



CZECH TECHNICAL UNIVERSITY IN PRAGUE
Faculty of Nuclear Sciences and Physical Engineering



Active Adaptive Algorithmic Quantification of Preferences

Aktivní adaptivní algoritmická kvantifikace preferencí

Master thesis

Author: **Bc. Tereza Siváková**
Supervisor: **Ing. Miroslav Kárný, DrSc.**
Academic year: 2021/2022

ZADÁNÍ DIPLOMOVÉ PRÁCE

Student: Bc. Tereza Siváková
Studijní program: Aplikace přírodních věd
Studijní obor: Aplikované matematicko-stochastické metody
Název práce (česky): Aktivní adaptivní algoritmická kvantifikace preferencí
Název práce (anglicky): Active Adaptive Algorithmic Quantification of Preferences

Pokyny pro vypracování:

- 1) Prohlubte si znalosti bayesovského odhadování.
- 2) Prohlubte si znalosti plně pravděpodobnostního návrhu (PPN) rozhodovacích strategií.
- 3) Doplněte si přehled stavu problematiky algoritmického získávání preferencí založeném na aktivní interakci s rozhodujícím.
- 4) K pasivnímu řešení založeném na PPN a průběžném odhadování, které jste rozvíjela v rámci výzkumného úkolu, přidejte aktivní vrstvu, která umožňuje dle reakcí rozhodujícího nastavovat volitelné parametry vrstvy základní. I pro tuto vrstvu zachovejte metodologii vrstvy základní.
- 5) Navržené řešení implementujte v systému Matlab a proveďte extenzivní vyhodnocení kvality navrženého řešení.

Doporučená literatura:

- 1) C. Boutilier, A POMDP Formulation of Preference Elicitation Problems. AAAI-02 Proceedings, AAAI, 2002, 239-246.
- 2) J. Branke, S. Corrente, S. Greco, and W. Gutjahr, Efficient pairwise preference elicitation allowing for indifference, Computers & Operations Research, vol. 88, no. Suppl. C, 2017, 175-186.
- 3) M. Kárný and T.V. Guy, Preference Elicitation within Framework of Fully Probabilistic Design of Decision Strategies, IFAC International Workshop on Adaptive and Learning Control Systems - ALCOS, vol. 52, no. 29, 2019, 239-244.
- 4) M. Kárný and M. Ruman, Preference Elicitation for Markov Decision Processes within Fully Probabilistic Design Framework, IEEE Tran. Systems, Man, and Cybernetics: Systems, 2021, submitted
- 5) V. Peterka, Bayesian system identification, in Eykhoff, P. (Ed.), Trends and Progress in System Identification. Perg. Press, 1981, 239-304.
- 6) G. Pigozzi, A. Tsoukiàs, and P. Viappiani, Preferences in artificial intelligence, Annals of Mathematics and Artificial Intelligence, vol. 77, no. 3, 2016, 36-401.

Jméno a pracoviště vedoucího diplomové práce:

Ing. Miroslav Kárný, DrSc.

ÚTIA AV ČR, v.v.i., Pod Vodárenskou věží 4, 182 00, Praha 8

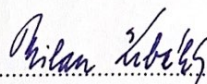
Jméno a pracoviště konzultanta:


Datum zadání diplomové práce: 28.2.2021

Datum odevzdání diplomové práce: 5.1.2022

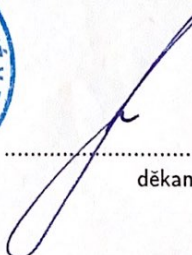
Doba platnosti zadání je dva roky od data zadání.

V Praze dne 28.2.2021


.....
garant oboru


.....
vedoucí katedry




.....
děkan

Acknowledgment:

I would like to thank my supervisor Ing. Miroslav Kárný, DrSc. for his expert guidance, helpfulness, professional and human approach to leading my research project. I express my gratitude for his language assistance. I would like to thank Ing. Tatiana Valentine Guy, Ph.D. for the support of this project with grant MŠMT LTC18075 and EU-COST Action CA16228.

Author's declaration:

I declare that this Master thesis is entirely my own work and I have listed all the used sources in the bibliography.

Prague, January 5, 2022

Tereza Siváková

Název práce:

Aktivní adaptivní algoritmická kvantifikace preferencí

Autor: Tereza Siváková

Obor: Aplikované matematicko-stochastické metody

Druh práce: Diplomová práce

Vedoucí práce: Ing. Miroslav Kárný, DrSc., ÚTIA AV ČR, v.v.i.

Abstrakt: Tato diplomová práce se zabývá dynamickým rozhodováním za použití plně pravděpodobnostního návrhu. Tento návrh modeluje uzavřenou rozhodovací smyčku splňující agentovy preference pomocí *ideální distribuce chování*, která přiřazuje vysoké hodnoty pravděpodobnosti preferovanému chování a malé hodnoty pravděpodobnosti nežádoucímu chování. Následně nalezneme optimální rozhodovací politiku pomocí minimalizace Kullback-Leiblerovy divergence reálné distribuce chování a ideální distribuce chování. Optimální politika pak vybere v každém kroku, při pozorovaném stavu, uzavřené smyčky, takovou akci, díky které se systém s nejvyšší pravděpodobností posune do preferovaného stavu. V této práci se také zabýváme možností přidání vyvážené preference na volbu akcí. Kromě výše zmíněného se zabýváme zpětnou vazbou agenta na vývoj rozhodování. Agent známkuje známkami od 1 do 5 jako ve škole, jak se mu poslušnost stavů a akcí líbí. Přidáváme optimalizační vrstvu, která nastavuje vrstvu základní tak, aby bylo co nejvíce vyhověno agentovým preferencím vyjádřenými známkami.

Klíčová slova: bayesovské odhadování, rozhodování, plně pravděpodobnostní návrh, politika, preference, zjišťování preferencí

Title:

Adaptive Algorithmic Quantification of Preferences

Author: Tereza Siváková

Abstract: This thesis studies optimal decision making with the focus on preferences quantified for fully probabilistic design (FPD). FPD introduces the so-called *ideal behaviour distribution*, which has high probability values of preferred behaviour and low probability values of inappropriate behaviours. By minimizing the Kullback-Leibler divergence of the real behaviour distribution and the ideal behaviour distribution an optimal decision policy is found. The policy in every time epoch and for the observed closed-loop state selects the action, thanks to which the system transits to the preferred state with the highest probability. This research also studies preferences targeting actions as well as contradicting preferences. In addition to the above, we deal with the agent's feedback to decision-making. The agent grades the achieved behaviour by marks from 1 to 5 as in school, as he likes the sequence of states and actions. We are adding an optimization closed-loop that tunes the basic closed-loop to meet the agent's preferences expressed by marks as much as possible.

Key words: Bayesian estimation, decision-making, fully probabilistic design, policy, preference, preference elicitation

Contents

Introduction	7
1 Preliminaries	13
1.1 Notions and definitions	13
1.2 Markov decision process	14
1.3 Decision making via fully probabilistic design	15
1.4 Bayesian learning of Markov Chain	17
2 Generic and Specific PE in FPD	20
2.1 The generic choice of optimal ideal model of the system	21
2.2 The generic choice of optimal ideal decision rule	21
2.3 The specific choice of \mathbb{M}^i making $\mathbb{C}^i \neq \emptyset$	23
2.3.1 The specific choice of \mathbb{m}^i	24
2.4 Algorithmic summary for discrete-valued states and actions	27
3 Preference elicitation as a dialogue with the user	29
4 Experiments	31
4.1 Common Simulation and Evaluation Options	31
4.2 Decision making without user's control	32
4.3 Decision making with user's control	34
4.4 Comparison of costs and responses in all experiments	37
5 Conclusions	40
A Additional graphs	42

Introduction

In this master's thesis, we deal with the quantification of preferences in the theory of dynamic decision-making. Decision-making (DM) is an important part of every man, institution, firm and societal activity; further on, we refer to them as agents. An agent must make hundreds of decisions per day, less or more important. Our work concerns cases in which we inspect, how to make optimal decisions to achieve some predetermined goal. The key problem addressed here is how to describe correctly mathematically a certain, incompletely specified, target and our other conditions for the DM problem, then find the optimal decisions aiming to achieve this goal

Many articles concern this preference elicitation (PE) for various DM set-ups, for example [7], [5] or [9]. The PE is a process of obtaining the quantitative description of the agent's preferences. It is an essential part of DM. We need to know what the agent wants to give them the optimal decisions they may use as their actions. In this thesis, we try to solve it in a non-standard way. We use the fully probabilistic design (FPD), which is an alternative to the usual DM based on the Markov decision process (MPD). Compared to our previous work, we also tune input parameters by asking queries during the DM process. This should allow the agent to fine-tune their preferences and gain insight into the DM problem.

To characterize our work technically, we must explain the (MDP), see [23], that provides the mathematical framework of DM. The MDP works on a system, which is a 'cut out' part of the world. It is described by transition probability density between individual states, in which the system may occur. The MDP calls the DM goal as the preferred state. Thus, a set of all possible goals is a set of states. The set of individual decisions that can be made is the set of actions. The transition between the individual states depends on the selected action. A key feature of Markov's DM process is that it does not matter in what state the system was a year or a week ago. It only depends on the latest state of the system and on the action, which is selected.

The agent observes the system, this allows him to determine the latest state (the more general case of partially observable state from [3] is left aside for simplicity). Based on the observation, the agent selects the action. The chosen action is selected according to the agent's decision policy (the adopted widespread synonym for strategy). The policy maps states on actions. The connection between the agent and randomly behaving system is called a closed-loop. It is sketched in Figure 1.

The policy consists of a sequence of decision rules, which are applied in every DM step. The decision rule says which action will be chosen. The decision rule can be randomized and it can be strategically selected to optimally gain a certain goal. The design of decision policy is influenced by other factors and information that the agent carries: a model of the system and agent's preferences. The agent's model of the system expresses the agent's assumptions about the reaction of the system to the action. The agent gets the model of the system before they start solving the decision problem or estimates it during the DM process by Bayesian learning, see in [22]. In Bayesian learning, the agent uses a parametric model of the system. It is a parametric description of transition probability. Their knowledge about the parameters

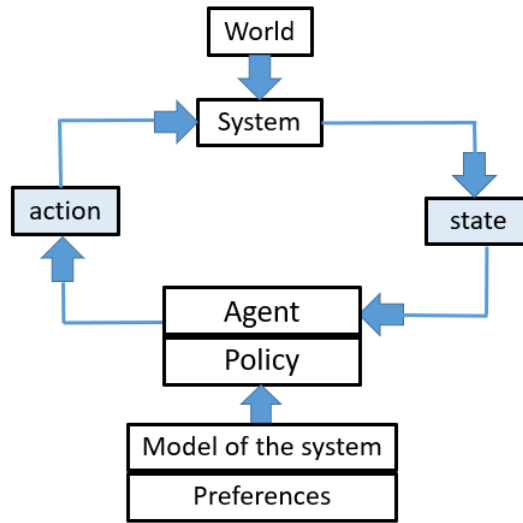


Figure 1: Scheme of the closed-loop: model of the system and preference serve to policy design

is given by a prior probability density (pd) ¹ and by the past observed data. Then the agent corrects its incomplete knowledge of the parameters by Bayes' rule, which together with the parametric model gives it the predictive pd. We assume that actions and states evolve, but the transition probability density is time-invariant.

The closer the predictive pd to the real system the better. If the agent knows the transition probability density between the system states, they can easily select the action that causes the transition to the preferred state with the highest probability. They are also capable to find the optimal decision rule more easily. The optimal decision rule selects the action that moves the system to the preferred state with the highest probability. The policy is evaluated by using dynamic programming, can be found in [1], which is a technique for solving a complex problem by dividing it into a collection of simpler partial problems. This method is used from the end of the planning period (horizon) to the beginning. Also, it should be noted that the world influences the transition probability density of the system because the system is a part of the world and if something influences the world it influences the system as well.

The fact that preferences affect policy is perhaps already obvious because optimized decision rule is selected based on preferences, based on the user's goal. In MDP the preferences are expressed by the utility function.

The optimal policy should be the one that maximizes the expected utility function. The utility function quantitatively expresses the 'utility' of the choice of action and it helps to mathematically measure the quality of the choice and thus of the decision rule. The preferred states have higher utility for the agent. So we want to maximize the expected utility/reward of every selected action and the following state or we want to minimize the expected loss. The loss can be interpreted as a price that has to be paid for deviation from preferred states and actions.

The utility function is compiled based on preferences and the 'compilation' process forms the core of preference elicitation.

¹The probability density (pd) means probability density function for continuous values and probability for discrete values. When handling them, we use often integral notation that means summing in the discrete case.

So before the beginning of the decision-making, we need to obtain the set of possibly observed states, the set of admissible actions, the model of the system, the utility function describing preferences. Then the decision-making is just an optimization problem. We need to find a sequence of decision rules that minimizes the expected loss, measure the last observation (state) and generate the action according to the optimal decision rule.

In this thesis, we work with a fully probabilistic design (FPD), see in [15], instead of MDP. FPD is an alternative to MDP. FPD uses probability distribution instead of the utility function. Agent's preferences are reflected in the ideal probability distribution. FPD tries to get the real density as close as possible to the ideal probability distribution. The ideal probability distribution assigns high values of probability to preferred states and low values to undesirable states. The optimal policy is then evaluated as an argument of the minimum of the Kullback-Leibler divergence, can be found in [18], of the real probability distribution and the ideal probability distribution. When we talk about preference elicitation it is easier to work with this mathematical framework than with MDP as it is easier to work with ideal probability distribution than with the utility function. If the preferences are explicitly specified at the beginning of the decision-making process, the task was solved satisfactorily as was shown in our previous work [14, 26]. However, if the agent has multiple preferences, the task no longer has a unique solution and it depends only on the agent, which solution they prefer. Also, it can happen that the preferences can not be reached because the agent has unfulfillable targets. The agent preferences do not always have to be achievable because the transition probability density of the system and the probability distribution may differ from the agent's expectations. The agent may have completely unrealistic wishes. For example, winning a marathon without ever running and preparing. It is possible because the agent can have some hidden talent, but in reality, this probability is minuscule.

The main task of preference elicitation (PE) is to mathematically describe the user's preferences and optimally solve the DM problem. Preference elicitation is a process of extracting the necessary information on preferences from the user. It is based on asking queries about the user's preferences. Every query costs the elicitation system some price, that is why it is very important to ask wisely and limit the number of queries to compile the ideal probability distribution as accurately as possible with the lowest costs.

Many articles deal with preference elicitation in various ways and use it for different DM problems. For example in [3], PE is solved for partially observable Markov decision processes, which is the most general MDP, where only observation is known and not the specific state. Moreover, in [10, 21] PE is solved for a group recommender system, which tries to elicit a recommendation based on preferences of individual users that satisfy all groups of individual users. In medicine [19] elicits patient preferences in shared decision-making. There are groups of specific situations, in which the PE is used in some way. We focus on observable states and one user's preferences. The terms *user*² and *agent* in this PE task merge.

The main problems are contradicting preferences, duration and complexity of PE. The user may have unreasonable goals, which cannot be achieved under the conditions of this decision-making process. Allow us to give a few examples to make it easier to understand PE.

²The word "user" is used more often when we talk about PE.

Heating example The user wants to have a temperature of 22°C in a given room and they do not want to pay a lot of money for heating. The given room is a system, the states are individual temperatures, the action is to turn up the heater or down and the world is the real surrounding world of the room. If it is cold outside, it is needed to heat more, turn up the heater, until the required room temperature has been reached, and vice versa.

Let it be cold outside. We have 19°C in the room and we want to have 22°C , but it means that we must heat but we do not want to pay a lot of money. This is a contradictory preference. We need to find a balance between the two conditions that will work best for the user. Because if we only consider the condition for the action (pay less) it could happen that we would not heat at all and we would not reach the preferred state 22°C . And if we only consider the condition for the preferred state, we will pay a lot. So we need to also consider a state, where we would not notice such a difference from the preferred state, we would save more money and we would reach both conditions. For example, we cannot feel such a difference when there is 21 or $21,5^{\circ}\text{C}$ in the room versus 22°C and we can save more money for heat only for 21 or $21,5^{\circ}\text{C}$.

It is necessary to find a solution that would optimize both conditions. But every user is different and they can prefer different combinations. They can also prefer 20°C because they would rather get dressed and save money when they find out how much the heating costs. But we do not know that, so we need to ask questions and find out what combination is the best for the user.

Another example could be from a medical environment.

Pandemic example We have a spreading pandemic and we want to save as many lives as possible. We want to restrict the movement of people and prevent the spread of a pandemic. On the other hand, we do not want to affect the economy or education. It is a very complicated decision problem. We also need to find a balance between saving lives and affecting the economy and education. Here again, it is up to the user to which requirement he attaches more importance.

Or from the political environment.

Governing example We have a ruling political party that wants to be re-elected. For example, it can spend money campaigning and getting voters before the election on its side. On the other hand, it must have enough money in the treasury to be able to continue to govern well.

The biggest problems are:

- the user cannot fully quantify their preferences especially in a multi-attribute decision-making task
- the preference elicitation is prone to contradictions preferences
- the user is unwilling to spend too much deliberation effort and time on this hard decision-making sub-task.

All these examples face uncertainty and users have inconsistent preferences. We need to find the balance between the preferences to satisfy all user's wishes³ as much as possible. This can be achieved by additional questions and subsequent fine-tuning of the parameters of the preference description (loss or ideal probability density) so that the user is as satisfied as possible. We need to find a constellation, where no further improvements are possible.

³We use the term preference for more mathematical description and the wish for an informal human expression.

The idea is to reduce the uncertainty about the user’s preferences. The elicitation system has an available set of queries, and each query associates a finite set of possible responses. We can have two states which we compare and the query can be ‘Do you like the first observation more than the other?’, as in [11], then the response is Yes/No. Or we can ask ‘Compare these two observations.’ The responses can be ‘I prefer first/I prefer second/I do not know.’ It can also be done on a finite set of chosen observations, as in [27], and the query can be ‘Choose the best observation.’

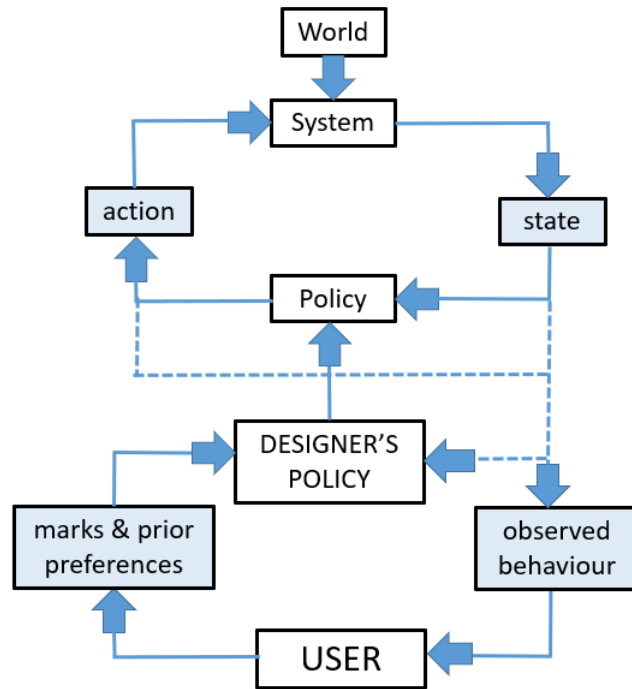


Figure 2: Scheme of the upper-level closed-loop for PE

In this thesis, we focus both on the initial phase of preferences description and then we ask additional questions dynamically during DM. The foreseen choice of interactions with the agent is more dynamic than usual. It shows a finite set of observations every unit of time and asks if the agent likes it. The response is a number from 1 to 5 with the meaning marks at school. So the agent values how much they like the sequence of the observations which we show them. And the process can end at a time when no improvement will be possible and the marks will ideally converge to 1. As said above, the agent can have inconsistent preferences so that the marks may approach 1 but do not need to be 1. We must allow the sequence of observations long enough to make it clear that no improvement is possible. Also, every agent will mark differently even if they will have the same wishes at the beginning. Every agent can be differently demanding or they will understand during the DM process that they formulated their wishes incorrectly. PE is very individual and it depends on the emotions and perception of the agent.

PE has some resemblance to the estimation of the model of the system. We try to learn about user’s preferences by observing their marking and based on marks we select parameters of the ideal probability density. It is a kind of closed-loop as well where parameters (sequences of states and actions of the main DM closed-loop) are states and marks are actions. The scheme is in Figure 2.

It is important to have at least some knowledge of preferences before the beginning of decision-making, finer updating can be made during the process.

In FPD, PE consists of:

- a transformation of user's wishes into a non-empty set of prospective ideal pds
- a choice of the optimal ideal pd within this set that adds as few additional user's preferences as possible.

The solution:

- unambiguously combines multiple attributes, see in [17]
- provides an ambitious, but potentially reachable, goal of the policy design
- suppresses contradictions and decreases the PE-related deliberation burden on the user
- simplifies the PE controlled by queries, see in [5, 7], by decreasing the number of optional meta-parameters

In this thesis, we provide a methodology on how to compile the ideal probability density and solve a decision-making problem. The structure of this thesis can be described as follows. Chapter 1 presents a brief introduction to the mathematical model of MDPs, FPD, and Bayes' learning. In core Chapter 2, we find the ideal pd. Finally, we summarize the knowledge of the algorithm. It is used twice because we have two closed-loops, the main closed-loop and the meta-closed-loop. Then, we talk about PE and the dialogue with the user in Chapter 3 and we find the appropriate tools. This solution is then tested in simulations in Chapter 4. Finally, the conclusion summarizes the work that has been done and suggests future research.

Chapter 1

Preliminaries

1.1 Notions and definitions

There are some notions and notations in the table below.

notion	notation
naturals numbers	\mathbb{N}
a set of elements m	\mathbb{M}
m is an element of a set \mathbb{M}	$m \in \mathbb{M}$
a function	f
a probability of x or a probability density	$p(x)$
cardinality of the set \mathbb{M}	$ \mathbb{M} $
an empty set	\emptyset
definition by equality	\equiv
proportionality	\propto

Table 1.1: Basic notation

- Probability density (pd) is the probability density function for continuous-valued x and the probability for discrete-valued x . The use of integrals for probability density is to be understood as sums in the discrete case.
- A conditional probability density $p(x|y)$ of x under the condition y is related to the joint pd $p(x, y)$ and marginal pd $p(y)$ by the equation $p(x|y) = \frac{p(x, y)}{p(y)}$ for $p(y) > 0$.
- A chain rule for two events x, y is the equation $p(x, y) = p(x|y)p(y)$, which is just another expression of the definition of conditional probability density.
- The argument of the maxima is defined as $\text{Argmax}_{x \in \mathbb{M}} [f(x)] \equiv \{x | x \in \mathbb{M} \wedge \forall y \in \mathbb{M} : f(y) \leq f(x)\}$. The argument of the minima is similar.
- A p -norm $\|f\|_p$ for $p > 1$ of a real-valued function $f(x)$ on \mathbb{X} reads $\|f\|_p \equiv \left[\int_{\mathbb{X}} |f(x)|^p dx \right]^{\frac{1}{p}}$.
- The support of the pd p is defined by $\text{supp}[p] \equiv \{x \in \mathbb{M} : p(x) > 0\}$.

Definition 1.1.1. (Kullback-Leibler divergence, see in [18]) If p, q are two pds on a set of \mathbb{X} , then the similarity of these two pds is measured by the Kullback-Leibler (KL) divergence

$$D(p||q) \equiv \int_{x \in \mathbb{X}} p(x) \ln \left(\frac{p(x)}{q(x)} \right) dx. \quad (1.1)$$

$D \geq 0$ for all p, q and the equality $D(p||q) = 0$ occurs if and only if $p = q$ for almost all $x \in \mathbb{X}$. These properties are proved in [18].

Theorem 1.1.1. (Hölder's inequality, see in [24]) Let \mathbb{X} be a measurable space with Lebesgue measure μ . Let $f, g : \mathbb{X} \rightarrow [0, \infty)$ be measurable functions on \mathbb{X} , $1 < p, q < \infty$. Let $\frac{1}{p} + \frac{1}{q} = 1$. Then, Hölder's inequality holds

$$\int_{\mathbb{X}} |f(x)g(x)| dx \leq \left(\int_{\mathbb{X}} |f(x)|^p dx \right)^{\frac{1}{p}} \left(\int_{\mathbb{X}} |g(x)|^q dx \right)^{\frac{1}{q}} = \|f\|_p \|g\|_q. \quad (1.2)$$

The equality is reached for $|f|^p$ proportional to $|g|^q$ on \mathbb{X} .

1.2 Markov decision process

Definition 1.2.1. A Markov decision process (MDP) is defined by an ordered set of five elements $\{\mathbb{T}, \mathbb{S}, \mathbb{A}, m, l\}$ and policy π , where:

- The element \mathbb{T} stands for a discrete, finite set of decision epochs $\mathbb{T} \equiv \{0, 1, 2, \dots, |\mathbb{T}|\}$, $|\mathbb{T}| \in \mathbb{N}$
- The element \mathbb{S} denotes a finite set of possible observable states of the system. The set would be $\mathbb{S} \equiv \{s^1, s^2, \dots, s^{|\mathbb{S}|}\}$, for $|\mathbb{S}| \in \mathbb{N}$, where states are $s^j \in \mathbb{S}$, for $\forall j \leq |\mathbb{S}| \in \mathbb{N}$ and $\mathbb{S} \subset \mathbb{R}$ is a subset of a space of real numbers \mathbb{R} .
- The element \mathbb{A} denotes a finite set of possible actions $\mathbb{A} \equiv \{a^1, a^2, \dots, a^{|\mathbb{A}|}\}$, for $|\mathbb{A}| \in \mathbb{N}$, where actions are $a^j \in \mathbb{A}$, for $\forall j \leq |\mathbb{A}| \in \mathbb{N}$ where $\mathbb{A} \subset \mathbb{R}$ is a subset of a space of real numbers \mathbb{R} .
- The element m is the transition pd, meaning that $m(s_t|a_t, s_{t-1}) \geq 0$ and $\int_{s_t \in \mathbb{S}} m(s_t|a_t, s_{t-1}) ds_t = 1$, while $m(s_t|a_t, s_{t-1}, v_{t-1}) = m(s_t|a_t, s_{t-1})$ for $\forall s_t, s_{t-1} \in \mathbb{S}$, $\forall a_t \in \mathbb{A}$, $\forall t \in \mathbb{T}$, where $v_{t-1} \equiv (a_{t-1}, s_{t-2}, a_{t-2}, \dots)$ are past observations to time $t - 1$: $a_{t-1}, s_{t-2}, a_{t-2}, \dots, s_1, a_1, s_0$.
- A real valued function $l = l(\tilde{s}, a, s)$ is called loss function, where $\tilde{s}, s \in \mathbb{S}, a \in \mathbb{A}$.
The function assigns a loss to the transition from the state s under the action a to the state \tilde{s} .
- Policy π is a sequence of probability densities $r_t(a|s)$, where actions $a = a_t$ are conditioned by states $s = s_{t-1}$ and $s \in \mathbb{S}, a \in \mathbb{A}, t \in \mathbb{T}$. They are called decision rules. They fulfill $r_t(a|s) \geq 0$ a $\int_{a \in \mathbb{A}} r_t(a|s) da = 1$ for $\forall a \in \mathbb{A}, \forall s \in \mathbb{S}$.

Comment: We would like to mention for clarity that the dimension of the transition pd for the discrete case is $|\mathbb{S}| \times |\mathbb{A}| \times |\mathbb{S}|$ and the dimension of decision rules is $|\mathbb{A}| \times |\mathbb{S}|$.

As it is written in the Introduction, the decision-making problem is defined on the closed-loop formed by the system and the agent. The system is a “cut out” part of the world. The system is described by transition probability density between the states, in which the system can occur. The agent (the decision-maker) observes the state of the system and based on it they choose the action that further influences the transition of the system to the other state. The action is chosen by the agent’s policy. The policy π consists of its sequence of decision rules $r(a|s)$. The decision rule decides, which action will be chosen.

We need to find the decision rule in every step of the decision making, thanks to which the system moves with the highest probability to the state that the agent wants. The best-chosen decision rule we will call optimal decision rule.

The optimal decision rule chooses the action, which moves the system with the lowest loss to the preferred state. The loss function can be interpreted as the energy that has to be released or the price that has to be paid to perform the action and for a deviation from the preferred state. The information about the preferred state is given before the beginning of the decision-making.

The transition pd m is called the model of the system. It expresses the agent’s assumptions about the system. Then, they choose (by the model) the action, which moves the system with the highest probability density to the preferred state. The model has to be given before the beginning of the decision making and then we will improve it by Bayes learning 1.4 during the decision process. In the considered discrete cases, the model of the system is described by transition probability between the states.

The transition probability is given only by the current state and the chosen action $m(\tilde{s}|a, s, v) = m(\tilde{s}|a, s)$. This is the key property of the Markov decision process.

So before the decision making we have to know

- the set of possibly observed states \mathbb{S}
- the set of admissible actions \mathbb{A}
- the model of the system $m(\tilde{s}|a, s)$
- the loss $l(\tilde{s}, a, s)$ quantifying the information about preferences, typically characterized by a set of preferred states \mathbb{S}^i and a set of preferred actions \mathbb{A}^i

Then the decision making is just an optimization problem and we are looking for optimal policy. We have to

- find a sequence of decision rules that minimizes the expected loss
- measure the last observation (state) and generate the action according to the optimal decision rule

1.3 Decision making via fully probabilistic design

As it is written above, we try to solve the decision problem. We have already described the mathematical framework that is the most often used for solving the DM problems in Section 1.2. It refers to the agent associated with its uncertain system into the closed-loop. The standard Markov decision process solution is based on the minimization of the expected values of the loss function (or the maximization of the expected utility). It is the basis for the majority of the systematic solutions for decision-making tasks. However, it has various limitations (e.g. computational complexity or difficulty of combining partial preferences). Therefore, an alternative was sought and a fully probabilistic design (FPD), can be found in [14], was proposed.

FPD is less computationally demanding in terms of this thesis and its main advantage is that it introduces the so-called ideal probability density of behaviour, that reflects the agent’s preferences and to

which the real probability density of behaviour tries to get closer. It means that the agent says the preferences and based on that the ideal probability of behaviour (defined below) is compiled. The probability of behaviour consists of the transition pd and the decision rules, as in the Markov decision process, from which it was derived. Then, the real pd of the behaviour and the ideal pd tries to get as close as possible by selecting an appropriate policy. This proposal is based on the fact that the minimization of the loss function can be understood as an attempt to the influence probability density of the closed-loop variables.

We have outlined where the fully probabilistic design comes from and what it has in common with the Markov decision process, and now we move on to the more mathematical definitions of the design.

So first the agent observes the state of the system in every decision epoch and according to it they choose the action $a_t \in \mathbb{A} \neq \emptyset$ at every discrete time $t \in \mathbb{T}$, which influences the closed-loop by stimulating transitions of the system from the state $s_{t-1} \in \mathbb{S}$ to state $s_t \in \mathbb{S}$. A collection of actions and states to the horizon $T \equiv |\mathbb{T}|$ describing the behaviour of the closed-loop (agent-system) is

$$b \equiv (s_0, a_1, s_1, a_2, s_2, \dots, a_T, s_T) \in \mathbb{B}. \quad (1.3)$$

The choice of the actions $a_t, t \in \mathbb{T}$ is made by the decision policy of the agent $\pi \in \Pi$ consisting of the sequence of the decision rules

$$\pi \equiv (r(a_1|s_0), r(a_2|s_1), \dots, r(a_T|s_{T-1})). \quad (1.4)$$

As it can be seen in the definition of the decision rule $r(a_t|s_{t-1})$, the decision rule chooses the next action based on the state, in which the system is situated at the moment.

The behaviour of closed-loop is fully described by the probability density $c^\pi(b)$ depending on the policy π . The probability density can be written by the chain rule as

$$c^\pi(b) = \prod_{t \in \mathbb{T}} m(s_t|a_t, s_{t-1})r(a_t|s_{t-1}),^1 \quad (1.5)$$

where probability density $m(s_t|a_t, s_{t-1})$ describes transitions of the states of the system. It is the model of the system. It stores the agent's assumptions about the transitions of the system's states. It can be known or it can be estimated and improved during the decision process according to the observed states and chosen actions. This will be more explained in Section 1.4. The sequence of the probability densities is briefly denoted

$$m \equiv (m(s_1|a_1, s_0), m(s_2|a_2, s_1), \dots, m(s_T|a_T, s_{T-1})). \quad (1.6)$$

This defines pd modelling the real behaviour of the agent. Now let us focus on the ideal behaviour, which gives the FPD peculiarity and "simplicity" and it is the main property of FPD. As it was said before FPD uses ideal probability density, which reflects the agent's preferences. The ideal pd gives high values of pds to preferred states and actions and low values to undesirable states and actions. The ideal model of the system m^i should have high values of transition pd to the preferred state. If we denote the preferred state as s^j , the ideal model should fulfil follows.

$$m^i(s^i|a_t, s_{t-1}) \geq m^i(s^j|a_t, s_{t-1}), \quad (1.7)$$

where $s^i \neq s^j \in \mathbb{S}$ for $\forall j \leq n \in \mathbb{N}$, where n indicates the number of all states.

The FPD tries to get the real probability density (1.5) as close as possible to the ideal probability density written as follows

$$c^i(b) = \prod_{t \in \mathbb{T}} m^i(s_t|a_t, s_{t-1})r^i(a_t|s_{t-1}), \quad (1.8)$$

¹The initial state s_0 is given before the beginning of the decision-making and we take it as deterministic. The initial state also cannot be influenced by the agent.

where $m^i(s_t|a_t, s_{t-1})$ is an ideal model of the system representing the ideal dynamic of the system (transition probability density between states) and $r^i(a_t|s_{t-1})$ is an ideal decision rule. Both of these factors of the ideal probability density of behaviour give high values to the preferred ones. So the ideal model of the system gives high probability values to transitions to the preferred states and the ideal decision rule gives high probability values to the preferred actions. The ideal probability density of behaviour reflects the agent's preferences and has the same factors as the real probability density (1.5). We try to bring each real factor as close as possible to the ideal factors. How to find the ideal factors is described in Chapter 2.

The optimal policy is found as the argument of the minimum of the Kullback-Leibler divergence, defined in Definition 1.1.1, of the real probability density to the ideal one. The ideal pd expresses the agent's preferences. So it has to be given before the decision making. But often it is not given unambiguously, this case is addressed in detail in the next chapter.

The two models of the closed-loop $c^\pi(b)$, $c^i(b)$ can be brought near by minimization of the Kullback-Leibler divergence. The optimal decision policy, can be found in [14], is given by

$$\pi^o \in \text{Arg min}_{\pi \in \Pi} D(c^\pi || c^i) = \text{Arg min}_{\pi \in \Pi} \int_{b \in \mathbb{B}} c^\pi(b) \ln \left(\frac{c^\pi(b)}{c^i(b)} \right) db. \quad (1.9)$$

Theorem 1.3.1. (FPD, see in [25]) Decision rules, which constitute the optimal decision policy π^o , are computed as follows

$$r^o(a_t|s_{t-1}) \equiv r^i(a_t|s_{t-1}) \frac{\exp[-d(a_t, s_{t-1})]}{h(s_{t-1})}, \quad (1.10)$$

where

$$\begin{aligned} d(a_t, s_{t-1}) &\equiv \int_{s_t \in \mathbb{S}} m(s_t|a_t, s_{t-1}) \ln \left[\frac{m(s_t|a_t, s_{t-1})}{h(s_t)m^i(s_t|a_t, s_{t-1})} \right] ds_t \\ h(s_{t-1}) &\equiv \int_{a_t \in \mathbb{A}} r^i(a_t|s_{t-1}) \exp[-d(a_t, s_{t-1})] da_t, \end{aligned} \quad (1.11)$$

$t = T, T - 1, \dots, 1$ and $h(s_t) \in [0, 1]$, $h(s_T) \equiv 1$.

Backwards recursion starts in $h(s_T) = 1 \geq h(s_t)$, $\forall t \in \mathbb{T}$. The achieved minimum is

$$\min_{\pi \in \Pi} D(c^\pi || c^i) = -\ln(h(s_0)). \quad (1.12)$$

The proof can be found in [25].

Remark The $-\ln(h(s_0))$ is the FPD version of value function in MDP, defined in [23].

1.4 Bayesian learning of Markov Chain

The FPD presupposes the knowledge of the model of the system for finding the optimal policy. But if we do not have enough knowledge about the model of the system we have to estimate it. It can be guessed at first subjectively and the initial ideas can be improved by Bayes' learning, can be found in [22], during the decision making.

So we have a parametric model of the system $m(s_t|a_t, K_{t-1}, \theta)$, where $K_{t-1} \equiv (s_{t-1}, a_{t-1}, \dots, s_1, a_1, s_0)$ are the past observation up to time $t - 1$.

Theorem 1.4.1. (Bayesian Learning, see in [13], [22]) Let θ be unknown to the designed decision rules. This postulates independence of a_t and θ when conditioned by the knowledge K_{t-1} (cf. natural conditions of control)

$$r(a_t|K_{t-1}) = f(a_t|\theta, K_{t-1}) \iff r(\theta|a_t, K_{t-1}) = f(\theta|K_{t-1}). \quad (1.13)$$

The parametric model $m(s_t|a_t, K_{t-1}, \theta)$ with this unknown² $\theta \in \Theta$ is given. Then, the predictive pd reads

$$m(s_t|a_t, K_{t-1}) = \int_{\Theta} m(s_t|a_t, K_{t-1}, \theta) p(\theta|K_{t-1}) d\theta, \quad (1.14)$$

It uses the Bayesian parameter estimation, which evolves posterior pd $p(\theta|K_{t-1})$, $t > 1$. It is the sufficient statistic for constructing parameter estimators. The data updating evolves $p(\theta|K_{t-1})$, independently of $r(a_t|K_{t-1})$,

$$p(\theta|K_t) = \frac{m(s_t|a_t, K_{t-1}, \theta) p(\theta|K_{t-1})}{m(s_t|a_t, K_{t-1})} \propto m(s_t|a_t, K_{t-1}, \theta) p(\theta|K_{t-1}). \quad (1.15)$$

The recursion is initiated by the prior pd $p(\theta) \equiv p(\theta|a_1, K_0) = p(\theta|K_0)$.

The assumed parametric model belongs to the exponential family, i.e. it has the form

$$m(s_t|a_t, K_{t-1}, \theta) = m(\Psi_t, \theta) = A(\theta) \exp\langle B(\Psi_t), C(\theta) \rangle, \quad (1.16)$$

where Ψ_t is data vector combining s_t and a regression vector $\psi_t = \psi(a_t, K_{t-1})$. $A(\cdot)$ is a non-negative scalar function defined on Θ , $B(\cdot)$, $C(\cdot)$ are functions of $\langle \cdot, \cdot \rangle$ -compatible and finite dimensions on sets of data vectors and parameter. The symbol $\langle \cdot, \cdot \rangle$ is the scalar-valued function, linear in the first argument.

We assume the Markov property $\psi_t \equiv (a_t, s_{t-1})$ and the finite amount of realizations of data vector. Then, the most general parametric model is Markov chain with unknown transition pds, described via Kronecker's delta

$$m(s_t|\psi_t, \Theta) = \prod_{\Psi=(s,\psi)} \theta_{s|\psi}^{\delta(\Psi, \Psi_t)} = \exp\left[\sum_{\Psi=(s,\psi)} \underbrace{\delta(\Psi, \Psi_t)}_{B_{s|\psi}(\Psi)} \underbrace{\ln(\theta_{s|\psi})}_{C_{s|\psi}(\Theta)} \right]$$

$$\theta \in \Theta \equiv \{ \theta_{s|\psi} > 0, \sum_{s \in \mathbb{S}} \theta_{s|\psi} = 1 \}. \quad (1.17)$$

It belongs to the exponential family and has Dirichlet pd as the conjugate prior

$$p(\theta) = Di_{\theta}(V_0) = \prod_{\psi \in \Psi^*} \frac{\prod_{s \in \mathbb{S}} \theta_{s|\psi}^{V_0(s|\psi)-1}}{Be(V_0(\cdot|\psi))} \quad (1.18)$$

$$Be(V(\cdot|\psi)) \equiv \frac{\prod_{s \in \mathbb{S}} \Gamma(V(s|\psi))}{\Gamma(\sum_{s \in \mathbb{S}} V(s|\psi))},$$

where $\Gamma(\cdot)$ is gamma function. It can be re-written as follows

$$p(\theta|K_0) \propto \prod_{\Psi} \theta_{s|\psi}^{V_0(s|\psi)-1} = \exp \left[\sum_{\Psi=(s,\psi)} V_0(s|\psi) \ln(\theta_{s|\psi}) \right] = p(\theta|V_0). \quad (1.19)$$

Obviously, then the Bayes rule (1.15) preserves the shape and

$$p(\theta|K_t) = p(\theta|V_t) \quad (1.20)$$

for $V_t(s|\psi) = V_{t-1}(s|\psi) + \delta(\Psi, \Psi_t)$, where $\Psi_t = (s_t, \psi_t) = (s_t, a_t, s_{t-1})$ is made of the observed data.

This provides the needed learning.

²Lack of fonts forced us to violate our agreement on the set notation

Theorem 1.4.2. (Bayesian Learning of Markov Chain)

Let the initial value V_0 of the statistic V be given, $V_0 > 0$, and let s_t, a_t, s_{t-1} be specific measured values at the time $t \in \mathbb{T}$. Then the value of the statistic V_t updates as

$$V_t(\tilde{s}|a, s) = \begin{cases} V_{t-1}(\tilde{s}|a, s) & (\tilde{s}, a, s) \neq (s_t, a_t, s_{t-1}) \\ V_{t-1}(\tilde{s}|a, s) + 1 & (\tilde{s}, a, s) = (s_t, a_t, s_{t-1}) \end{cases} \quad (1.21)$$

where $s_t, s_{t-1}, \tilde{s}, s \in \mathbb{S}, a, a_t \in \mathbb{A}$.

The model of the system is the predictive pd

$$m(\tilde{s}|a, s, V = V_{t-1}) = \frac{V(\tilde{s}|a, s)}{\sum_{\tilde{s} \in \mathbb{S}} V(\tilde{s}|a, s)}. \quad (1.22)$$

Its used certainly-equivalent approximation fixes V_{t-1} during the optimization of decision rules, i.e. sets $m(\tilde{s}|a, s) \approx m(\tilde{s}|a, s, V)$.

Chapter 2

Generic and Specific PE in FPD

This chapter is based on the research from the article [16]. Its theorems are presented with proofs as [16] is yet unpublished. We are going to define the equations of how to assemble the components of the ideal distribution. At first, we will talk about the ideal distribution in general and then we will define equations first for the ideal model of the system and then for the ideal decision rule.

The ideal pd \mathbf{c}^i (1.8) expresses the agent's preferences. PE within FPD is to find the optimal ideal pd \mathbf{c}^{io} . However, the specification of preferences is not always complete due to misrepresentation to the agent of preference. The generic incomplete description of the the agent's preferences partially characterizes the suitable ideal joint pd. The incompleteness of the description implies that the set

$$\mathbb{C}^i \equiv \{\text{ideal pds } \mathbf{c}^i(b), b \in \mathbb{B}, \text{ respecting agent's preferences}\} \quad (2.1)$$

includes more pds. On the other hand it can be also empty because of the agent's inconsistencies. The agent's preferences can be in contradiction or the agent can have unachievable goals.

The PE then consists of the choice of:

- the non-empty set \mathbb{C}^i that overcomes inconsistencies of agent's preferences
- the optimal closed-loop ideal pd \mathbf{c}^{io} from the set (2.1).

The PE principle from [15] recommends to choose as the optimal ideal closed-loop pd

$$\mathbf{c}^{io} \in \text{Arg min}_{\mathbf{c}^i \in \mathbb{C}^i} \min_{\pi \in \Pi} D(\mathbf{c}^\pi \| \mathbf{c}^i). \quad (2.2)$$

Its use in FPD ensures that no additional preferences are added to those expressed by the agent. Then, cf. (1.11), (1.12), (2.2), the optimal closed-loop ideal pd \mathbf{c}^{io} reads

$$\mathbf{c}^{io} \in \text{Arg min}_{\mathbf{c}^i \in \mathbb{C}^i} \min_{\pi \in \Pi} D(\mathbf{c}^\pi \| \mathbf{c}^i) \stackrel{(1.12)}{=} \text{Arg min}_{\mathbf{c}^i \in \mathbb{C}^i} (-\ln(h(s_0))) = \text{Arg min}_{\mathbf{c}^i \in \mathbb{C}^i} \ln\left(\frac{1}{h(s_0)}\right) \quad (2.3)$$

If $\ln\left(\frac{1}{h(s_0)}\right)$ should be minimal and $\ln()$ is increasing function then the argument must be also minimal. And the fraction $\frac{1}{h(s_0)}$ is minimal if $h(s_0)$ is maximal. Then the optimization can be written as follows

$$\text{Arg min}_{\mathbf{c}^i \in \mathbb{C}^i} \ln\left(\frac{1}{h(s_0)}\right) = \text{Arg min}_{\mathbf{c}^i \in \mathbb{C}^i} \frac{1}{h(s_0)} = \text{Arg max}_{\mathbf{c}^i \in \mathbb{C}^i} h(s_0) = \text{Arg max}_{\mathbf{c}^i \in \mathbb{C}^i} \int_{a_1 \in \mathbb{A}} r^i(a_1 | s_0) \exp[-d(a_1, s_0)] da_1. \quad (2.4)$$

In summary, the overall result is as follows

$$\begin{aligned} \mathbf{c}^{io} \equiv m^{io} r^{io} \in \text{Arg max} & \left[\max_{r^i \in \mathbb{R}^i} \int_{a_1 \in \mathbb{A}} r^i(a_1 | s_0) \exp[-\mathbf{d}(a_1, s_0)] da \right] \\ \mathbf{d}(a_1, s_0) = \int_{s_1 \in \mathbb{S}} & m(s_1 | a_1, s_0) \ln \left(\frac{m(s_1 | a_1, s_0)}{h(s_0) m^i(s_1 | a_1, s_0)} \right) ds_1, \end{aligned} \quad (2.5)$$

$h(s_0)$ comes from the previous backward recursion via step (1.11).

The minimization over a \mathbf{c}^i -factor ($\mathbf{c}^i(s_t | a_t, s_{t-1}) = m^i(s_t | a_t, s_{t-1}) r(a_t | s_{t-1})$) in any decision epoch $t \in \mathbb{T}$ and for any realized state s_{t-1} are formally identical. Therefore, we can suppress t and $s_{t-1} \in \mathbb{S}$ and deal with $m(s|a) \equiv m(s_t = s | a_t = a, s_{t-1})$, $m^i(s|a) \equiv m^i(s_t = s | a_t = a, s_{t-1})$, $r(a) \equiv r(a_t = a | s_{t-1})$, $r^i(a) \equiv r^i(a_t = a | s_{t-1})$ and $h(s) = h(s_t = s)$.

The optimization (2.5) considers the given $h(s)$ and runs over \mathbb{M}^i (a set of m^i -s) of \mathbb{C}^i determined by a given r^i and over the set \mathbb{R}^i (a set of r^i -s) for which $\mathbf{c}^i = m^i r^i$ -factors are in the set

$$\{\mathbf{c}^i(s, a) : \mathbf{c}^i(s, a) = m^i(s|a) r^i(a), s \in \mathbb{S}, a \in \mathbb{A}, \text{respecting agent's preferences}\}. \quad (2.6)$$

We first perform the optimization (2.5) for a quite general choice of sets $\mathbb{M}^i, \mathbb{R}^i$ (Sections 2.1, 2.2). Then we specialize it to a specific but still general case.

2.1 The generic choice of optimal ideal model of the system

Theorem 2.1.1. (Optimal m^{io} -Factor) Let $r^i \in \mathbb{R}^i$ be a fixed ideal decision rule, which defines a non-empty cross-empty $\mathbb{M}^i \equiv \{m^i : m^i r^i \in \text{set (2.6)}\}$. Let $m^i(s|a) \in \mathbb{M}^i$ exist such that $\mathbf{d}(a) < \infty, \forall a \in \mathbb{A}$ (1.11). Then, the optimal ideal m^{io} -factor minimises $\mathbf{d}(a), s \in \mathbb{S}, a \in \mathbb{A}$, i.e.

$$m^{io}(s|a) \in \text{Arg max}_{m^i \in \mathbb{M}^i} \int_{\mathbb{A}} r^i(a) \exp[-\mathbf{d}(a)] da = \text{Arg min}_{m^i \in \mathbb{M}^i} \mathbf{d}(a). \quad (2.7)$$

Proof We first start with a simple consideration and then we will formally prove it. For the fixed r^i the integral is maximal if the $\exp[-\mathbf{d}(a)]$ is maximal. And because e^{-x} is decreasing function of $x \geq 0$, the maximum of $\exp[-\mathbf{d}(a)]$ is reached by the minimal $\mathbf{d}(a)$.

And formally, for $\mathbb{M}^i \neq \emptyset$ and any $a \in \mathbb{A}$, a minimiser $m^{i*} \in \mathbb{M}^i$ of $\mathbf{d}(a) \geq 0$ exists giving the value $\mathbf{d}^*(a) \leq \mathbf{d}(a)$, where $\mathbf{d}(a)$ is the value obtained for an arbitrary $m^i \in \mathbb{M}^i$ and the same h see (1.11). This implies that $\mathbf{d}^*(a) < \infty$ and $\exp(-\mathbf{d}^*(a)) \geq \exp(-\mathbf{d}(a))$. Multiplication of this inequality by $r^i(a) \geq 0$ and the integration over the set \mathbb{A} implies that $m^{io} = m^{i*}$. \square

2.2 The generic choice of optimal ideal decision rule

The decision rule decides, which action will be chosen. So the decision rules must work on the set of admissible actions. Then, the support $\text{supp}[r]$ of an admissible r -factor should be included in the set of possible actions \mathbb{A} . The form of the FPD-optimal r^o -factor of the optimal ideal pd \mathbf{c}^{io} , Theorem 1.3.1 implies that $\text{supp}[r^o] \subseteq \text{supp}[r^i]$. Therefore, only the ideal r^i -factors

$$r^i \in \mathbb{R}^i \equiv \left\{ r^i : \text{supp}[r^i] = \mathbb{A} \right\} \quad (2.8)$$

keep actions $a \in \mathbb{A}$ and exclude none.

Consequently, (2.8) is the generic constraint on the set $\mathbb{R}^i \equiv \{r^i : m^{io} r^i \in (2.6) \text{ while } m^{io} \text{ is given by (2.7)}\}$.

Theorem 2.2.1. (Optimal r^{io} -Factor Meeting (2.8)) Let assumptions of Theorem 2.1.1 hold and for a scalar $p > 1$

$$\mathbb{R}^i \equiv \left\{ r^i : \text{supp}[r^i] = \mathbb{A} \text{ and } \|r^i\|_p \equiv \left[\int_{\mathbb{A}} (r^i(a))^p da \right]^{1/p} < \infty \right\} \quad (2.9)$$

while

$$|\mathbb{A}| \equiv \int_{\mathbb{A}} da < \infty. \quad (2.10)$$

Then, the optimal ideal r^{io} -factor reads, cf. (1.11), (2.7)

$$\begin{aligned} r^{io} &\propto \chi_{\mathbb{A}}(a) \exp[-\nu d^o(a)], \quad \nu \equiv \frac{1}{p-1}, \\ d^o(a) &\equiv \int_{\mathbb{S}} m(s|a) \ln \left(\frac{m(s|a)}{h(s)m^{io}(s|a)} \right) ds \stackrel{(2.7)}{\leq} d(a) \end{aligned} \quad (2.11)$$

where $\chi_{\mathbb{A}}(a)$ is the indicator function of the set \mathbb{A} . The r^{io} -factor (2.11) belongs to (2.9) and meets (2.8).

Proof The non-negativity of $d(a)$ implies that $\exp(-d(a)) \in [0, 1]$, which with (2.10) provides that $\|\exp(-d^o)\|_q < \infty$ on \mathbb{A} for $q \equiv \frac{p}{p-1} = p\nu$. We are looking for the optimal ideal decision rule which will fulfil the equation

$$r^{io}(a) \in \text{Arg max}_{r^i \in \mathbb{R}^i} \int_{\mathbb{A}} r^i(a) \exp[-d^o(a)] da.$$

At first, we will bound the integral using of Hölder's inequality defined in Theorem 1.1.1,

$$\begin{aligned} \int_{\mathbb{A}} r^i(a) \exp[-d^o(a)] da &= \int_{\mathbb{A}} |r^i(a) \exp[-d^o(a)]| da \leq \left(\int_{\mathbb{A}} |r^i(a)|^p da \right)^{\frac{1}{p}} \left(\int_{\mathbb{A}} |\exp[-d^o(a)]|^q da \right)^{\frac{1}{q}} = \\ &= \|r^i(a)\|_p \|\exp[-d^o(a)]\|_q, \end{aligned} \quad (2.12)$$

We talk about pd so all of these functions are non-negative, i.e. we do not need absolute values. To have both sides of (2.12) equal, we need the factors integrated on the left-hand side linearly dependent. So it can be seen that

$$\begin{aligned} (r^{io}(a))^p &\propto (\exp[-d^o(a)])^q \\ r^{io}(a) &\propto (\exp[-d^o(a)])^{\frac{q}{p}} \\ r^{io}(a) &\propto \exp\left[-\frac{q}{p}d^o(a)\right] = \exp[-\nu d^o(a)]. \end{aligned} \quad (2.13)$$

The finiteness of $d^o(a) < \infty$ on \mathbb{A} is guaranteed by assumptions of Theorem 2.1.1. This makes r^{io} (2.13) positive on \mathbb{A} therefore (2.8) is met. \square

Remarks

- The generic constraint (2.8) implies that the ideal r^i -factors support exploration, which makes Bayesian learning efficient. It is well seen on Section 1.4: if some action $a \in \mathbb{A}$ is unused the corresponding $V(\tilde{s}, a, s)$ does not evolve. No action from \mathbb{A} is a priori forbidden.
- The value function $-\ln(h(s))$ from Theorem 1.3.1 influences the r^{io} -factor (2.11) via $d^o(a)$ but not the m^{io} -factor.

2.3 The specific choice of \mathbb{M}^i making $\mathbb{C}^i \neq \emptyset$

This section makes the generic solutions of Sections 2.1, 2.2 more specific and guarantees that $\mathbb{C}^i = \emptyset$. The optimal ideal r^{io} -factor is uniquely given by choice of m^{io} (and by the opted ν) via (2.11). The description of the agent's preferences is thus reduced to those given by a non-empty set \mathbb{M}^i . This means that a wide range of practical cases will be covered with a few additional PE-oriented queries. Our specific case concerns the following general wish of the agent.

The agent wants to reach given sets of ideal states \mathbb{S}^i and actions \mathbb{A}^i , $\emptyset \neq \mathbb{S}^i \subset \mathbb{S}$, $\emptyset \neq \mathbb{A}^i \subseteq \mathbb{A}$. (2.14)

This requirement is quantified as the preference to assign the highest probability to the set of ideal states \mathbb{S}^i and actions \mathbb{A}^i (2.14) by closing the loop of the given model of the system m and of the optimal ideal decision rule r^{io} . So we define the optimized functional by the equation

$$\int_{\mathbb{A}} \rho(a) r^{io}(a) da \equiv \int_{\mathbb{A}} \left[\int_{\mathbb{S}} \chi_{\mathbb{S}^i}(s) m(s|a) ds + w \chi_{\mathbb{A}^i}(a) \right] r^{io}(a) da \text{ and we want it as large as possible.} \quad (2.15)$$

The introduced weight $w \in \mathbb{W} \equiv [0, \infty)$ parametrize how much the agent prefers to stay in \mathbb{A}^i relative to being in \mathbb{S}^i .

The inspected problem has a meaningful solution if

$$\rho(a) = \int_{\mathbb{S}} \chi_{\mathbb{S}^i}(s) m(s|a) ds + w \chi_{\mathbb{A}^i}(a) > 0, \text{ on } \mathbb{A}. \quad (2.16)$$

If the functional (2.15) is large, then the probability of the preferred sets is also large. The part $\int_{\mathbb{S}} \chi_{\mathbb{S}^i}(s) m(s|a) ds$ forces the model of the system to have the highest probability of the set \mathbb{S}^i . And the part $w \chi_{\mathbb{A}^i}(a) r^{io}(a)$ should guarantee that the ideal decision rule will choose the actions from the set \mathbb{A}^i relatively often. The weight w balances these probabilities.

Remarks

- The weight is here fixed at $0 \leq w < \infty$. Its fine-tuning is to be made by PE controlled by additional queries. However, the weight cannot be too high, because then only the set of actions \mathbb{A}^i would be preferred and staying in \mathbb{S}^i would be neglected. Therefore, we will examine the weight $w \in [0, 1]$.
- The function determining $\rho(a)$ qualitatively plays the role of the reward of MDP. Our construction of the optimal ideal pd c^{io} quantifies the agent's preferences in an ambitious but realistic way.

Maximization of (2.15) with r^{io} given by (2.11) is complicated and it will be addressed in a few steps.

Theorem 2.3.1. (Optimal Value of d^o) Under assumptions of Theorem 2.2.1, covering those of Theorem 2.1.1, and under (2.16), the optimal ideal model m^{io} fulfilling (2.15) determines $d^o(a)$, giving $r^{io} = r^i(m^{io})$ (2.11), $a \in \mathbb{A}$, as a function meeting the equation

$$d^o(a) = d^o(\bar{a}) + \ln \left(\frac{\rho(\bar{a})}{\rho(a)} \right) \equiv d^o(\bar{a}) + \ln \left(\frac{\max_{a \in \mathbb{A}}(\rho(a))}{\rho(a)} \right), \quad \bar{a} \in \text{Arg max}_{a \in \mathbb{A}}(\rho(a)) \equiv \bar{\mathbb{A}}. \quad (2.17)$$

Proof By construction $\|r^{io}\|_p < \infty$ and the finite volume of \mathbb{A} (2.10) implies $\|\rho\|_q < \infty$ for $q = \frac{p}{p-1} = pv$. Hölder's inequality, Theorem 1.1.1, applied to (2.15)

$$\int_{\mathbb{A}} |\rho(a) r^{io}(a)| da \leq \left(\int_{\mathbb{A}} |(r^{io}(a))|^p da \right)^{\frac{1}{p}} \left(\int_{\mathbb{A}} |\rho(a)|^q da \right)^{\frac{1}{q}} \quad (2.18)$$

implies that the inequality becomes equality on \mathbb{A} if the arguments are linearly dependent. For $\nu = \frac{q}{p}$

$$(r^{io}(a))^p = \kappa^p (\rho(a))^q$$

$$r^{io}(a) = \kappa \rho^{\frac{q}{p}}(a) = \kappa \rho^\nu(a) \stackrel{(2.11)}{=} \frac{\exp[-\nu d^o(a)]}{\int_{\mathbb{A}} \exp[-\nu d^o(a)] da}, \quad a \in \mathbb{A}. \quad (2.19)$$

The factor κ is determined by the normalization requirement

$$\kappa = \frac{1}{\int_{\mathbb{A}} \rho^\nu(a) da} > 0 \quad (2.20)$$

due to the finite volume of the set of actions. Also, the normalizing factor

$$J(d^o) \equiv \int_{\mathbb{A}} \exp(-\nu d^o(a)) da \in (0, \infty) \quad (2.21)$$

due to the finiteness of d^o and the finite volume of the action set. The logarithmic version of equation (2.19)

$$\begin{aligned} \ln(\rho^\nu(a)) &= \ln\left(\frac{\exp[-\nu d^o(a)]}{\kappa J(d^o)}\right) \implies \ln(\rho^\nu(a)) + \ln(\kappa J(d^o)) = -\nu d^o(a) \\ d^o(a) &= -\frac{1}{\nu} \ln(\kappa J(d^o)) - \ln(\rho(a)) \equiv \Phi - \ln(\rho(a)), \quad a \in \mathbb{A}. \end{aligned} \quad (2.22)$$

The scalar value Φ does not depend on a specific action value. Thus, it has to meet (2.22) for any fixed \bar{a} . The choice we made is

$$\Phi = d^o(\bar{a}) + \ln\left(\max_{a \in \mathbb{A}}(\rho(a))\right), \quad \bar{a} \in \text{Arg max}_{a \in \mathbb{A}}(\rho(a)) \Leftrightarrow \bar{a} \in \text{Arg min}_{a \in \mathbb{A}} d^o(a), \quad (2.23)$$

which gives (2.17). □

2.3.1 The specific choice of m^i

Theorem 2.3.2 (Solvability of (2.17)). Under (2.16) and $|\mathbb{A}| < \infty$, the smallest $d^o(\bar{a})$ exists such that (2.17) has a solution $m^{io}(s|a)$, $s \in \mathbb{S}$, $\forall a \in \mathbb{A}$ (2.23).

Proof Properties of the KLD conditioned by $a \in \mathbb{A}$ (and implicitly on s_{t-1}) imply that the values $d^o(a) \in [-\int_{\mathbb{S}} m(s|a) \ln(h(s)) ds, \infty] \subset [0, \infty]$. Indeed, the option $m^{io}(s|a) \equiv m(s|a)$ attains the lower bound. The upper bound is reached for $m^{io}(s|a)$ singular with respect to $m(s|a)$, i.e. being zero on a subset of \mathbb{S} to which $m(s|a)$ assigns a positive probability. Thus, the smallest $d^o(\bar{a})$ guaranteeing solvability of (2.17) $\forall a \in \mathbb{A}$ is

$$0 \leq d^o(\bar{a}) \leq \max_{a \in \mathbb{A}} \int_{\mathbb{S}} m(s|a) \ln\left[\frac{\rho(a)}{\rho(\bar{a})h(s)}\right] ds. \quad (2.24)$$

Because it applies

$$\max_{a \in \mathbb{A}} \int_{\mathbb{S}} m(s|a) \ln\left[\frac{\rho(a)}{\rho(\bar{a})h(s)}\right] ds \geq \int_{\mathbb{S}} m(s|\bar{a}) \ln\left[\frac{\rho(\bar{a})}{\rho(\bar{a})h(s)}\right] ds = \int_{\mathbb{S}} m(s|\bar{a}) \ln\left[\frac{1}{h(s)}\right] ds \quad (2.25)$$

for $|\mathbb{A}| < \infty$, $h(s) \in (0, 1]$ and $\rho(a) > 0$. The maximum in (2.24) is finite and the range of $d^o(\bar{a})$ implies existence of $m^{io}(s|\bar{a})$ with $d^o(\bar{a})$ (2.24). \square

The ideal m^{io} gives $d^o(a)$ (1.11) and $r^{io}(m^{io})$ via (2.7). The next proposition provides it for generic pds $m(s|a)$. It requires to find m^{io} giving d^o (2.17) on \mathbb{A} .

Theorem 2.3.3. (m^{io} Meeting (2.15), Generic $m(s|a)$) Let $m(s|a)$, for some $a \in \mathbb{A}$, be non-uniform on \mathbb{S} and Theorem (2.2.1) hold. Then, the m^{io} -factor meeting (2.15) has the form

$$m^i(s|a) = \frac{m(s|a) \exp(-e(a)m(s|a))}{\int_{\mathbb{S}} m(s|a) \exp(-e(a)m(s|a)) ds}. \quad (2.26)$$

It is well defined at least for

$$|\mathbb{S}| \equiv \int_{\mathbb{S}} ds < \infty. \quad (2.27)$$

The real valued $e(a)$ in (2.26) is the existing solution of $L(e(a)) = R(a)$. For $d^o(\bar{a})$ meeting (2.24) with $\bar{a} \in \text{Arg max}_{a \in \mathbb{A}}$, the left- and right-hand sides of this equation are

$$\begin{aligned} L(e(a)) &\equiv e(a)\Lambda(a) + \ln\left(\int_{\mathbb{S}} m(s|a) \exp[-e(a)m(s|a)] ds\right), \\ \Lambda(a) &\equiv \int_{\mathbb{S}} m^2(s|a) ds > 0 \\ R(a) &\equiv -\int_{\mathbb{S}} m(s|a) \ln\left(\frac{m(s|a)}{h(s)}\right) ds + d^o(\bar{a}) + \ln\left(\frac{\rho(\bar{a})}{\rho(a)}\right), \quad \bar{a} \in \text{Arg max}_{a \in \mathbb{A}} \rho(a) \equiv \bar{\mathbb{A}}. \end{aligned} \quad (2.28)$$

Proof Substituting (1.11) into the equation (2.17) we get

$$d^o(\bar{a}) + \ln\left(\frac{\max_{a \in \mathbb{A}}(\rho(a))}{\rho(a)}\right) = \int_{\mathbb{S}} m(s|a) \ln\left[\frac{m(s|a)}{h(s)m^i(s|a)}\right] ds \quad (2.29)$$

$$-\int_{\mathbb{S}} m(s|a) \ln(m^i(s|a)) ds = \int_{\mathbb{S}} m(s|a) \ln\left(\frac{h(s)}{m(s|a)}\right) ds + d^o(\bar{a}) + \ln\left(\frac{\max_{a \in \mathbb{A}}(\rho(a))}{\rho(a)}\right). \quad (2.30)$$

For a fixed $a \in \mathbb{A}$ and non-uniform $m(s|a)$, the equation (2.30) is integral equation for an unknown function $-\ln(m^i(s|a))$. We want to find its particular solution $-\ln(m^i(s|a)) = e(a)m(s|a) + v(a)$ with the optional scalar-valued $e(a)$ and $v(a)$. Formally, $-\ln(m^i(s|a))$ has a constituent $o(s|a)$ orthogonal to $m(s|a)$, i.e. meeting $\int_{\mathbb{S}} m(s|a)o(s|a) ds = 0$. This part has no influence on the $d^o(a)$ value so we did not need to use it in generic case. The inspected form of $-\ln(m^i(s|a))$ and the normalization give (2.26).

By substituting the equation for $m^i(s|a)$ (2.26) into (2.30) for $a \in \mathbb{A} \setminus \bar{\mathbb{A}}$ we get the equations for the opted $e(a)$ in (2.26)

$$\begin{aligned}
L(\mathbf{e}(a)) &= R(a), \text{ where} \\
L(\mathbf{e}(a)) &\equiv \mathbf{e}(a)\Lambda(a) + \ln\left(\int_{\mathbb{S}} m(s|a) \exp[-\mathbf{e}(a)m(s|a)]ds\right) \underbrace{\int_{\mathbb{S}} m(s|a)ds}_{=1} \\
&= \mathbf{e}(a)\Lambda(a) + \ln\left(\int_{\mathbb{S}} m(s|a) \exp[-\mathbf{e}(a)m(s|a)]ds\right), \\
\Lambda(a) &\equiv \int_{\mathbb{S}} m^2(s|a)ds > 0 \\
R(a) &\equiv -\int_{\mathbb{S}} m(s|a) \ln\left(\frac{m(s|a)}{h(s)}\right)ds + d^o(\bar{a}) + \ln\left(\frac{\max_{a \in \mathbb{A}} \rho(a)}{\rho(a)}\right) \geq 0 \\
&= -\int_{\mathbb{S}} m(s|a) \ln\left(\frac{m(s|a)}{h(s)}\right)ds + d^o(\bar{a}) + \ln\left(\frac{\rho(\bar{a})}{\rho(a)}\right), \quad \bar{a} \in \text{Arg max}_{a \in \mathbb{A}} \rho(a) \equiv \bar{\mathbb{A}}.
\end{aligned}$$

Under (2.16), the right-hand side $R(a)$ is bounded on \mathbb{A} . It remains to verify existence of $\mathbf{e}(a)$ solving (2.28) for each $a \in \mathbb{A}$. The first derivative of $L(\mathbf{e}(a))$ with respect to $\mathbf{e}(a)$, $a \in \mathbb{A}$, reads

$$\begin{aligned}
\frac{dL(\mathbf{e}(a))}{d\mathbf{e}(a)} &= \Lambda(a) + \frac{1}{\int_{\mathbb{S}} m(s|a) \exp[-\mathbf{e}(a)m(s|a)]ds} \frac{d\left(\int_{\mathbb{S}} m(s|a) \exp[-\mathbf{e}(a)m(s|a)]ds\right)}{d\mathbf{e}(a)} \\
&= \Lambda(a) + \frac{1}{\int_{\mathbb{S}} m(s|a) \exp[-\mathbf{e}(a)m(s|a)]ds} \int_{\mathbb{S}} \frac{\partial}{\partial \mathbf{e}(a)} m(s|a) \exp[-\mathbf{e}(a)m(s|a)]ds \\
&= \Lambda(a) + \frac{1}{\int_{\mathbb{S}} m(s|a) \exp[-\mathbf{e}(a)m(s|a)]ds} \int_{\mathbb{S}} m(s|a) \exp[-\mathbf{e}(a)m(s|a)](-m(s|a))ds \\
&= \Lambda(a) - \int_{\mathbb{S}} m(s|a)m^i(s|a)ds. \tag{2.31}
\end{aligned}$$

The second derivative is the positive variance of $m(s|a)$ concerning $m^i(s|a)$

$$\begin{aligned}
\frac{d^2L(\mathbf{e}(a))}{d\mathbf{e}^2(a)} &= -\int_{\mathbb{S}} m^2(s|a) \frac{\exp[-\mathbf{e}(a)m(s|a)](-m(s|a))}{\int_{\mathbb{S}} m(s|a) \exp[-\mathbf{e}(a)m(s|a)]ds} ds \\
&\quad - \frac{\int_{\mathbb{S}} m^2(s|a) \exp[-\mathbf{e}(a)m(s|a)]ds}{\left(\int_{\mathbb{S}} m(s|a) \exp[-\mathbf{e}(a)m(s|a)]ds\right)^2} \int_{\mathbb{S}} m(s|a) \exp[-\mathbf{e}(a)m(s|a)](-m(s|a))ds \\
&= \int_{\mathbb{S}} m^2(s|a)m^i(s|a)ds - \left[\int_{\mathbb{S}} m(s|a)m^i(s|a)ds\right]^2 > 0.
\end{aligned}$$

Thus, $L(\mathbf{e}(a))$ is a convex function of $\mathbf{e}(a)$, which is monotonous whenever the derivative (2.31) non-zero. The zero variance i.e. excluded constant $m(s|a)$. For $\mathbf{e}(a) = 0$, $L(0) = 0 \leq R(a)$ as $R(a) \geq 0$ due to (2.24) and (2.17). For the non-constant $m(s|a)$, $\lim_{\mathbf{e}(a) \rightarrow \infty} L(\mathbf{e}(a)) = \infty$ as $\Lambda(a) > 0$. The case $\Lambda(a) = 0$ is excluded by the normalisation $\int_{\mathbb{S}} m(s|a)ds = 1$.

Thus, the left-hand side $L(\mathbf{e}(a))$, continuously dependent on $\mathbf{e}(a)$, intersects $R(a)$ at most for two values of $\mathbf{e}(a)$ solving the inspected equation. The solution leading to the smaller (non-negative) value $d^o(a)$ is the proper one. The strict convexity guarantees that the numerical search for the solution is trouble-less. \square

Theorem 2.3.4. [m^{io} Meeting (2.15), Uniform $m(s|a)$] For uniform pd $m(s|a)$ on \mathbb{S} with $|\mathbb{S}| < \infty$, the optimal m^{io} -factor meeting (2.17) has the form

$$m^i(s|a) = \frac{\exp[-\mathbf{e}(a)\mathbf{o}(s|a)]}{\int_{\mathbb{S}} \exp[-\mathbf{e}(a)\mathbf{o}(s|a)]ds} \quad (2.32)$$

for an arbitrary non-zero $\mathbf{o}(s|a)$ with $\int_{\mathbb{S}} \mathbf{o}(s|a)ds = 0$. The real valued $\mathbf{e}(a)$ is that of the pair existing solutions of (2.33), which makes the corresponding $\mathbf{d}^o(a)$ smaller.

$$\begin{aligned} L(\mathbf{e}(a)) &\equiv \ln \left[\frac{\int_{\mathbb{S}} \exp[-\mathbf{e}(a)\mathbf{o}(s|a)]ds}{|\mathbb{S}|} \right] \\ &= R(a) \equiv \mathbf{d}^o(\bar{a}) + \int_{\mathbb{S}} m(s|a) \ln \left[\frac{h(s)\rho(\bar{a})}{\rho(a)} \right] ds. \end{aligned} \quad (2.33)$$

Proof Let us consider $a \in \mathbb{A}$ with a uniform $m(s|a)$. Then, (2.17) with $\mathbf{d}^o(\bar{a})$ given by (2.24) is Fredholm's integral equation for the unknown function $\ln(\frac{m(s|a)}{m^{io}(s|a)})$, $s \in \mathbb{S}$. Its particular solution is searched in the form $\ln(\frac{m(s|a)}{m^{io}(s|a)}) = \mathbf{e}(a)\mathbf{o}(s|a) + \mathbf{v}(a)$. The choice $\int_{\mathbb{S}} \mathbf{o}(s|a)ds = 0$ makes $\mathbf{o}(s|a)$ orthogonal to the uniform $m(s|a)$ and gives (2.32). The definition of $\mathbf{d}(a)$ (1.11) and the equation (2.17) provides (2.33).

Inspection of the 1st and 2nd derivatives of $L(\mathbf{e}(a))$ in (2.33) with respect to $\mathbf{e}(a)$ shows that it is convex function for inevitably non-constant $\mathbf{o}(s|a)$.

The left-hand side $L(\mathbf{e}(a))$ of (2.33) is zero for $\mathbf{e}(a) = 0$, while right-hand side is non-negative for $\mathbf{d}^o(\bar{a})$ (2.24). Also, $\lim_{\mathbf{e}(a) \rightarrow \pm\infty} L(\mathbf{e}(a)) = \infty$ as $\mathbf{o}(s)$ must be negative (positive) on a subset of \mathbb{S} of a positive volume. This implies nature and existence of the solution of (2.33). \square

2.4 Algorithmic summary for discrete-valued states and actions

The obtained optimization is summarized in Algorithm 1. It is written for the simple case of the closed-loop with a finite amount of possible states and actions. We show the overall evaluation structure without the need to cope with nontrivial integrations and potential violations of finiteness assumptions (2.10), (2.27). The conditioning state $\tilde{s} = s_{t-1}$ is explicitly written there.

Algorithm 1 FPD with Preference Quantification for Behaviours with a Finite Number of Realisations

Inputs

- ✓ Finite sets of states \mathbb{S} and actions \mathbb{A} , subsets of ideal states $\emptyset \neq \mathbb{S}^i \subset \mathbb{S}$ and actions $\emptyset \neq \mathbb{A}^i \subset \mathbb{A}$
- ✓ The relative weight $w \geq 0$ of sets $\mathbb{S}^i, \mathbb{A}^i$ % (2.15)
- ✓ Environment model $m(s|a, \tilde{s}), s, \tilde{s} \in \mathbb{S}, a \in \mathbb{A}$
- ✓ Design horizon T , the exploration controlling $\nu > 1$ and the value function $h(s) \equiv 1, \forall s \in \mathbb{S}$ % (1.11)

Evaluation of h-independent variables
For $\tilde{s} \in \mathbb{S}$ do
For $a \in \mathbb{A}$ do

$$\rho(a|\tilde{s}) = \sum_{s \in \mathbb{S}^i} m(s|a, \tilde{s}) + \chi_{\mathbb{A}^i}(a)w \quad \% (2.15)$$

$$\Lambda(a|\tilde{s}) \equiv \sum_{s \in \mathbb{S}} m^2(s|a, \tilde{s}) \quad \% (2.28)$$

end of $a \in \mathbb{A}$

$$\bar{a}(\tilde{s}) \in \text{Arg max}_{a \in \mathbb{A}} \rho(a|\tilde{s}) \quad \% (2.17)$$

$$\bar{\rho}(\tilde{s}) = \rho(\bar{a}(\tilde{s})|\tilde{s}) \quad \% (2.17)$$

end of $\tilde{s} \in \mathbb{S}$
Design cycle for $t = T, T - 1, \dots, 1$:
Evaluation of h-dependent variables
For $\tilde{s} \in \mathbb{S}$ do

$$d^o(\bar{a}(\tilde{s})) \equiv \max \left\{ 0, \max_{a \in \mathbb{A}} \left[\sum_{s \in \mathbb{S}} m(s|a, \tilde{s}) \ln \left[\frac{\rho(a|\tilde{s})}{\bar{\rho}(\tilde{s})h(s)} \right] \right] \right\} \quad \% (2.24)$$

For $a \in \mathbb{A}$ do

$$d^o(a|\tilde{s}) = d^o(\bar{a}(\tilde{s})) + \ln \left(\frac{\bar{\rho}(\tilde{s})}{\rho(a|\tilde{s})} \right) \quad \% (2.17)$$

If $m(s|a, \tilde{s})$ is not uniform

$$R(a|\tilde{s}) = d^o(a|\tilde{s}) + \sum_{s \in \mathbb{S}} m(s|a, \tilde{s}) \ln(h(s)) \quad \% (2.28)$$

$$\text{Find } e(a|\tilde{s}) \text{ in } R(a|\tilde{s}) = e(a|\tilde{s})\Lambda(a|\tilde{s}) + \ln \left(\sum_{s \in \mathbb{S}} \exp[-e(a|\tilde{s})m(s|a, \tilde{s})] \right) \quad \% (2.28)$$

$$m^{io}(s|a, \tilde{s}) \propto \exp(-e(a|\tilde{s})m(s|a, \tilde{s})) \quad \% (2.26)$$

else

 Choose $o(s)$ such that $\sum_{s \in \mathbb{S}} o(s) = 0$

$$\text{Find } e(a|\tilde{s}) \text{ in } \ln \left[\sum_{s \in \mathbb{S}} \frac{\exp[-e(a|\tilde{s})o(s)]}{|\mathbb{S}|} \right] = d^o(\bar{a}(\tilde{s})) + \frac{1}{|\mathbb{S}|} \sum_{s \in \mathbb{S}} \ln \left[\frac{h(s)\bar{\rho}(\tilde{s})}{\rho(a|\tilde{s})} \right]$$

$$\text{Set } m^{io}(s|a) \propto \exp[-e(a|\tilde{s})o(s)]. \quad \% (2.32)$$

end if on uniform m

$$r^{io}(a|\tilde{s}) = \exp[-\nu d^o(a|\tilde{s})] \quad \% (2.11)$$

end of $a \in \mathbb{A}$

$$r^{io}(a|\tilde{s}) = \frac{r^{io}(a|\tilde{s})}{\sum_{a \in \mathbb{A}} r^{io}(a|\tilde{s})}, a \in \mathbb{A} \quad \% (2.11)$$

$$n(\tilde{s}) = \sum_{a \in \mathbb{A}} r^{io}(a|\tilde{s}) \exp[-d^o(a|\tilde{s})] \quad \% (1.11)$$

$$r^o(a|\tilde{s}) = \frac{\exp[-(\nu+1)d^o(a|\tilde{s})]}{n(\tilde{s})}, a \in \mathbb{A} \quad \% (1.11)$$

end of $\tilde{s} \in \mathbb{S}$

$$h(s) = n(s), \forall s \in \mathbb{S} \quad \% (1.11)$$

end of the design cycle
Outputs

- ✓ Optimal ideal m^{io} -factors over whole design horizon
 - ✓ Optimal ideal r^{io} -factors over whole design horizon
 - ✓ Optimal decision rules r^o -factors over whole design horizon.
-

Chapter 3

Preference elicitation as a dialogue with the user

Before the DM, a model of the system and preferences have to be specified. Because we already described how to define the model of the system or how to use Bayesian learning to estimate the model in Section 1.4, it remains to specify how we will find out the preferences of the agent. In the previous thesis, the agent specified the state and action, which are preferred, before the beginning of the DM. There was this problem if the agent¹ wished two opposite things. In this case, we needed to choose the weight w part of (2.15), which determines how much the user prefers to stay in set A^i relative to being in S^i . Furthermore, the user could not give feedback during the DM. It may have been that the user has specified its goals and obtained something else. That is why we added dialogue with the user during the DM. So the user will have control over his preferences and over the result of the DM.

The DM with PE described in Chapter 2., referred to as the basic DM, deals with two types of inputs:

- ✓ those directly describing the basic DM, which include:
 - ▶ the state S and action A sets;
 - ▶ the wishes-expressing ideal sets $S^i \subset S$ and $A^i \subseteq A$;
- ✓ more technical, strategy-influencing, inputs that include:
 - ▶ the weight $w \geq 0$ balancing the relative importance of ideal sets, see (2.15);
 - ▶ the scalar $\nu > 1$ balancing exploitation with exploitation (duality, [8, 20]).

Fine modifications of ideal sets S^i , A^i or the design horizon $|T|$ are other potential inputs of Alg. 1. For example, the user will not prefer one state/action but the whole neighbourhood of the state/action, in discrete space it would be the nearest surrounding states/actions. The user can change his preferences and extend the sets because in some cases the user will not “feel” the difference. These changes are unconsidered here. At least, if we let change more parameters, the problem will get much difficult because of the dimensionality and it will be more time-consuming. In this thesis, we present what the dialogue with the user may look like and the part with more parameters changes we let for further researches.

Thus, the rest of this work focuses just on the pair w , ν . Its optimal choice depends on: ▶ subjective user’s preferences; ▶ the user’s attitude to the basic DM; ▶ emotions, etc., all together on user’s mental state. The dependence is complex and the mental state can hardly be directly measured and quantified.

¹In this chapter the agent will be called user as it is usual for PE.

Two users can have the same preferences expressed by the sets $\mathbb{S}^i, \mathbb{A}^i$, but they can respond differently. One of them may reduce demands and the solution will be absolutely different for each of them. Thus, it is necessary to relate the optional inputs to the explicitly-expressed user's satisfaction. The user is asked to judge the DM quality reached for various choices of inputs. This is the domain of classical PE [6] that often elicits preferences about a static DM and interactively queries the user. Even advanced versions, represented by [3], become cumbersome in the targeted basic *dynamic* DM. This makes us adopt the next user-driven way that consists of formulating and solving an appropriate FPD meta-task.

The user assigns (satisfaction) marks, serving as the (meta-)state $S_T \in \bar{\mathbb{S}}$, to the behaviour caused by the policy, designed via Alg. 1 for trial values of the optional inputs (here, (w, ν)). Their changes A_T are as the (meta-)action. The actions are generated by (meta-)policy gained by Alg. 1. It runs more slowly than the basic DM, $T \in \{\bar{T}, 2\bar{T}, \dots\} \subset \mathbb{T}$ given by a step $\bar{T} > 1$. The applied zero-order holder keeps the latest user's marking as the current state. This makes the user quite free and allows the user to stop the interactions according to their will.

This simple idea has to cope with the possible infinite regress, i.e. Alg. 1 at meta-level needs meta-inputs opted via a meta-PE, etc. Also, the curse of dimensionality [2] endangers applicability as the opted inputs are multiple and continuous-valued. The following way counteracts both obstacles.

We decided to ask queries after every time epoch $\bar{T} > 1$, but also the queries could be asked irregularly after some multiples of the \bar{T} .

The use of zero-order holder copes with the expected irregularity of user's responses. It makes realistic the time-invariance of the model $M(S_T|A_T, S_{T-\bar{T}}, \Theta) := \Theta_{S_T|A_T, S_{T-\bar{T}}}$ needed for learning this meta-model, cf. the beginning of Sec. 2.4.

The choice of the ordinal scale of marks $\bar{\mathbb{S}} := \{1, \dots, |\bar{\mathbb{S}}| := 5\}$ suffices for expressing "satisfaction degree". A rich, cross-domain, experience, e.g. in marketing [4] or in European Credit and Accumulation System, confirms this. The mark $S = 1$ is taken as the best one, which unambiguously defines the ideal set $\bar{\mathbb{S}}^i := \{1\}$.

By construction, the outcomes of the basic DM depend smoothly on the discussed inputs. Thus, changes $A := (\Delta w, \Delta \nu)$ of inputs (w, ν) can be selected in a finite set $\bar{\mathbb{A}} := \{(\Delta w, \Delta \nu)\}$ of discrete values. The natural flexible options are

$$\Delta w \in \{-\bar{w}, 0, \bar{w}\}, \quad \Delta \nu \in \{-\bar{\nu}, 0, \bar{\nu}\}, \quad \bar{w}, \bar{\nu} > 0. \quad (3.1)$$

Alg. 1 is to guarantee that opted inputs stay within their admissible ranges ($w \geq 0, \nu > 0$). The used simple clipping at boundaries of (3.1) seems to suffice. No other demands exist with respect to action. Thus, $\bar{\mathbb{A}} = \bar{\mathbb{A}}^i$ and $W = 0$ (meta-twin to w in (2.15)). The last input to the meta-use of Alg. 1 is the counterpart of ν . This input cares about exploration that has to be stimulated at both levels. It makes no sense to choose a different value at the meta-level. Thus, ν is common at both levels: a slightly delayed value ν_{T-1} is at disposal when designing the new one.

The appearance of $\bar{T}, \bar{w}, \bar{\nu}$ demonstrates the danger of infinite regress. At present, it is cut by force and they are chosen heuristically. They, however, cover, the first step in a conceptual solution that: ► lets appear only meta-inputs that have a weak influence on results; ► tunes them via a universal adaptive minimization of miss-modelling error [12].

Chapter 4

Experiments

There are several experiments in this chapter. Experiments primarily illustrate the presented theory. Additional sample examples are in [26]. We present a realistic example with a heating system.

4.1 Common Simulation and Evaluation Options

Simulated environment is chosen to be $15 \times 7 \times 15$ given by $|\mathbb{S}| = 15$ and $|\mathbb{A}| = 7$. It is created by learning the transition pd $p(s_t|a_t, s_{t-1})$. The system is simulated with 10^5 real values y_t stimulated by independently generated discrete actions in $\mathbb{A} := \{1, \dots, 7\}$. The states $s_t \in \mathbb{S} := \{1, \dots, 15\}$ are gained via an affine mapping of discretized values of the real-valued y_t generated by equation ($y_0 = 0$)

$$y_t = 0.99y_{t-1} + 0.05a_t - 0.125 + 0.05\varepsilon_t.$$

There, ε_t is the white, zero-mean, normal noise. It has a unit variance.

Experiments: DM results without and with the user's control are compared. DM without the user's control is the basic DM with no meta-level and preferences expressed by the ideal sets $\mathbb{S}^i, \mathbb{A}^i$ (2.14) and by fixed options w, v (2.11),(2.15). DM with the user's control solves the basic DM supported by the second-layer implementing the solution of the meta-DM task as described in Chapter 3. The DM with user's control gives the user the chance to express their satisfaction every ten steps, $\bar{T} = 10$. The satisfaction is quite subjective. It is demonstrated by presenting selected results for different users.

Experimental conditions (see below) are set to make results comparable. The users are informed about the key common conditions, i.e. the preferred state and preferred action, the price paid for the respective action values, see Table 4.1 and for state values in Table 4.2 and Table 4.3. The prices express the deviation from the preferred action/state. We objectively want to minimize the sum of prices. So the experiment with the lowest price is considered to be objectively the best. We decided to have two different prices for states because we want to express that the final evaluation of results can depend on how the price is defined. The first price neglects small deviations and is prone to large deviations. The second price values multiple deviations for the same price. It guarantees that it will depend more on whether there is the deviation and not so much on how big it is. Then, the experiments with more occurrences in the preferred state will be evaluated as the best experiments and not experiments with the smallest deviations.

Table 4.1: The price paid for individual action values

action	1	2	3	4	5	6	7
price	3	2	1	0	1	2	3

Table 4.2: The price 1 paid for individual state values

state	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
price 1	3	2.5	2	1.5	1	0.5	0	0.5	1	1.5	2	2.5	3	3.5	4

Table 4.3: The price 2 paid for individual state values

state	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
price 2	3	3	2	2	1	1	0	1	1	2	2	3	3	4	4

4.2 Decision making without user's control

Experiment 1. The user's preference is $\mathbb{S}^i = \{7\}$ and no extra preference is expressed on actions, $\mathbb{A}^i = \mathbb{A}$.

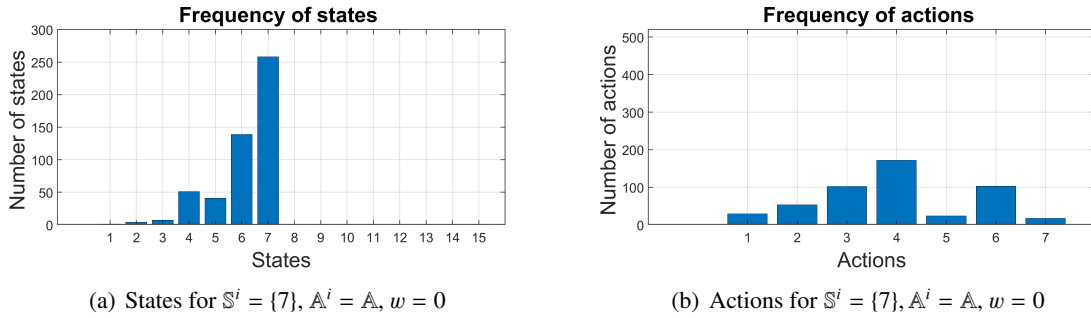


Figure 4.1: Exp. 1: states and actions in DM without user's control and no preference on actions.

Discussed results: The results are in Fig. 4.1. The wished state occurs the most often as we want. Because this experiment has no extra preference on actions, the actions are chosen just to fulfil the preference on states. All action values were realized with no extreme dominance of one value. It can however be seen that it will probably not be difficult to get good results when requesting action $\mathbb{A}^i = \{4\}$.

Experiment 2. The user’s preference is $\mathbb{S}^i = \{7\}$ while requiring the actions to be in “zero energy” set $\mathbb{A}^i = \{4\}$. The weight value $w = 0.3$ (2.15) is fixed to express the latter preference.

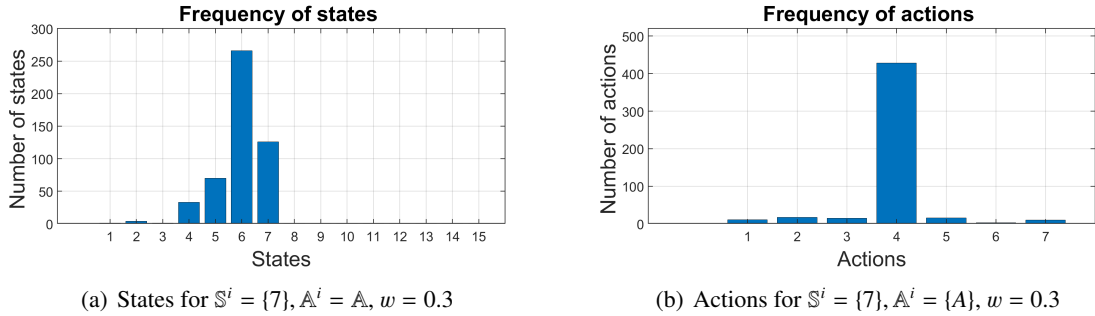


Figure 4.2: Exp. 2: states and actions in DM without user’s control and with the preference on actions.

Discussed results: The results are in Fig. 4.2. As it can be seen, the wished state does not occur as often as in Exp. 1 due to the additional preference on actions. For $w = 0.3$, the wished action occurs the most often and the number of the wished action is much higher than in Exp. 1. The comparison of Exp. 5 and Exp. 6 shows that these two experiments contradict each other. We need to find the balance between these preferences. When these two preferences can not be fulfilled together, the user needs to choose, which preference is more important for them.

Experiment 3. The user’s preference is again $\mathbb{S}^i = \{7\}, \mathbb{A}^i = \{4\}$ as in Exp. 2. The extreme weight $w = 10$ is tried.

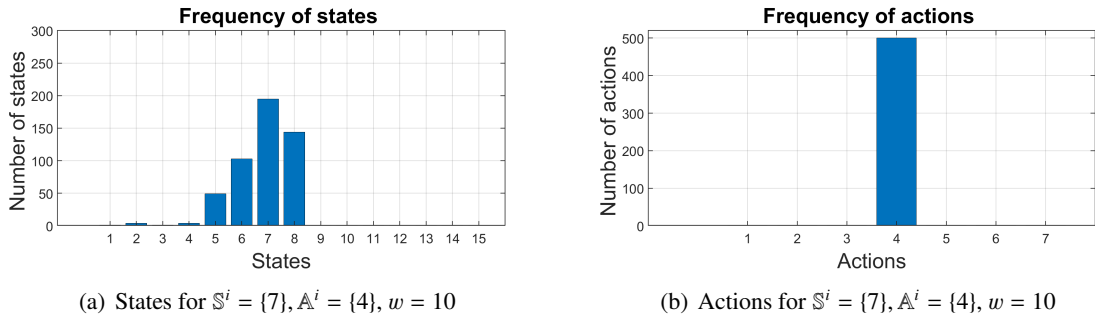


Figure 4.3: Exp. 3: states and actions in DM without user’s control and with a hard preference on actions

Discussed results: The results are in Fig. 4.3. As we expect the target state $\mathbb{S}^i = \{7\}$ is reached less often than in the previous case. The “harmonized” state $\{8\}$ is visited more often than before. The stress on the desired actions is surely too high. We can see that no non-preferred action is chosen. There is no balance between the preferences. It is generally dangerous as the found policy lacks the explorative capability. The same dangerous behaviour was observed for all $w \geq 1$.

4.3 Decision making with user's control

We want to find a balance between contradiction preferences. The parts with additional preference are kind of hard because the user can not before the decision making explain or show how much they prefer to stay in the preferred state relative to selecting the preferred action. And every user is different and has a different view on it. So for one user would be better to have the situation without additional preference and for someone else would be better if they get the preference on actions and do not care about the preferred state. And for someone, it might be better if the weight w was completely different. That is why we worked on the decision making with the user's control. The user gives feedback and based on that the weight w is selected and also the free parameter of exploration ν adapts. Then, we can say even during the decision making, if the user is satisfied and what are their preferences.

Experiment 4. The user's preference is $\mathbb{S}^i = \{7\}$, $\mathbb{A}^i = \{4\}$. Neither the weight w nor ν are fixed and the 1st user marks the seen closed-loop behaviour.

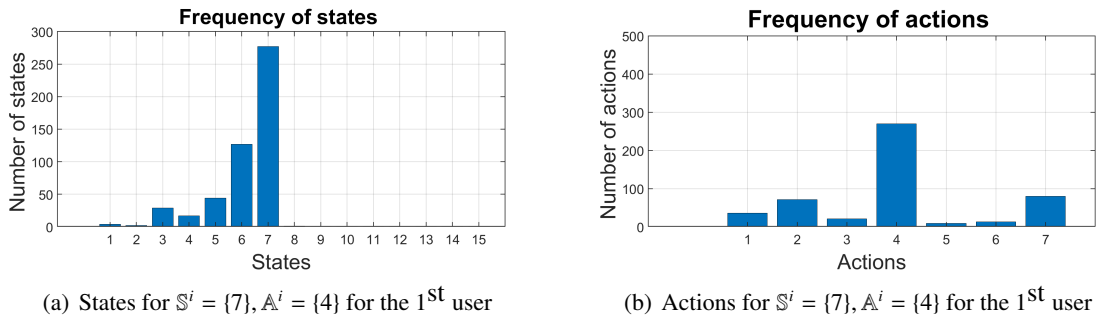


Figure 4.4: Exp. 4: states and actions in DM with the 1st user control

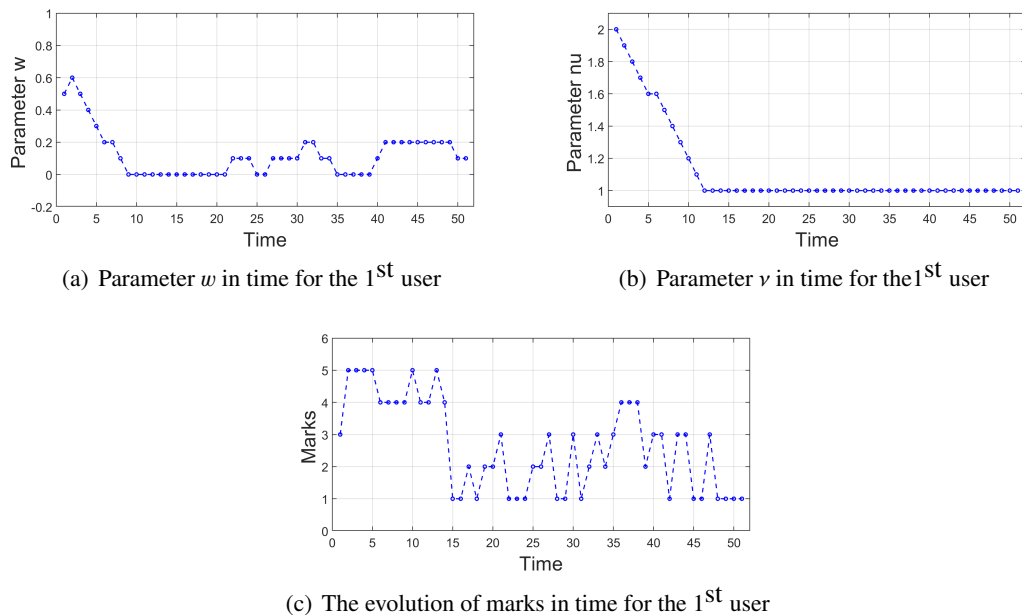


Figure 4.5: Exp. 4: The evolution of parameters of the 1st user in time.

Discussed results: The results are in Fig. 4.4. As it can be seen the preferred state occurs the most often. Compared to Exp. 1. without the user's control, this experiment gives better results. The preferred state occurs more often and so does the preferred action. In Fig. 4.5 there is the evolution of parameters w , ν and the user's marks. The parameter ν converges to 1. The parameter w moves around 0 to 0.2. The mark 1 appears quite often, so the 1st user is satisfied.

Experiment 5. The user's preference is $\mathbb{S}^i = \{7\}$, $\mathbb{A}^i = \{4\}$. Neither the weight w nor ν are fixed and the 2nd user marks the seen closed-loop behaviour.

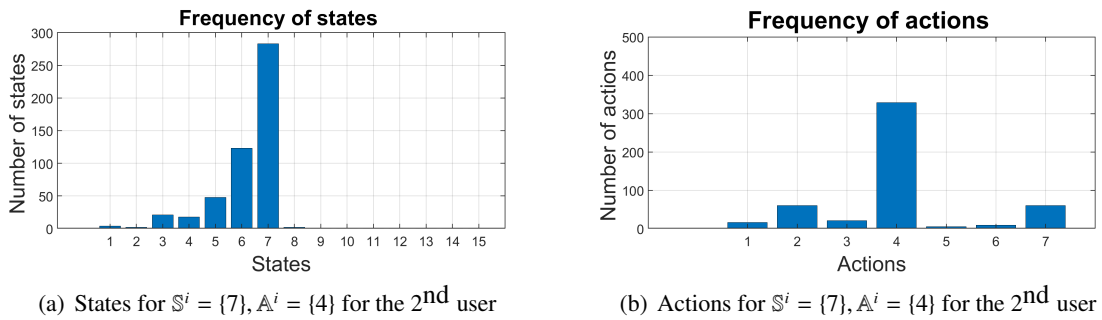


Figure 4.6: Exp. 5: states and actions in DM with the 2nd user control

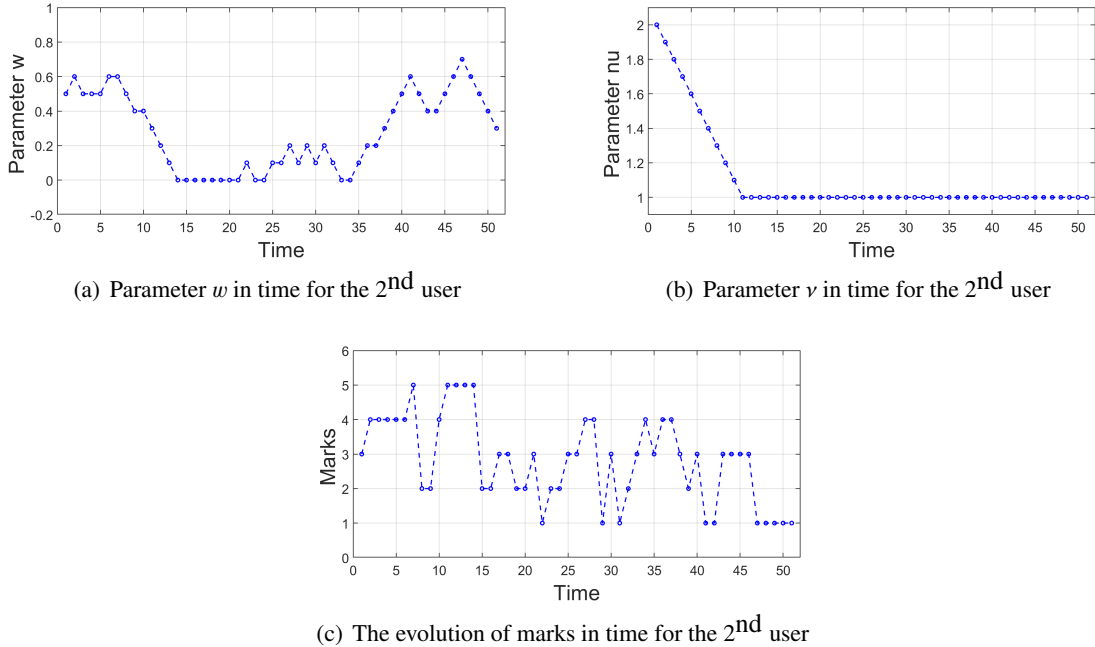


Figure 4.7: Exp. 5: The evolution of parameters of the 2nd user in time.

Discussed results: The results are in Fig. 4.6. This user selects the preferred action even more often than the first one. The preferred state does not occur significantly differently from the first one. So for

our objective comparison, these results are better, because this user pays a lesser price than the first one, see Table 4.4.

This user is not as satisfied as the first one if you see the marks in Fig. 4.7. However, it might also mean that the second user is very strict or that the first one is tolerant and settles for less. In any case, we can say that the more strict user gets better objective results. The parameter ν converges to the value 1 faster on the other hand the weight w is still tuned.

Experiment 6. In this experiment we compare evolution of states and actions in time for the 1st user and the 2nd user.

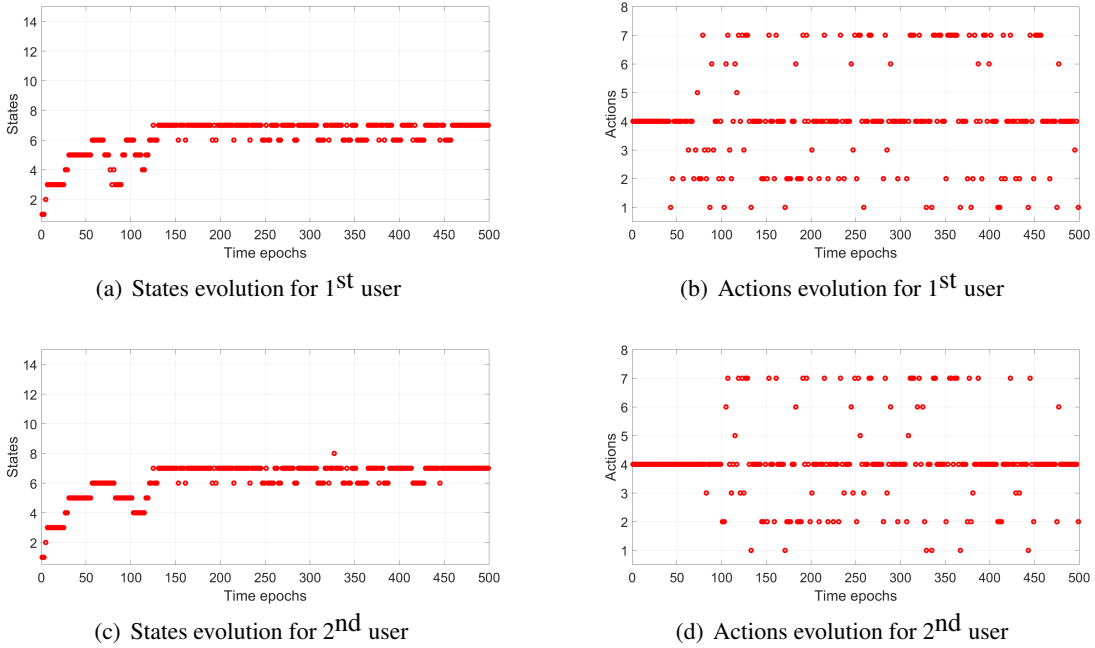
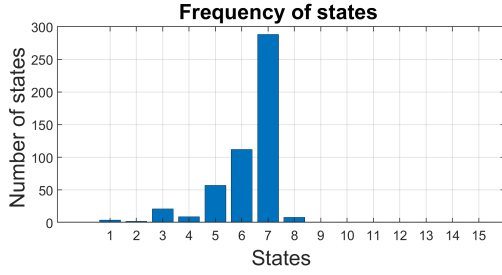


Figure 4.8: Evolution of states and actions for the 1st and 2nd user.

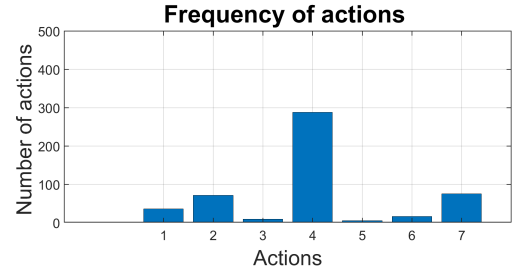
Discussed results: The results are in the Fig. 4.8. We display every second measurement for clarity. The evolution of states is more consistent for the second user. The states increase and decrease just once and after that, they stabilize in contrast with the first user, whose states increase and decrease twice before they stabilize. The evolution of actions is also more consistent for the 2nd user. Other actions (differing from the preferred action) are chosen less often. For example, action 1 is chosen much less often for the 2nd user than for the 1st user.

Experiment 7. In this experiment we show results of some other users. The user's preference is still $\mathbb{S}^i = \{7\}$, $\mathbb{A}^i = \{4\}$. Neither the weight w nor ν are fixed and the 3-5th users mark the seen closed-loop behaviour.

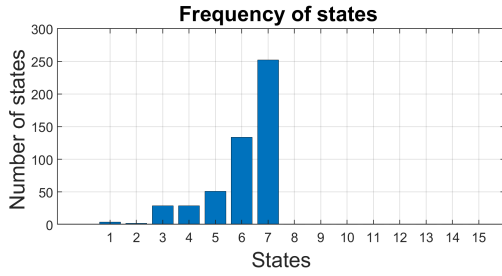
Discussed results: The users 3-5 were asked to have preferences $\mathbb{S}^i = \{7\}$, $\mathbb{A}^i = \{4\}$ and they were let to decide what preference will be more important for them.



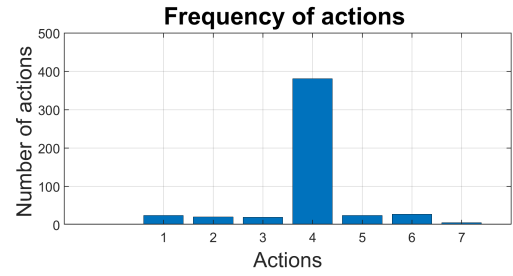
(a) States for $\mathbb{S}^i = \{7\}, \mathbb{A}^i = \{4\}$ for the 3rd user



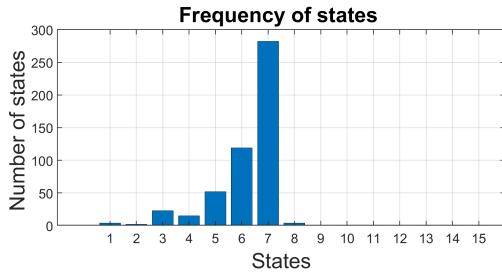
(b) Actions for $\mathbb{S}^i = \{7\}, \mathbb{A}^i = \{4\}$ for the 3rd user



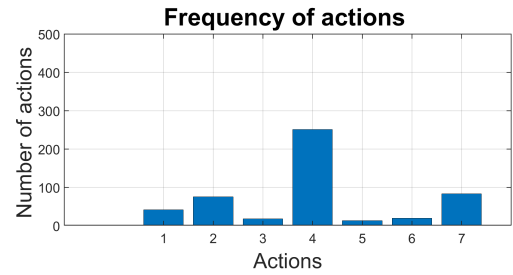
(c) States for $\mathbb{S}^i = \{7\}, \mathbb{A}^i = \{4\}$ for the 4th user



(d) Actions for $\mathbb{S}^i = \{7\}, \mathbb{A}^i = \{4\}$ for the 4th user



(e) States for $\mathbb{S}^i = \{7\}, \mathbb{A}^i = \{4\}$ for the 5th user



(f) Actions for $\mathbb{S}^i = \{7\}, \mathbb{A}^i = \{4\}$ for the 5th user

Figure 4.9: Exp. 7: states and actions in DM with the 3-5nd user control

The main differences are seen in the frequency of the state 8 and the frequency of the preferred action 4. We show that users are different. If five users get the same task: mark the sequence of states and actions with the same preference, they get significantly different results. Some of the users are strict and they are not satisfied even with better results than tolerant users accepting worse results. For example, the 4th user, obviously, is more interested in having the preferred action than to have the preferred state.

The results are discussed in the next section. We assess, which user is satisfied the most. And the evolution of tuned parameters can be seen in Appendix.

4.4 Comparison of costs and responses in all experiments

Table 4.4 shows the prices paid for actions and states in all experiments. It confirms expectations, including the desirable influence of users.

Discussed results: All of these results are easily comparable within to the Table 4.4. The first three results are for the experiments without the user's control with fixed parameters and then five experiments

Table 4.4: The price paid for actions in all experiments

Exp. no	Opted Parameters	The price of actions	The price 1 of states	The price 2 of states	The number of selections of the preferred action	The number of occurrences of the preferred state
1.	$w = 0, \nu = 1$	576	214	311	172	258
2.	$w = 0.3, \nu = 1$	134	267.5	419	428	126
3.	$w = 10, \nu = 1$	0	193.5	321	500	195
4.	1 st user	546	208.5	282	270	277
5.	2 nd user	392	195.5	269	329	283
6.	3 rd user	224	236.5	319	381	252
7.	4 th user	521	189.5	255	288	288
8.	5 th user	591	199	269	251	282

with the user's control. These experiments with user's control are done with five different people of different age, sex, education. In the results, there can be seen the diversity of the users' marks and thus the users' view on the pair of preferences. As it is said above the more strict users get better results according to our "objective" comparison. On the other hand, the more tolerant users are more satisfied and that is what we want. We want to get the results, which satisfy the user.

The strategy is chosen to compare the results by prices that will be paid for deviation from preferred state and preferred action. It is described more in Tables 4.1, 4.2 and 4.3. Objectively, we want to minimize the sum of prices. So the experiment with the both (action and state) lowest prices brings the best "objective" result. We also add the number of occurrences of the preferred state and action.

At first, we compare the results for the **price 1** of states. By the price 1, the best results give the experiments 3, 5, 7, 8. If we take into account the price paid for deviation from the preferred action, obviously, the best result is Experiment 3. Another good result is obtained for the 2nd user. It is interesting that the best result is Experiment 3., which has almost the lowest number of occurrences in the preferred state. This is because there is a big amount of occurrences of states 6 and 8, which have a low price 0.5. The small deviations are allowed for the state price 1. However, it could be a good result, because we do not pay any price for action and the occurrence of extreme values is pretty low.

That is why we decided to take into account the number of occurrences of the preferred state and add the state **price 2**. The lowest prices paid for states are for experiments 5, 7, 8. And the lowest prices paid for actions are for experiments 2, 3, 5, 6. So the best result for price 2 is Experiment 5 with the 2nd user's feedback. That is the most strict user. This user has also the second biggest number of occurrences of preferred state and the fourth biggest number of selection of preferred action. This could be objectively the best option.

We would like to emphasize that we present two "objective" comparisons. And for both of them, the best results are different. It is important to have in mind what kind of problem is solved and what price would fit. Furthermore, the occurrence of the preferred state should be taken into account.

The question remains, if the user was so strict, because they wanted to have the best results or if they were not satisfied enough and wanted better possible results. This part with the user's control is kind of hard because we can not know if the user has unachievable preferences or if this is their strategy to force the closed-loop to tune the preferences more. The human factor adds more uncertainty to this problem.

Other experiments: We tried more experiments and we have tried to observe imperfections and possibilities to improve our algorithm and the design of the meta-closed-loop. We have come to the conclusion that it could be better to take as the state of meta-closed-loop the difference between the previous mark and present mark than the absolute value of the present mark. Then, the meta-closed-loop will care about differences, the state will be a difference of marks and the action will be a difference of free parameters. It could give better results and the values of free parameters could be more consistent. The problem is still the dimensionality.

We also tried a more simple system, namely $3 \times 3 \times 3$. The problem with this system was that the meta-system was much more complex. It was hard to learn the meta-system and influence the simple system in the right way. Another problem with small systems is their poor flexibility. There are few possible states and differences of the parameters that can influence the system extremely and not gradually.

Extreme parameter values can also be the problem because the weight cannot be less than zero. This is why it can stay at this value and it may not be the best result. Then, the parameter is no longer tuned and it is not possible to learn the relation of tuned parameters and user's marks well and find the best result. In further research, we would like to work on these problems and to find the way out.

Chapter 5

Conclusions

We studied the theory of optimal decision-making. We introduced the Markov decision process and then we worked with a Fully probabilistic design. We showed how to use Bayes learning for estimating the model of the system. We have found the optimal decision rule based on the given preferences using FPD. We found the ideal probability density of the behaviour. We added the meta-closed-loop and the ability for the user to control decision-making, to fine-tune the free parameters to satisfy the user's preferences the most. The meta-loop changes the free parameters based on the user's feedback (marks). The main advantage is that the user does not have to understand the decision-making algorithm and they just mark the sequences of states and actions.

The algorithm needs as inputs: ► the set of allowed actions; ► specification of the wished state and actions sets; ► the on-line satisfaction marking by the user that judges behaviour improvements caused by changes of the exploration option ν and of the scalar weights w balancing importance the ideal states and actions; ► on-line learned and adapting state-transition models at both levels.

We have coded a program using the theory and prepared the simulation environment. All of these experiments are simulated with a $15 \times 7 \times 15$ system. We used a heat equation and this task can correspond to reality. We have chosen several experiments to illustrate the theory described above and we have simulated them using Matlab. We have divided the experiments into two parts. The first part shows the decision-making without the user's control with fixed parameters chosen by the user. The possible disadvantage of this part could be that the user has to have an understanding of the algorithm and the theory because they need to select the parameters themselves. That is why the second part is with the user's control, where the parameters are free and are fine-tuned during the decision-making. So the user does not have to understand the theory and the parameters are chosen by the algorithm.

The proposed solution addressed a very difficult problem of preference elicitation in dynamic DM. We tried to find the parameters to satisfy the user's preferences. We presented one example of preferences for five different users and also three experiments with fixed parameters. Because the experiments were done with different users, the results are different for the same marginal preferences. Again, the most important is the satisfaction of the user. The parameters are selected based on the user's feedback and if the user's marks converge to 1, the user is satisfied. The biggest advantage of the user's control is that the user does not have to understand the algorithm as is mentioned above. We evaluated experiments for two different comparison prices. For the first price of states, the experiment without the user's control is the best and for the second price of states, the experiment with the user's control is the best. This emphasizes the importance of how the user values the deviations.

We would like to stress that not every decision-making problem with user's preferences can be satisfactorily solved. The user may have preferences, that cannot be reached on the system. They can have

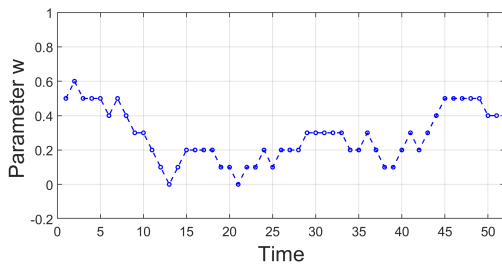
unattainable goals. The system can also have large dimensions and we will not be able to evaluate it in real-time.

Future research In the future, we would like to focus on improving the dialogue with the user. Also, we want to investigate if it would be better to work with differences in marks than with absolute values. Then, we would like to find out how to solve the problems with extreme values of free parameters. We would like to study the dialogues with more free parameters. For example, extension of the sets of preferred states and actions. Because the user can give more importance to their preference on actions and does not care about small deviation from the preferred state. Then, it will be good to extend the set of preferred states to have even better results. Last but not least we want to break the curse of dimensionality.

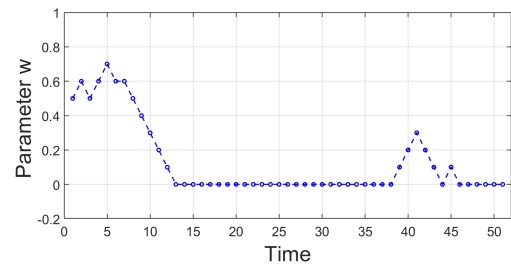
Appendix A

Additional graphs

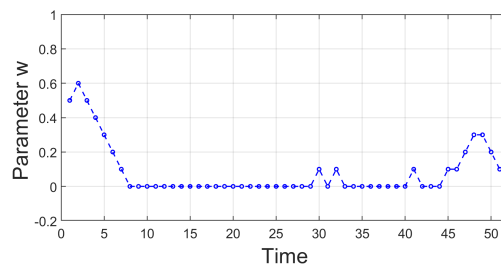
In this chapter there are additional graphs of the experiments. They are moved here in order to make basic text readable and allow the interested readers to see additional details. The graphs concern evolution of free parameters for 3rd - 5th user.



(a) Parameter w in time for the 3rd user

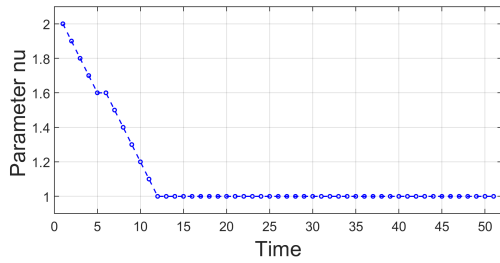


(b) Parameter w in time for the 4th user

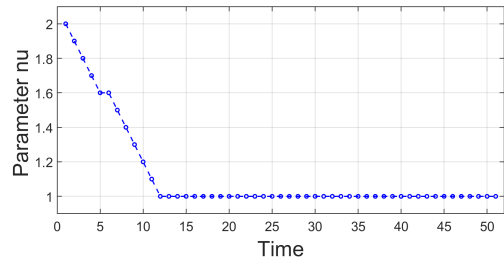


(c) Parameter w in time for the 5th user

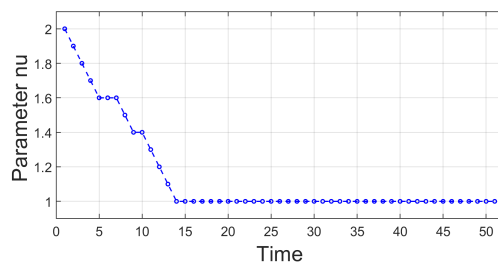
Figure A.1: The evolution of the parameters w (2.15) for the 3rd - 5th user.



(a) Parameter ν in time for the 3rd user

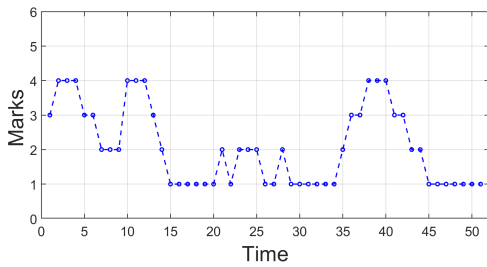


(b) Parameter ν in time for the 4th user

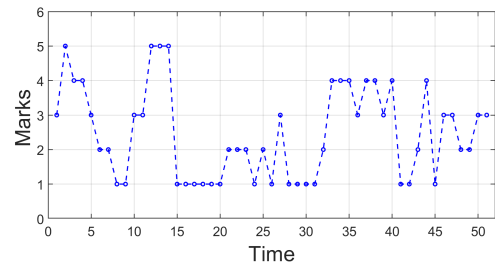


(c) Parameter ν in time for the 5th user

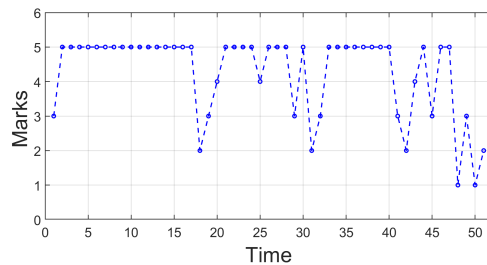
Figure A.2: The evolution of the parameters ν (2.11) for the 3rd - 5th user.



(a) The evolution of marks in time for the 3rd user



(b) The evolution of marks in time for the 4th user



(c) The evolution of marks in time for the 5th user

Figure A.3: The evolution of the 3rd - 5th user's marks.

Bibliography

- [1] R. Bellman. *Dynamic Programming*. Princeton Uni. Press, N.Y., 1957.
- [2] R.E. Bellman. *Adaptive Control Processes*. Princeton U. Press, NJ, 1961.
- [3] C. Boutilier. A POMDP formulation of preference elicitation problems. In *Proc. of the 18th National Conf. on AI, AAAI-2002*, pages 239–246, Edmonton, AB, 2002.
- [4] I. Brace. *Questionnaire design: How to plan, structure and write survey material for effective market research*. Kogan Page, London, 2004.
- [5] J. Branke and et al. Efficient pairwise preference elicitation allowing for indifference. *Computers & Oper. Res.*, 88(Suppl. C):175 – 186, 2017.
- [6] L. Chen and P. Pearl. Survey of preference elicitation methods. Technical Report IC/2004/67, HCI Group Ecole Polytechnique Federale de Lausanne, Switzerland, 2004.
- [7] J. Drummond and C. Boutilier. Preference elicitation and interview minimization in stable matchings. In *Proc. of 28th AAAI Conf. on AI*, pages 645 – 653, 2014.
- [8] A.A. Feldbaum. Theory of dual control. *Autom. Remote Control*, 21-22(9-2), 1960-61.
- [9] K. Gajos and D.S. Weld. Preference elicitation for interface optimization. In *Proc. of the 18th annual ACM Symp. on User interface and Technology, UIST-2005*, pages 173 – 182. ACM, N.Y., 2005.
- [10] I. Garcia, S. Pajares, L. Sebastia, and E. Onaindia. Preference elicitation techniques for group recommender systems. *Information Sciences*, 189:155–175, 2012.
- [11] Shengbo Guo and Scott Sanner. Real-time multiattribute Bayesian preference elicitation with pairwise comparison queries. In *Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics*, pages 289–296, Chia Laguna Resort, Sardinia, Italy, 2010. JMLR Workshop and Conference Proceedings.
- [12] M. Kárný. Towards on-line tuning of adaptive-agent’s multivariate meta-parameter. *Int. J. of Machine Learning and Cybernetics*, 12(9):2717–2731, 2021.
- [13] M. Kárný, J. Böhm, T.V. Guy, L. Jirsa, I. Nagy, P. Nedoma, and L. Tesař. *Optimized Bayesian Dynamic Advising: Theory and Algorithms*. Springer, London, UK, 2006.
- [14] M. Kárný and T.V. Guy. Fully probabilistic control design. *SCL*, 55:259–265, 2006.

- [15] M. Kárný and T.V. Guy. Preference elicitation within framework of fully probabilistic design of decision strategies. In *IFAC Int. Workshop on Adaptive and Learning Control Systems*, volume 52, pages 239–244, 2019.
- [16] M. Kárný and T. Siváková. Model-based preference quantification. *IEEE Trans Cybernetics*, 2021. submitted.
- [17] R.L. Keeney and H. Raiffa. *Decisions with Multiple Objectives: Preferences and Value Tradeoffs*. J. Wiley & Sons, 1976.
- [18] S. Kullback and R. Leibler. On information and sufficiency. *Ann Math Stat*, 22:79–87, 1951.
- [19] A.M.D. Landmark, E.H. Ofstad, and J. Svenneig. Eliciting patient preferences in shared decision-making (sdm): Comparing conversation analysis and sdm measurements. *Patient Education and Counseling*, 100:2081–2087, 2017.
- [20] Ali Mesbah. Stochastic model predictive control with active uncertainty learning: A survey on dual control. *Annual Reviews in Control*, 45:107 – 117, 2018.
- [21] L. Naamani-Dery, M. Kalech, L. Rokach, and B. Shapira. Reducing preference elicitation in group decision making. *Expert Systems with Applications: An International Journal*, 61:246–261, 2016.
- [22] V. Peterka. Bayesian system identification. In P. Eykhoff, editor, *Trends and Progress in System Identification*, pages 239–304. Perg. Press, 1981.
- [23] M.L. Puterman. *Markov Decision Processes: Discrete Stochastic Dynamic Programming*. Wiley, 2005.
- [24] M.M. Rao. *Measure Theory and Integration*. J. Wiley, 1987.
- [25] J. Šindelář, I. Vajda, and M. Kárný. Stochastic control optimal in the Kullback sense. *Kybernetika*, 44(1):53–60, 2008.
- [26] T. Siváková and M. Kárný. Algorithmic choice of feasible preferences. Technical Report 2384, ÚTIA AVČR, 2019. in Czech.
- [27] P. Viappiani and C. Boutilier. Optimal Bayesian recommendation sets and myopically optimal choice query sets. *Advances in Neural Information Processing Systems 23*, pages 2352–2360, 2010.