Czech Technical University in Prague
Faculty of Nuclear Sciences and Physical Engineering

Applicable Adaptive Discounted Fully Probabilistic Design of Decision Strategy

Aplikovatelný adaptivní diskontovaný plně pravděpodobnostní návrh rozhodovací strategie

Master's Thesis

Author: **Soňa Molnárová**

Supervisor: **Ing. Miroslav Kárný, DrSc.**

Academic year: 2023/2024

ZADÁNÍ DIPLOMOVÉ PRÁCE

I. OSOBNÍ A STUDIJNÍ ÚDAJE

Příjmení: **Molnárová** Jméno: **Soňa** Osobní číslo: **476644**

Fakulta/ústav: **Fakulta jaderná a fyzikálně inženýrská**

Zadávající katedra/ústav: **Katedra matematiky**

Studijní program: **Aplikované matematicko-stochastické metody**

II. ÚDAJE K DIPLOMOVÉ PRÁCI

Název diplomové práce:

Aplikovatelný adaptivní diskontovaný plně pravděpodobnostní návrh rozhodovací strategie

Název diplomové práce anglicky:

Applicable Adaptive Discounted Fully Probabilistic Design of Decision Strategy

Pokyny pro vypracování:

1. Vytvořte přehled rozhodovacích problémů, v nichž hraje roli diskontování.
2. Dolaďte teorii plně pravděpodobnostního návrhu s diskontováním [1], [2].
3. Shrňte bayesovské odhadování a předpovídání v dynamické exponenciální rodině modelů [3] a to včetně bayesovsky interpretovaného a odhadovaného zapomínání [4], [5].
4. Zvolte vhodnou verzi realizovatelné kombinace průběžného odhadování s uvažovaným návrhem optimální strategie [6].
5. Navrhněte možné postupy volby faktoru řídícího diskontovaný plně pravděpodobnostní návrh. Například zvažte jeho nastavovaní jako meta-akci[7].
6. Dle potřeby doplňte teoretické výsledky simulačně.

Seznam doporučené literatury:

(části vybrané po dohodě se školitelem)

[1] S. Molnárová, Discounted Fully Probabilistic Design of Decision Strategy, Bc. Thesis, FNSPE CTU, 2022

[2] M. Kárný, S. Molnárová, Discounted Fully Probabilistic Design of Decision Policies, IEEE Tran. on SMC, 2023, submitted

[3] V. Peterka, Bayesian System Identification, In P. Eykhoff, Trends and Progress in System Identification, Pergamon Press, Oxford, 1981,239-303

[4] R. Kulhavý, M.B. Zarrop, On a general concept of forgetting, Int. J. of Control, 58(4), 1993, 905-924

[5] M. Kárný, Minimum Expected Relative Entropy Principle, Proc. of the 18th ECC, 35-40, Sankt Petersburg, 2020, 35-40

[6] A. Mesbah, Stochastic model predictive control with active uncertainty learning: A Survey on dual control, Annual Reviews in Control 45, 2018, 107-117

[7] M. Kárný, Towards on-line tuning of adaptive-agent's multivariate meta-parameter, International Journal of Machine Learning and Cybernetics, 12(9) 2021, 2717-2731

Jméno a pracoviště vedoucí(ho) diplomové práce:

Ing Miroslav Kárný, DrSc. ÚTIA AV ČR Praha

Jméno a pracoviště druhé(ho) vedoucí(ho) nebo konzultanta(ky) diplomové práce:

Datum zadání diplomové práce: **31.10.2023** Termín odevzdání diplomové práce: **10.05.2024**

Platnost zadání diplomové práce: **31.10.2025**

Masáková

Ing Miroslav Kárný, DrSc.
podpis vedoucí(ho) práce

prof. Ing. Zuzana Masáková, Ph.D.
podpis vedoucí(ho) ústavu/katedry

doc. Ing. Václav Čuba, Ph.D.
podpis děkana(ky)

III. PŘEVZETÍ ZADÁNÍ

Diplomantka bere na vědomí, že je povinna vypracovat diplomovou práci samostatně, bez cizí pomoci, s výjimkou poskytnutých konzultací. Seznam použité literatury, jiných pramenů a jmen konzultantů je třeba uvést v diplomové práci.

16. 11. 2023
Datum převzetí zadání

Molnárová
Podpis studentky

Acknowledgment

I would like to thank Ing. Miroslav Kárný, DrSc. for his expert guidance, patience and willingness to go above and beyond to help me throughout the years that I have studied under his supervision. I would also like to thank Ing. Tatiana Valentine Guy, Ph.D. for allowing me to become a part of the Adaptive Systems department where I worked as a research assistant while working on this thesis.
This work has been supported by EU-COST Action CA21169.

Author's declaration

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

Prague, May 10, 2024

Soňa Molnárová

Molnárová

Názov práce: **Aplikovatelný adaptivní diskontovaný plně pravděpodobnostní návrh rozhodovací strategie**
Autor: Soňa Molnárová
Obor: Aplikované matematicko stochastické metody
Druh práce: Diplomová práce
Vedúci práce: Ing. Miroslav Kárný, DrSc.,
ÚTIA AV ČR, v.v.i., Pod Vodárenskou věží 4, 182 00 Praha 8

Abstrakt: Práca sa zaoberá problematikou nižšej uplatniteľnosti budúcich výnosov, inak nazývanej diskontovanie, pri využití plne pravdepodobnostného návrhu rozhodovacej stratégie (PPN). PPN získava optimálnu stratégiu pre riešenie rozhodovacích úloh iba prostredníctvom pravdepodobnostných distribúcií, v čom spočíva jeho hlavná výhoda. Štandardne sa rozhodovacie úlohy riešia pomocou Markovských rozhodovacích procesov (MRP), ktoré PPN taktiež zahŕňa. Postupy riešenia diskontovaných MRP už navrhnuté boli. PPN má však výhodu pri riešení úloh s neznámym modelom systému. Vďaka svojej pravdepodobnostnej podstate totiž PPN získava presnejšie odhady tohto modelu. Po predchádzajúcom rozšírení PPN o diskontovanie a odhadovanie modelu systému sa táto práca zameriava na preskúmanie vplyvu diskontovania na rozhodovacie procesy a jeho možné výhody v úlohách s neznámymi modelmi systému.

Kľúčové slová: Bayesovské odhadovanie, diskontovanie, potlačenie vplyvu približného modelovania, pravdepodobnostný návrh stratégií, rozhodovanie, zabúdanie

Title: **Applicable Adaptive Discounted Fully Probabilistic Design of Decision Strategy**
Author: Soňa Molnárová

Abstract: The thesis addresses the issue of decreased utility of future rewards, referred to as discounting, while utilizing fully probabilistic design (FPD) of decision strategies. FPD obtains the optimal strategy for decision tasks using only probability distributions, which is its main asset. The standard way of solving decision tasks is provided by Markov decision processes (MDP), which FPD covers as a special case. Methods of solving discounted MDPs have already been introduced. However, the use of FPD might be advantageous when solving tasks with an unknown system model. Due to its probabilistic nature, FPD is able to obtain a more precise estimation of this model. After previously introducing discounting and system model estimation to FPD, the thesis now examines the effect of discounting on decision processes and its possible advantages when dealing with an unknown system model.

Key words: Bayesian estimation, decision making, discounting, forgetting, probabilistic strategy design, suppression of approximate modeling impact

Contents

List of Symbols

$\lvert\mathcal{S}\rvert$	cardinality of set $\mathcal{S}$
$\mathrm{x} \in \mathcal{S}$	x is an element of $\mathcal{S}$
$p(\mathrm{x})$	probability density function p value at point x
$\hat{\theta}$	point estimate of parameter θ
$\bigcup \mathcal{S}_i$	union of sets $\mathcal{S}_i$
$\mathcal{S}_y \subset \mathcal{S}_x$	set $\mathcal{S}_y$ is a subset of set $\mathcal{S}_x$
$\times$	symbol for Cartesian product of sets
$\wedge$	logical conjunction
H_t	data up to time t
$\ln(\bullet)$	natural logarithm of $\bullet$
$\equiv$	definition by assignment
$\delta(\mathrm{a}, \mathrm{b})$	Kronecker's delta function of a and b,
$\mathrm{D}(p, q)$	Kullback-Leibler divergence of probability density function p to probability density function q

Chapter 1

Introduction

Each day, every person needs to make a countless decisions to reach his or her set goals. Once a certain decision is made, it influences the decision maker's environment, referred to as the system, causing it to shift from its current state. Understanding how the system changes under the impact of the chosen decisions, or actions, allows for process optimization using standard probability calculus, as addressed in the theory of Markov decision processes [31].

To understand how the system evolves over time, it is necessary to know its model. The system model is represented by the probability density function, which describes how the system transitions between different states under the impact of chosen actions. Unfortunately, knowledge of this model may often be lacking. Bayesian statistical methods [27] are frequently used to estimate the system model. In Bayesian estimation, models are equipped with parameters, considered to be random variables, and the probability density of these parameters is updated after acquiring new data at each step of the process.

An especially challenging case of model estimation arises when parameter values are not constant but vary over time. In such cases, an approach known as forgetting [18] is adopted to make decisions regarding the studied parameters. The more significant the parameter change in the given step of the process was considered to be, the more of the accumulated information about the parameters is forgotten. Rather subtle changes are mostly neglected, whereas very abrupt changes result in disregarding most of the accumulated information. If all gathered information is forgotten, the model needs to be re-estimated from scratch.

Another potential issue is the increasing dimensionality of the task when numerical computations are attempted. As each step of the process brings new information about the parameter, keeping it all can quickly disrupt the feasibility of the algorithm. A widely used method to avoid this inconvenience is to replace the unknown parameters with their most likely estimates, improving computational feasibility. This approach is known as the certainty-equivalence strategy, see [16].

Once the system model estimate is known, it is possible to proceed to the optimization of the process discussed earlier. Process optimization involves selecting actions that prompt the system to move to the decision maker's preferred states. By assigning a loss function to all state transitions under the influence of actions, it is possible to choose the action related to the minimal expected loss function. This practice is used in Markov decision processes.

Another, more general, approach is to construct a decision strategy, which is a sequence of probability densities called decision rules. These describe the process of action selection in probabilistic terms. This is where the fully probabilistic design of decision strategy [15] gets its name. It minimizes the distance between the real system behavior described by system models and decision rules, and their ideal equivalents. The ideals model the most optimistic scenario of system evolution, which may not be attainable in reality.

In comparison with the standard Markov approach, the fully probabilistic formulation is advantageous due to its generality. All Markov decision tasks can be arbitrarily well approximated by a fully probabilistic task, but some fully probabilistic tasks cannot be formulated as Markov decision processes.

Decision tasks can also suffer from improper excitation, meaning that some actions and states may not be chosen at all throughout the process. This results in incomplete description of the system behavior, potentially leading to poor results. This is because the omitted actions might have led to better system performance compared to the chosen actions [20]. However, the randomized decision rules of the fully probabilistic design introduce a degree of exploration. Actions are generated randomly, although there are still preferences towards certain values in the form of probability bias.

Despite these advantages, Markov decision processes, being the more commonly used method for decision problem solving, are equipped with more mathematical embellishments, one of these being discounting. The lack of discounting has already been overcome in [24]. The resulting discounted fully probabilistic design of decision strategy worked with a known state transition model. As mentioned earlier, such a model is rarely known. System model estimation was incorporated in [25], allowing this newly established discounted formulation to gain adaptivity and become applicable in real-life decision tasks.

The aim of this finalizing work was to study the relationship between discounting and forgetting. A set of hypotheses was formulated and tested to determine if a connection between the two could be found and if an improvement in performance could be achieved. The results are presented in Section 3.5. However, before formulating the results, the basic theoretical foundation had to be laid out. Section 2.3 recalls the theory regarding parameter estimation, while Section 2.5 discusses the issue of varying parameters and forgetting. Previously established discounting is reintroduced in Section 3.4, after reintroducing the standard fully probabilistic design in Section 3.2, together with its connection to Markov decision processes discussed in Section 3.3. The conclusion summarizes the contributions of the work as well as its main results.

Chapter 2

Bayesian Statistics

Since decisions can be influenced by all sorts of uncontrollable factors, decision processes occurring in nature can be extremely complex. Furthermore many of these factors are unknown to an outer observer. The outcomes of such processes can therefore seem to be random (otherwise called stochastic) rather than generated based on some fixed set of deterministic rules.

Standard way to deal with random processes is to give them a probabilistic description. Each of the process' possible outcomes is said to occur with a certain probability. It is therefore said that the outcomes follow a certain probability distribution.

Besides their characteristic functional form, probability distributions are also given by their specific parameters. The functional form is usually assumed to be known, e.g. normal. However, its parameters are seldom known and need to be estimated.

Classical methods such as maximum likelihood [33] can produce biased and imprecise estimates for some distributions [22]. On the other hand Bayesian statistics treats parameters as random variables and derives them by their probability distribution. This way the probability of the best parameter value is non-zero even when the distribution does not peak directly at this value. This makes Bayesian approach to parameter estimation more robust to bias. Moreover probabilistic description logically leads to addressed decision making.

2.1 Basic Relations

As probability distribution uniquely specifies its corresponding probability density function[1] [1], it is possible to model the parameters using densities instead. The purpose of this section is to give a brief overview of frequently utilized operations when dealing with them.

Random outcomes are realizations of certain measurable functions called random variables. All possible outcomes then form the range of the given random variable. Probability densities are defined on measurable sets containing the mentioned outcomes.

For a non-negative function p_X to be considered a probability density of the random variable X it needs to follow normalization condition given below

$$\int_{\mathcal{S}_x} p_X(\mathrm{x})\mathrm{dx} = 1,$$

where $\mathcal{S}_x$ is the range of X. This equation describes the case when X is a continuous random variable. Integration shall be replaced with summation in discrete case.

[1] Probability density function will be oftentimes referred to as probability density or simply density in the later text. Probability distribution will be also shortened and referred to as distribution.

Densities can be used to express probability of some event occurrence as well. Let $\mathcal{S}_y$ be a non-empty measurable subset of $\mathcal{S}_x$, i.e. $\mathcal{S}_y \subset \mathcal{S}_x$. Probability P of X falling in set $\mathcal{S}_y$ is then defined as

$$\mathrm{P}(X \in \mathcal{S}_y) = \int\limits_{\mathcal{S}_y} p_X(\mathrm{x})\mathrm{dx}, \tag{2.1}$$

It is yet to be stressed that random variables can also have multidimensional range. In this case it is possible to modify the probability density function as follows. Let $\mathcal{S}_x = \mathcal{S}_a \times \mathcal{S}_b$ be the Cartesian product of sets $\mathcal{S}_a$ and $\mathcal{S}_b$. Then the probability density function of X can be expressed as

$$p_X(\mathrm{x}) = p_{A,B}(\mathrm{a}, \mathrm{b}). \tag{2.2}$$

Function (2.2) is called *joint* probability density function of random variables A and B.

Having the knowledge of joint density $p_{A,B}$ it would be useful to determine the probability density p_B of B. Utilizing the above mentioned joint density and probability definition (2.1)

$$\mathrm{P}(B \in \mathcal{S}_y) = \mathrm{P}(A \in \mathcal{S}_a \wedge B \in \mathcal{S}_b) = \int\limits_{\mathcal{S}_a \times \mathcal{S}_y} p_{A,B}(\mathrm{a}, \mathrm{b})\mathrm{d}(\mathrm{a}, \mathrm{b}). \tag{2.3}$$

The first two expressions in (2.3) are equivalent as it is sure that a random event $A = \mathrm{a}$ for value a in $\mathcal{S}_\mathcal{A}$ will happen.

It has already been derived that the previous probability can also be written as in (2.1). Comparing these two equations yields

$$p_B(\mathrm{b}) = \int\limits_{\mathcal{S}_a} p_{A,B}(\mathrm{a,b})\mathrm{da}, \tag{2.4}$$

with the use of Fubini's theorem [9] on the integral in (2.3). Probability density (2.4) is called *marginal.* As usual, integration above needs to be replaced by summation for discrete random variables.

Another frequent task is to determine how the joint density changes after certain information about one of the two random variables is obtained. In other words, knowing $p_{A,B}$, what happens with the probability density once we have learned that B equals b. As there is no reason to change the expectations towards random variable A it would make sense for the new density to be proportional to the original one, i.e.

$$p_{A|B}(\mathrm{a}|\mathrm{b}) = \lambda p_{A,B}(\mathrm{a,b}), \tag{2.5}$$

for all a in $\mathcal{S}_a$ and the known b, λ being a suitable constant. Probability density $p_{A|B}$ is called *conditional* and it describes a two-dimensional variable $X = (A, B)$ when the uncertainty about one of its components (B in this case) is removed.

Once again for $p_{A|B}$ to be considered a probability density it needs to follow the normalization rule, therefore

$$1 = \int\limits_{\mathcal{S}_a} p_{A|B}(\mathrm{a}|\mathrm{b})\mathrm{da} = \int\limits_{\mathcal{S}_a} \lambda p_{A,B}(\mathrm{a,b})\mathrm{a} = \lambda p_B(\mathrm{b}),$$

where the last equation follows from the marginalization (2.4) and homogeneity of integrals. Proper value of the λ coefficient has therefore been derived. Its substitution back to (2.5) yields the complete formula valid for $p_B(\mathrm{b}) > 0$, i.e.

$$p_{A|B}(\mathrm{a}|\mathrm{b}) = \frac{p_{A,B}(\mathrm{a,b})}{p_B(\mathrm{b})}. \tag{2.6}$$

Let C be a new one-dimensional continuous random variable with range $\mathcal{S}_C$. It enters the relations (2.4) and (2.6) in the following manner

$$p_{B|C}(\mathrm{b}|\mathrm{c}) = \int\limits_{\mathcal{S}_a} p_{A,B|C}(\mathrm{a},\mathrm{b}|\mathrm{c})\mathrm{da},$$
$$p_{A,B|C}(\mathrm{a},\mathrm{b}|\mathrm{c}) = p_{A|B,C}(\mathrm{a}|\mathrm{b},\mathrm{c})p_{B|C}(\mathrm{b}|\mathrm{c}).$$

The latter of the two can be rewritten by exchanging random variable A for random variable B as

$$p_{A,B|C}(\mathrm{a},\mathrm{b}|\mathrm{c}) = p_{B|A,C}(\mathrm{b}|\mathrm{a},\mathrm{c})p_{A|C}(\mathrm{a}|\mathrm{c}).$$

Using the above formulas, probability density function conditional on both B and C can be contained in the form

$$p_{A|B,C}(\mathrm{a}|\mathrm{b},\mathrm{c}) = \frac{p_{B|A,C}(\mathrm{b}|\mathrm{a},\mathrm{c})p_{A|C}(\mathrm{a}|\mathrm{c})}{\int\limits_{\mathcal{S}_a} p_{B|A,C}(\mathrm{b}|\mathrm{a},\mathrm{c})p_{A|C}(\mathrm{a}|\mathrm{c})\mathrm{da}}.$$

The indexation by the respective random variable can make the notation quite cumbersome as seen in the previous paragraphs. For the sake of simplicity, random variables will from now on be identified solely by the arguments of their respective density. Using the declared new notation it is possible to rewrite the previous relation in a more readable manner as

$$p(\mathrm{a}|\mathrm{b},\mathrm{c}) = \frac{p(\mathrm{b}|\mathrm{a},\mathrm{c})p(\mathrm{a}|\mathrm{c})}{\int\limits_{\mathcal{S}_a} p(\mathrm{b}|\mathrm{a},\mathrm{c})p(\mathrm{a}|\mathrm{c})\mathrm{da}}. \tag{2.7}$$

This equation is known as Bayes' formula.

Densities in the previous paragraphs already dealt with multiple (specifically three) random variables at once. Now let there be n random variables $X_1, ..., X_n$, n being an arbitrary natural number. By repeatedly applying the conditional probability density definition (2.6) it is possible to decompose their joint density in the following way

$$p(\mathrm{x}_1, ..., \mathrm{x}_n) = \prod_{i=2}^{n} p(\mathrm{x}_i|\mathrm{x}_{i-1}, ..., \mathrm{x}_1)p(\mathrm{x}_1) \equiv \prod_{i=1}^{n} p(\mathrm{x}_i|\mathrm{x}_{i-1}, ..., \mathrm{x}_1), \tag{2.8}$$

where the second expression interprets $p(\mathrm{x}_1) = p(\mathrm{x}_1|\mathrm{x}_0)$. This decomposition is also known as chain rule and shall be made frequent use of in the following text.

2.2 System Model

Before any sort of estimation can take place, there need to be available data to base this estimation on. These data can be generally divided into inputs, or *actions*, and outputs, or partially observed *states*. Outputs equal to states in this work. Actions form the part of the data which is directly passed on to the system while states present the response of the system. This means they can be observed only passively and influenced purely through the chosen actions. All data observed up to time t, t included, will be denoted by

$$\mathrm{H}_t = (\mathrm{s}_t, \mathrm{a}_t, \mathrm{s}_{t-1}, \mathrm{a}_{t-1}, ..., \mathrm{s}_1, \mathrm{a}_1, \mathrm{s}_0),$$

where a_t denotes the action influencing the system at time t while s_t is the system state at the given time. This collection carries the information about the history of the system and describes how the system evolves in time.

Knowing the initial state of the system s_0, it is desirable to describe the next n steps of its evolution. Using chain rule (2.8) derived in the previous section this can be done by the use of a joint probability density p in the following way,

$$p(\mathrm{H}_n|\mathrm{s}_0) = \prod_{t=1}^{n} p(\mathrm{s}_t, \mathrm{a}_t|\mathrm{H}_{t-1}) = \prod_{t=1}^{n} p(\mathrm{s}_t|\mathrm{a}_t, \mathrm{H}_{t-1})p(\mathrm{a}_t|\mathrm{H}_{t-1}) \equiv \prod_{t=1}^{n} m(\mathrm{s}_t|\mathrm{a}_t, \mathrm{H}_{t-1})r(\mathrm{a}_t|\mathrm{H}_{t-1}). \quad (2.9)$$

Here, probability density $m(\mathrm{s}_t|\mathrm{a}_t, \mathrm{H}_{t-1})$ represents how the current state depends on the past history of the system. It *models* the state transitions dependent on the chosen actions and system history, hence its mnemonic notation m. Density $r(\mathrm{a}_t|\mathrm{H}_{t-1})$ is used to decide the next action once the history up to the point $t-1$ has been observed. It will be referred to as *decision rule* and will be denoted by r. A sequence of decision rules forms a *decision strategy.*

To construct a system model means to find the conditional probability densities introduced in (2.9) and explained in the previous paragraph. Such task becomes more complex once the model includes unknown parameters. In that case a direct use of $m(\mathrm{s}_t|\mathrm{a}_t, \mathrm{H}_{t-1}, \theta)$, θ being the unknown parameters with the values in $\mathcal{S}_\theta$, can only be made after the elimination of θ. Knowing range $\mathcal{S}_\theta$ the elimination is possible utilizing the marginalization formula (2.4)

$$m(\mathrm{s}_t|\mathrm{a}_t, \mathrm{H}_{t-1}) = \int_{\mathcal{S}_\theta} m(\mathrm{s}_t, \theta|\mathrm{a}_t, \mathrm{H}_{t-1}))\mathrm{d}\theta = \int_{\mathcal{S}_\theta} m(\mathrm{s}_t|\mathrm{a}_t, \mathrm{H}_{t-1}, \theta)p(\theta|\mathrm{a}_t, \mathrm{H}_{t-1})\mathrm{d}\theta, \quad (2.10)$$

where the new density $p(\theta|\mathrm{a}_t, \mathrm{H}_{t-1})$ describes the uncertainty of parameters θ. Relation (2.10) holds provided that θ is continuous. Otherwise the integration would be replaced by summation.

To conclude this section an exact specification of what exactly is meant by a system model needs to be provided. Quoting [27], any mathematical description which defines the set of conditional probability densities m for the time period required through a finite collection of parameters is called a system model. The choice of model defining the conditional probabilities featured in (2.9) and subsequent parameter estimation in (2.10) will be crucial for state prediction and decision making in general.

2.3 Parameter Estimation

After acquiring the action-state (input-output) data H_t up to some finite epoch t, the question to be answered is how to extract the information about the unknown parameter θ contained in the data, i.e. how to calculate probability density $p(\theta|\mathrm{H}_t)$. In Bayesian statistics the aim is not to find a point estimate $\hat{\theta}$ using one of the classical statistical methods but to evaluate the probability density as a whole. This makes sense as the main part of decision problems is to predict the next state, which is an operation that does not require the knowledge of any point estimates. Utilizing these would only result in lowering the decision accuracy.

The goal is to predict the next state purely based on the past history of the system. This can be achieved using marginalization and chain rule

$$m(\mathrm{s}_{t+1}|\mathrm{a}_{t+1}, \mathrm{H}_t) = \int_{\mathcal{S}_\theta} m(\mathrm{s}_{t+1}|\mathrm{a}_{t+1}, \mathrm{H}_t, \theta)p(\theta|\mathrm{a}_{t+1}, \mathrm{H}_t)\mathrm{d}\theta,$$

where the first factor in the integrand above is given by the model structure. Hence it is possible to move onto prediction once the density $p(\theta|\mathrm{a}_{t+1}, \mathrm{H}_t)$ is found.

In many real life applications the estimation is required in real time. This means that no fixed amount of data is available. More and more data is provided with each epoch. Were it not so, it would be possible to perform a one-shot estimation, meaning all available data would be processed at once.

Both of these cases can be treated at the same time in the following way. Provided probability density $p(\theta|\mathrm{H}_{t_1})$ and data H_t, where $t_1 < t$, find density $p(\theta|\mathrm{H}_t)$. Here, setting $t_1 = 0$ leads to one-shot estimation while setting $t_1 = t - 1$ for all t defines a recursive relation for real time estimation. Thus using Bayes formula (2.7) the sought after probability density can be obtained as

$$p(\theta|\mathrm{H}_t) = \frac{p(\mathrm{H}_{t_1+1}^t|\mathrm{H}_{t_1}, \theta)p(\theta|\mathrm{H}_{t_1})}{\int\limits_{\mathcal{S}_\theta} p(\mathrm{H}_{t_1+1}^t|\mathrm{H}_{t_1}, \theta)p(\theta|\mathrm{H}_{t_1})\mathrm{d}\theta}, \tag{2.11}$$

where $\mathrm{H}_{t_1+1}^t$ is a symbol denoting all state-action data acquired in a discrete time interval $\{t_1 + 1, t_1 + 2, ..., t\}$.

To solve the presented problem, $p(\mathrm{H}_{t_1+1}^t|\mathrm{H}_{t_1}, \theta)$ has to be expressed through known probability densities. This will be done in the most general way where the actions are dependent on the past states, therefore also possibly dependent on the parameters θ - it is said that actions are generated in a closed loop.

The first expression in the numerator of (2.11) gains the form

$$p(\mathrm{H}_{t_1+1}^t|\mathrm{H}_{t_1}, \theta) = \prod_{\tau=t_1+1}^{t} m(\mathrm{s}_\tau|\mathrm{a}_\tau, \mathrm{H}_{\tau-1}, \theta)r(\mathrm{a}_\tau|\mathrm{H}_{\tau-1}, \theta) \tag{2.12}$$

after the application of chain rule. First probability density $m(\mathrm{s}_\tau|\mathrm{a}_\tau, \mathrm{H}_{\tau-1}, \theta)$ included in the product is given by the model structure whereas the second probability density, namely $r(\mathrm{a}_\tau|\mathrm{H}_{\tau-1}, \theta)$, describes the action generation.

Let us now consider a situation when no more information about θ is available for opting a_τ than that given by data $\mathrm{H}_{\tau-1}$. This is the case when, for instance, the observer who observes the system and the decision maker who influences it are the same person. Since all knowledge available about the parameters is provided by the data, the last conditional density in (2.12) can be simplified as

$$r(\mathrm{a}_\tau|\mathrm{H}_{\tau-1}, \theta) = r(\mathrm{a}_\tau|\mathrm{H}_{\tau-1}). \tag{2.13}$$

Provided that this condition holds, let us study the joint probability density of parameters θ and actions a_τ. Once again using chain rule, this probability density can be expressed as

$$p(\theta, \mathrm{a}_\tau|H_{\tau-1}) = p(\theta|\mathrm{a}_\tau, \mathrm{H}_{\tau-1})r(\mathrm{a}_\tau|\mathrm{H}_{\tau-1}) = r(\mathrm{a}_\tau|\theta, \mathrm{H}_{\tau-1})p(\theta|\mathrm{H}_{\tau-1}). \tag{2.14}$$

The equivalent conditions (2.13) and (2.14) are called the *natural conditions of control*[2]. They can be interpreted in the following manner. Provided that data $\mathrm{H}_{\tau-1}$ are given, the parameters and the inputs are conditionally independent.

[2] Natural conditions of control do not always have to be met. For example, if the decision maker possesses more information about the parameters than the observer, the observer is able to acquire this additional information by watching the decision maker. Equality (2.13) therefore does not hold in this case. However, for the needs of this work further generalization will not be necessary and so the natural conditions of control will be assumed to hold at all times.

Utilizing the previous conditions, conditional probability density of the unknown parameters given the past history of the system $p(\theta|\mathrm{H}_t)$, see (2.11), can be formulated in the following simplified way. Firstly, the product (2.12) is substituted into (2.11). Since densities $r(\mathrm{a}_\tau|\mathrm{H}_{\tau-1},\theta)$ are conditionally independent of the parameters (2.13), they can be taken out of the integration in (2.11). They subsequently cancel out with the same probability densities substituted into the numerator giving the final formula

$$p(\theta|\mathrm{H}_t) = \frac{\prod\limits_{\tau=t_1+1}^{t} m(\mathrm{s}_\tau|\mathrm{a}_\tau,\mathrm{H}_{\tau-1},\theta)p(\theta|\mathrm{H}_{t_1})}{\int\limits_{\mathcal{S}_\theta}\prod\limits_{\tau=t_1+1}^{t} m(\mathrm{s}_\tau|\mathrm{a}_\tau,\mathrm{H}_{\tau-1},\theta)p(\theta|\mathrm{H}_{t_1})\mathrm{d}\theta}. \tag{2.15}$$

Knowing conditional density (2.15), it is possible to describe the behavior of the parameters. This equation answers the question regarding the information extraction contained in the data. In the following subsections this density will be expressed for the case of fixed and growing data.

2.3.1 Fixed Amount of Data

Setting t_1 to zero in (2.15), the formula becomes

$$p(\theta|\mathrm{H}_t) = \frac{\prod\limits_{\tau=1}^{t} m(\mathrm{s}_\tau|\mathrm{a}_\tau,\mathrm{H}_{\tau-1},\theta)p(\theta)}{\int\limits_{\mathcal{S}_\theta}\prod\limits_{\tau=1}^{t} m(\mathrm{s}_\tau|\mathrm{a}_\tau,\mathrm{H}_{\tau-1},\theta)p(\theta)\mathrm{d}\theta} \equiv \frac{L_t(\theta,\mathrm{H}_t)p(\theta)}{\int\limits_{\mathcal{S}_\theta} L_t(\theta,\mathrm{H}_t)p(\theta)\mathrm{d}\theta}. \tag{2.16}$$

Here $L_t(\theta,\mathrm{H}_t)$ stands for the significant part of the likelihood function. The term significant part refers to the fact that factors $r(\mathrm{a}_\tau|\mathrm{H}_{\tau-1},\theta) \equiv r(\mathrm{a}_\tau|\mathrm{H}_{\tau-1})$ are absent for τ lesser than t.

The equality $t_1 = 0$ turned the conditional probability density $p(\theta|\mathrm{H}_{t_1})$ in (2.15) into prior probability density $p(\theta)$. It is therefore possible to interpret the formula (2.16) as a correction of prior information by objective data H_t.

2.3.2 Growing Data

Real time estimation is obtained by setting t_1 in (2.11) to $t-1$. This yields

$$p(\theta|\mathrm{H}_t) = \frac{m(\mathrm{s}_t|\mathrm{a}_t,\mathrm{H}_{t-1},\theta)p(\theta|\mathrm{H}_{t-1})}{\int\limits_{\mathcal{S}_\theta} m(\mathrm{s}_t|\mathrm{a}_t,\mathrm{H}_{t-1},\theta)p(\theta|\mathrm{H}_{t-1})\mathrm{d}\theta}.$$

Here the denominator is actually a formula for deriving the probability density of the state in the next epoch, specifically

$$\int\limits_{\mathcal{S}_\theta} m(\mathrm{s}_t|\mathrm{a}_t,\mathrm{H}_{t-1},\theta)p(\theta|\mathrm{H}_{t-1})\mathrm{d}\theta = m(\mathrm{s}_t|\mathrm{a}_t,\mathrm{H}_{t-1}). \tag{2.17}$$

The next state is therefore predicted based on data H_{t-1} from the previous epoch. Hence once a new action is determined, a new state is observed and a new data pair $(\mathrm{s}_t,\mathrm{a}_t)$ is received. The set of all data is afterwards expanded to epoch t, i.e. H_t is now available, and used to acquire an updated conditional density of parameters

$$p(\theta|\mathrm{H}_t) = \frac{m(\mathrm{a}_t|\mathrm{a}_t,\mathrm{H}_{t-1},\theta)p(\theta|\mathrm{H}_{t-1})}{m(\mathrm{s}_t|\mathrm{a}_t,\mathrm{H}_{t-1})}. \tag{2.18}$$

Such recalculation for each time step t can be computationally demanding when performing numerical simulations. Simplification in form of posterior density $p(\theta|\mathrm{H}_t)$ would therefore come in handy. A certain simplification can be done by replacing the still-growing data H_t by a suitable sufficient statistic. The topic of sufficient statistics will be covered later in section 2.4.

Equation (2.18) describes how to proceed in order to update posterior densities of the unknown parameters θ. However, the decision maker's focus often lays purely on state prediction without any interest in the parameters of the obtained density. In such case it would seem reasonable to omit all forms of parameter estimation and focus directly on predicting the next states resulting in saved computation time and memory.

This is possible using an altered formula (2.11) where t_1 is set[3] to be equal to some positive time t_0, i.e. $t_1 = t_0 > 0$. The formula then gets the form

$$m(\mathrm{s}_t|\mathrm{a}_t, \mathrm{H}_{t-1}) = \frac{\int_{\mathcal{S}_\theta} m(\mathrm{s}_t|\mathrm{a}_t, \mathrm{H}_{t-1}, \theta) L_{t-1}(\theta, \mathrm{H}_{t-1}) p(\theta|\mathrm{H}_{t_0}) \mathrm{d}\theta}{\int_{\mathcal{S}_\theta} L_{t-1}(\theta, \mathrm{H}_{t-1}) p(\theta|\mathrm{H}_{t_0}) \mathrm{d}\theta}. \tag{2.19}$$

As a reminder, the significant part of the likelihood function denoted by L_{t-1} defined in subsection 2.3.1 has the form

$$L_{t-1}(\theta, \mathrm{H}_{t-1}) = \prod_{\tau=t_0+1}^{t-1} m(\mathrm{s}_\tau|\mathrm{a}_\tau, \mathrm{H}_{\tau-1}, \theta).$$

Relating to this definition a recursive formula of the significant part of the likelihood function can be obtained as

$$L_t(\theta, \mathrm{H}_t) = m(\mathrm{s}_t|\mathrm{a}_t, \mathrm{H}_{t-1}, \theta) L_{t-1}(\theta, \mathrm{H}_{t-1}).$$

Utilizing the previous knowledge and denoting the integral in the denominator of (2.19) as $I_{t-1}(\mathrm{H}_{t-1})$, the recursive formula for the conditional density of the next state can be formulated as

$$m(\mathrm{s}_t|\mathrm{a}_t, \mathrm{H}_{t-1}) = \frac{I_t(\mathrm{H}_t)}{I_{t-1}(\mathrm{H}_{t-1})}.$$

However, as s_t and a_t are still unknown, it is necessary to mention that the integral in the numerator is a function of these two random variables as well. Rewritten, the final prediction formula of the states in the next epoch in real-time estimation is given by

$$m(\mathrm{s}_t|\mathrm{a}_t, \mathrm{H}_{t-1}) = \frac{I_t(\mathrm{s}_t, \mathrm{a}_t, \mathrm{H}_{t-1})}{I_{t-1}(\mathrm{H}_{t-1})}. \tag{2.20}$$

In other words, in order to perform real-time estimation while eliminating parameters, the integral I_t has to be expressible as a function of the most recent state-action pair.

2.3.3 Prior $p(\theta|\mathrm{H}_0)$

As the integral in the denominator of (2.18) does not depend on parameters, recursive formula of the conditional parameter density can be expressed using proportionality symbol, i.e.

[3] This is the standard process when the densities $m(\mathrm{s}_t|\mathrm{a}_t, \mathrm{H}_{t-1}, \theta)$ are not defined by the model structure right from the beginning of the process but from some time $t_0 > 0$. The topic is otherwise known as the problem of initial data, see [27].

$$p(\theta|\mathrm{H}_t) \propto m(\mathrm{s}_t|\mathrm{a}_t, \mathrm{H}_{t-1}, \theta)p(\theta|\mathrm{H}_{t-1}) \propto ... \propto \prod_{\tau=1}^{t} m(\mathrm{s}_\tau|\mathrm{a}_\tau, \mathrm{H}_{\tau-1}, \theta)p(\theta|\mathrm{H}_0). \tag{2.21}$$

Here the expression H_0 contains not only the initial state of the system s_0 but also any possible prior knowledge about the parameters. Densities $p(\theta|\mathrm{H}_0)$ are taken to be close to uniform in case of no available prior knowledge.

Up until now probability densities $m(\mathrm{s}_t|\mathrm{a}_t, \mathrm{H}_{t-1}, \theta)$ were considered to depend on all available system history. In many decision processes, however, decisions can be influenced only by the means of the most recent knowledge of the system - its past state and action opted in that state. Such system can be described using Markov chain model below

$$m(\mathrm{s}_\tau|\mathrm{a}_\tau, \mathrm{H}_{\tau-1}, \theta) = m(\mathrm{s}_\tau|\mathrm{a}_\tau, \mathrm{s}_{\tau-1}, \theta) \equiv \theta_{\mathrm{s}_\tau|\mathrm{a}_\tau,\mathrm{s}_{\tau-1}}.$$

The first equality above sets the knowledge of the last observed state s_{t-1} equal to the knowledge of the whole history of how the system evolved in time. This is known as Markov property. It says that the system evolution is independent of its history. Unless mentioned otherwise, Markov property will be considered to hold for both the system model and the decision rules at all times.

The number of unknown values introduced by the last equality is finite for discrete case[4]. Let the cardinality of the state set $\mathcal{S}_{\mathcal{S}}$ be an arbitrary natural number n greater than two[5]. Since $\theta_{\mathrm{s}_\tau|\mathrm{a}_\tau,\mathrm{s}_{\tau-1}}$ denote values of probability densities, they need to satisfy the non-negativity and normalization conditions

$$\theta_{\mathrm{s}_\tau|\mathrm{a}_\tau,\mathrm{s}_{\tau-1}} \geq 0, \qquad \sum_{\mathrm{s}_\tau=1}^{n} \theta_{\mathrm{s}_\tau|\mathrm{a}_\tau,\mathrm{s}_{\tau-1}} = 1, \tag{2.22}$$

for any $\mathrm{s}_\tau, \mathrm{s}_{\tau-1}$ in $\mathcal{S}_{\mathcal{S}}$ and a_τ in $\mathcal{S}_{\mathcal{A}}$. The finite, possibly zero, number of a-values is j.

Using this parametrisation it is possible to rewrite conditional densities $p(\theta|\mathrm{H}_t)$ from (2.21) in the following manner,

$$\begin{aligned} p(\theta|\mathrm{H}_t) &\propto \prod_{\tau=1}^{t} \theta_{\mathrm{s}_\tau|\mathrm{a}_\tau,\mathrm{s}_{\tau-1}} p(\theta|\mathrm{H}_0) = \prod_{\tau=1}^{t} \prod_{\mathrm{s}} \prod_{\mathrm{a}} \prod_{\mathrm{s}'} \theta_{\mathrm{s}|\mathrm{a},\mathrm{s}'}^{\delta(\mathrm{s}_\tau,\mathrm{s})\delta(\mathrm{a}_\tau,\mathrm{a})\delta(\mathrm{s}_{\tau-1},\mathrm{s}')} p(\theta|\mathrm{H}_0) \\ &\equiv \prod_{\mathrm{s}} \prod_{\mathrm{a}} \prod_{\mathrm{s}'} \theta_{\mathrm{s}|\mathrm{a},\mathrm{s}'}^{\Delta_{t;\mathrm{s}|\mathrm{a},\mathrm{s}'}} p(\theta|\mathrm{H}_0), \end{aligned}$$

where exponent $\Delta_{t;\mathrm{s}|\mathrm{a},\mathrm{s}'}$ denotes the sum of all Kronecker's delta functions above over all time epochs up to time t, i.e.

$$\Delta_{t;\mathrm{s}|\mathrm{a},\mathrm{s}'} \equiv \sum_{\tau=1}^{t} \delta(\mathrm{s}_\tau, \mathrm{a})\delta(\mathrm{a}_\tau, \mathrm{a})\delta(\mathrm{a}_{\tau-1}, \mathrm{s}').$$

[4] Numerical simulations presented later in the work will be restricted to discrete case as well.

[5] Strict inequality is used only so that the integral calculation in (2.26) would not automatically reduce to Beta function after taking the latter two products out of the integration and using the introduced x-substitution. Integral calculation is therefore more general. Discarding generality, the state set cardinality may very well be equal to two as well.

It would make sense for the initial density $p(\theta|\mathrm{H}_0)$ to be of the same form as the conditional probability density $p(\theta|\mathrm{H}_t)$. For this purpose a new prior statistic $\nu_{0;\mathrm{s}|\mathrm{a},\mathrm{s}'}$ is introduced and the initial density above is then expressed as

$$p(\theta|\mathrm{H}_0) \propto \prod_{\mathrm{s}} \prod_{\mathrm{a}} \prod_{\mathrm{s}'} \theta_{\mathrm{s}|\mathrm{a},\mathrm{s}'}^{\nu_{0;\mathrm{s}|\mathrm{a},\mathrm{s}'}-1}. \tag{2.23}$$

The meaning of the subtraction of one will become self-explanatory at the end of the section, see (2.29) later.

Having determined the exact form of the initial density $p(\theta|\mathrm{H}_0)$, density $p(\theta|\mathrm{H}_t)$ can now be expressed as

$$p(\theta|\mathrm{H}_t) \propto \prod_{\mathrm{s}} \prod_{\mathrm{a}} \prod_{\mathrm{s}'} \theta_{\mathrm{s}|\mathrm{a},\mathrm{s}'}^{\Delta_{t;\mathrm{s}|\mathrm{a},\mathrm{s}'}+\nu_{0;\mathrm{s}|\mathrm{a},\mathrm{s}'}-1} \equiv \prod_{\mathrm{s}} \prod_{\mathrm{a}} \prod_{\mathrm{s}'} \theta_{\mathrm{s}|\mathrm{a},\mathrm{s}'}^{\nu_{t;\mathrm{s}|\mathrm{a},\mathrm{s}'}-1}. \tag{2.24}$$

The exponent in (2.24) can be found recursively according to the rule

$$\begin{aligned}\nu_{t;\mathrm{s}|\mathrm{a},\mathrm{s}'} &= \Delta_{t;\mathrm{s}|\mathrm{a},\mathrm{s}'} + \nu_{0;\mathrm{s}|\mathrm{a},\mathrm{s}'} = \Delta_{t-1;\mathrm{s}|\mathrm{a},\mathrm{s}'} + \nu_{0;\mathrm{s}|\mathrm{a},\mathrm{s}'} + \delta(\mathrm{s}_t,\mathrm{s})\delta(\mathrm{a}_t,\mathrm{a})\delta(\mathrm{s}_{t-1},\mathrm{s}') \\ &= \nu_{t-1;\mathrm{s}|\mathrm{a},\mathrm{s}'} + \delta(\mathrm{s}_t,\mathrm{s})\delta(\mathrm{a}_t,\mathrm{a})\delta(\mathrm{s}_{t-1},\mathrm{s}'). \end{aligned} \tag{2.25}$$

Thus for the given state-action-state triplet the current value of the exponent is either equal to the exponent from the previous epoch or is greater by one. This is true for any given time.

The last expression in (2.24) is precisely the integrand which enters the integration in (2.17). Thanks to the normalization condition in (2.22), the last parameter $\theta_{n|\mathrm{a},\mathrm{s}'}$ can be expressed through the previous $n-1$ parameters as

$$\theta_{n|\mathrm{a},\mathrm{s}'} = 1 - \sum_{\mathrm{s}=1}^{n-1} \theta_{\mathrm{s}|\mathrm{a},\mathrm{s}'},$$

for any available action a and state s′. Since neither the integration nor multiplication run over the time variable, it will be omitted from further notation to somewhat improve its readability. All of the symbols below are thought to represent their current value at time t.

Utilizing the normalization condition and the new notation, integral (2.17) then gets the form

$$\underset{\substack{\sum_{\mathrm{s}=1}^{n-1}\theta_{\mathrm{s}|\mathrm{a},\mathrm{s}'}\leq 1 \\ \theta_{\mathrm{s}|\mathrm{a},\mathrm{s}'}\geq 0}}{\int\cdots\int} \prod_{\mathrm{s}=1}^{n-1} \prod_{\mathrm{a}=1}^{j} \prod_{\mathrm{s}'=1}^{n} \theta_{\mathrm{s}|\mathrm{a},\mathrm{s}'}^{\nu_{\mathrm{s}|\mathrm{a},\mathrm{s}'}-1} \left(1 - \sum_{\mathrm{s}=1}^{n-1} \theta_{\mathrm{s}|\mathrm{a},\mathrm{s}'}\right)^{\nu_{n|\mathrm{a},\mathrm{s}'}-1} \mathrm{d}\theta. \tag{2.26}$$

The first state product upper bound only goes up to $n-1$ as the last θ symbol has been separated from the rest of the product and rewritten with the use of the remaining symbols. θ symbol in the differential then denotes all $\theta_{\mathrm{s}|\mathrm{a},\mathrm{s}'}$ variables for all $(n-1) \times j \times (n-1)$ state-action-state combinations.

By further separating the last but one θ from both the product and the summation hints the recursive approach towards the solution of the integration task. After the rest of the sum is taken out of the parentheses a new substitution

$$x_{n-1} \equiv \frac{\theta_{n-1|\mathrm{a},\mathrm{s}'}}{1 - \sum_{\mathrm{s}=1}^{n-2} \theta_{\mathrm{s}|\mathrm{a},\mathrm{s}'}}, \tag{2.27}$$

can take place. Integral (2.26) then transforms into

$$\int\cdots\int_{\substack{\sum_{\mathrm{s}=1}^{n-2}\theta_{\mathrm{s|a,s'}}\le 1\\ \theta_{\mathrm{s|a,s'}}\ge 0}}\int_0^1 \prod_{\mathrm{s}=1}^{n-2}\prod_{\mathrm{a}=1}^{j}\prod_{\mathrm{s}'=1}^{n}\theta_{\mathrm{s|a,s'}}^{\nu_{\mathrm{s|a,s'}}-1}\,x_{n-1}^{\nu_{n-1\mathrm{|a,s'}}-1}\cdots$$

$$\cdots(1-x_{n-1})^{\nu_{n\mathrm{|a,s'}}-1}\left(1-\theta_{n-2\mathrm{|a,s'}}-\sum_{\mathrm{s}=1}^{n-3}\theta_{\mathrm{s|a,s'}}\right)^{\nu_{n\mathrm{|a,s'}}+\nu_{n-1\mathrm{|a,s'}}-1}\mathrm{d}\mathbf{x}_{n-1}\mathrm{d}\tilde{\theta}.$$

Symbol $\tilde{\theta}$ represents the remaining $(n-2)\times j\times(n-2)$ θ-variables whereas $\mathbf{x}_{n-1}$ in bold stands for variables x_{n-1} for all possible actions a and states s', see (2.27). Separating the last element of the sum in the boundary condition from the rest can serve as a hint for finding the new integration limits. Separation and subsequent division leads to

$$1\ge\theta_{n-1\mathrm{|a,s'}}+\sum_{\mathrm{s}=1}^{n-2}\theta_{\mathrm{s|a,s'}}\ge 0 \iff 1\ge\frac{\theta_{n-1\mathrm{|a,s'}}}{1-\sum_{\mathrm{s}=1}^{n-2}\theta_{\mathrm{s|a,s'}}}=x_{n-1}\ge 0.$$

Recursively continuing this way, the final form of integral (2.26) can be obtained as

$$\int_0^1\cdots\int_0^1\prod_{\mathrm{s}=1}^{n-1}\prod_{\mathrm{a}=1}^{j}\prod_{\mathrm{s}'=1}^{n}x_{\mathrm{s}}^{\nu_{\mathrm{s|a,s'}}-1}(1-x_{\mathrm{s}})^{\sum_{\mathrm{k}=1}^{n-\mathrm{s}+1}\nu_{\mathrm{k|a,s'}}-n-\mathrm{s}-2}\mathrm{d}\mathbf{x}_{n-1}\cdots\mathrm{d}\mathbf{x}_1, \tag{2.28}$$

which is the integral form of multivariate beta function $\mathrm{Beta}(\nu_{1|1,1},...,\nu_{n|j,n})\equiv\mathrm{Beta}(\nu)$. The final parameter estimation model (2.18) therefore acquires the following form and follows Dirichlet's distribution thanks to the minus one subtraction, i.e.

$$p(\theta|\mathrm{H}_t)=\prod_{\mathrm{s}=1}^{n-1}\prod_{\mathrm{a}=1}^{j}\prod_{\mathrm{s}'=1}^{n}\frac{m(\mathrm{s|a,s'},\theta)^{\nu_{t;\mathrm{s|a,s'}}-1}}{\mathrm{Beta}(\nu)}\sim\mathrm{Dir}(\nu). \tag{2.29}$$

2.4 Conjugate Families of Distributions

For the derived theory to be of real practical use in numerical simulations, the data need to be compressed in such way that the information stored inside stays complete whereas the dimension of the problem is reduced to some fixed natural number. This can be done through the means of a *sufficient statistic*. Statistic in general is a notion denoting any quantity computed from the available data values. Statistic $T_t\equiv T_t(\mathrm{H}_t)$ is said to be sufficient for a certain random variable X when it fulfills the relation

$$p(\mathrm{x}|\mathrm{H}_t)=p(\mathrm{x}|T_t). \tag{2.30}$$

That way all the information provided by the growing data can be represented by a fixed number of values. In other words, all the information about random variable X is contained in the statistic T_t. Storing the entire data vector H_t which is generally of a high dimension thus becomes redundant.

Property (2.30) intuitively does not need to hold for all probability densities. Let T_t be a sufficient statistic for some family of distributions $\{p(\mathrm{H}_t|\theta)|\theta\in\mathcal{S}_\theta\}$, where θ are the unknown parameters with a certain distribution which will also be denoted by p for simplicity. Following the Bayes' rule the formula for the posterior probability density of these parameters gets the form

$$p(\theta|\mathrm{H}_t) = \frac{p(\mathrm{H}_t|\theta)p(\theta)}{\int\limits_{\mathcal{S}_\theta} p(\mathrm{H}_t|\tilde{\theta})p(\tilde{\theta})\mathrm{d}\tilde{\theta}} \equiv q(T_t(\mathrm{H}_t), \theta), \tag{2.31}$$

where q is a symbol for the new reference probability density. The existence of sufficient statistic T_t was utilized in the final equality of (2.31). It is now possible to isolate the likelihood function from this expression as

$$p(\mathrm{H}_t|\theta) = \frac{q(T_t(\mathrm{H}_t), \theta)}{p(\theta)} \int\limits_{\mathcal{S}_\theta} p(\mathrm{H}_t|\tilde{\theta})p(\tilde{\theta})\mathrm{d}\tilde{\theta} \equiv R(T_t(\mathrm{H}_t), \theta)U(\mathrm{H}_t). \tag{2.32}$$

Expression (2.32) says that the probability density functions from the family introduced above can be factored into a form, where the part of the function which depends on the data in a way other than through the sufficient statistic, can be isolated as a stand-alone function U of H_t. Furthermore, this statement can be reversed. Substitution of (2.32) into (2.31) yields

$$p(\theta|\mathrm{H}_t) = \frac{U(\mathrm{H}_t)R(T_t(\mathrm{H}_t), \theta)p(\theta)}{\int\limits_{\mathcal{S}_\theta} U(\mathrm{H}_t)R(T_t(\mathrm{h}_t), \tilde{\theta})p(\tilde{\theta})\mathrm{d}\tilde{\theta}}. \tag{2.33}$$

Here the function U is independent of parameters $\tilde{\theta}$ and can therefore be taken outside of the integrand in the denominator and canceled out with the same function in the numerator. The posterior probability density (2.33) then becomes dependent on data only through the means of statistic T_t which makes it sufficient.

Therefore T_t can be identified as sufficient for a certain family of distributions when its probability density functions can be factored according to (2.32). This statement is otherwise known as factorization theorem.

Understanding the term sufficient statistic naturally leads to the question of finding its form for data which follow a certain probability distribution. For instance, arithmetic mean and standard deviance expressed using arithmetic mean multiplied by its corresponding degrees of freedom, i.e.

$$\frac{1}{n}\sum_{i=1}^{n} \mathrm{x}_i \qquad \text{and} \qquad \sum_{i=1}^{n}(\mathrm{x}_i - \bar{\mathrm{x}}_n)^2,$$

$\bar{\mathrm{x}}_n$ being arithmetic mean of $\mathrm{x}_1, ..., \mathrm{x}_n$, form a two-dimensional sufficient statistic of finite dimension for Gaussian family of distributions.

In general, not all sufficient statistics need to be of fixed dimension. In order to find distributions which will be usable for computations in real time for large samples of data, the set of all sufficient statistics needs to be limited to the ones which can be updated in real time similarly to the example above. Factorization theorem provided a way of checking if a sufficient statistic exists for a given distribution. Now we need a tool to decide whether the dimension of this distribution's statistic stays fixed as the data grows. Such tool can be found in the theory of *conjugate families of distributions.*

A prior probability density is said to come from a conjugated family of distributions when its posterior probability density also comes from this family. An important point to be made is that this family needs to be rich enough to provide enough densities to properly represent the prior distribution of the unknown parameters.

A theorem independently formulated by Darmois, Koopman [17] and Pitman [28] claims that provided that the support of a given probability density does not change with its varying parameters then there exists a sufficient statistic of fixed dimension if and only if[6] the probability density comes from an exponential family of distributions. System $\{p(\mathrm{H}_t|\theta)|\theta \in \mathcal{S}_\theta\}$ of densities is considered an exponential family when its elements have the form

$$p(\mathrm{H}_t|\theta) = c(\theta)h(\mathrm{H}_t)\mathrm{e}^{T_t(\mathrm{H}_t)Q(\theta)}. \tag{2.34}$$

Here $c, h \geq 0$, T and Q are known functions of respective arguments. The multivariate T, Q are combined via dot-product. In other words, functions from the exponential family can be factored in a way so that the data in the exponent of the exponential are present only in the form of a sufficient statistic.

The above exposition puts light on the fact that the most generally parameterized Markov chain model treated in previous section possesses a sufficient statistic of a fixed dimension. Indeed, simple manipulation allows to further reshape the model in the following manner,

$$m(\mathrm{s}_t|\mathrm{a}_t, \mathrm{s}_{t-1}, \theta) \equiv \prod_{\mathrm{s},\mathrm{a},\mathrm{s}'} \theta_{\mathrm{s}|\mathrm{a},\mathrm{s}'}^{\delta(\mathrm{s},\mathrm{s}_t)\delta(\mathrm{a},\mathrm{a}_t)\delta(\mathrm{s}',\mathrm{s}_{t-1})} = \exp\left(\sum_{\mathrm{s},\mathrm{a},\mathrm{s}'} \delta(\mathrm{s},\mathrm{s}_t)\delta(\mathrm{a},\mathrm{a}_t)\delta(\mathrm{s}',\mathrm{s}_{t-1})\ln(\theta_{\mathrm{s}|\mathrm{a},\mathrm{s}'})\right).$$

Comparing this result with (2.34), specific forms of the functions from the exponential form can be obtained, namely

$$\begin{aligned} c(\theta) &= 1, & Q(\theta) &= \ln(\theta), \\ h(\mathrm{H}_t) &= 1, & T_t(\mathrm{H}_t) &= \delta(\mathrm{H}_t), \end{aligned}$$

with the sum representing the dot product of the parametric function with the Kronecker's sufficient statistic given on a specific system history H_t.

2.5 Forgetting

In previous text it was derived that the propagation of the parameter density has the following structure,

$$p(\theta|\mathrm{H}_t) \propto m(\mathrm{s}_t|\mathrm{a}_t, \mathrm{H}_{t-1}, \theta)p(\theta|\mathrm{H}_{t-1}).$$

This formula sufficiently describes a scenario where no changes in parameter values are expected to be observed.

A problem arises, however, when parameter values vary in time. Generally a model describing these changes is necessary. Using the definition of conditional probability density as well as marginalization rule, a new relation describing this case can be obtained

$$p(\theta_{t+1}|\mathrm{H}_t) = \int_{\mathcal{S}_\theta} p(\theta_{t+1}|\theta_t, \mathrm{H}_t)p(\theta_t|\mathrm{H}_t)\mathrm{d}\theta_t, \tag{2.35}$$

with $p(\theta_{t+1}|\theta_t, \mathrm{H}_t)$ modelling the evolution of parameter estimates, rather often being an unknown probability density. Time indexation is now included to emphasize possible value variations. History variable H_t in equation (2.35) stays constant, i.e. no new information has yet been gained, while the parameters have evolved from θ_t to θ_{t+1}.

[6] Sufficient statistic of fixed dimension also exists for the family of uniform distributions. However, densities from this family do not fulfill the regularity condition of a bound support regardless of the parameter values.

A new problem arises. Without the knowledge of conditional probability density $p(\theta_{t+1}|\theta_t, \mathrm{H}_t)$, the exact means to provide data-based parameter estimation for the next epoch cannot be provided. Hence the standard probability calculus falls short and the task at hand needs to be solved by different means.

Two most extreme hypotheses about the fate of the probability density (2.35) include

$$p(\theta_{t+1}|\mathrm{H}_t) = \begin{cases} p(\theta_t|\mathrm{H}_t) & \text{in optimistic case,} \\ p(\theta_t|\mathrm{H}_0) & \text{in pessimistic case.} \end{cases} \tag{2.36}$$

Symbolic notation $p(\theta_{t+1}|\mathrm{H}_t) = p(\theta_t|\mathrm{H}_t)$ represents the hypothesis, that the parameters in the next time epoch remain constant, i.e. $p(\theta_{t+1} = \theta|\mathrm{H}_t) = p(\theta_t = \theta|\mathrm{H}_t)$. This refers to the optimistic parameter evolution outcome as no parameter change has occurred. The same notational rule applies to the second row of (2.36). This pessimistic case refers to a situation when the parameter variation is so high it becomes impossible to predict the future values. Return to prior information is then inevitable.

Let us denote the set of all available parameter probability densities on $\mathcal{S}_\theta$ as $\mathcal{S}_p$. One of the hypotheses (2.36) is chosen based on the means of a so-called *forgetting operator* F. It yields the final parameter model $p(\theta_{t+1}|\mathrm{H}_t)$ by combining hypotheses (2.36) in the following manner,

$$p(\theta_{t+1}|\mathrm{H}_t) = \mathrm{F}[p(\theta_t|\mathrm{H}_t), p(\theta_t|\mathrm{H}_0)] \equiv \operatorname*{argmin}_{p\in\mathcal{S}_p}[\lambda_t \mathrm{D}(p, p(\theta_t|\mathrm{H}_t)) + (1-\lambda_t)\mathrm{D}(p, p(\theta_t|\mathrm{H}_0))]. \tag{2.37}$$

In the last expression above, λ_t and $1-\lambda_t$ are non-negative weights specified in advance. They denote probabilities of the respective hypotheses. Symbol $\mathrm{D}(\cdot,\cdot)$ stands for a known divergence measure of two probability densities. Different divergence measures can be found for instance in [3]. Forgetting operator defined this way then obtains a solution which minimizes expectation of a loss function given by the said divergence.

Remark 1. Operator construction given by (2.37) is said to be based on *Bayes principle.* There are other options how to define F such as barycentre principle or minimum distance principle. These can be both expressed in a way which leads to the Bayes expression above, see [18]. It is thus sufficient to focus on the introduced definition without any loss of generality.

For the construction to be complete, it is necessary to define the divergence D. It can be shown that one such plausible choice is given by *Kullback-Leibler divergence* [19], also known as relative entropy,

$$\mathrm{D}(p,q) = \int_{\mathcal{S}_\theta} p(\theta) \ln\left(\frac{p(\theta)}{q(\theta)}\right) \mathrm{d}\theta, \tag{2.38}$$

where p and q are both probability densities defined on set $\mathcal{S}_\theta$, q being strictly positive. It was proven in [18] that this divergence pick leads to a unique solution of (2.35) in the form

$$p(\theta_{t+1}|\mathrm{H}_t) \propto [p(\theta_t|\mathrm{H}_t)]^{\lambda_t}[p(\theta_t|\mathrm{H}_0)]^{1-\lambda_t}, \tag{2.39}$$

provided that densities $p(\theta_t|\mathrm{H}_t)$ and $p(\theta_t|\mathrm{H}_0)$ are not mutually orthogonal, meaning their product is non-zero almost everywhere.

Remark 2. Solution (2.39) can also be generalized for a higher number of hypotheses n, meaning n greater than two. If this is a matter of interest for the reader, he is referred to [13]. Here, minimum expected relative entropy is introduced to solve what is called meta-decision tasks, where each hypothesis is assigned its own probability density.

By using the conjugated Dirichlet's distribution of parameters (2.29) discussed in the previous chapter, final form of probability density (2.35) can be expressed as

$$\begin{aligned} p(\theta_{t+1}|\mathrm{H}_t) &\propto [p(\theta_t|\mathrm{H}_t)]^{\lambda_t}[p(\theta_t|\mathrm{H}_0)]^{1-\lambda_t} \\ &\propto \prod_{s=1}^{n-1}\prod_{a=1}^{j}\prod_{s'=1}^{n} m(\mathrm{s}|\mathrm{a},\mathrm{s}',\theta_{t+1})^{\lambda_t\nu_{t;\mathrm{s}|\mathrm{a},\mathrm{s}'}+(1-\lambda_t)\nu_{0;\mathrm{s}|\mathrm{a},\mathrm{s}'}-1}, \end{aligned} \tag{2.40}$$

recalling that the exponent $\nu_{0;\mathrm{s}|\mathrm{a},\mathrm{s}'}$ refers to the probability density $p(\theta_t|\mathrm{H}_0)$ as all other information was discarded, or *forgotten*, in its case.

Constants λ_t and $1-\lambda_t$ were said to be probabilities of the two hypotheses (2.36). Their values are therefore to lay somewhere in the interval $[0,1]$. Raising parameter densities to the power lesser than one can be interpreted as their flattening, i.e. lowering the importance of accumulated knowledge. Present outcomes are thus regarded as more important. The same logic applies to future outcomes in discounting tasks, which will be covered later on.

Chapter 3

Fully Probabilistic Design

In everyday decision making the goal is to make choices which lead to the most favorable results. In order to do this, it is necessary to know how the system evolves in time given different states and actions to influence it. In other words, it is necessary to know the system dynamics.

Unveiling it was the aim of the previous chapter. Now that the question of system description is off the table it is possible to move on to the task of identifying the most profitable actions for each step of a decision process. Markov decision processes [31] present a powerful tool for solving decision-making tasks. The core of the approach is based on maximizing an expected reward function or, alternatively, minimizing an expected loss function.

Let the said loss function be denoted as $L(\mathrm{s}_t, \mathrm{a}_t, \mathrm{s}_{t-1})$ for an arbitrary state-action-state triplet. Not including more distant history of the system than the three given arguments is justified when future system evolution is independent of its history. This fact is referred to as Markov property.

The state-action-state triplet in $L(\mathrm{s}_t, \mathrm{a}_t, \mathrm{s}_{t-1})$ with the smallest addition to the loss function for a given decision *epoch* t specifies the action a_t to be used for the said epoch granting optimal system evolution. A visualization scheming this process can be seen in Figure 3.1 below.

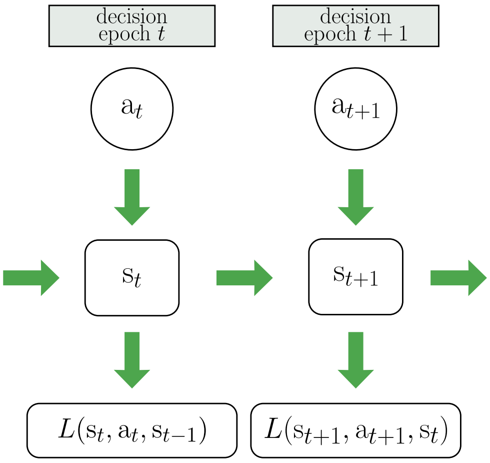

decision epoch t
decision epoch $t+1$
a_t
a_{t+1}
s_t
s_{t+1}
$L(\mathrm{s}_t, \mathrm{a}_t, \mathrm{s}_{t-1})$
$L(\mathrm{s}_{t+1}, \mathrm{a}_{t+1}, \mathrm{s}_t)$

Figure 3.1: System evolution in time inducing loss L in each epoch.

Perhaps one of the biggest challenges in resolving decision tasks is uncertainty. Virtually in all real-life applications the real system model remains unknown and its identification[1], see Section 2.3, is only a more or less exact representation of the true system. Decision tasks using this incomplete information will always be influenced by its imprecision.

One of the driving forces of the research conduced in the span of this work from [24] up until now was to investigate whether it would be possible to somehow decrease this modeling error. The most straightforward way would be improving the model quality by making it as close to the real model as possible. That is, obtaining parameter estimates which best represent the system dynamics.

Generally, the more is known about how the system transitions between different states in time, the better the estimates. In order to acquire sufficient knowledge about these state transitions, the system dynamics needs to be *explored* as much as possible.

Given an action which is considered to yield optimal results in the given epoch with respect to the loss function, it might be profitable to opt for a different action so that more information about the system dynamics can be gained. Parameter estimation then becomes more precise and comprehensive. This practice is referred to as the *exploration principle* or system *excitation.*

However, such activity is in conflict with the system's optimal performance. Exploration must not exceed a certain limit for the decision process not to depart too far from the desired evolution. Combining these two contradictory directions results in what is called an *exploration and exploitation principle*[2]. Different ways how to incorporate it into a decision task can be found in Mesbah's survey [23].

Markov decision processes can suffer from insufficient excitation due to the deterministic nature of their optimal values. On the other hand fully probabilistic design independently introduced first in [11] and later in [7] has the favorable quality of generating actions based on probability density functions previously labeled as decision rules, see (2.9). Probabilistic nature of such actions naturally conforms the exploration and exploitation principle, giving fully probabilistic design a notable advantage over the standard Markov approach.

Nevertheless as it will be shown in the next section, the dimension of probability densities required in both fully probabilistic design and Markov decision processes can grow rapidly when the system model is unknown. Before introducing the standard version of the fully probabilistic design it will therefore be necessary to provide a method on how to overcome this obstacle.

3.1 Certainty-Equivalence Strategy

Assuming that the system model is known, the problem of new data prediction can be modeled according to

$$p(\mathrm{s}_t, \mathrm{a}_t | \mathrm{H}_{t-1}) = m(\mathrm{s}_t | \mathrm{a}_t, \mathrm{s}_{t-1}) r(\mathrm{a}_t | \mathrm{s}_{t-1})$$

if Markov property holds for the system model m as well as the decision rules r. Data prediction for the next epoch is therefore conditioned merely on the last observed state instead of the whole history H_{t-1} present on the left hand side.

However, when dealing with uncertainty, it becomes necessary to include parameters in the system model. This way it becomes $m(\mathrm{s}_t | \mathrm{a}_t, \mathrm{s}_{t-1}, \theta_{t-1})$, where θ_{t-1} can vary in time as discussed in Section 2.5. Time indexation is present to emphasize this fact. Forgetting showed a promising

[1] System identification is a term from control theory which describes the process of estimating the model parameters.

[2] In control theory referred to as dual control.

way to deal with parameter changes. Their probability density was derived to be

$$p(\theta_t|\mathrm{H}_{t-1}) \propto \prod_{\mathrm{s}=1}^{n-1} \prod_{\mathrm{a}=1}^{j} \prod_{\mathrm{s}'=1}^{n} m(\mathrm{s}|\mathrm{a},\mathrm{s}',\theta_t)^{\lambda_{t-1}\nu_{t-1;\mathrm{s}|\mathrm{a},\mathrm{s}'}+(1-\lambda_{t-1})\nu_{0;\mathrm{s}|\mathrm{a},\mathrm{s}'}-1}.$$

Statistic ν_{t-1} can be obtained recursively according to the formula (2.25), ν_0 is a prior statistic and λ_{t-1} and $1-\lambda_{t-1}$ are known probabilities, see the discussion under (2.40). With the knowledge of parameter probability density, it would then be possible to express the sought after system model as

$$m(\mathrm{s}_t|\mathrm{a}_t,\mathrm{s}_{t-1},\mathrm{H}_{t-1}) = m(\mathrm{s}_t|\mathrm{a}_t,\mathrm{s}_{t-1},\nu_{t-1}) = \int_{\mathcal{S}_\theta} m(\mathrm{s}_t|\mathrm{a}_t,\mathrm{s}_{t-1},\theta)p(\theta|\nu_{t-1})d\theta. \tag{3.1}$$

In order to predict the next state s_t, not only the past state s_{t-1} and the present action a_t are utilized. Statistic ν_{t-1} entering the estimation through the parameter probability density p is also needed. Furthermore once the future state s_t has been estimated, the new value of the statistic can be calculated after evaluating the product $\delta(\mathrm{s}_t,\mathrm{s})\delta(\mathrm{a}_t,\mathrm{a})\delta(\mathrm{s}_{t-1},\mathrm{s}')$ in (2.25).

In other words, knowledge of past values of ν is necessary in order to predict the future, whereas the future values become known once the predictions have been made. To focus merely on the system states would therefore lead to an incomplete evolution description. Rather than the new state alone, its coupling with the next value of the ν statistic is observed. The two of them together form a *hyper-state* in each epoch. System model under hyper-states then becomes

$$m(\mathrm{s}_t,\nu_t|\mathrm{a}_t,\mathrm{s}_{t-1},\nu_{t-1}). \tag{3.2}$$

As elegant as such solution may seem, it quickly hits the wall when faced with commonly available computational power. Numerical handling regarding hyper-states suffer from their extensive dimensionality. The most widespread tactic to overcome this computational shortcoming is called *certainty-equivalence strategy*, [16]. In this approach, unknown parameters θ are replaced with the most likely ν estimates

$$\hat{\theta} \equiv \frac{\nu_{t-1;\mathrm{s}|\mathrm{a},\mathrm{s}'}}{\sum_{\mathrm{s}=1}^{n}\nu_{t-1;\mathrm{s}|\mathrm{a},\mathrm{s}'}}. \tag{3.3}$$

Due to its computational feasibility as opposed to the hyper-state model (3.2) as well as its closeness to the integral (3.1), approximation

$$m(\mathrm{s}_t|\mathrm{a}_t,\mathrm{s}_{t-1},\nu_{t-1}) \approx m(\mathrm{s}_t|\mathrm{a}_t,\mathrm{s}_{t-1},\hat{\theta}), \tag{3.4}$$

enters the computations in place of the precise system model. This method is suitable for well-estimated slowly varying parameters.

3.2 Standard Formulation and Solution

Decision rules r and system model m serve to describe the decision process in an arbitrary time step. However, the ultimate decision-making goal is not to focus on individual epochs but to optimize the process as a whole. It is therefore desirable to express the entire system evolution using a single probability density function.

Let us consider a decision process with n epochs. The time set $\mathcal{S}_t$ therefore consists of the first n natural numbers, i.e. $\{1, 2, ..., n\}$. A new argument called *behavior*, denoted b, will be introduced using the defining relation of a *closed-loop* probability density c^r as

$$c^r(\mathrm{b}) \equiv c^r(\mathrm{s}_n, \mathrm{a}_n, \mathrm{s}_{n-1}, \mathrm{a}_{n-1}, ..., \mathrm{s}_1, \mathrm{a}_1, \mathrm{s}_0) = \prod_{\tau=1}^{n} m(\mathrm{s}_\tau|\mathrm{a}_\tau, \mathrm{s}_{\tau-1}, \hat{\theta})r(\mathrm{a}_\tau|\mathrm{s}_{\tau-1}) \equiv m(\mathrm{b})r(\mathrm{b}). \quad (3.5)$$

Term closed loop is a notion from control theory denoting a tuple consisting of the decision maker and the system. Density (3.5) assigns probabilities to different realizations of the system evolution which consist of its authentic state transitions and action selections. It hence describes the system's *real* behavior. Superscript r aims to highlight the dependence on decision rules.

Having knowledge of how the system evolves in time, it is possible to optimize its performance. This means that actions which influence the closed loop are not selected completely randomly but in such a way so that the closed loop tends to end up in the decision maker's desired states. Before trying to optimize the decision process, however, it is necessary to have some preferences regarding how it should *ideally* behave. Ideal closed-loop density

$$c^i(\mathrm{b}) \equiv c^i(\mathrm{s}_n, \mathrm{a}_n, \mathrm{s}_{n-1}, \mathrm{a}_{n-1}, ..., \mathrm{s}_1, \mathrm{a}_1, \mathrm{s}_0) = \prod_{\tau=1}^{n} m^i(\mathrm{s}_\tau|\mathrm{a}_\tau, \mathrm{s}_{\tau-1})r^i(\mathrm{a}_\tau|\mathrm{s}_{\tau-1}) \equiv m^i(\mathrm{b})r^i(\mathrm{b}) \quad (3.6)$$

quantifies these preferences by introducing ideal system model m^i and ideal decision rules r^i. State transitions given by m^i define the desired evolution of the system which is assumed to be known and does not need to be estimated. That is why ideal system model lacks the parameter estimate seen in the closed-loop density (3.5).

Remark 3. The task of model identification was successfully finished in Section 2.5 and subsequently approximated in Section 3.1 so that it could be used in real world applications. Model parameters will hence not be the primary focus of this chapter. In order to save some space in the derivations below, their estimates $\hat{\theta}$ will no longer be included in model arguments. This measure was taken merely for notation simplification purposes. It should be remembered that what is essentially meant by $m(\mathrm{s}_t|\mathrm{a}_t, \mathrm{s}_{t-1})$ is always the original probability density $m(\mathrm{s}_t|\mathrm{a}_t, \mathrm{s}_{t-1}, \hat{\theta})$.

Generally speaking, it may not be possible for the system to behave exactly according to the ideal depicted in (3.6). Optimizing the process therefore means to choose such actions, which would minimize the distance between the densities (3.5) and (3.6). One possible metric which measures this distance can be Kullback-Leibler divergence (2.38) and its use is implied by axiomatics, see [12]. Different examples of measures can be found for instance in [3].

Finally, since the decision problem formulation is probabilistic, the optimizing actions are chosen via the means of optimal decision rules which will be denoted by r^o,

$$r^o \in \underset{r}{\operatorname{argmin}}\, \mathrm{D}(c^r, c^i) = \int_{\mathcal{S}_b} c^r(\mathrm{b}) \ln\left(\frac{c^r(\mathrm{b})}{c^i(\mathrm{b})}\right) \mathrm{db}. \quad (3.7)$$

Integration above runs through the set of all behaviors $\mathcal{S}_b$.

Decision rules which follow the relation (3.7) are said to be gained through fully probabilistic design. Before deriving their exact form, an additive version of Kullback-Leibler divergence as well as dynamic programming need to be formulated.

Lemma 3.2.1. Let there be a loss function defined as

$$L^r(\mathrm{s}_t, \mathrm{a}_t, \mathrm{s}_{t-1}) \equiv \ln\left(\frac{m(\mathrm{s}_t|\mathrm{a}_t, \mathrm{s}_{t-1})r(\mathrm{a}_t|\mathrm{s}_{t-1})}{m^i(\mathrm{s}_t|\mathrm{a}_t, \mathrm{s}_{t-1})r^i(\mathrm{a}_t|\mathrm{s}_{t-1})}\right), \quad (3.8)$$

where superscript r highlights the dependence of L^r on decision rules. Then for any arbitrary decision rule r Kullback-Leibler divergence of the two closed-loop densities (3.5) and (3.6) can be expressed in an additive form

$$\begin{aligned}\mathrm{D}(c^r, c^i) &= \sum_{t\in\mathcal{S}_t} \int_{(\mathcal{S}_s,\mathcal{S}_a,\mathcal{S}_s)} c^r(\mathrm{s}_t, \mathrm{a}_t, \mathrm{s}_{t-1}) L^r(\mathrm{s}_t, \mathrm{a}_t, \mathrm{s}_{t-1}) \mathrm{d}(\mathrm{s}_t, \mathrm{a}_t, \mathrm{s}_{t-1}) \\ &= \sum_{t\in\mathcal{S}_t} \int_{\mathcal{S}_s} c^r(\mathrm{s}_{t-1}) \left[\int_{(\mathcal{S}_s,\mathcal{S}_a)} m(\mathrm{s}_t|\mathrm{a}_t, \mathrm{s}_{t-1}) r(\mathrm{a}_t|\mathrm{s}_{t-1}) \ln\left(\frac{m(\mathrm{s}_t|\mathrm{a}_t, \mathrm{s}_{t-1}) r(\mathrm{a}_t|\mathrm{s}_{t-1})}{m^i(\mathrm{s}_t|\mathrm{a}_t, \mathrm{s}_{t-1}) r^i(\mathrm{a}_t|\mathrm{s}_{t-1})} \right) \mathrm{d}(\mathrm{s}_t, \mathrm{a}_t) \right] \mathrm{d}\mathrm{s}_{t-1},\end{aligned} \tag{3.9}$$

where $c^r(\mathrm{s}_{t-1})$ is independent of $r(\mathrm{a}_\tau|\mathrm{s}_{\tau-1})$ for all τ greater or equal to t.

Proof. The proof is quite straightforward. Substituting the closed-loop densities into the Kullback-Leibler divergence definition (2.38) yields

$$\begin{aligned}\mathrm{D}(c^r(\mathrm{b}), c^i(\mathrm{b})) &= \int_{(\mathcal{S}_s,\mathcal{S}_a)^n} c^r(\mathrm{s}_n, \mathrm{a}_n, ..., s_1, a_1) \ln\left(\prod_{t\in\mathcal{S}_t} \frac{m(\mathrm{s}_t|\mathrm{a}_t, \mathrm{s}_{t-1}) r(\mathrm{a}_t|\mathrm{s}_{t-1})}{m^i(\mathrm{s}_t|\mathrm{a}_t, \mathrm{s}_{t-1}) r^i(\mathrm{a}_t|\mathrm{s}_{t-1})} \right) \mathrm{d}(\mathrm{s}_n, \mathrm{a}_n, ..., \mathrm{s}_1, \mathrm{a}_1) \\ &= \int_{(\mathcal{S}_s,\mathcal{S}_a)^n} \sum_{t\in\mathcal{S}_t} c^r(\mathrm{s}_n, \mathrm{a}_n, ..., \mathrm{s}_1, \mathrm{a}_1) \ln\left(\frac{m(\mathrm{s}_t|\mathrm{a}_t, \mathrm{s}_{t-1}) r(\mathrm{a}_t|\mathrm{s}_{t-1})}{m^i(\mathrm{s}_t|\mathrm{a}_t, \mathrm{s}_{t-1}) r^i(\mathrm{a}_t|\mathrm{s}_{t-1})} \right) \mathrm{d}(\mathrm{s}_n, \mathrm{a}_n, ..., \mathrm{s}_1, \mathrm{a}_1),\end{aligned}$$

where the last equation stems from the product property of logarithms. Symbol $(\mathcal{S}_s, \mathcal{S}_a)^n$ stands for n-fold Cartesian product of sets $\mathcal{S}_s$ and $\mathcal{S}_a$.

Since only finite decision processes are considered, it is possible to utilize the linearity of integrals property and exchange the order of the summation and integration, resulting in

$$\mathrm{D}(c^r(\mathrm{b}), c^i(\mathrm{b})) = \sum_{t\in\mathcal{S}_t} \int_{(\mathcal{S}_s,\mathcal{S}_a)^n} c^r(\mathrm{s}_n, \mathrm{a}_n, ..., \mathrm{s}_1, \mathrm{a}_1) \ln\left(\frac{m(\mathrm{s}_t|\mathrm{a}_t, \mathrm{s}_{t-1}) r(\mathrm{a}_t|\mathrm{s}_{t-1})}{m^i(\mathrm{s}_t|\mathrm{a}_t, \mathrm{s}_{t-1}) r^i(\mathrm{a}_t|\mathrm{s}_{t-1})} \right) \mathrm{d}(\mathrm{s}_n, \mathrm{a}_n, ..., \mathrm{s}_1, \mathrm{a}_1).$$

As the expression inside the logarithm of the integrand above depends only on s_t, a_t and s_{t-1}, the rest of the states and actions present in the argument of density c^r can integrate out creating a marginal closed-loop probability density. Using the L^r definition (3.8) from the lemma, the whole expression simplifies into

$$\mathrm{D}(c^r(\mathrm{b}), c^i(\mathrm{b})) = \sum_{t\in\mathcal{S}_t} \int_{(\mathcal{S}_s,\mathcal{S}_a,\mathcal{S}_s)} c^r(\mathrm{s}_t, \mathrm{a}_t, \mathrm{s}_{t-1}) L^r(\mathrm{s}_t, \mathrm{a}_t, \mathrm{s}_{t-1}) \mathrm{d}(\mathrm{s}_t, \mathrm{a}_t, \mathrm{s}_{t-1}),$$

which is precisely the form seen in the first row of (3.9).

The second row of (3.9) can be easily obtained from the first one using chain rule (2.8) on the closed-loop density, i.e.

$$c^r(\mathrm{s}_t, \mathrm{a}_t, \mathrm{s}_{t-1}) = c^r(\mathrm{s}_t|\mathrm{a}_t, \mathrm{s}_{t-1}) c^r(\mathrm{a}_t|\mathrm{s}_{t-1}) c^r(\mathrm{s}_{t-1}) \equiv m(\mathrm{s}_t|\mathrm{a}_t, \mathrm{s}_{t-1}) r(\mathrm{a}_t|\mathrm{s}_{t-1}) c^r(\mathrm{s}_{t-1}),$$

as well as Fubini's theorem for multidimensional integration [32].

□

Additive form of Kullback-Leibler divergence with r-dependent losses motivated the following theorem generalizing dynamic programming [2]. It is an optimization method which can be used to recursively solve many types of tasks by splitting them into smaller sub-tasks.

Theorem 3.2.2 (Dynamic programming)**.** Let us define a *value function* v as $v(\mathrm{s}_{t-1}) \equiv$

$$\min_{\{r(\mathrm{a}_\tau|\mathrm{s}_{\tau-1}),\tau\geq t\}} \sum_{\tau\geq t} \int_{(\mathcal{S}_s,\mathcal{S}_a,\mathcal{S}_s)} L^r(\mathrm{s}_\tau,\mathrm{a}_\tau,\mathrm{s}_{\tau-1}) m(\mathrm{s}_\tau|\mathrm{a}_\tau,\mathrm{s}_{\tau-1}) r(\mathrm{a}_\tau|\mathrm{s}_{\tau-1}) c^r(\mathrm{s}_{\tau-1}|\mathrm{s}_{t-1}) \mathrm{d}(\mathrm{s}_\tau,\mathrm{a}_\tau,\mathrm{s}_{\tau-1}). \tag{3.10}$$

Here, the r-indexation in loss function L^r emphasizes its dependence on decision rules $r(\mathrm{a}_\tau|\mathrm{s}_{\tau-1})$ but only up to the epoch t. Value function v then follows a backward functional recursion

$$v(\mathrm{s}_{t-1}) = \min_{\{r(\mathrm{a}_t|\mathrm{s}_{t-1})\}} \int_{(\mathcal{S}_s,\mathcal{S}_a)} \left[L^r(\mathrm{s}_t,\mathrm{a}_t,\mathrm{s}_{t-1}) + v(\mathrm{s}_t)\right] m(\mathrm{s}_t|\mathrm{a}_t,\mathrm{s}_{t-1}) r(\mathrm{a}_t|\mathrm{s}_{t-1}) \mathrm{d}(\mathrm{s}_t,\mathrm{a}_t), \tag{3.11}$$

which starts with the value $v(s_n) = 0$.

Proof. By isolating the first term of the sum in the definition (3.10), the minimization becomes

$$v(\mathrm{s}_{t-1}) = \min_{\{r(\mathrm{a}_t|\mathrm{s}_{t-1})\}} \left[\int_{(\mathcal{S}_s,\mathcal{S}_a,\mathcal{S}_s)} L^r(\mathrm{s}_t,\mathrm{a}_t,\mathrm{s}_{t-1}) m(\mathrm{s}_t|\mathrm{a}_t,\mathrm{s}_{t-1}) r(\mathrm{a}_t|\mathrm{s}_{t-1}) c^r(\mathrm{s}_{t-1}|\mathrm{s}_{t-1}) \mathrm{d}(\mathrm{s}_t,\mathrm{a}_t,\mathrm{s}_{t-1}) \right.$$
$$\left. + \min_{\{r(\mathrm{a}_\tau|\mathrm{s}_{\tau-1}),\tau\geq t+1\}} \sum_{\tau\geq t+1} \int_{(\mathcal{S}_s,\mathcal{S}_a,\mathcal{S}_s)} L^r(\mathrm{s}_\tau,\mathrm{a}_\tau,\mathrm{s}_{\tau-1}) m(\mathrm{s}_\tau|\mathrm{a}_\tau,\mathrm{s}_{\tau-1}) r(\mathrm{a}_\tau|\mathrm{s}_{\tau-1}) c^r(\mathrm{s}_{\tau-1}|\mathrm{s}_{t-1}) \mathrm{d}(\mathrm{s}_\tau,\mathrm{a}_\tau,\mathrm{s}_{\tau-1}) \right].$$

In the nested minimization above, epoch $t-1$ is only present through the state variable s_{t-1} in $c^r(\mathrm{s}_{\tau-1}|\mathrm{s}_{t-1})$. Since the summation runs from $t+1$, both the loss function and the densities relate to epoch t and higher. Using marginalization and chain rule, this density can be rewritten so that it relates to epoch t via the state variable s_t,

$$c^r(\mathrm{s}_{\tau-1}|\mathrm{s}_{t-1}) = \int_{\mathcal{S}_s} c^r(\mathrm{s}_{\tau-1},\mathrm{s}_t|\mathrm{s}_{t-1}) \mathrm{d}\mathrm{s}_t = \int_{\mathcal{S}_s} c^r(\mathrm{s}_{\tau-1}|\mathrm{s}_t,\mathrm{s}_{t-1}) c^r(\mathrm{s}_t|\mathrm{s}_{t-1}) \mathrm{d}\mathrm{s}_t,$$

which will be useful later on.

Since Markov property is considered to hold throughout this work, further history is regarded to be of no influence on the system. Density $c^r(\mathrm{s}_{\tau-1}|\mathrm{s}_t,\mathrm{s}_{t-1})$ therefore reduces to $c^r(\mathrm{s}_{\tau-1}|\mathrm{s}_t)$ and the discussed conditional density $c^r(\mathrm{s}_{\tau-1}|\mathrm{s}_{t-1})$ becomes

$$c^r(\mathrm{s}_{\tau-1}|\mathrm{s}_{t-1}) = \int_{\mathcal{S}_s} c^r(\mathrm{s}_{\tau-1}|\mathrm{s}_t) c^r(\mathrm{s}_t|\mathrm{s}_{t-1}) \mathrm{d}\mathrm{s}_t.$$

Before substituting this manipulated density back into the value function, it will be necessary to come up with a shorter notation. Let L^r_t denote the loss function value[3] $L^r(\mathrm{s}_t,\mathrm{a}_t,\mathrm{s}_{t-1})$ and similarly for m_t and r_t. Only the time index is present to denote the epoch of the arguments whereas arguments themselves are omitted in order to save space. Identifying the integrable arguments is possible by the means of differential elements.

Utilizing the c^r density reformulation and the new notation, the value function gets the form

$$v(\mathrm{s}_{t-1}) = \min_{\{r_t\}} \left[\int_{(\mathcal{S}_s,\mathcal{S}_a)} L^r_t m_t r_t \mathrm{d}(\mathrm{s}_t,\mathrm{a}_t) \right.$$
$$\left. + \min_{\{r_\tau,\tau\geq t+1\}} \sum_{\tau\geq t+1} \int_{(\mathcal{S}_s,\mathcal{S}_a,\mathcal{S}_s,\mathcal{S}_s)} L^r_\tau m_\tau r_\tau c^r(\mathrm{s}_{\tau-1}|\mathrm{s}_t) c^r(\mathrm{s}_t|\mathrm{s}_{t-1}) \mathrm{d}(\mathrm{s}_t,\mathrm{s}_\tau,\mathrm{a}_\tau,\mathrm{s}_{\tau-1}) \right].$$

[3] Otherwise referred to simply as loss.

Here, the integrand inside the nested minimization is independent of the decision rules r_τ for τ grater than t. A part of the expression hence can be taken out of the integration, resulting in

$$\begin{aligned} v(\mathrm{s}_{t-1}) = \min_{\{r_t\}} \Bigg[\int_{(\mathcal{S}_s,\mathcal{S}_a)} & L_t^r m_t r_t \mathrm{d}(\mathrm{s}_t, \mathrm{a}_t) \\ + \int_{\mathcal{S}_s} \Bigg[\min_{\{r_\tau, \tau \geq t+1\}} \sum_{\tau \geq t+1} \int_{(\mathcal{S}_s,\mathcal{S}_a,\mathcal{S}_s)} & L_\tau^r m_\tau r_\tau c^r(\mathrm{s}_{\tau-1}|\mathrm{s}_t) \mathrm{d}(\mathrm{s}_\tau, \mathrm{a}_\tau, \mathrm{s}_{\tau-1}) \Bigg] c^r(\mathrm{s}_t|\mathrm{s}_{t-1}) \mathrm{d}\mathrm{s}_t \Bigg]. \end{aligned}$$

The nested brackets above hold the value function $v(\mathrm{s}_t)$ for the next recursion step.

Applying the same logic as before but expanding the density using actions this time, $c^r(\mathrm{s}_t|\mathrm{s}_{t-1})$ becomes

$$\begin{aligned} c^r(\mathrm{s}_t|\mathrm{s}_{t-1}) = \int_{\mathcal{S}_a} c^r(\mathrm{s}_t, \mathrm{a}_t|\mathrm{s}_{t-1}) \mathrm{d}\mathrm{a}_t &= \int_{\mathcal{S}_a} c^r(\mathrm{s}_t|\mathrm{a}_t, \mathrm{s}_{t-1}) c^r(\mathrm{a}_t|\mathrm{s}_{t-1}) \mathrm{d}\mathrm{a}_t \\ &= \int_{\mathcal{S}_a} m(\mathrm{s}_t|\mathrm{a}_t, \mathrm{s}_{t-1}) r(\mathrm{a}_t|\mathrm{s}_{t-1}) \mathrm{d}\mathrm{a}_t. \end{aligned}$$

Substituting the acquired integral back into the previous expression yields the final backward functional recursion (3.11),

$$\begin{aligned} v(\mathrm{s}_{t-1}) &= \min_{\{r_t\}} \left[\int_{(\mathcal{S}_s,\mathcal{S}_a)} L_t^r m_t r_t \mathrm{d}(\mathrm{s}_t, \mathrm{a}_t) + \int_{(\mathcal{S}_s,\mathcal{S}_a)} v(\mathrm{s}_t) m_t r_t \mathrm{d}(\mathrm{s}_t, \mathrm{a}_t) \right] \\ &= \min_{\{r_t\}} \int_{(\mathcal{S}_s,\mathcal{S}_a)} [L_t^r + v(\mathrm{s}_t)] m_t r_t \mathrm{d}(\mathrm{s}_t, \mathrm{a}_t). \end{aligned}$$

Finally, the equality of $v(\mathrm{s}_n)$ to zero stems from the fact that the decision process is finite. After arriving at the final epoch, no new action is selected. The sum in the value function definition (3.10) is therefore empty after setting t equal to n.

□

Remark 4. The motivation behind the value function definition is the following. In Theorem 3.2.2, $L^r(\mathrm{s}_\tau, \mathrm{a}_\tau, \mathrm{s}_{\tau-1})$ represents the loss function value for the given epoch τ. The total loss is then the sum over the entire decision process, i.e.

$$\sum_{\tau=1}^{n} L^r(\mathrm{s}_\tau, \mathrm{a}_\tau, \mathrm{s}_{\tau-1}).$$

Consider its minimization with the dynamic programming approach described on page 28, i.e. taking into account only the last $n - t$ loss summands. When performing the minimization of its expected value, the exact definition of the value function (3.10) is obtained.

Now that the preliminaries have been stated and proven, it is possible to perform fully probabilistic design, which provides the exact form of the decision rules forming the optimal decision strategy in (3.7).

Theorem 3.2.3 (Fully probabilistic design)**.** Backward functional recursion given by

$$d(\mathrm{a}_t, \mathrm{s}_{t-1}) \equiv \int_{\mathcal{S}_s} m(\mathrm{s}_t|\mathrm{a}_t, \mathrm{s}_{t-1}) \ln \left(\frac{m(\mathrm{s}_t|\mathrm{a}_t, \mathrm{s}_{t-1})}{h(\mathrm{s}_t) m^i(\mathrm{s}_t|\mathrm{a}_t, \mathrm{s}_{t-1})} \right) \mathrm{d}\mathrm{s}_t, \tag{3.12}$$

$$h(\mathrm{s}_{t-1}) \equiv \int_{\mathcal{S}_a} r^i(\mathrm{a}_t|\mathrm{s}_{t-1}) \exp\left(-d(\mathrm{a}_t, \mathrm{s}_{t-1})\right) \mathrm{d}\mathrm{a}_t, \tag{3.13}$$

where $h(\mathrm{s}_{t-1})$ belongs into the interval $[0,1]$ and begins with the value of $h(\mathrm{s}_n)$ equal to one, minimizes the value function from dynamic programming Theorem 3.2.2 defined as $v(\mathrm{s}_t) \equiv -\ln(h(\mathrm{s}_t))$. For closed-loop densities (3.5) and (3.6), the reached minimum is

$$v(\mathrm{s}_0) = \mathrm{D}(c^{r^o}, c^i). \tag{3.14}$$

Optimal decision rules forming the optimal closed-loop density c^{r^o} can be obtained as

$$r^o(\mathrm{a}_t|\mathrm{s}_{t-1}) = \frac{r^i(\mathrm{a}_t|\mathrm{s}_{t-1})\exp(-d(\mathrm{a}_t,\mathrm{s}_{t-1}))}{h(\mathrm{s}_{t-1})}. \tag{3.15}$$

Proof. After setting the time index equal to zero, the sum in the value function definition (3.10) runs through the whole time set $\mathcal{S}_t$ and $v(\mathrm{s}_0)$ minimization is then identical to the relation (3.7).

Let the loss function L^r in the dynamic programming Theorem 3.2.2 be defined as (3.8), i.e.

$$L^r(\mathrm{s}_t,\mathrm{a}_t,\mathrm{s}_{t-1}) \equiv \ln\left(\frac{m(\mathrm{s}_t|\mathrm{a}_t,\mathrm{s}_{t-1})r(\mathrm{a}_t|\mathrm{s}_{t-1})}{m^i(\mathrm{s}_t|\mathrm{a}_t,\mathrm{s}_{t-1})r^i(\mathrm{a}_t|\mathrm{s}_{t-1})}\right).$$

This definition is correct since $L^r(\mathrm{s}_t,\mathrm{a}_t,\mathrm{s}_{t-1})$ does not depend on decision rules others than the ones describing the action selection in the epoch t. Considering the loss function in this form and defining value function as the logarithm of (3.13), it becomes

$$\begin{aligned} v(\mathrm{s}_{t-1}) &= \min_{\{r(\mathrm{a}_t|\mathrm{s}_{t-1})\}} \int\limits_{(\mathcal{S}_s,\mathcal{S}_a)} \left[\ln\left(\frac{m(\mathrm{s}_t|\mathrm{a}_t,\mathrm{s}_{t-1})r(\mathrm{a}_t|\mathrm{s}_{t-1})}{m^i(\mathrm{s}_t|\mathrm{a}_t,\mathrm{s}_{t-1})r^i(\mathrm{a}_t|\mathrm{s}_{t-1})}\right) + v(\mathrm{s}_t)\right] m(\mathrm{s}_t|\mathrm{a}_t,\mathrm{s}_{t-1})r(\mathrm{a}_t|\mathrm{s}_{t-1})\mathrm{d}(\mathrm{s}_t,\mathrm{a}_t) \\ &= \min_{\{r(\mathrm{a}_t|\mathrm{s}_{t-1})\}} \int\limits_{(\mathcal{S}_s,\mathcal{S}_a)} \left[\ln\left(\frac{m(\mathrm{s}_t|\mathrm{a}_t,\mathrm{s}_{t-1})}{m^i(\mathrm{s}_t|\mathrm{a}_t,\mathrm{s}_{t-1})h(\mathrm{s}_t)}\right) + \ln\left(\frac{r(\mathrm{a}_t|\mathrm{s}_{t-1})}{r^i(\mathrm{a}_t|\mathrm{s}_{t-1})}\right)\right] m(\mathrm{s}_t|\mathrm{a}_t,\mathrm{s}_{t-1})r(\mathrm{a}_t|\mathrm{s}_{t-1})\mathrm{d}(\mathrm{s}_t,\mathrm{a}_t), \end{aligned}$$

where the first equality above utilizes the dynamic programming relation (3.11).

Previous derivations hint the choice of the value function as the logarithm of (3.13). This way $h(\mathrm{s}_t)$ could have been included in the logarithm of the system models which all depend on s_t. The logarithm of decision rules independent of s_t now stand on their own, which allows it to be taken out of the integration through the state set $\mathcal{S}_s$. This computation yields

$$\begin{aligned} \int\limits_{\mathcal{S}_s} m(\mathrm{s}_t|\mathrm{a}_t,\mathrm{s}_{t-1})r(\mathrm{a}_t|\mathrm{s}_{t-1})\ln\left(\frac{r(\mathrm{a}_t|\mathrm{s}_{t-1})}{r^i(\mathrm{a}_t|\mathrm{s}_{t-1})}\right)\mathrm{d}\mathrm{s}_t &= r(\mathrm{a}_t|\mathrm{s}_{t-1})\ln\left(\frac{r(\mathrm{a}_t|\mathrm{s}_{t-1})}{r^i(\mathrm{a}_t|\mathrm{s}_{t-1})}\right)\int\limits_{\mathcal{S}_s} m(\mathrm{s}_t|\mathrm{a}_t,\mathrm{s}_{t-1})\mathrm{d}\mathrm{s}_t \\ &= r(\mathrm{a}_t|\mathrm{s}_{t-1})\ln\left(\frac{r(\mathrm{a}_t|\mathrm{s}_{t-1})}{r^i(\mathrm{a}_t|\mathrm{s}_{t-1})}\right). \end{aligned}$$

Continuing in the value function computation while utilizing the previous knowledge, it gets the form

$$\begin{aligned} v(\mathrm{s}_{t-1}) = \min_{\{r(\mathrm{a}_t|\mathrm{s}_{t-1})\}} \int\limits_{\mathcal{S}_a} r(\mathrm{a}_t|\mathrm{s}_{t-1}) &\left[\ln\left(\frac{r(\mathrm{a}_t|\mathrm{s}_{t-1})}{r^i(\mathrm{a}_t|\mathrm{s}_{t-1})}\right)\right. \\ &\left. + \int\limits_{\mathcal{S}_s} m(\mathrm{s}_t|\mathrm{a}_t,\mathrm{s}_{t-1})\ln\left(\frac{m(\mathrm{s}_t|\mathrm{a}_t,\mathrm{s}_{t-1})}{m^i(\mathrm{s}_t|\mathrm{a}_t,\mathrm{s}_{t-1})h(\mathrm{s}_t)}\right)\mathrm{d}\mathrm{s}_t\right]\mathrm{d}\mathrm{a}_t, \end{aligned}$$

where the second integral is precisely the d function definintion (3.13). Using this auxiliary notation, the value function simplifies into

$$v(\mathrm{s}_{t-1}) = \min_{\{r(\mathrm{a}_t|\mathrm{s}_{t-1})\}} \int_{\mathcal{S}_a} r(\mathrm{a}_t|\mathrm{s}_{t-1}) \left[\ln \left(\frac{r(\mathrm{a}_t|\mathrm{s}_{t-1})}{r^i(\mathrm{a}_t|\mathrm{s}_{t-1})} \right) + d(\mathrm{a}_t, \mathrm{s}_{t-1}) \right] \mathrm{da}_t.$$

Ignoring d, it can be seen that the expression above notably resembles the definition of Kullback-Leibler divergence (2.38). In order to include d in the logarithm, certain adjustments need to be made, specifically

$$d(\mathrm{a}_t, \mathrm{s}_{t-1}) = -[-d(\mathrm{a}_t, \mathrm{s}_{t-1})] = -\ln\left(\exp[-d(\mathrm{a}_t, \mathrm{s}_{t-1})]\right).$$

This expression can be substituted back to the value function which then becomes

$$v(\mathrm{s}_{t-1}) = \min_{\{r(\mathrm{a}_t|\mathrm{s}_{t-1})\}} \int_{\mathcal{S}_a} r(\mathrm{a}_t|\mathrm{s}_{t-1}) \left[\ln \left(\frac{r(\mathrm{a}_t|\mathrm{s}_{t-1})}{r^i(\mathrm{a}_t|\mathrm{s}_{t-1}) \exp[-d(\mathrm{a}_t, \mathrm{s}_{t-1})]} \right) \right] \mathrm{da}_t.$$

In order for the expression above to be considered a Kullback-Leibler divergence, the denominator inside the logarithm needs to be a probability density. Such criterion can only be guaranteed by a normalization factor, which is precisely the function h given by (3.13). Thus

$$\begin{aligned} \ln \left(\frac{r(\mathrm{a}_t|\mathrm{s}_{t-1})}{r^i(\mathrm{a}_t|\mathrm{s}_{t-1}) \exp[-d(\mathrm{a}_t, \mathrm{s}_{t-1})]} \right) &= \ln \left(\frac{r(\mathrm{a}_t|\mathrm{s}_{t-1}) h(\mathrm{s}_{t-1})}{r^i(\mathrm{a}_t|\mathrm{s}_{t-1}) \exp[-d(\mathrm{a}_t, \mathrm{s}_{t-1})]} \right) - \ln[h(\mathrm{s}_{t-1})] \\ &= \ln \left(\frac{r(\mathrm{a}_t|\mathrm{s}_{t-1}) h(\mathrm{s}_{t-1})}{r^i(\mathrm{a}_t|\mathrm{s}_{t-1}) \exp[-d(\mathrm{a}_t, \mathrm{s}_{t-1})]} \right) + v(\mathrm{s}_{t-1}). \end{aligned}$$

The logarithm of $h(\mathrm{s}_{t-1})$ can be written as the value function for the given epoch. Substitution of this expression back to the integral reminiscing Kullback-Leibler divergence then yields

$$v(\mathrm{s}_{t-1}) = \min_{\{r(\mathrm{a}_t|\mathrm{s}_{t-1})\}} \int_{\mathcal{S}_a} r(\mathrm{a}_t|\mathrm{s}_{t-1}) \left[\ln \left(\frac{r(\mathrm{a}_t|\mathrm{s}_{t-1}) h(\mathrm{s}_{t-1})}{r^i(\mathrm{a}_t|\mathrm{s}_{t-1}) \exp[-d(\mathrm{a}_t, \mathrm{s}_{t-1})]} \right) + v(\mathrm{s}_{t-1}) \right] \mathrm{da}_t.$$

As the integration runs through actions a_t, it is possible to move $v(\mathrm{s}_{t-1})$ in front of the integral. Decision rule density then integrates out to unity, which enables the value function on the right hand side to cancel out with the one on the left hand side. Kullback-Leibler divergence minimum is then received as

$$0 = \min_{\{r(\mathrm{a}_t|\mathrm{s}_{t-1})\}} \int_{\mathcal{S}_a} r(\mathrm{a}_t|\mathrm{s}_{t-1}) \left[\ln \left(\frac{r(\mathrm{a}_t|\mathrm{s}_{t-1}) h(\mathrm{s}_{t-1})}{r^i(\mathrm{a}_t|\mathrm{s}_{t-1}) \exp[-d(\mathrm{a}_t, \mathrm{s}_{t-1})]} \right) \right] \mathrm{da}_t \equiv \mathrm{D}\left[r(\mathrm{a}_t|\mathrm{s}_{t-1}), r^o(\mathrm{a}_t|\mathrm{s}_{t-1}) \right].$$

The final equality above is used to define the optimal decision rules as stated in the theorem, i.e. (3.15).

All that remains to show is that the function $h(\mathrm{s}_t)$ falls into the interval $[0, 1]$ for any given epoch t. For this purpose, the concave logarithmic loss will be modified into a convex $x \ln x$ form and the statement will be then proven based on Jensen's inequality [8]. Once again the space restriction will not allow the complete notation and so its shortened version utilizing only time indexation will be used, recall the proof of Theorem 3.2.2. Hence

$$\begin{aligned}v_{t-1} &= \min_{\{r_\tau,\tau\geq t\}} \sum_{\tau\geq t} \int\limits_{(\mathcal{S}_s,\mathcal{S}_a,\mathcal{S}_s)} \ln\left(\frac{m_\tau r_\tau}{m^i_\tau r^i_\tau}\right) m_\tau r_\tau c^r_{\tau-1} \mathrm{d}(\mathrm{s}_\tau,\mathrm{a}_\tau,\mathrm{s}_{\tau-1}) \\ &= \min_{\{r_\tau,\tau\geq t\}} \sum_{\tau\geq t} \int_{\mathcal{S}_s} \left[\int_{(\mathcal{S}_a,\mathcal{S}_s)} \ln\left(\frac{m_\tau r_\tau}{m^i_\tau r^i_\tau}\right) \frac{m_\tau r_\tau}{m^i_\tau r^i_\tau} m^i_\tau r^i_\tau d(\mathrm{s}_\tau,\mathrm{a}_\tau)\right] c^r_{\tau-1} \mathrm{d}\mathrm{s}_{\tau-1}.\end{aligned}$$

Expression in the square brackets above stands for the expected value of the product of the logarithm and the fraction term, that is

$$\int\limits_{(\mathcal{S}_a,\mathcal{S}_s)} \ln\left(\frac{m_\tau r_\tau}{m^i_\tau r^i_\tau}\right) \frac{m_\tau r_\tau}{m^i_\tau r^i_\tau} m^i_\tau r^i_\tau d(\mathrm{s}_\tau,\mathrm{a}_\tau) \equiv \mathrm{E}^i\left[\frac{m_\tau r_\tau}{m^i_\tau r^i_\tau} \ln\left(\frac{m_\tau r_\tau}{m^i_\tau r^i_\tau}\right)\right].$$

Here, E^i denotes the mentioned expected value and the superscript i relates it to its centering densities m^i_τ and r^i_τ.

The expected value argument already has the desired $x \ln x$ form and it is therefore possible to apply Jensen's inequality, yielding

$$\begin{aligned}\mathrm{E}^i\left[\frac{m_\tau r_\tau}{m^i_\tau r^i_\tau} \ln\left(\frac{m_\tau r_\tau}{m^i_\tau r^i_\tau}\right)\right] &\geq \mathrm{Id}\left[\mathrm{E}^i\left[\frac{m_\tau r_\tau}{m^i_\tau r^i_\tau}\right]\right] \ln\left[\mathrm{E}^i\left[\frac{m_\tau r_\tau}{m^i_\tau r^i_\tau}\right]\right] \\ = \left[\int_{(\mathcal{S}_s,\mathcal{S}_a)} \frac{m_\tau r_\tau}{m^i_\tau r^i_\tau} m^i_\tau r^i_\tau \mathrm{d}(\mathrm{s}_\tau,\mathrm{a}_\tau)\right] &\ln\left[\int_{(\mathcal{S}_s,\mathcal{S}_a)} \frac{m_\tau r_\tau}{m^i_\tau r^i_\tau} m^i_\tau r^i_\tau \mathrm{d}(\mathrm{s}_\tau,\mathrm{a}_\tau)\right] = 0.\end{aligned}$$

The zero equality follows from the fact that the centering densities in the integral cancel out with the denominator and the numerator then integrates to unity. Logarithm of this same expression then equals zero.

□

Up until now only loss functions dependent on decision rules were considered. Markov decision processes discussed in the beginning of the chapter, however, deal with loss functions independent of the decision strategy and its rules. The next theorem therefore states what happens to dynamic programming when dealing with this type of loss function. Index r will be omitted in order to emphasize the independence.

Theorem 3.2.4. Let there be a function φ defined as

$$\varphi(\mathrm{a}_t,\mathrm{s}_{t-1}) \equiv \int\limits_{\mathcal{S}_s} L(\mathrm{s}_t,\mathrm{a}_t,\mathrm{s}_{t-1}) m(\mathrm{s}_t|\mathrm{a}_t,\mathrm{s}_{t-1}) \mathrm{d}\mathrm{s}_t, \tag{3.16}$$

where L represents a loss function independent of decision rules r. Optimal strategy (3.7) is then deterministic and generates optimizing actions a^o according to

$$\mathrm{a}^o_t(\mathrm{s}_{t-1}) \in \operatorname*{Argmin}_{\mathrm{a}_t\in\mathcal{S}_a} \varphi(\mathrm{a}_t,\mathrm{s}_{t-1}). \tag{3.17}$$

These optimizing actions are dependent on the current state of the system in the given epoch.

Proof. The proof will be done by the means of mathematical induction performed on recursion (3.11) from dynamic programming. Since the recursion is backward, the base case needs to be shown in the final epoch of the process, i.e.

$$v(\mathrm{s}_{n-1}) = \min_{\{r(\mathrm{a}_n|\mathrm{s}_{n-1})\}} \int\limits_{(\mathcal{S}_s,\mathcal{S}_a)} \Big[L(\mathrm{s}_n,\mathrm{a}_n,\mathrm{s}_{n-1}) + v(\mathrm{s}_n)\Big] m(\mathrm{s}_n|\mathrm{a}_n,\mathrm{s}_{n-1}) r(\mathrm{a}_n|\mathrm{s}_{n-1}) \mathrm{d}(\mathrm{s}_n,\mathrm{a}_n).$$

Theorem 3.2.2 says that the value function in the final epoch equals zero. This statement serves as the base case in the previous relation. Utilizing the φ function definition (3.16), the relation then becomes

$$v(\mathrm{s}_{n-1}) = \min_{\{r(\mathrm{a}_n|\mathrm{s}_{n-1})\}} \int\limits_{\mathcal{S}_a} \varphi(\mathrm{a}_n,\mathrm{s}_{n-1}) r(\mathrm{a}_n|\mathrm{s}_{n-1}) \mathrm{d}\mathrm{a}_n.$$

The minimization in $v(\mathrm{s}_{n-1})$ is performed over the set of all decision rules available for the given epoch. The integral above is therefore minimum possible with respect to r. Action choice (3.17) from the theorem statement leads to a value of function φ which is minimum in the action variable. Combination of minimization in decision rules and actions then yields a value function which is optimal for the given state.

Now it is time to prove the inductive step $t \to t-1$. Value function (3.11) can be split into two terms and utilizing the φ definition it turns into

$$v(\mathrm{s}_{t-1}) = \min_{\{r(\mathrm{a}_t|\mathrm{s}_{t-1})\}} \left[\int\limits_{\mathcal{S}_a} \varphi(\mathrm{a}_t,\mathrm{s}_{t-1}) r(\mathrm{a}_t|\mathrm{s}_{t-1}) \mathrm{d}\mathrm{a}_t + \int\limits_{\mathcal{S}_s} v(\mathrm{s}_t) m(\mathrm{s}_t|\mathrm{a}_t,\mathrm{s}_{t-1}) \mathrm{d}\mathrm{s}_t \int\limits_{\mathcal{S}_a} r(\mathrm{a}_t|\mathrm{s}_{t-1}) \mathrm{d}\mathrm{a}_t\right].$$

Here, the integral of the decision rule density r over the whole action set $\mathcal{S}_a$ equals one. As $v(\mathrm{s}_t)$ is independent of decision rules $r(\mathrm{a}_t|\mathrm{s}_{t-1})$, the second integral can be taken out of the minimization. Hence all that is left to minimize is the first integral

$$v(\mathrm{s}_{t-1}) = \min_{\{r(\mathrm{a}_t|\mathrm{s}_{t-1})\}} \left[\int\limits_{\mathcal{S}_a} \varphi(\mathrm{a}_t,\mathrm{s}_{t-1}) r(\mathrm{a}_t|\mathrm{s}_{t-1}) \mathrm{d}\mathrm{a}_t\right] + \int\limits_{\mathcal{S}_s} v(\mathrm{s}_t) m(\mathrm{s}_t|\mathrm{a}_t,\mathrm{s}_{t-1}) \mathrm{d}\mathrm{s}_t.$$

Choosing actions according to (3.17) minimizes the entire value function with regards to actions and the same logic as in the base case applies. Function v is therefore optimal for all epochs.

□

3.3 Fully Probabilistic Design and Markov Decision Processes

Fully probabilistic design as well as Markov decision processes represent two different approaches to solving decision tasks. It will be the purpose of this section to show the relationship between them, while also highlighting some advantages fully probabilistic design holds.

Markov decision process can be defined as a tuple consisting of sets of all possible states and actions, transition probability density which fulfills the Markov property and expected immediate reward. Choosing an arbitrary t from the time set $\mathcal{S}_t$ and utilizing the notation of the previous section, this tuple can be written as

$$(\mathcal{S}_s, \mathcal{S}_a, m(\mathrm{s}_t|\mathrm{a}_t,\mathrm{s}_{t-1}), L(\mathrm{s}_t,\mathrm{a}_t,\mathrm{s}_{t-1}))\,,$$

where system model m serves as the transition probability density for the given epoch. Replacing reward with expected loss is justified provided that there exists an unequivocal relation between the two. This work will cover the case where the expected loss is equal to negative expected reward, i.e.

$$L(\mathrm{s}_t, \mathrm{a}_t, \mathrm{s}_{t-1}) = -R(\mathrm{s}_t, \mathrm{a}_t, \mathrm{s}_{t-1}),$$

R denoting the said reward function. Besides that, Markov decision processes' loss also must be independent of decision rules r.

The goal of both Markov decision processes and fully probabilistic design is to optimize the process so that the system ends up in the decision maker's desired states. These are usually fixed for each epoch but there may not be a clear preference towards actions to choose. Fully probabilistic design quantifies this fact by setting the ideal decision rules equal to the real closed-loop rules, that is

$$r^i(\mathrm{a}_t|\mathrm{s}_{t-1}) \equiv r(\mathrm{a}_t|\mathrm{s}_{t-1}). \tag{3.18}$$

Equivalence (3.18) introduced in [14] is called *leave to the fate option* and it can be shown that fully probabilistic design tasks which follow this condition reduce to Markov decision processes. Recalling the form of the loss function (3.8) from fully probabilistic design,

$$L^r(\mathrm{s}_t, \mathrm{a}_t, \mathrm{s}_{t-1}) \equiv \ln\left(\frac{m(\mathrm{s}_t|\mathrm{a}_t, \mathrm{s}_{t-1})r(\mathrm{a}_t|\mathrm{s}_{t-1})}{m^i(\mathrm{s}_t|\mathrm{a}_t, \mathrm{s}_{t-1})r^i(\mathrm{a}_t|\mathrm{s}_{t-1})}\right),$$

it is easy to see that under leave to the fate option, the loss becomes independent of decision rules r. The total loss is obtained as

$$L(\mathrm{b}) \equiv \sum_{t\in\mathcal{S}_t} L_t(\mathrm{s}_t, \mathrm{a}_t, \mathrm{s}_{t-1}) = \sum_{t\in\mathcal{S}_t} \ln\left(\frac{m(\mathrm{s}_t|\mathrm{a}_t, \mathrm{s}_{t-1})}{m^i(\mathrm{s}_t|\mathrm{a}_t, \mathrm{s}_{t-1})}\right). \tag{3.19}$$

where $L_t(\mathrm{s}_t, \mathrm{a}_t, \mathrm{s}_{t-1})$ represents a partial loss for the given epoch t. The optimal strategy of such process is then deterministic according to Theorem 3.2.4.

Inversion of (3.19) might make the task conversion from fully probabilistic design to a Markov decision process seem quite straightforward. Moving the sum in (3.19) inside the logarithm, thus getting products of the two system models and subsequently expressing the ideal system model yields

$$m^i(\mathrm{b}) = m(\mathrm{b})\exp[-L(\mathrm{b})].$$

Hence, opting for the ideal density m^i to be proportionate to the product of the real system model and exponential of the expected loss function might seem as the right choice when trying to convert the design approach.

In reality, for models proportionate to the said product this practice leads to Kullback-Leibler divergence of the form

$$D(c^r||m^i r) = \mathrm{E}^r[L(\mathrm{b})] + \mathrm{E}^r[\ln(\Phi(\mathrm{a}_n, ..., \mathrm{a}_1))], \tag{3.20}$$

where Φ is a normalization factor which ensures that the proportionate m^i is in fact a probability density, specifically

$$\Phi(\mathrm{a}_n, ..., \mathrm{a}_1) = \int_{\mathcal{S}_s^n} m(\mathrm{b})\exp\left(-L(\mathrm{b})\right)\mathrm{d}(\mathrm{s}_n, ..., \mathrm{s}_1), \tag{3.21}$$

symbol $\mathcal{S}_s^n$ denoting an n-fold Cartesian product of set $\mathcal{S}_s$.

It can be seen in (3.20) that what is being minimized is not the expected loss function as desired. Minimization of its sum with the additional term (3.21) is performed instead. The relationship between the two approaches is therefore a little more complicated. It is described in the next theorem.

Theorem 3.3.1. 1. There are fully probabilistic design tasks having no standard Markov decision process equivalent.

2. Let there be a Markov decision process with stabilizing strategy r^s such that the expected loss averaged over this strategy is finite. It is then possible to approximate such process with a fully probabilistic design task using the same system model m and ideal closed-loop density of the form
$$c^{i\lambda}(\mathrm{b}) \equiv \frac{\tilde{c}(\mathrm{b})\exp\left[-L(\mathrm{b})/\lambda\right]}{\int_{\mathcal{S}_b}\tilde{c}(\mathrm{b})\exp\left[-L(\mathrm{b})/\lambda\right]\mathrm{db}}, \tag{3.22}$$
defined for λ positive and density $\tilde{c}$ positive on the set of all behaviors $\mathcal{S}_b$. Meanwhile it is assumed that

   (a) density $c^{i\lambda}$ is well defined when the denominator in (3.22) is finite and

   (b) for all deterministic strategies r such that the expected loss function averaged over r is finite it holds that the Kullback-Leibler divergence of the product mr to density $\tilde{c}$ is also finite.

Proof. 1. Markov decision processes require loss functions strictly defined according to (3.19). They cannot therefore accommodate decision tasks with other types of loss functions.

2. Bearing in mind the assumptions about λ and density $\tilde{c}$ mentioned in (2b), let us define two strategies on the set of all possible strategies $\mathcal{S}_r$,
$$r^o \in \operatorname*{Argmin}_{\mathcal{S}_r} \mathrm{E}^r[L], \text{ where } \mathrm{E}^{r^o}[L] \leq \mathrm{E}^{r^s}[L] < \infty, \tag{3.23}$$
$$r^{o\lambda} \in \operatorname*{Argmin}_{\mathcal{S}_r}[\mathrm{E}^r[L] + \lambda\mathrm{D}(mr,\tilde{c})] < \infty. \tag{3.24}$$
Minimization of the expected loss function in (3.23) corresponds with the minimization in Markov decision processes. The additional term in (3.24) then connects the problem of minimizing the expected loss function with fully probabilistic design.

It is possible to manipulate the argument of the λ-minimization (3.24) in a way that yields
$$\mathrm{E}^r[L] + \lambda\mathrm{D}(mr,\tilde{c}) = \lambda\int_{\mathcal{S}_b} m(\mathrm{b})r(\mathrm{b})\ln\left(\frac{m(\mathrm{b})r(\mathrm{b})}{\tilde{c}(\mathrm{b})\exp\left(\frac{-L(\mathrm{b})}{\lambda}\right)}\right).$$
This form is achievable by multiplying the loss function L by one and applying identity, i.e.
$$\frac{\lambda}{\lambda}L = -\lambda\ln\left(\exp\left(\frac{-L}{\lambda}\right)\right).$$
As the denominator in (3.22) is finite, it can be used in yet another manipulation - this time a zero addition, since
$$\ln\left(\int_{\mathcal{S}_b}\tilde{c}(\mathrm{b})\exp\left(\frac{-L(\mathrm{b})}{\lambda}\right)\mathrm{db}\right) - \ln\left(\int_{\mathcal{S}_b}\tilde{c}(\mathrm{b})\exp\left(\frac{-L(\mathrm{b})}{\lambda}\right)\mathrm{db}\right) = 0.$$

This expression will serve as a normalizing factor of the denominator in the integral above. Logarithm of a finite definite integral is a constant. It can be therefore taken out of the integration, which allows the product of densities m and r to integrate to unity. In the other term, the normalizing integral is combined with the denominator creating the density $c^{i\lambda}$. The argument of (3.24) then becomes

$$\mathrm{E}^{r}[L] + \lambda \mathrm{D}(mr, \tilde{c}) = \lambda \int_{\mathcal{S}_b} m(\mathrm{b})r(\mathrm{b}) \ln\left(\frac{m(\mathrm{b})r(\mathrm{b})}{c^{i\lambda}(\mathrm{b})}\right) \mathrm{db} - \mathrm{const} = \lambda \mathrm{D}(mr, c^{i\lambda}) - \mathrm{const}.$$

Since neither multiplication by a positive constant nor subtracting a constant changes the result of minimization, it is possible to rewrite the relation (3.24) as

$$r^{o\lambda} \in \operatorname*{Argmin}_{r \in \mathcal{S}_r} \mathrm{D}(mr, c^{i\lambda}).$$

Utilizing the definitions (3.23) and (3.24) as well as the non-negativity property of Kullback-Leibler divergence, it is possible to obtain the following set of inequalities

$$\begin{aligned} 0 \le \mathrm{E}^{r^{o\lambda}}[L] - \mathrm{E}^{r^{o}}[L] &\le \mathrm{E}^{r^{o\lambda}}[L] + \lambda \mathrm{D}(mr^{o\lambda}, \tilde{c}) - \mathrm{E}^{r^{o}}[L] \le \\ &\le \mathrm{E}^{r^{o}}[L] + \lambda \mathrm{D}(mr^{o}, \tilde{c}) - \mathrm{E}^{r^{o}}[L] = \lambda \mathrm{D}(mr^{o}, \tilde{c}) \overset{\lambda \to 0_+}{\longrightarrow} 0_+, \end{aligned}$$

where the symbol $\overset{\lambda \to 0_+}{\longrightarrow} 0_+$ denotes right convergence. This statement holds as long as Kullback-Leibler divergence is finite, which is, however, guaranteed from the assumption (2b).

In other words, the expected loss function connected with fully probabilistic design converges to the Markov decision process expected loss function. Written mathematically

$$\mathrm{E}^{r^{o\lambda}}[L] \longrightarrow \mathrm{E}^{r^{o}}[L].$$

This means that every Markov decision process optimal strategy can be arbitrarily well approximated by a strategy acquired using fully probabilistic design.

□

3.4 Discounted Formulation and Solution

Discounting is a term used to describe the decreasing preference of future rewards or, alternatively, losses. These are said to be *discounted* in order to reflect their present value. This phenomenon can be observed in many forms of human behavior from ordinary future planning to impulse control problems seen for example in eating habits [30] or substance abuse [21].

Indeed, one would intuitively prefer being given 10 dollars immediately rather than one week later. Consider that the present value of the 10 dollars next week is 1 dollar. In order to obtain the present value of the future reward, it needs to be multiplied by 0.1. This number which relates future rewards to their present value is called *discounting factor* and the principle of value decrease in relation to future is referred to as *temporal* discounting.

On the other hand, *probabilistic* discounting deals with rewards which are perceived as uncertain. Smaller amount of money, for instance being given 5 dollars with no uncertainty, could be preferred by many individuals before 10 dollars they may not be sure to receive. It was

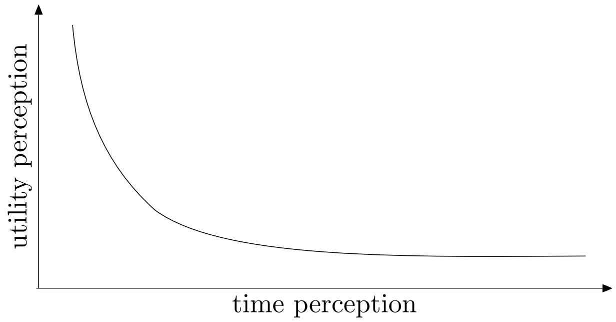


Figure 3.2: An example of decreasing time preference.

shown in [6] that both temporal and probabilistic discounting can be treated using the same mathematical models. For this reason it is possible to limit the study to the former case.

Discounting finds frequent application in investment analysis [10] for obtaining the present value of future cash flows. Other applications in finance include benefit-cost analysis [5], which aims to analyze whether a project is economically feasible by comparing its discounted future benefits with present losses. Capital budgeting [34] is another closely related application. The benefits of purchasing new equipment, expanding facilities or running new projects need to be weighed out by their present costs.

Although discounting is generally the most frequently utilized in economic fields, since the end of the last century it has also found its application in environmental economics [4], [26] for matters such as modeling greenhouse gas emissions. All of these applications are also discussed in [29].

Markov decision processes already incorporate the discounting factor and can therefore be used to solve the above mentioned tasks. Fully probabilistic design, however, lacked the necessary apparatus for dealing with problems which require discounting. This shortcoming was overcome in the bachelor's project [24]. The goal of this section and its subsections is hence to recollect the derivation of discounted fully probabilistic design.

3.4.1 Discounted Fully Probabilistic Design Is Non-Trivial

Discounted fully probabilistic design aims to weigh out the future losses as previously discussed. The weights used therefore serve as the discounting factors. The goal is hence to minimize a Kullback-Leibler divergence of the following form

$$\mathrm{D}(c^r, c^{iw}) \equiv \mathrm{E}^r\left[\sum_{t\in\mathcal{S}_t} w(\mathrm{s}_{t-1})\ln\left(\frac{m(\mathrm{s}_t|\mathrm{a}_t,\mathrm{s}_{t-1})r(\mathrm{a}_t|\mathrm{s}_{t-1})}{m^i(\mathrm{s}_t|\mathrm{a}_t,\mathrm{s}_{t-1})r^i(\mathrm{a}_t|\mathrm{s}_{t-1})}\right)\right], \tag{3.25}$$

where $w(\mathrm{s}_{t-1})$ serve as the weights for the given partial loss function. Ideal closed-loop density c^{iw} now possesses one more superscript in order to emphasize discounting applied in the form of weights.

Due to the fact that the time set $\mathcal{S}_t$ is finite, it is possible to exchange the order of the summation and integration in (3.25) yielding the integral form

$$\mathrm{D}(c^r, c^{iw}) = \sum_{t\in\mathcal{S}_t} \int_{(\mathcal{S}_s,\mathcal{S}_a)} m(\mathrm{s}_t|\mathrm{a}_t,\mathrm{s}_{t-1})r(\mathrm{a}_t|\mathrm{s}_{t-1})\ln\left(\frac{m(\mathrm{s}_t|\mathrm{a}_t,\mathrm{s}_{t-1})r(\mathrm{a}_t|\mathrm{s}_{t-1})}{m^i(\mathrm{s}_t|\mathrm{a}_t,\mathrm{s}_{t-1})r^i(\mathrm{a}_t|\mathrm{s}_{t-1})}\right)^{w(\mathrm{s}_{t-1})} \mathrm{d}(\mathrm{a}_t,\mathrm{s}_t).$$

Discounting weights w above have turned into exponents using the logarithmic power rule.

For the previous entry to follow the definition of Kullback-Leibler divergence (2.38), the centering density and the density in the numerator must be the same. This is achievable by redistributing some of the numerator into the denominator, i.e.

$$\ln\left(\frac{m_t r_t}{m_t^i r_t^i}\right)^{w_{t-1}} = \ln\left(\frac{m_t r_t}{[m_t^i r_t^i]^{w_{t-1}}[m_t r_t]^{1-w_{t-1}}}\right).$$

The simplified notation from the proof of Theorem 3.2.2 was used once again in order to save space.

All that is left to do is to ensure that the denominator is a probability density. A normalization factor therefore needs to be added and subtracted. Density c^{iw} then becomes

$$c^{iw}(\mathrm{b}) \equiv \prod_{t\in\mathcal{S}_t} \frac{[m^i(\mathrm{s}_t|\mathrm{a}_t,\mathrm{s}_{t-1})r^i(\mathrm{a}_t|\mathrm{s}_{t-1})]^{w(\mathrm{s}_{t-1})}[m(\mathrm{s}_t|\mathrm{a}_t,\mathrm{s}_{t-1})r(\mathrm{a}_t|\mathrm{s}_{t-1})]^{1-w(\mathrm{s}_{t-1})}}{\int\limits_{(\mathcal{S}_s,\mathcal{S}_a)} [m^i(\mathrm{s}_t|\mathrm{a}_t,\mathrm{s}_{t-1})r^i(\mathrm{a}_t|\mathrm{s}_{t-1})]^{w(\mathrm{s}_{t-1})}[m(\mathrm{s}_t|\mathrm{a}_t,\mathrm{s}_{t-1})r(\mathrm{a}_t|\mathrm{s}_{t-1})]^{1-w(\mathrm{s}_{t-1})}\mathrm{d}(\mathrm{s}_t,\mathrm{a}_t)}. \tag{3.26}$$

Let us denote the normalizing integral above as

$$\Phi(\mathrm{s}_{t-1}) \equiv \int\limits_{(\mathcal{S}_s,\mathcal{S}_a)} [m^i(\mathrm{s}_t|\mathrm{a}_t,\mathrm{s}_{t-1})r^i(\mathrm{a}_t|\mathrm{s}_{t-1})]^{w(\mathrm{s}_{t-1})}[m(\mathrm{s}_t|\mathrm{a}_t,\mathrm{s}_{t-1})r(\mathrm{a}_t|\mathrm{s}_{t-1})]^{1-w(\mathrm{s}_{t-1})}\mathrm{d}(\mathrm{s}_t,\mathrm{a}_t). \tag{3.27}$$

The choice of ideal closed-loop density as (3.26) then leads to a minimization of Kullback-Leibler divergence with an additional term

$$\mathrm{D}(c^r, c^{iw}) = \mathrm{E}^r\left[\sum_{t\in\mathcal{S}_t}\left(w(\mathrm{s}_{t-1})\ln\left(\frac{m(\mathrm{s}_t|\mathrm{a}_t,\mathrm{s}_{t-1})r(\mathrm{a}_t|\mathrm{s}_{t-1})}{m^i(\mathrm{s}_t|\mathrm{a}_t,\mathrm{s}_{t-1})r^i(\mathrm{a}_t|\mathrm{s}_{t-1})}\right) + \ln\left(\Phi(\mathrm{s}_{t-1})\right)\right)\right]. \tag{3.28}$$

Discounted fully probabilistic design formulation is therefore non-trivial and cannot be achieved through simply weighing down the future losses.

3.4.2 Correct Formulation and Solution

Let us introduce a new binary variable *pointer*, denoted p_t for the given epoch t, which influences the ideal densities m^i and r^i in the following manner

$$m^i(\mathrm{s}_t|\mathrm{a}_t,\mathrm{p}_t,\mathrm{s}_{t-1}) = \begin{cases} m^i(\mathrm{s}_t|\mathrm{a}_t,\mathrm{s}_{t-1}), & \text{if } \mathrm{p}_t = 1,\\ m(\mathrm{s}_t|\mathrm{a}_t,\mathrm{s}_{t-1}), & \text{if } \mathrm{p}_t = 0,\end{cases} \tag{3.29}$$

$$r^i(\mathrm{a}_t|\mathrm{p}_t,\mathrm{s}_{t-1}) = \begin{cases} r^i(\mathrm{a}_t|\mathrm{s}_{t-1}), & \text{if } \mathrm{p}_t = 1,\\ r(\mathrm{a}_t|\mathrm{p}_t,\mathrm{s}_{t-1}), & \text{if } \mathrm{p}_t = 0.\end{cases} \tag{3.30}$$

Pointer choice equal to zero sets the ideal densities to be identical to their real counterparts. Such choice corresponds with the leave to the fate option (3.18). System model m in (3.29) is independent of pointers, which was discussed in [27].

This is not true for decision rules, cf. (3.29) and (3.30) for the zero pointer case. In order to emphasize this fact, pointers are present as one of the conditional arguments.

Let the weights $w(\mathrm{s}_{t-1})$ be defined in a way so that they express the probability of pointers equal to one based on the current state of the system, i.e.

$$w(\mathrm{s}_{\tau-1}) \equiv p(\mathrm{p}_\tau = 1|\mathrm{s}_{\tau-1}) \quad \text{and} \quad w^i(\mathrm{s}_{\tau-1}) \equiv p^i(\mathrm{p}_\tau = 1|\mathrm{s}_{\tau-1}), \tag{3.31}$$

where the superscript i denotes the weights of an ideal closed loop. Probability densities corresponding with the zero pointers can then be expressed as $1 - w(\mathrm{s}_{t-1})$ thanks to the pointers' binary character. Analogical pattern applies for the ideal weights.

Due to the introduction of pointers, behaviors need to be redefined, as they now carry one more variable. Remembering the leave to the fate options (3.29) and (3.30), weighted closed-loop density can be defined as

$$\begin{aligned}\mathrm{c}^{rw}(\mathrm{b}) \equiv & c^r(\mathrm{s}_n, \mathrm{a}_n, \mathrm{p}_n, \mathrm{s}_{n-1}, \mathrm{a}_{n-1}, \mathrm{p}_{n-1}, ..., \mathrm{s}_1, \mathrm{a}_1, \mathrm{p}_1, \mathrm{s}_0) \\ = & \prod_{\tau=1}^{n} m(\mathrm{s}_\tau|\mathrm{a}_\tau, \mathrm{s}_{\tau-1}) r(\mathrm{a}_\tau|\mathrm{p}_\tau, \mathrm{s}_{\tau-1}) w^{\mathrm{p}_\tau}(\mathrm{s}_{\tau-1})(1 - w(\mathrm{s}_{\tau-1}))^{1-\mathrm{p}_\tau}, \end{aligned} \tag{3.32}$$

while its ideal equivalent has the following structure

$$\begin{aligned} c^{iw}(\mathrm{b}) = & \prod_{\tau=1}^{n} [m^i(\mathrm{s}_\tau|\mathrm{a}_\tau, \mathrm{s}_{\tau-1}) r^i(\mathrm{a}_\tau|\mathrm{s}_{\tau-1}) w^i(\mathrm{s}_{\tau-1})]^{\mathrm{p}_\tau} \\ & \times [m(\mathrm{s}_\tau|\mathrm{a}_\tau, \mathrm{s}_{\tau-1}) r(\mathrm{a}_\tau|\mathrm{p}_\tau, \mathrm{s}_{\tau-1})(1 - w^i(\mathrm{s}_{\tau-1}))]^{1-\mathrm{p}_\tau}. \end{aligned} \tag{3.33}$$

These discounted closed loop models result in the desired minimization of Kullback-Leibler divergence (3.25) with an additional term. Specifically it reads

$$\begin{aligned} \mathrm{D}(c^{rw}, c^{iw}) = & \mathrm{E}^r \left[\sum_{\tau=1}^{n} w(\mathrm{s}_{\tau-1}) \ln \left(\frac{m(\mathrm{s}_\tau|\mathrm{a}_\tau, \mathrm{s}_{\tau-1}) r(\mathrm{a}_\tau|\mathrm{p}_\tau = 1, \mathrm{s}_{\tau-1})}{m^i(\mathrm{s}_\tau|\mathrm{a}_\tau, \mathrm{s}_{\tau-1}) r^i(\mathrm{a}_\tau|\mathrm{s}_{\tau-1})} \right) \right] \\ & + \sum_{\tau=1}^{n} \mathrm{D}\left([w(\mathrm{s}_{\tau-1}), 1 - w(\mathrm{s}_{\tau-1})], [w^i(\mathrm{s}_{\tau-1}), 1 - w^i(\mathrm{s}_{\tau-1})] \right), \end{aligned} \tag{3.34}$$

where the second sum of divergencies zeroes out in case of the leave to the fate option applied to weights. This means that the weights w are set equal to their ideals w^i. In other words, there is no preference in the choice of pointers.

The choice of closed-loop densities in the form (3.32) and (3.33) therefore offers a promising way of formulating the discounted fully probabilistic design. All that remains is to show how to construct the decision rules r in (3.34). This will be the aim of the following theorem.

Theorem 3.4.1 (Discounted fully probabilistic design)**.** Discounted fully probabilistic design optimal decision rules which minimize Kullback-Leibler divergence of the closed loop model (3.32) to its ideal counterpart (3.33)

- coincide with the optimal strategy from the standard formulation in case that pointers equal one, and
- become deterministic for the zero pointer case. Optimal actions are selected directly as

$$\mathrm{a}^o(\mathrm{s}_{t-1}) \in \operatorname*{argmin}_{\mathrm{a}_t \in \mathcal{S}_a} k(\mathrm{a}_t, \mathrm{s}_{t-1}), \text{ with } k(\mathrm{a}_t, \mathrm{s}_{t-1}) = \int_{\mathcal{S}_s} m(\mathrm{s}_t|\mathrm{a}_t, \mathrm{s}_{t-1}) \ln \left(\frac{1}{h(\mathrm{s}_t)} \right) \mathrm{d}\mathrm{s}_t, \tag{3.35}$$

being the minimized function. Normalizing factors

$$h(\mathrm{s}_{t-1}) = h^i(\mathrm{s}_{t-1}) w^i(\mathrm{s}_{t-1}) + \kappa(\mathrm{s}_{t-1})(1 - w^i(\mathrm{s}_{t-1})) \text{ and} \tag{3.36}$$

$$h^i(\mathrm{s}_{t-1}) = \int_{\mathcal{S}_a} r^i(\mathrm{a}_t|\mathrm{s}_{t-1}) \exp(-d(\mathrm{a}_t, \mathrm{s}_{t-1})) \mathrm{d}\mathrm{a}_t, \tag{3.37}$$

ensuring the backward functional recursion to the initial state s_0 and

$$d(\mathrm{a}_t, \mathrm{s}_{t-1}) = \int\limits_{\mathcal{S}_s} m(\mathrm{s}_t|\mathrm{a}_t, \mathrm{s}_{t-1}) \ln\left(\frac{m(\mathrm{s}_t|\mathrm{a}_t, \mathrm{s}_{t-1})}{h(\mathrm{s}_t) m^i(\mathrm{s}_t|\mathrm{a}_t, \mathrm{s}_{t-1})}\right) \mathrm{ds}_t \text{ and} \tag{3.38}$$

$$\kappa(\mathrm{s}_{t-1}) = \exp(-k(\mathrm{a}^o(\mathrm{s}_{t-1}), \mathrm{s}_{t-1})), \tag{3.39}$$

being the final two auxiliary functions. Pointers are then generated according to the optimal weights

$$w^o(\mathrm{s}_{t-1}) = w^i(\mathrm{s}_{t-1}) \frac{h^i(\mathrm{s}_{t-1})}{h(\mathrm{s}_{t-1})}.$$

Proof. Substituting extended actions $(\mathrm{a}_t, \mathrm{p}_t)$ into Theorem 3.2.3 yields

$$d(\mathrm{a}_t, \mathrm{p}_t, \mathrm{s}_{t-1}) = \int\limits_{\mathcal{S}_s} m(\mathrm{s}_t|\mathrm{a}_t, \mathrm{p}_t, \mathrm{s}_{t-1}) \ln\left(\frac{m(\mathrm{s}_t|\mathrm{a}_t, \mathrm{p}_t, \mathrm{s}_{t-1})}{h(\mathrm{s}_t) m^i(\mathrm{s}_t|\mathrm{a}_t, \mathrm{p}_t, \mathrm{s}_{t-1})}\right) \mathrm{ds}_t.$$

Under the leave to the fate option (3.29) density m^i in the denominator cancels out with m^i in the numerator, leaving only h. Remembering that pointers have no influence on the system model, see the discussion under (3.29) on page 39, together with their binary character, function d becomes

$$\begin{aligned} d(\mathrm{a}_t, \mathrm{p}_t, \mathrm{s}_{t-1}) = {} & \mathrm{p}_t \int\limits_{\mathcal{S}_s} m(\mathrm{s}_t|\mathrm{a}_t, \mathrm{s}_{t-1}) \ln\left(\frac{m(\mathrm{s}_t|\mathrm{a}_t, \mathrm{s}_{t-1})}{h(\mathrm{s}_t) m^i(\mathrm{s}_t|\mathrm{a}_t, \mathrm{s}_{t-1})}\right) \mathrm{ds}_t \\ & + (1 - \mathrm{p}_t) \int\limits_{\mathcal{S}_s} m(\mathrm{s}_t|\mathrm{a}_t, \mathrm{s}_{t-1}) \ln\left(\frac{1}{h(\mathrm{s}_t)}\right) \mathrm{ds}_t \end{aligned}$$

Here, the first integral is precisely the d from standard fully probabilistic design (3.13) whereas the second integral is the function k from the theorem statement, see (3.35). Taking into account the multiplication by p_t and $1 - \mathrm{p}_t$, the extended d function can be expressed as

$$d(\mathrm{a}_t, \mathrm{p}_t, \mathrm{s}_{t-1}) = \begin{cases} d(\mathrm{a}_t, \mathrm{s}_{t-1}), & \text{for } \mathrm{p}_t = 1 \\ k(\mathrm{a}_t, \mathrm{s}_{t-1}), & \text{for } \mathrm{p}_t = 0. \end{cases} \tag{3.40}$$

Moving on to the normalizing factor h, it can be once again obtained by substituting the extended actions back into the h from standard fully probabilistic design, i.e.

$$h(\mathrm{s}_{t-1}) = \int\limits_{\mathcal{S}_a} \sum_{\mathrm{p}_t = 0}^{1} r^i(\mathrm{a}_t, \mathrm{p}_t|\mathrm{s}_{t-1}) \exp[-d(\mathrm{a}_t, \mathrm{p}_t, \mathrm{s}_{t-1})] \mathrm{da}_t.$$

The summation is present to eliminate the dependency of h on pointers.

Extended d can be seen above whereas ideal decision rules with extended actions can be reformulated using chain rule and weights definition (3.31), thus

$$r^i(\mathrm{a}_t, \mathrm{p}_t|\mathrm{s}_{t-1}) = r^i(\mathrm{a}_t|\mathrm{p}_t, \mathrm{s}_{t-1}) p^i(\mathrm{p}_t|\mathrm{s}_{t-1}) = \begin{cases} r^i(\mathrm{a}_t|\mathrm{s}_{t-1}) w^i(\mathrm{s}_{t-1}), & \text{for } \mathrm{p}_t = 1 \\ r(\mathrm{a}_t|\mathrm{p}_t = 0, \mathrm{s}_{t-1})(1 - w^i(\mathrm{s}_{t-1})), & \text{for } \mathrm{p}_t = 0, \end{cases} \tag{3.41}$$

where density $r^i(\mathrm{a}_t|\mathrm{p}_t, \mathrm{s}_{t-1})$ was expressed using leave to the fate option (3.30).

The final form of function h can be obtained by substituting the extended d from (3.40) and ideal decision rules (3.41) back to its defining integral. Therefore

$$\begin{aligned} h(\mathrm{s}_{t-1}) = {} & w^i(\mathrm{s}_{t-1}) \int\limits_{\mathcal{S}_a} r^i(\mathrm{a}_t|\mathrm{s}_{t-1}) \exp[-d(\mathrm{a}_t, \mathrm{s}_{t-1})] \mathrm{d}\mathrm{a}_t \\ & + (1 - w^i(\mathrm{s}_{t-1})) \int\limits_{\mathcal{S}_a} r(\mathrm{a}_t|\mathrm{p}_t = 0, \mathrm{s}_{t-1}) \exp[-k(\mathrm{a}_t, \mathrm{s}_{t-1})] \mathrm{d}\mathrm{a}_t. \end{aligned} \tag{3.42}$$

Apparently the case with pointers equal to one in the first row of (3.42) coincides with the h from standard fully probabilistic design. However, the case with zero pointers is not familiar and it will be paid more attention to in the following paragraphs.

It can be seen that function k obtained from function d for the zero pointer case notably resembles the definition of function φ from Theorem 3.2.4. The assumption of independence of the loss function, i.e. the logarithm of inverse h from (3.35) in this case, on decision rules holds. Therefore in epochs with zero pointers, the actions are generated deterministically according to

$$\mathrm{a}^o(\mathrm{s}_{t-1}) \in \operatorname*{Argmin}_{\mathrm{a}_t \in \mathcal{S}_a} k(\mathrm{a}_t, \mathrm{s}_{t-1}) = \operatorname*{Argmin}_{\mathrm{a}_t \in \mathcal{S}_a} \int\limits_{\mathcal{S}_s} m(\mathrm{s}_t|\mathrm{a}_t, \mathrm{s}_{t-1}) \ln \left(\frac{1}{h(\mathrm{s}_t)} \right) \mathrm{d}\mathrm{s}_t. \tag{3.43}$$

Decision rules r in (3.42) are hence deterministic and generate optimal actions a^o dependent on the current state of the system.

Substituting this knowledge back into (3.42), the final form of the normalizing factor h reads

$$h(\mathrm{s}_{t-1}) = w^i(\mathrm{s}_{t-1}) h^i(\mathrm{s}_{t-1}) + (1 - w^i(\mathrm{s}_{t-1})) \kappa(\mathrm{s}_{t-1}),$$

where κ represents the exponential of function k with optimal actions a^o, see (3.39).

All that is left to show is the form of optimal decision rules r^o. Having derived $d(\mathrm{a}_t, \mathrm{p}_t, \mathrm{s}_{t-1})$ and $r^i(\mathrm{a}_t, \mathrm{p}_t|\mathrm{s}_{t-1})$ for the discounted design, it is possible to substitute these into (3.15), yielding

$$\begin{aligned} r^o(\mathrm{a}_t, \mathrm{p}_t|\mathrm{s}_{t-1}) = {} & \mathrm{p}_t \frac{r^i(\mathrm{a}_t|\mathrm{p}_t = 1, \mathrm{s}_{t-1}) p^i(\mathrm{p}_t = 1|\mathrm{s}_{t-1}) \exp[-d(\mathrm{a}_t, \mathrm{p}_t = 1, \mathrm{s}_{t-1})]}{h(\mathrm{s}_{t-1})} \\ & + (1 - \mathrm{p}_t) \frac{r(\mathrm{a}_t|\mathrm{p}_t = 0, \mathrm{s}_{t-1}) p^i(\mathrm{p}_t = 0|\mathrm{s}_{t-1}) \exp[-d(\mathrm{a}_t, \mathrm{p}_t = 0, \mathrm{s}_{t-1})]}{h(\mathrm{s}_{t-1})}. \end{aligned}$$

Once again utilizing the derived ideal decision rules with extended actions (3.41), the final form of r^o reads

$$r^o(\mathrm{a}_t, \mathrm{p}_t|\mathrm{s}_{t-1}) = \begin{cases} \dfrac{w^i(\mathrm{s}_{t-1}) r^i(\mathrm{a}_t|\mathrm{s}_{t-1}) \exp[-d(\mathrm{a}_t, \mathrm{s}_{t-1})]}{h(\mathrm{s}_{t-1})} & \text{if } \mathrm{p}_t = 1, \\ \dfrac{(1 - w^i(\mathrm{s}_{t-1})) r(\mathrm{a}_t|\mathrm{p}_t = 0, \mathrm{s}_{t-1}) \exp[-k(\mathrm{a}_t, \mathrm{s}_{t-1})]}{h(\mathrm{s}_{t-1})} & \text{if } \mathrm{p}_t = 0. \end{cases} \tag{3.44}$$

These relations can be directly used to obtain optimal weights w^o by integrating over the actions set. Let us demonstrate this on the less complex case with pointers equal to one, i.e.

$$w^o(\mathrm{s}_{t-1}) = \int\limits_{\mathcal{S}_a} r^o(\mathrm{a}_t, \mathrm{p}_t = 1|\mathrm{s}_{t-1}) \mathrm{d}\mathrm{a}_t = \frac{w^i(\mathrm{s}_{t-1})}{h(\mathrm{s}_{t-1})} \int\limits_{\mathcal{S}_a} r^i(\mathrm{a}_t|\mathrm{s}_{t-1}) \exp[-d(\mathrm{a}_t, \mathrm{s}_{t-1})] \mathrm{d}\mathrm{a}_t.$$

The last integral above is precisely the normalizing factor from standard fully probabilistic design (3.37). Optimal weights can hence be expressed simply as

$$w^o(\mathrm{s}_{t-1}) = w^i(\mathrm{s}_{t-1})\frac{h^i(\mathrm{s}_{t-1})}{h(\mathrm{s}_{t-1})}. \tag{3.45}$$

Optimal weights for pointers equal to zero can be then obtained directly as $1 - w^o(\mathrm{s}_{t-1})$.

Now that the optimal decision rules with extended actions are known, it might be useful to move the pointers into the conditional part, yielding $r^o(\mathrm{a}_t|\mathrm{p}_t = 1, \mathrm{s}_{t-1})$. Such density is necessary when both the current state of the system as well as the current pointer are known and the decision maker wishes to predict the best action choice based on this available information. Conditional density r^o can be expressed using chain rule as

$$r^o(\mathrm{a}_t|\mathrm{p}_t = 1, \mathrm{s}_{t-1}) = \frac{r^o(\mathrm{a}_t, \mathrm{p}_t = 1|\mathrm{s}_{t-1})}{w^o(\mathrm{s}_{t-1})} = \frac{r^i(\mathrm{a}_t|\mathrm{s}_{t-1})\exp[-d(\mathrm{a}_t, \mathrm{s}_{t-1})]}{h^i(\mathrm{s}_{t-1})}.$$

The last equality above was acquired after substituting (3.44) and (3.45) for r^o and w^o respectively.

The previous computations hold for pointers equal to one. The zero case can be derived analogically once again substituting r^o definition (3.41) and $1 - w^o$, thus

$$r^o(\mathrm{a}_t|\mathrm{p}_t = 0, \mathrm{s}_{t-1}) = \frac{r^o(\mathrm{a}_t, \mathrm{p}_t = 0|\mathrm{s}_{t-1})}{1 - w^o(\mathrm{s}_{t-1})} = \frac{r^o(\mathrm{a}_t|\mathrm{p}_t = 0, \mathrm{s}_{t-1})\exp[-k(\mathrm{a}_t, \mathrm{s}_{t-1})]}{\kappa(\mathrm{s}_{t-1})}.$$

As both the left and the right hand side contain density $r^o(\mathrm{a}_t|\mathrm{p}_t = 0, \mathrm{s}_{t-1})$, in order for the equality to hold, the exponential of k needs to be equal to κ. It can be seen from the definition (3.39) that this is only true for actions (3.43). Decision rules $r^o(\mathrm{a}_t|\mathrm{p}_t = 0, \mathrm{s}_{t-1})$ therefore generate deterministic optimal actions turning $\exp(k)$ into κ.

□

3.5 Numerical Simulations

This part numerically illustrates the theory and verifies the working hypothesis on equality of the optimal forgetting and discounting factor. This hypothesis was already studied in the research project [25], where there was discovered dependency of the simulation on the chosen random number generator seed value. This work therefore handles the dependency by averaging multiple simulations with different seeds to see if they converge to desired results.

In the previous subsection it was discussed that the weights (3.31) serve as the discounting factors relating future losses to their current value. The motivation of setting the discounting factor as weights is the following. After applying the decision rule version with the leave to the fate option (3.30) it can be seen in the resulting form of the Kullback-Leibler divergence (3.34) that the only decision rule densities with pointers equal to one enter the optimization. The zero case discards the density from the optimization by canceling out it out with the density in the numerator. That is where the variable pointer gets its name. It serves as a random selector of the epochs in which the process shall be optimized.

Forgetting factor λ_t in (2.40) expresses the probability that no model variations occurred in between times $t - 1$ and t. Estimated model should thus represent an appropriate description of reality whenever the forgetting factor equals one. It would therefore seem reasonable to keep optimizing the process whenever the estimated model is considered to hold. In probabilistic terms, this means to set the discounting factor equal to one whenever the forgetting factor happens to be one.

Our question was if this hypothesis is sound. Forgetting factor value could then serve as a good estimate of the discounting factor. The hypothesis was tested in the mentioned research project where the results did not seem favorable.

During the course of the work we came up with yet another hypothesis regarding the effect of discounting on decision processes when the model variations are caused by imprecise modeling. System models acquired through model estimation, see Section 2.3, are always imperfect when describing the real system behavior. These imperfect estimates enter the optimal decision rule design discussed in Theorem 3.2.3. However, new states are generated according to a valid system model m. When this valid model varies in time, the quality of its estimate decreases and becomes even more imprecise over time. The loss accumulates and the process performance drops.

This is why it might be profitable to omit some epochs from the optimization in order to limit the accumulation of loss in systems with time inconsistent models. Stopping the optimization before the valid and the estimated model stride too far away from each other might solve this problem and eventually improve the overall performance. This thought motivates our simulations.

Simulation Setup

The decision horizon n was set to 200 time epochs. Next, a state set containing three available states and an action set with two available actions were chosen as

$$\mathcal{S}_s = \{1, 2, 3\} \quad \text{and} \quad \mathcal{S}_a = \{1, 2\}.$$

The initial state of the system s_0 was taken to be the state 1.

In order to test the hypothesis of equality of the discounting and forgetting factor values, denoted w and λ respectively, a discrete grid of these values was defined. It reads

$$w, \lambda \in \{0, 0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9, 1\}.$$

The dropping of the time indices embodies the fact that both factors were believed to remain constant in the course of time.

Before launching the simulations, it is necessary to define ideal probability densities r^i and m^i as well as the valid system model m. Their exact definitions can be seen in Tables 3.1, 3.2 and 3.3 below. These were formulated in a way that would emphasize the preferred state-action combination, which was set to be the couple

$$(\mathrm{s}_t, \mathrm{a}_t) = (3, 2) \quad \text{for} \quad t = 1, 2, 3, ..., n.$$

This means that the decision maker wishes to keep reaching state 3 as much as possible during the course of the process.

Since one of the goals was to test the effect of discounting on decision processes with unknown and varying system model, this variance also needed to be incorporated. New states were therefore generated from a linearly degenerating model given by $\frac{t}{n} m^{0.85} + \frac{n-t}{n} m$, which was then normalized.

$r^i(\mathrm{a}_t\vert\mathrm{s}_{t-1})$	$\mathrm{a}_t = 1$	$\mathrm{a}_t = 2$
$\mathrm{s}_{t-1} = 1$	0.4167	0.5833
$\mathrm{s}_{t-1} = 2$	0.4286	0.5714
$\mathrm{s}_{t-1} = 3$	0.3077	0.6923

Table 3.1: Ideal decision rule matrix.

$m(\mathrm{s}_t\vert\mathrm{a}_t, \mathrm{s}_{t-1})$	$\mathrm{a}_t = 1$	$\mathrm{a}_t = 2$
$\mathrm{s}_t = 1$, $\mathrm{s}_{t-1} = 1$	0.1951	0.2381
$\mathrm{s}_t = 1$, $\mathrm{s}_{t-1} = 2$	0.3333	0.3256
$\mathrm{s}_t = 1$, $\mathrm{s}_{t-1} = 3$	0.3409	0.3542
$\mathrm{s}_t = 2$, $\mathrm{s}_{t-1} = 1$	0.3171	0.3571
$\mathrm{s}_t = 2$, $\mathrm{s}_{t-1} = 2$	0.2857	0.2093
$\mathrm{s}_t = 2$, $\mathrm{s}_{t-1} = 3$	0.4091	0.3958
$\mathrm{s}_t = 3$, $\mathrm{s}_{t-1} = 1$	0.4878	0.4048
$\mathrm{s}_t = 3$, $\mathrm{s}_{t-1} = 2$	0.3810	0.4651
$\mathrm{s}_t = 3$, $\mathrm{s}_{t-1} = 3$	0.2500	0.2500

Table 3.2: Real system model matrix.

$m^i(\mathrm{s}_t\vert\mathrm{a}_t, \mathrm{s}_{t-1})$	$\mathrm{a}_t = 1$	$\mathrm{a}_t = 2$
$\mathrm{s}_t = 1$, $\mathrm{s}_{t-1} = 1$	0.3158	0.2286
$\mathrm{s}_t = 1$, $\mathrm{s}_{t-1} = 2$	0.3070	0.3409
$\mathrm{s}_t = 1$, $\mathrm{s}_{t-1} = 3$	0.3774	0.1154
$\mathrm{s}_t = 2$, $\mathrm{s}_{t-1} = 1$	0.2632	0.2857
$\mathrm{s}_t = 2$, $\mathrm{s}_{t-1} = 2$	0.3728	0.2955
$\mathrm{s}_t = 2$, $\mathrm{s}_{t-1} = 3$	0.3302	0.1923
$\mathrm{s}_t = 3$, $\mathrm{s}_{t-1} = 1$	0.4211	0.4857
$\mathrm{s}_t = 3$, $\mathrm{s}_{t-1} = 2$	0.3202	0.3636
$\mathrm{s}_t = 3$, $\mathrm{s}_{t-1} = 3$	0.2925	0.6923

Table 3.3: Ideal system model matrix.

Finally, all 200 steps were run under 50 different seeds ranging from 1 to 50. The results were recorded in form of percentage occurrences of each state out of the 200 epochs. The 50 sets of results were subsequently averaged in order to eliminate the dependency on the random number generator. For clarity, they are plotted in form of heat maps. Now that all the preliminaries have been stated, it is possible to demonstrate the results and verify the following.

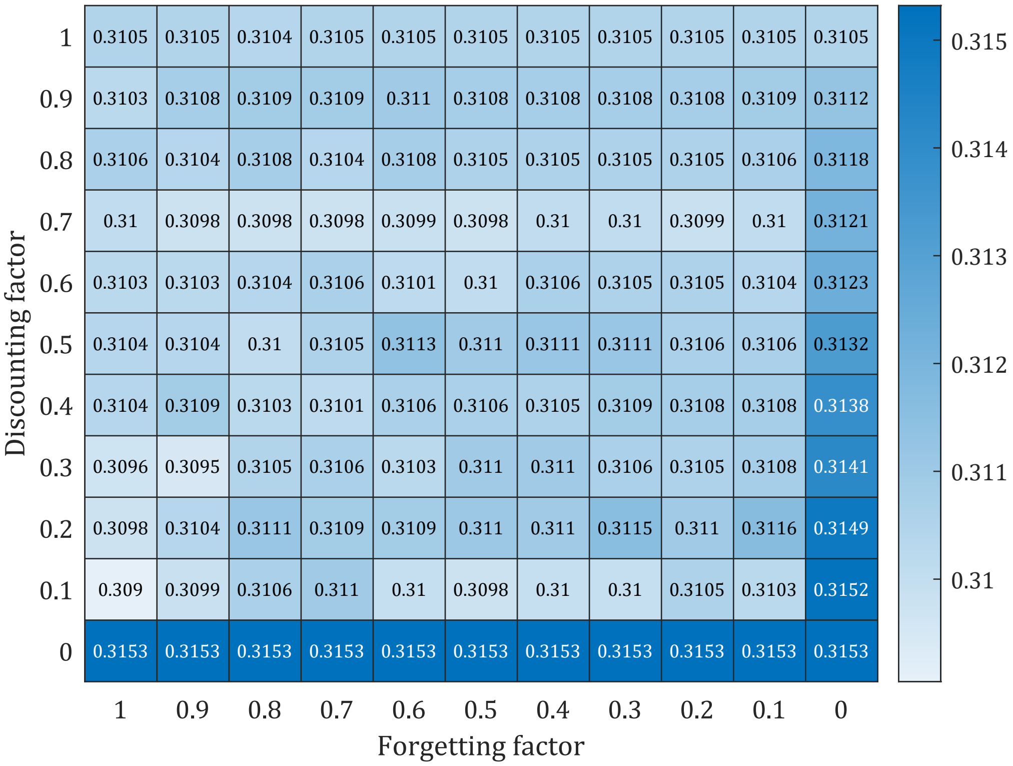
Hypothesis that forgetting factor is an adequate estimate of the discounting factor

The hypothesis of the forgetting factor to be a valid estimate of the discounting factor would correspond with the best results observed on the diagonal of the preferred state's heat map.

Since the preferences were chosen as the state 3 and action 2, it can be seen in Tables 3.2 and 3.3 that the third state has higher values of probability densities than the remaining ones. The same applies to the second action in Table 3.1. However, as the leave to the fate option (3.30) annuls[4] the action preference in some of the epochs, the preference on actions is not quite as consistent as it is in case of states.

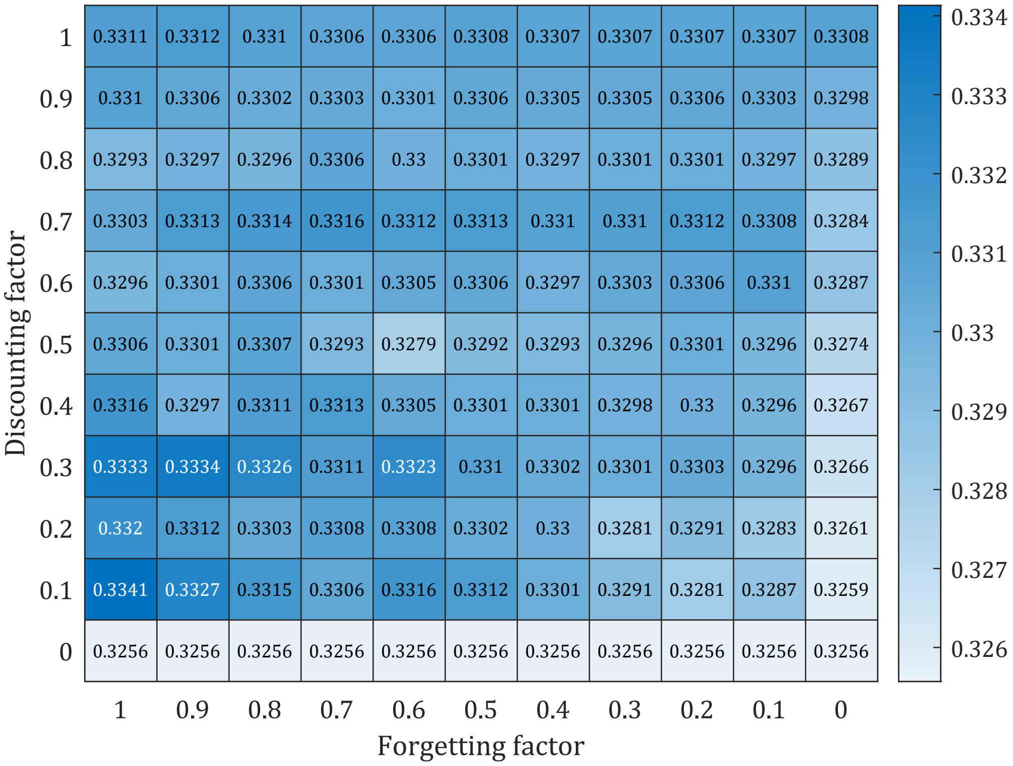
Regarding the hypothesis of discounting factor estimation, the best results in each row of the heat map Figure 3.5 appeared near the diagonal. However, the differences between the highest and lowest percentage occurrence were mostly in the order of third decimal place. They were therefore not sufficiently convincing to draw reliable conclusions. Even after removing the dependence on random number generator seed observed in the previous work, this hypothesis cannot be convincingly accepted or refuted.

[4] See the discussion before Simulation Setup.



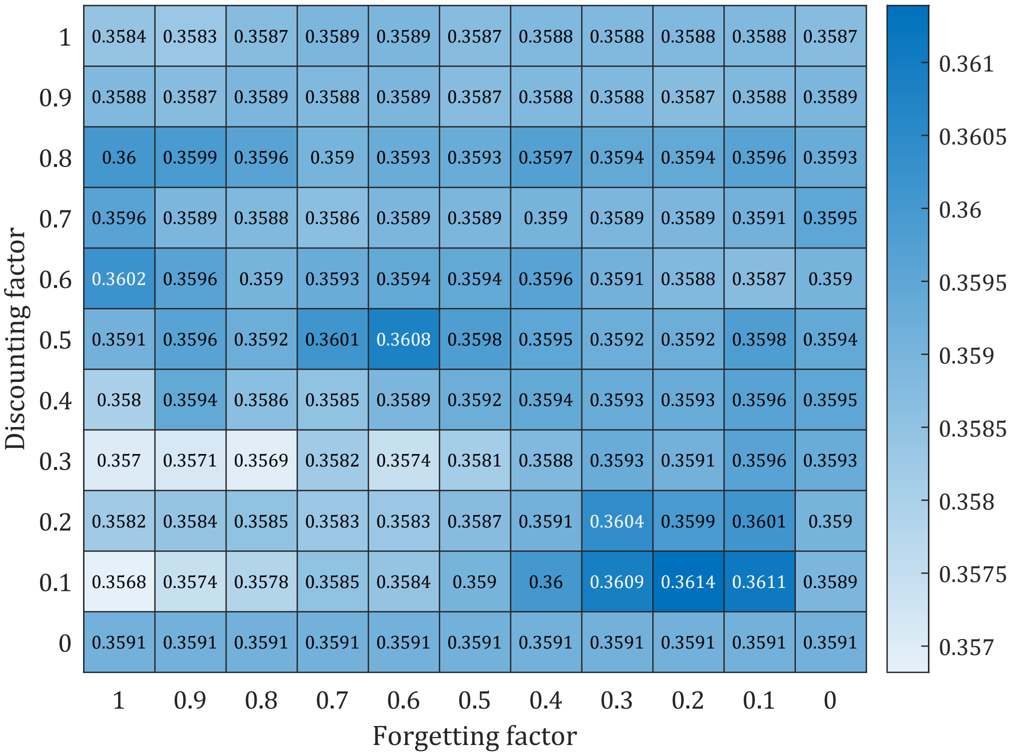
Discounting factor \ Forgetting factor	1	0.9	0.8	0.7	0.6	0.5	0.4	0.3	0.2	0.1	0
1	0.3105	0.3105	0.3104	0.3105	0.3105	0.3105	0.3105	0.3105	0.3105	0.3105	0.3105
0.9	0.3103	0.3108	0.3109	0.3109	0.311	0.3108	0.3108	0.3108	0.3108	0.3109	0.3112
0.8	0.3106	0.3104	0.3108	0.3104	0.3108	0.3105	0.3105	0.3105	0.3105	0.3106	0.3118
0.7	0.31	0.3098	0.3098	0.3098	0.3099	0.3098	0.31	0.31	0.3099	0.31	0.3121
0.6	0.3103	0.3103	0.3104	0.3106	0.3101	0.31	0.3106	0.3105	0.3105	0.3104	0.3123
0.5	0.3104	0.3104	0.31	0.3105	0.3113	0.311	0.3111	0.3111	0.3106	0.3106	0.3132
0.4	0.3104	0.3109	0.3103	0.3101	0.3106	0.3106	0.3105	0.3109	0.3108	0.3108	0.3138
0.3	0.3096	0.3095	0.3105	0.3106	0.3103	0.311	0.311	0.3106	0.3105	0.3108	0.3141
0.2	0.3098	0.3104	0.3111	0.3109	0.3109	0.311	0.311	0.3115	0.311	0.3116	0.3149
0.1	0.309	0.3099	0.3106	0.311	0.31	0.3098	0.31	0.31	0.3105	0.3103	0.3152
0	0.3153	0.3153	0.3153	0.3153	0.3153	0.3153	0.3153	0.3153	0.3153	0.3153	0.3153

Figure 3.3: Occurrences of state 1 for different factor combinations.



Discounting factor \ Forgetting factor	1	0.9	0.8	0.7	0.6	0.5	0.4	0.3	0.2	0.1	0
1	0.3311	0.3312	0.331	0.3306	0.3306	0.3308	0.3307	0.3307	0.3307	0.3307	0.3308
0.9	0.331	0.3306	0.3302	0.3303	0.3301	0.3306	0.3305	0.3305	0.3306	0.3303	0.3298
0.8	0.3293	0.3297	0.3296	0.3306	0.33	0.3301	0.3297	0.3301	0.3301	0.3297	0.3289
0.7	0.3303	0.3313	0.3314	0.3316	0.3312	0.3313	0.331	0.331	0.3312	0.3308	0.3284
0.6	0.3296	0.3301	0.3306	0.3301	0.3305	0.3306	0.3297	0.3303	0.3306	0.331	0.3287
0.5	0.3306	0.3301	0.3307	0.3293	0.3279	0.3292	0.3293	0.3296	0.3301	0.3296	0.3274
0.4	0.3316	0.3297	0.3311	0.3313	0.3305	0.3301	0.3301	0.3298	0.33	0.3296	0.3267
0.3	0.3333	0.3334	0.3326	0.3311	0.3323	0.331	0.3302	0.3301	0.3303	0.3296	0.3266
0.2	0.332	0.3312	0.3303	0.3308	0.3308	0.3302	0.33	0.3281	0.3291	0.3283	0.3261
0.1	0.3341	0.3327	0.3315	0.3306	0.3316	0.3312	0.3301	0.3291	0.3281	0.3287	0.3259
0	0.3256	0.3256	0.3256	0.3256	0.3256	0.3256	0.3256	0.3256	0.3256	0.3256	0.3256

Figure 3.4: Occurrences of state 2 for different factor combinations.



Discounting factor \ Forgetting factor	1	0.9	0.8	0.7	0.6	0.5	0.4	0.3	0.2	0.1	0
1	0.3584	0.3583	0.3587	0.3589	0.3589	0.3587	0.3588	0.3588	0.3588	0.3588	0.3587
0.9	0.3588	0.3587	0.3589	0.3588	0.3589	0.3587	0.3588	0.3588	0.3587	0.3588	0.3589
0.8	0.36	0.3599	0.3596	0.359	0.3593	0.3593	0.3597	0.3594	0.3594	0.3596	0.3593
0.7	0.3596	0.3589	0.3588	0.3586	0.3589	0.3589	0.359	0.3589	0.3589	0.3591	0.3595
0.6	0.3602	0.3596	0.359	0.3593	0.3594	0.3594	0.3596	0.3591	0.3588	0.3587	0.359
0.5	0.3591	0.3596	0.3592	0.3601	0.3608	0.3598	0.3595	0.3592	0.3592	0.3598	0.3594
0.4	0.358	0.3594	0.3586	0.3585	0.3589	0.3592	0.3594	0.3593	0.3593	0.3596	0.3595
0.3	0.357	0.3571	0.3569	0.3582	0.3574	0.3581	0.3588	0.3593	0.3591	0.3596	0.3593
0.2	0.3582	0.3584	0.3585	0.3583	0.3583	0.3587	0.3591	0.3604	0.3599	0.3601	0.359
0.1	0.3568	0.3574	0.3578	0.3585	0.3584	0.359	0.36	0.3609	0.3614	0.3611	0.3589
0	0.3591	0.3591	0.3591	0.3591	0.3591	0.3591	0.3591	0.3591	0.3591	0.3591	0.3591

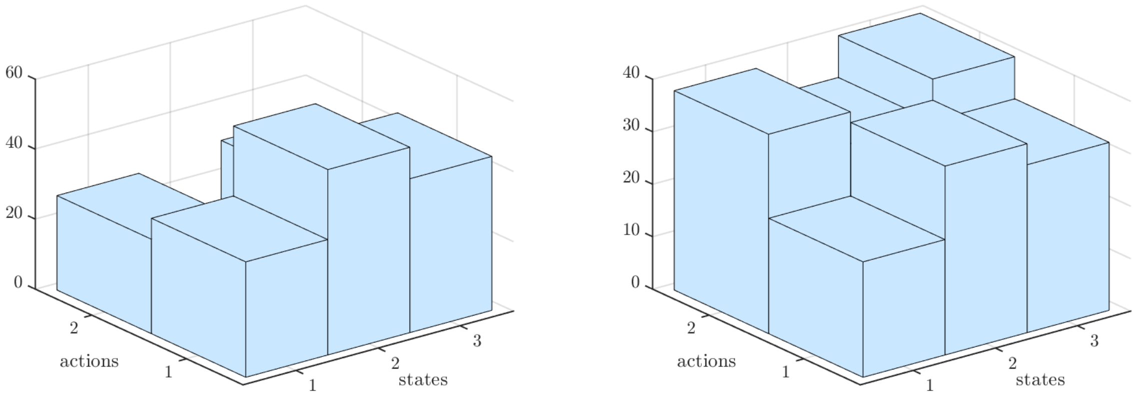
Figure 3.5: Occurrences of state 3 for different factor combinations.

Hypothesis that discounting factor improves model performance under imprecise modeling

As discussed in Simulation Setup, modeling error was achieved by modifying the density m, which entered the fully probabilistic design. The system model used for generating new states and the model used for decision rule design were hence two different densities. The mismatch between the models might have caused loss accumulation which decreased the system's performance.

The hypothesis here was to test whether discounting increased the occurrence of the preferred state throughout the process compared to the case when no discounting was used, that is, to test if the process' performance improved under discounting. Standard fully probabilistic design is obtained when setting the discounting factor to one. We were therefore comparing the first row of the heat map in Figure 3.5 with the remaining rows to see if higher occurrence could be found.

The best results overall were found for the combination of discounting factor 0.1 and forgetting factor 0.2. This simulation is presented in Figure 3.6 in form of histograms. The simulation shown in Figure 3.6a has an absolute occurrence of state 3 equal to 74. Comparing this result with Figure 3.6b, which contains the best results out of all forgetting factor values, it still has the absolute occurrence of the preferred state equal to 72. Even for the best possible choice of the forgetting factor, discounting hence still managed to slightly improve the overall performance.



(a) Discounting factor 0.1 and forgetting factor 0.2.

(b) Discounting factor 1 and forgetting factor 0.7.

Figure 3.6: Comparing the best results obtained with and without discounting (be aware of different scales).

Summary

The goal of the current section was to check hypotheses regarding how to set the discounting factor value for optimal performance as well as examining the effect of discounting in tasks with time-varying system models.

When studying the first hypothesis, the best results were not observed for the exact same value of the two factors, however, the highest occurrences could be seen around the diagonal of the studied heat map. Despite this fact, the difference in values between these optimal occurrences and occurrences detected for more distant factor combinations have been very subtle. In order to carry out conclusions about the truthfulness of the hypothesis, more significant differences would need to be observed.

The second idea was to check whether discounting helps when dealing with time-inconsistent system models. When comparing the values from the first row in Figure 3.5 with the rest of the heat map, it can be seen that discounting improved the system performance for the combination of discounting and forgetting factor of 0.1 and 0.2 respectively. The change in percentage was, however, not big enough for this result to be considered significant.

Furthermore, it can be seen in all of the provided figures that the choice of discount factor in some cases led to performance deterioration when compared to the non-discounted case. It is hence important to pay close attention when choosing the discounting factor value in order to prevent this situation.

Chapter 4

Conclusion

Fully probabilistic design of decision strategy was equipped with discounting to reflect the decreasing preference of future gains. In order to ensure the design's adaptive property, Bayesian estimation was incorporated to allow a construction of parametric models. They face the uncertainty of modeling whenever no exact information is given about how the system evolves over time. In order to further increase the design's adaptivity, forgetting has been incorporated to address the issue of time varying models.

Besides reintroducing the theoretical foundation established in the previous works [24] and [25], the thesis also examined the effect of discounting on decision processes under uncertainty. The objective was to verify if discounting improved the performance of the process when dealing with an unknown system model. The main idea was to stop the optimization before the model could significantly degrade, which would result in an increased loss. A slight improvement could have been observed, however, the change was not significant enough to carry out serious conclusions.

The hypothesis of the forgetting factor being an adequate choice of the discounting factor after handling the dependency on random number generator seed discovered in the previous work was not denied but could have been supported only weakly. The best results were observed near the same value of both factors, however, the differences were not very significant. The question of careful handling of the two factor values was also raised as some combinations led to obvious performance decrease.

The thesis brought together the research on fully probabilistic design, discounting and system model estimation, enabling it to cover much wider range of decision tasks. Relationship between discounting and model estimation was also examined. Similarly to the previous hypothesis, however, it

Further research might hence focus on the loss function of the given process, which might hint a more promising way of choosing the optimal factor values. The work has also been limited to elaborating decision processes with discrete states and actions. In order to further generalize the fully probabilistic design, continuous spaces should be examined as well, which would further improve the applicability of the probabilistic approach.

Bibliography

[1] R. B. Ash. *Basic Probability Theory.* Courier Corporation, 2008.

[2] R. Bellman. *Dynamic Programming.* Princeton University Press, 1957.

[3] I. Csiszár. Information-type measures of difference of probability distributions and indirect observations. *Studia Scientific Mathetica Hungary*, 2:299–318, 1967.

[4] M. C. Davidson. Climate change and the ethics of discounting. *WIREs Climate Change*, 6(4):401–412, 2015.

[5] B. D. F. and Z. R. O. Appropriate discounting for benefit-cost analysis. *Journal of Benefit-Cost Analysis*, 2(2):1–20, 2011.

[6] L. Green and J. Myerson. A discounting framework for choice with delayed and probabilistic rewards. *Psychol Bull.*, 130(5):769–792, 2004.

[7] P. Guan, M. Raginsky, and R. Willett. Online Markov decision processes with Kullback-Leibler control cost. In *Am. Control Conference*, pages 1388–1393. IEEE, June 2012.

[8] A. Guessab and G. Schmeisser. Necessary and sufficient conditions for the validity of jensen's inequality. *Archiv der Mathematik*, 100:561–570, 2013.

[9] P. R. Halmos. *Measure Theory.* Springer Science+Business Media, 1974.

[10] J. Hirshleifer. Risk, the discount rate, and investment decisions. *The American Economic Review*, 51(2):112–120, 1961.

[11] M. Kárný. Towards fully probabilistic control design. *Automatica*, 32:1719–1722, 1996.

[12] M. Kárný. Axiomatisation of fully probabilistic design revisited. *SCL*, 141:104719, 2020.

[13] M. Kárný. Minimum expected relative entropy principle. In *Proc. of the 18th ECC*, pages 35–40, Sankt Petersburg, 2020.

[14] M. Kárný, J. Böhm, T. V. Guy, L. Jirsa, I. Nagy, P. Nedoma, and L. Tesař. *Optimized Bayesian Dynamic Advising: Theory and Algorithms.* Springer, 2006.

[15] M. Kárný and T. Guy. Fully probabilistic control design. *Systems & Control Letters*, 55(4):259–265, 2006.

[16] M. Kárný, A. Halousková, J. Böhm, R. Kulhavý, and P. Nedoma. Design of linear quadratic adaptive control: theory and algorithms for practice. *Kybernetika*, 21 (1a):3–96, 1995.

[17] B. O. Koopman. On distributions admitting a sufficient statistic. *Transactions of the American Mathematical Society*, 39(3):399–409, 1936.

[18] R. Kulhavý and M. B. Zarrop. On a general concept of forgetting. *International Journal of Control*, 58(4):905–924, 1993.

[19] S. Kullback. and R. A. Leibler. On Information and Sufficiency. *Annals of Mathematical Statistics*, 22(1):79–86, 1951.

[20] P. Kumar. A survey on some results in stochastic adaptive control. *SIAM J. Control and Applications*, 23:399–409, 1985.

[21] J. MacKillop, M. Amlung, and L. e. a. Few. Delayed reward discounting and addictive behavior: a meta-analysis. *Psychopharmacology*, 216:305–321, 2011.

[22] K. Mardia, H. Southworth, and C. Taylor. On bias in maximum likelihood estimators. *Journal of Statistical Planning and Inference*, 76(1-2):31–39, 1999.

[23] A. Mesbah. Stochastic model predictive control with active uncertainty learning: A survey on dual control. *Annual Reviews in Control*, 45:107–117, 2018.

[24] S. Molnárová. Discounted Fully Probabilistic Design of Decision Strategy. Bachelor's project, 2022.

[25] S. Molnárová. Adaptive Discounted Fully Probabilistic Design of Decision Strategy. Research project, 2023.

[26] E. Moxnes. Discounting, climate and sustainability. *Ecological Economics*, 102:158–166, 2014.

[27] V. Peterka. Bayesian system identification. In P. Eykhoff, editor, *Trends and Progress in System Identification*, pages 239–303. Pergamon Press, 1981.

[28] E. J. G. Pitman. Sufficient statistics and intrinsic accuracy. *Mathematical Proceedings of the Cambridge Philosophical Society*, 32(4):567–579, 1936.

[29] P. R. Portney and J. P. Weyant. *Discounting and Intergenerational Equity.* Routledge, 1999.

[30] M. Price, M. Lee, and S. Higgs. Impulsivity, eating behaviour and performance on a delay discounting task. *Appetite*, 71:483, 2013.

[31] M. Puterman. *Markov Decision Processes.* John Wiley & Sons, 1994.

[32] M. Rao. *Measure Theory and Integration.* John Wiley & Sons, 1987.

[33] R. J. Rossi. *Mathematical Statistics: An Introduction to Likelihood Based Inference.* John Wiley & Sons, 2018.

[34] D. Sinclair. Capital budgeting decisions using the discounted cash flow method. *Canadian Journal of Anesthesia*, 57:704–705, 2010.