quality (see Table 1) with comparable computational effort.

Table 1. Performance quality

Method	Cost Function $J(t_{min}, t_{max}) = \Sigma_{t=t_{min}}^{t_{max}}(\theta_{t+1}^2 + \phi_{t+1}^2 + (\theta_t^s)^2 + (\theta_t^c)^2)$		
	Total cost $J(0, 200)$	Transient cost $J(0, 20)$	Persistent cost $J(100, 200)$
LQG	3.19623	3.10324	0.04751
PD	6.53832	6.37771	0.08482
LLFPD, $h = 1$	3.50326	3.19382	0.11641
LLFPD, $h = 2$	3.46451	3.22531	0.09185
LLFPD, $h = 5$	3.38828	3.20451	0.07306
LLFPD, $h = 10$	3.34851	3.21622	0.06081

5 Concluding remarks

The paper describes lazy fully probabilistic design of DM strategies. The idea is based on searching similar previously experienced closed-loop models. The similarity criterion is maximisation of the current DM preferences. Instead of searching over the whole action space, the approach investigates the previously experienced DM tasks only. The solution can be of help even when past ideal models are unknown.

The proposed solution significantly decreases: i) the computational load needed by FPD and other design techniques; ii) danger of choosing inappropriate DM preferences that are based on little or no knowledge of the environment. Moreover switching between different controllers can have weak stability (see [18]) while LL FPD provides stable closed-loop behaviour. The lazy FPD also allows for efficient preference elicitation, (especially when no prior knowledge is available), see [19, 20]. In this case suitable past ideal models can be used as ideal for the current DM task.

LL FPD approach designs an efficient optimising single-agent DM that does not depend on perfect knowledge of the environment and thus can create a reliable base for multi-agent systems. The approach also gives a way how to transfer ideals/models between different agents solving similar DM tasks on the same environment. This ability is highly demanded in many real-world applications where knowledge transfer cannot be easily ensured.

Acknowledgement: The authors would like to thank Miroslav Kárný for valuable discussions and comments. The research has been partially supported by the Czech Science Foundation, project GA16-09848S.

References

1. Kárný, M., Böhm, J., Guy, T.V., Jirsa, L., Nagy, I., Nedoma, P., Tesař, L.: Optimized Bayesian dynamic advising: Theory and algorithms. Springer, London (2006)
2. Kárný, M.: Towards fully probabilistic control design. Automatica **32**(12) (1996) 1719–1722
3. Savage Leonard, J.: The foundations of statistics. NY, John Wiley (1954) 188–190
4. Bellman, R.: Adaptive control processes. Princeton Press, NJ (1961)
5. Roll, J., Nazin, A., Ljung, L.: Nonlinear system identification via direct weight optimization. Automatica **41**(3) (2005) 475–490
6. Powell, W.B.: Approximate Dynamic Programming: Solving the curses of dimensionality. Volume 703. John Wiley & Sons (2007)
7. Aha, D.: Artificial Intelligence Review–Special Issue on Lazy Learning, 11 (1-5) (1997)
8. Aamodt, A., Plaza, E.: Case-based reasoning: Foundational issues, methodological variations, and system approaches. AI communications **7**(1) (1994) 39–59
9. Bitanti, S., Picci, G.: Identification, adaptation, learning. NATO ASI Series F on Computer and Systems Sciences (1996)
10. Weiss, K., Khoshgoftaar, T.M., Wang, D.: A survey of transfer learning. Journal of Big Data **3**(1) (2016) 9

11. Klenk, M., Aha, D.W., Molineaux, M.: The case for case-based transfer learning. AI Magazine **32**(1) (2011) 54
12. Kullback, S., Leibler, R.A.: On information and sufficiency. The annals of mathematical statistics **22**(1) (1951) 79–86
13. Peterka, V.: Bayesian approach to system identification. Trends and Progress in System Identification **1** (1981) 239–304
14. Kulhavy, R., Kárný, M.: Tracking of slowly varying parameters by directional forgetting. In: Proc. 9th IFAC World Congress. Volume 10. (1984) 78–83
15. Adachi, S., Hashimoto, S., Miyamori, G., Tan, A.: Autonomous flight control for a large-scale unmanned helicopter. IEEJ Transactions on Industry Applications **121**(12) (2001) 1278–1283
16. Gil, I.A., Barrientos, A., Del Cerro, J.: Attitude control of a minihelicopter in hover using different types of control. Revista Técnica de la Facultad de Ingeniería. Universidad del Zulia **29**(3) (2006)
17. Ambrosino, G., Celentano, G., Garofalo, F.: Decentralized pd controllers for tracking control of uncertain multivariable systems. IFAC Proceedings Volumes **18**(5) (1985) 1907–1911
18. Hou, Z.S., Wang, Z.: From model-based control to data-driven control: Survey, classification and perspective. Information Sciences **235**(Supplement C) (2013) 3 – 35
19. Kárný, M., Guy, T.: Preference elicitation in fully probabilistic design of decision strategies. In: Proc. of the 49th IEEE Conference on Decision and Control. (2010)
20. Braziunas, D., Boutilier, C.: Preference elicitation and generalized additive utility (nectar paper). In: Proc. of the 21st Nat. Conf. on AI (AAAI-06), Boston, MA (2006)