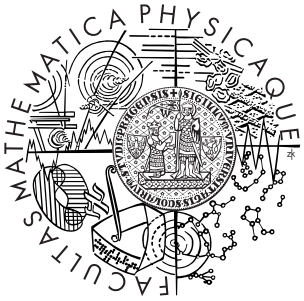
Univerzita Karlova v Praze
Matematicko-fyzikální fakulta

BAKALÁŘSKÁ PRÁCE



Ludvika Divišová

Vylepšení modelu dynamického rozhodování pomocí metody "Iteration spread in time".

Ústav teorie informace a automatizace Akademie věd České Republiky

Vedoucí bakalářské práce: Ing. Miroslav Kárný, DrSc.
Studijní program: Matematika, Obecná matematika

2008

Děkuji vedoucímu mé bakalářské práce, Ing. Miroslavu Kárnému, DrSc. za konzultace, cenné připomínky a návrhy a firmě Colloseum a.s. reprezentované Ing. Janem Zemanem za poskytnutí dat a výsledků výzkumů.
Tato práce byla podpořena grantem MŠMT 2C06001.

Prohlašuji, že jsem svou bakalářskou práci napsala samostatně a výhradně s použitím citovaných pramenů. Souhlasím se zapůjčováním práce a jejím zveřejňováním.

V Praze dne 12.12.2008 Ludvika Divišová

Contents

Název práce: Vylepšení modelu dynamického rozhodování pomocí metody "Iteration spread in time".
Autor: Ludvika Divišová
Katedra (ústav): Ústav teorie informace a automatizace, Akademie věd České Republiky
Vedoucí bakalářské práce: Ing. Miroslav Kárný, DrSc.
e-mail vedoucího: school@utia.cas.cz

Abstrakt: V předložené práci studujeme problematiku hledání nejlepšího rozhodnutí na základě určité předchozí zkušenosti se systémem. Využíváme k tomu dynamické programování a jeho aproximací. V práci shrnujeme teorii potřebnou k použití dynamického programování a zabýváme se její aplikací v případě obchodování s futures kontrakty, při kterém se snažíme najít nejlepší obchodní strategii (to je posloupnost rozhodnutí), která maximalizuje náš zisk, respektive minimalizuje ztrátovou funkci. Zavádíme pojem "Bellmanova funkce", vysvětlujeme nutnost aproximace této funkce, uvádíme jednu z již testovaných aproximačních metod společně s jejími výsledky a snažíme se navrhnout metodu, kterou bychom dosáhli nejlepší aproximace v rozumné době a s dostupnými výpočetními prostředky.

Klíčová slova: teorie rozhodování, dynamické programování, Bellmanova funkce, futures kontrakty

Title: Dynamic decision making via iterations spread in time
Author: Ludvika Divišová
Department: Institute of Information Theory and Automation, Academy of Sciences of the Czech Republic
Supervisor: Ing. Miroslav Kárný, DrSc.
Supervisor's e-mail address: school@utia.cas.cz

Abstract: In the present work we study the problem of finding the best decision based on our previous experience with the system. To solve this task, we use the dynamic programming and its approximations. In the work we summarize the theory needed for usage of the dynamic programming and we deal with its application on futures dealing trying to find best strategy, id est a sequence of decisions, maximizing our gain or minimizing the loss function. We introduce notion "Bellman function", explain why the approximation of this function is needed, demonstrate one of already tested approximation methods together with its results and we try to propose a method that would lead to the best approximation in suitable time and with available computation aids.

Keywords: theory of decision-making, dynamic programming, Bellman function, futures contracts

Chapter 1

Introduction

The main aim of this work is to familiarize readers with the theory of decision-making, the usage of dynamic programming in decision making and how it is possible to use this mathematical machinery in practice during dealing with futures contracts.
The reasons why we decided to show the decision-making theory on the case of futures dealing are simple: this problem is very actual, the financial background is highly interesting (with good strategy of dealing we can earn a lot of money) and mainly the Department of adaptive systems of The Institute of Information Theory and Automation in cooperation with company Colosseum a.s. has been dealing with the optimization of futures dealing for a long time, so it was possible to use wide spectrum of data, software for data computing and previous approaches to solve this optimization problem.
Mainly in the beginning of this work, the theory is presented very generally, so it can be used for different optimization tasks; not only for dealing with futures contracts. Later we show that the dynamic programming applied on problems from the real world can be very difficult and we have to use some approximation methods.
In this work we demonstrate one already tested approximation method and we present our proposal how to improve this method. However, this proposal is only one possible way of the approximation and should be treated as one small part of longtime running research, during which many quantitatively and qualitatively different approaches will be tested. These approaches are going to be examined according to the different points of view and appropriately combined to get better and stabilized strategy.

Let us shortly characterize the contents of chapters.
In Chapter 2, after we establish main notation, we present the basic and very important lemma of decision-making and all important propositions for dynamic programming. We define there the Bellman function and show one general method of its approximation called *Iterations spread in time*.
In next chapter, we characterize the futures dealing, define a gain function which we are trying to optimize and show by a simple example why the approximation is needed.
Chapter 4 contents finding the best form of approximation of the Bellman function, description of one possible method of approximation and the real proposal of this method improvement. At the end of this chapter our experiments are demonstrated.
Our conclusions are summarized in the last chapter.

Chapter 2

Principal theory of decision-making

In this chapter, we will summarize the basic theory of decision-making that helps the decision-maker to choose the best of his options. We will describe how we use the dynamic programming for finding the best option.

2.1 Main conventions and notions

First we present the list of notations we use in all text; for some quantities we present even some short characterization.

f means probability density function (pdf).

x^* is the set of all admissible values of x, $x \in x^*$.

$\hat{x}$ denotes the number of members in countable set x^*.

$\hat{t}$ is called time horizon.

x_t is quantity x at the time t, $t \in t^*$, where $t^* \equiv \{1, ..., \hat{t}\}$

a denotes the decision, $a \in a^*$. The task of decision-making is meaningful for $\hat{a} > 1$ only.

P_a is knowledge (experience) about the system available for choosing the decision $a \in a^*$.

F_a is knowledge (ignorance) about the system unavailable for choosing the decision $a \in a^*$.

Q means the set of all quantities that describe the behavior of the system. We can split the behavior with respect to any decision $a \in a^*$ into (P_{a^*}, a, F_{a^*}). Q^* consists of all possible realizations of Q and is called *system behavior*.

Δ_t means the innovation. It contains the quantities that are not included in P_{a_t} but are included in $P_{a_{t+1}}$. Often the innovation is exactly the observable output of the system.

R denotes the decision rule generating the decision a, $R : Q^* \to a^*$. R^o means the optimal decision rule giving us the best decision a^o.

$\mathbf{R}$ means a sequence of decision rules R_t, $\mathbf{R} \equiv \{R_t : Q^* \to a_t^*\}_{\{t \in t^*\}}$. $\mathbf{R}$ is called *strategy*.

Z is called *loss function* and quantifies the difference between decision-maker's aim and achievement.

$E(Z) \equiv E_{\mathbf{R}}(Z)$ means the expected loss and assigns a value in $[0, \infty]$ to every loss function Z and strategy $\mathbf{R}$. For expected loss we have

$$E(Z) = \int_{Q^*} Z(Q) f(Q) dQ \tag{2.1}$$

2.2 Dynamic programming

First we prepare one lemma creating the base of decision-making theory.

Proposition 2.1 (Basic decision-making lemma) The optimal admissible decision rule R^o, defining the optimal decision a^o,

$$R^o(P_{a^*}) \equiv a^o(P_{a^*}), \forall P_{a^*} \in P_{a^*}^*$$

minimizing the expected loss (2.1) can be constructed value-wise as follows. To each $P_{a^*} \in P_{a^*}^*$, a minimizing argument $a^o(P_{a^*})$ in

$$\min_{a \in a*} E[Z(P_{a^*}, a, F_{a^*}) \mid a, P_{a^*}] \tag{2.2}$$

is assigned as the value of the optimal decision rule corresponding to the considered argument. The reached minimum is

$$\min_{\{R:P^*_{a^*}\to a^*\}} E[Z(P_{a^*}, a, F_{a^*})] = E\left\{\min_{a\in a*} E[Z(P_{a^*}, a, F_{a^*}) \mid a, P_{a^*}]\right\}. \quad (2.3)$$

Proof: Can be seen in Kárný (2006).

The optimal strategy can be found by using a stochastic version of dynamic programming. It is a repetitive application of proposition (2.1).

Proposition 2.2 (Stochastic dynamic programming) The optimal strategy $\left\{R^o_t : P^*_{a^*_t} \to a^*_t\right\}_{t\in t^*} \in \left\{R_t : P^*_{a^*_t} \to a^*_t\right\}_{t\in t^*}$ acting on non-decreasing sequence of experience $P_{a^*_t} \subset P_{a^*_{t+1}}$ and minimizing the expected loss function $E[Z(Q)]$ can be constructed value-wise way. For every $t \in t^*$ and each $P_{a^*_t} \in P^*_{a^*_t}$, it is sufficient to take a minimizing argument $a^o(P_{a^*_t})$ in

$$\nu(P_{a^*_t}) = \min_{a_t\in a^*_t} E[\nu(P_{a^*_{t+1}}) \mid a_t, P_{a^*_t}], t \in t^* \quad (2.4)$$

as the decision generated by t-th rule of the optimal strategy, i.e. $a^o(P_{a^*_t}) = R^o_t(P_{a^*_t})$. The recursion (2.4) is performed in the backward manner against the course given by the increasing experience. It starts with

$$\nu(P_{a^*_{\hat{t}+1}}) \equiv E[Z(Q) \mid P_{a^*_{\hat{t}+1}}] \quad (2.5)$$

The reached minimum has the value

$$E[\nu(P_{a^*_1})] = \min_{\{R_t:P^*_{a^*_t}\to a^*_t\}_{t\in t^*}} E[Z(Q)].$$

The function ν is called *Bellman function*.

Proof: Can be seen in Kárný (2006).

From this moment we will assume that loss function is additive and depends only on the outputs of the system (the innovations) and on the decisions and does not depend on the internal parameters of the system (the ignorance F_a contains only unobserved data), that gives us following expression of loss function:

$$Z(Q) = \sum_{t=1}^{\hat{t}} z(\Delta(t), a(t)).$$

For this additive loss function we can formulate the proposition (2.2) as follows.

Proposition 2.3 (Dynamic programming for additive loss function) Let us consider additive loss function dependent only on the innovations and decisions. Then the optimal strategy $\left\{R_t^o : P_{a_t^*}^* \to a_t^*\right\}_{t\in t^*}$ acting on non-decreasing sequence of experience $\{P_{a_t^*}\}_{\{t\in t^*\}}$ and minimizing the expected additive loss function

$$E\left[\sum_{t=1}^{\hat{t}} z(\Delta(t), a(t))\right]$$

can be constructed value-wise. For all $t \in t^*$ and to each $P_{a_t^*} \in P_{a_t^*}^*$, a minimizing argument $a^o(P_{a_t^*}) = R_t^o(P_{a_t^*})$ in

$$\nu(P_{a_t^*}) = \min_{a_t \in a_t^*} E[z(\Delta(t), a(t)) + \nu(P_{a_{t+1}^*}) \mid a_t, P_{a_t^*}], t \in t^* \tag{2.6}$$

is assigned. The recursion (2.6) is performed in backward manner against the course given by the increasing experience, starting from

$$\nu(P_{a_{\hat{t}+1}^*}) \equiv 0.$$

the reached minimum has the value

$$E[\nu(P_{a_1^*})] = \min_{\left\{R_t : P_{a_t^*}^* \to a_t^*\right\}_{t\in t^*}} E[Z(P_{a_{\hat{t}+1}^*})].$$

Proof: Can be seen in Kárný (2006).

These two propositions give us the formal instruction how to find the optimal decision. However, for using the propositions we need to evaluate the expectation

$$E[\nu(P_{a_{t+1}^*}) \mid a_t, P_{a_t^*}] = \int_{\Delta_t^*} \nu(P_{a_t^*}, a_t, \Delta_t) f(\Delta_t \mid a_t, P_{a_t^*}) d\Delta_t, \forall t \in t^*. \tag{2.7}$$

The innovation Δ_t contains the quantities that can not be used for choosing a_t, those quantities belong to $P_{a_{t+1}^*}$.
As we can see from equation (2.7) we need to know $f(\Delta_t \mid a_t, P_{a_t^*})$ in every time $t \in t^*$, but that is rarely satisfied. So we try to estimate these pdfs using data we have from past, this process is called *learning* and is described in next section. The system of pdfs $\{f(\Delta_t \mid a_t, P_{a_t^*})\}_{t\in t^*}$ is called *outer model of the system.*

2.3 Learning

In this part we describe how to get outer model of the system we need for optimal decision-making. Let us prepare one technical proposition.

Proposition 2.4 (Bayes rule) For random quantities a, b, c it holds

$$f(a \mid b, c) = \frac{f(b \mid a, c) f(a \mid c)}{f(b \mid c)} = \frac{f(b \mid a, c) f(a \mid c)}{\int f(b \mid a, c) f(a \mid c) da} \tag{2.8}$$

Proof: From the definition of conditional pdf we get

$$f(a \mid b, c) = \frac{f(a,b|c)}{f(b|c)} = \frac{f(a,b|c)}{\int f(a,b|c) da}$$

Now we rewrite it using $f(a, b \mid c) = f(b \mid a, c) f(a \mid c)$.

First we will present the general case when the outer model relies on learning of time variant and directly unobservable quantities. This process is called *Bayesian filtering*. We will assume that

- the innovations Δ_t are related to experience $P_{a_t^*}$ and decisions a_t through the observation model

  $$\{f(\Delta_t \mid a_t, P_{a_t^*}, \Theta_t)\}_{t \in t^*} \tag{2.9}$$

  that is given up to unknown directly unobservable quantities $\Theta_t \in \Theta_t^* \subset F_{a_\tau^*}, \forall \tau \in t^*$.

- the evolution of the quantities $\Theta(\hat{t}) \in \Theta^*(\hat{t})$ is described by a known collection of pdfs

  $$\{f(\Theta_t \mid a_t, P_{a_t^*}, \Theta_{t-1})\}_{t \in t^*}. \tag{2.10}$$

- the quantities $\Theta(\hat{t})$ are unknown to the strategies considered. Formally we can express it as independency of a_t and Θ_t when conditioned on $P_{a_t^*}$

  $$f(a_t \mid P_{a_t^*}, \Theta_t) = f(a_t \mid P_{a_t^*}) \underbrace{\Leftrightarrow}_{(2.8)} f(\Theta_t \mid a_t, P_{a_t^*}) = f(\Theta_t \mid P_{a_t^*}) \tag{2.11}$$

- the initial values of $\Theta_0, P_{a_1^*}$ add nothing new to the prior information so that the prior pdf

$$f(\Theta_1) \equiv f(\Theta_1 \mid a_1, P_{a_1^*}, \Theta_0) = f(\Theta_1 \mid a_1, P_{a_1^*}) = f(\Theta_1 \mid P_{a_1^*})$$

  is the first term in the sequence of the pdfs (2.10).

Now we can write the main proposition for filtering.

Proposition 2.5 (Generalized Bayesian filtering) The outer model of the system $\{f(\Delta_t \mid a_t, P_{a_t^*})\}_{t \in t^*}$ is given by formula

$$f(\Delta_t \mid a_t, P_{a_t^*}) = \int_{\Theta_t^*} f(\Delta_t \mid a_t, P_{a_t^*}, \Theta_t) f(\Theta_t \mid P_{a_t^*}) d\Theta_t. \tag{2.12}$$

The evolution of the pdfs $f(\Theta_t \mid P_{a_t^*})$ is described by the following recursion that starts from the prior pdf $f(\Theta_1)$:

- Data updating

$$f(\Theta_t \mid P_{a_t^*}) = \frac{f(\Delta_t \mid a_t, P_{a_t^*}, \Theta_t) f(\Theta_t \mid P_{a_t^*})}{f(\Delta_t \mid a_t, P_{a_t^*})} \tag{2.13}$$

  that incorporates the innovation Δ_t and the decision a_t, and

- Time updating

$$f(\Theta_{t+1} \mid P_{a_{t+1}^*}) = \int_{\Theta_t^*} f(\Theta_{t+1} \mid a_{t+1}, P_{a_{t+1}^*}, \Theta_t) f(\Theta_t \mid P_{a_{t+1}^*}) d\Theta_t \tag{2.14}$$

  that reflects the time evolution $\Theta_t \to \Theta_{t+1}$.

Proof: Can be seen in Kárný (2006).

A special version of Bayesian filtering, for the case the internal quantities Θ_t are time invariant: $\Theta_t = \Theta, \forall t \in t^*$, is called *Bayesian estimation* and we call the common value Θ *parameter*. The observation model (2.9) has the form

$$\{f(\Delta_t \mid a_t, P_{a_t^*}, \Theta)\}_{t \in t^*}.$$

The proposition (2.3) can be rewritten for estimation as follows.

Proposition 2.6 (Generalized Bayesian estimation) The outer model of the system $\{f(\Delta_t \mid a_t, P_{a_t^*})\}_{t \in t^*}$ is given by formula

$$f(\Delta_t \mid a_t, P_{a_t^*}) = \int_{\Theta^*} f(\Delta_t \mid a_t, P_{a_t^*}, \Theta) f(\Theta \mid P_{a_t^*}) d\Theta. \tag{2.15}$$

The evolution of the pdf $f(\Theta \mid P_{a_t^*})$ is described by the recursion identical with data updating (2.13):

$$f(\Theta \mid P_{a_t^*}) = \frac{f(\Delta_t \mid a_t, P_{a_t^*}, \Theta) f(\Theta \mid P_{a_t^*})}{f(\Delta_t \mid a_t, P_{a_t^*})} \tag{2.16}$$

It starts from the prior pdf $f(\Theta) \equiv f(\Theta \mid P_{a_1^*}, a_1) = f(\Theta \mid P_{a_1^*})$.

Proof: Can be seen in Kárný (2006).

Now we know how to evaluate the expectation (2.7) and with that how to use the propositions (2.2) and (2.3).

2.4 Asymptotic of the dynamic programming

The asymptotic of the dynamic programming is realized for $\hat{t} \to \infty$. In this section we assume that

- loss function is additive
- loss function depends only on the innovations and on the decisions
- there is a finite-dimensional information state, i.e. $P_{a^*} \equiv x_{t-1} \equiv$ a finite-dimensional vector
- the partial loss depends only on x_t and a_t, $z(\Delta(t), a(t)) = z(x_t, a_t)$

Agreement 2.1 (Stabilizing strategy) Let us consider sequence of decision-making problems with the growing horizon $\hat{t} \to \infty$.
The strategy $\left\{R_t : P_{a_t^*}^* \to a_t^*\right\}_{t \in t^* \equiv \{1,\ldots,\hat{t}\}} \forall \hat{t} \in \{1, 2, ...\}$ is called *stabilizing strategy* if there is a finite constant c such that

$$E[z(x_t, a_t) \mid a_t, P_{a_t^*}] \leq c < \infty, t \in \{1, 2, ...\}. \tag{2.17}$$

Proposition 2.7 Let a stabilizing strategy exist. Then there is an optimal, stationary strategy, formed by a repetitive use of the same rule, for $\hat{t} \to \infty$ whose decisions are minimizing arguments in the formal analogy of (2.6)

$$\nu^{\infty}(x_{t-1}) + \kappa = \min_{a_t \in a_t^*} E[z(x_t, a_t) + \nu^{\infty}(x_t) \mid a_t, x_{t-1}] \tag{2.18}$$

with a constant $\kappa \leq c$ and a time invariant Bellman function ν^{∞}.

Proof: Can be seen in Kárný (2006).

It is obvious that the ideal planning goes through all our time horizon, but that can be very difficult. The simplest way how to make our decision-making easier is shortening of time horizon. First thought could be to reduce the horizon only to one step forward. However, in dynamic decision-making the effects and consequences of our decision become clear in longer time after the application of decision, so the decision appearing the best now can become wrong in longer perspective.
These facts lead us to a compromise; to the method called *running horizon*. We consider shorter time horizon (let us denote it $\tilde{t}$) and in this horizon we create the optimal strategy. But because the shorter horizon brings us only an approximation of ideal planning, we use only first decision from the strategy. After applying the decision we move in time about one time unit and get new data, we again create optimal strategy and again use only first decision of the strategy.

2.5 Iteration spread in time

Having $\hat{t} \to \infty$, we can set the running horizon $\tilde{t}$ to appropriate length. According to proposition (2.7) the optimal decision is minimizing argument in

$$\nu^{\infty}(x_{t-1}) + \kappa = \min_{a_t \in a_t^*} E[z(x_t, a_t) + \nu^{\infty}(x_t) \mid a_t, x_{t-1}].$$

This equation can be seen as an optimization of additive loss function $z(x_t, a_t)$ enlarged by stationary Bellman function $\nu^{\infty}(x_t)$ in one-step forward. As we know the Bellman function is time invariant, so this dynamic programming can be also seen as an iterative searching for its accurate form.

We use the Bellman function from time $t-1$ to get better estimation of the Bellman function in time t and with this more accurate Bellman function we go on with the method of running horizon. This method is called *Iteration spread in time* and we can describe the algorithm of this method as follows:

1. Appoint the starting form of the Bellman function.
2. For $t \in t^* = \{1, 2..., \hat{t}\}$ repeat:

a) Find decision rules $R_t, ..., R_{t+\tilde{t}}$ minimizing

$$E\left[\sum_{\tau=t}^{t+\tilde{t}} z(x_\tau, a_\tau) + \nu_{t-1}(x_{t+\tilde{t}}) \mid x_{t-1}\right] \tag{2.19}$$

for all forms of information state x_{t-1} with usage of the entire system model.

b) Take the final form of the Bellman function $\nu_t(x)$ as better approximation of the Bellman function.

c) Apply the decision $a_t = R_t(x_{t-1})$ for given specific state x_{t-1}.

d) Upgrade our experience with a_t and Δ_t.

With the theory of method *Iteration spread in time* we ended the first technical part of this work. In next chapters, we will pay more attention to a practical use and we will show why we use the approximation presented in last two sections.

Chapter 3

Futures dealing with using of dynamic programming

In this chapter we will apprise of what the futures contracts are and how the decision-making theory can help us with futures dealing. We will present an easy example to show why the approximation and estimation are needed. Starting this chapter we use some ideas presented in Kárný and Zeman (2008).
First of all let us explain the basics of futures dealing.

3.1 Futures dealing

The futures contract is an agreement between a seller and a buyer to make (as seller) and take (as buyer) delivery of a standardised quantity of a specific commodity or asset of standardized quality on particular date at a specified price. A futures contract gives the holder the obligation to make or take delivery under the terms of the contract.
The futures dealing takes place only on organized markets (like stock market), where the price of commodity is taken from the encountering of demand and supply. Through the futures contracts we can invest in many commodities like metals, energy, rope, grain etc. or in many financial instruments like currency, stock index, stocks and bond etc. All futures dealing can be summed up into short motto: "Buy cheap, sell expensive."
For better view into futures dealing we present a short example, that took place in USA in 1972: On the 5^{th} July 1972 there was a chance to buy a contracts for buying 5000 bushels of wheat in September for 1,5 USD/bushel

(means 7500 USD/1 contract). The investor thought that the price of wheat would grow up, so he bought 200 contracts. On the 3^{rd} August the price of wheat dramatically grew up to 1,72 USD/bushel, so the investor realize a profit 0,22 USD/bushel (means 1100 USD/1 contract). With 200 contracts the final profit is 220 000 USD.
In futures dealing the investors use their sense of business to find the optimal trading strategy. Moreover, they can - using right mathematical formulation of problem - use the theory we presented in chapter 2. In such case they will try to estimate the evolution of future prices of commodities and get the optimal strategy by using dynamic programming.

For applying the theory on futures dealing, we need few more notions and we need to make clear what exactly our previous notation means in this particular case.

t is time counted in days.

y_t is the price of the commodity in time t.

s_t denotes the number of units of the commodity we hold or owe in time $t \in t^*$. Obviously there are some restrictions for the amount of units we hold/owe, let us denote them $s_d, s_u,\ s_t \in [s_d, s_u]$.

a_t is action we can do in time $t \in t^*$. We buy or sell a_t units of the commodity for the price Cy_t where scaling factor C is given. For a_t it obviously holds:

$$a_t = s_t - s_{t-1} \tag{3.1}$$

Which gives us:

$$s_t = s_0 + \sum_{k=1}^{t} a_k \tag{3.2}$$

We denote the optimal action in time t as a_t^o.

K_t is the capital we have in time $t \in t^*$. For K_t we get:

$$K_t = K_{t-1} - Cy_t a_t - c|a_t| \tag{3.3}$$

Where c means the fee for trading.

g_t means the gain in time $t \in t^*$. Obviously the gain is the negative value of loss function: $g_t = -z_t$. So we will maximize instead of minimize. Let us specify the form of g_t. The gain consists of two parts: money (the difference between capital we had and capital we have now) and value of the units of commodity (the difference between the price of the units of commodity we held/owned and the price of the units of commodity we hold/own now):

$$g_t = K_t - K_0 + Cy_t s_t - Cy_0 s_0 \tag{3.4}$$

When we use the expressions (3.3) and (3.4) we get:

$$g_t = -\sum_{k=1}^{t} (C(y_k - y_t)a_k + c|a_k|) - C(y_0 - y_t)s_0 \tag{3.5}$$

If we express g_{t-1} exactly the same way and compare it, we get the recursion:

$$\begin{aligned} g_t &= g_{t-1} - \sum_{k=1}^{t-1} C(y_{t-1} - y_t)a_k - c|a_t| - C(y_{t-1} - y_t)s_0 \\ &= g_{t-1} - C(y_{t-1} - y_t)s_{t-1} - c|a_t| \end{aligned} \tag{3.6}$$

Let us define the initial gain $g_0 = 0$. Then with recursion (3.6) and with expression (3.1) of a_t it holds:

$$g_t = -\sum_{k=1}^{t} (C(y_{k-1} - y_k)s_{k-1} + c|a_k|) \tag{3.7}$$

As in section (2.4), we assume that the gain function is additive and that it depends on finite-dimensional information vector $d(t) = (d_1, \ldots, d_t), t \in t^*$, where $d_i = (y_i, a_i), i \in \{1, \ldots, t\}$.
Our main aim is to find optimal strategy which consists of rules R_t acting on the information state $P_t \equiv (y_t, d(t-1))$, $R_t : P_t \to a_t$ and which maximizes the expected value of the gain

$$E[g_t] = \textstyle\sum_{k=1}^{t} z(d(k))$$

where $z(d(k)) \equiv -C(y_{k-1} - y_k)s_{k-1} - c|a_k|, k \in \{1, \ldots, t\}$ are partial gains.

3.2 Motivation example

We present following simple example as a motivation for using the approximations in dynamic programming. This example shows us that the Bellman function is getting so difficult we are unable to evaluate it exactly.
We assume that the actual price y_t depends only on two previous values y_{t-1} and y_{t-2} and does not depend on our decisions. Let us make classic linear regression model and assume that the parameters in the model are known. Then we get

$$y_t = \beta_1 y_{t-1} + \beta_2 y_{t-2} + \epsilon_t \tag{3.8}$$

We also assume that we can buy or sell only one unit of commodity at time, that gives us following representation of a_t.

$$a_t = \begin{cases} 1 & \text{if we buy the commodity in time t} \\ 0 & \text{if we do nothing in time t} \\ -1 & \text{if we sell the commodity in time t} \end{cases}$$

At last we assume: the starting amount of held or owed units s_0 as well as ending amount $s_{\hat{t}}$ are zero; $s_0 = 0$, $s_{\hat{t}} = 0$.
We will make our strategy two steps forward, we proceed according to the proposition (2.3) with the recursion (2.6):

$$\nu(P_{a_t^*}) = \max_{a_t \in a_t^*} E[z(d(t)) + \nu(P_{a_{t+1}^*}) \mid a_t, P_{a_t^*}] \tag{3.9}$$

starting with

$$\nu(P_{a_{t^o+1}^*}) = 0.$$

Thanks to the relation (3.1), we can rewrite the Bellman function with s_t and y_t.
The recursion starts:
1^{st} step

$$\nu(s_1, s_2, s_3, y_1, y_2, y_3) = 0$$

2^{nd} step

$$\nu(s_1, s_2, y_1, y_2) = \max_{a_3 \in a_3^*} E[z(d(3)) + 0 \mid a_3, y_2, s_2, y_1, s_1]$$

With the definition of partial gain $z(d(t))$ in time t we get

$$\nu(s_1, s_2, y_1, y_2) =$$
$$\max_{a_3 \in a_3^*} E[-C(y_2 - (\beta_1 y_2 + \beta_2 y_1 + \epsilon_3))s_2 + c|a_3| + 0 \mid a_3, y_2, s_2, y_1, s_1]$$

From $a_3 = s_3 - s_2, s_3 = 0$ and $E[\epsilon_3 \mid a_3, y_2, s_2] = 0$ we get

$$a_3 = -s_2$$
$$\nu(s_1, s_2, y_1, y_2) = -C(y_2 - (\beta_1 y_2 + \beta_2 y_1))s_2 + c|s_2|$$

3^{rd} step

$$\nu(s_1, y_1) = \max_{a_2 \in a_2^*} E[z(d(2)) + \nu(s_1, s_2, y_1, y_2) \mid a_2, y_1, s_1]$$

As in the second step we can write
$\nu(s_1, y_1) = \max_{a_2 \in a_2^*} E[-C(y_1 - (\beta_1 y_1 + \beta_2 y_0 + \epsilon_2))s_1 - c|a_2| - C((1-\beta_1)(\beta_1 y_1 + \beta_2 y_0 + \epsilon_2) - \beta_2 y_1)(a_2 + s_1) + c|a_2 + s_1| \mid a_2, y_1, s_1]$
Let us mark the conditioned form as θ and divide the cases that can occur according to the values of s_1. According to our assumption $s_0 = 0$, so $s_1 \in \{-1, 0, 1\}$

a) $s_1 = -1$

Then the admissible actions and related values of θ are:
$a_2 = -1 : \theta = C(y_1 - (\beta_1 y_1 + \beta_2 y_0)) + c + 2C((1-\beta_1)(\beta_1 y_1 + \beta_2 y_0) - \beta_2 y_1)$
$a_2 = 0 : \theta = C(y_1 - (\beta_1 y_1 + \beta_2 y_0 +)) + C((1-\beta_1)(\beta_1 y_1 + \beta_2 y_0) - \beta_2 y_1) + c$
$a_2 = 1 : \theta = C(y_1 - (\beta_1 y_1 + \beta_2 y_0)) - c$

b) $s_1 = 0$

Then the admissible actions and related values of θ are:
$a_2 = -1 : \theta = C((1 - \beta_1)(\beta_1 y_1 + \beta_2 y_0) - \beta_2 y_1)$
$a_2 = 0 : \theta = 0$
$a_2 = 1 : \theta = -C((1 - \beta_1)(\beta_1 y_1 + \beta_2 y_0) - \beta_2 y_1)$

c) $s_1 = 1$

Then the admissible actions and related values of θ are:
$a_2 = -1 : \theta = -C(y_1 - (\beta_1 y_1 + \beta_2 y_0 + \epsilon_2)) + c$
$a_2 = 0 : \theta = -C(y_1 - (\beta_1 y_1 + \beta_2 y_0)) - C((1-\beta_1)(\beta_1 y_1 + \beta_2 y_0) - \beta_2 y_1) + c$
$a_2 = 1 : \theta = -C(y_1 - (\beta_1 y_1 + \beta_2 y_0)) - 2C((1-\beta_1)(\beta_1 y_1 + \beta_2 y_0) - \beta_2 y_1) + c$

Even though we made many assumptions and restrictions the Bellman function $\nu(s_1, y_1)$ is quite difficult. If we made our strategy for more steps, we would get into trouble with evaluating the expected value. That is why the approximation is needed, especially when we are dealing with problems from the real world which are obviously much more difficult than presented example.

Chapter 4

Proposal of improving the method "Iteration spread in time"

In this chapter we will use the notations from chapter 3, we will deal with the form of the Bellman function, present the method of its approximation, which is already tested on real data, together with our proposal of approximation.

4.1 Form of the Bellman function

First and very important step is to choose the right form of the Bellman function. As we said before and as the motivation example showed it is rarely possible to exactly evaluate the Bellman function. We present here a sensible choice of a finitely parameterized class of function where we search for approximation of the Bellman function.
Let us assume that from our observation of the system we got the linear autoregression model

$$y_t = \beta^T \psi_{t-1} + \epsilon_t,$$

where $\psi_{t-1} = (y_{t-1}, \ldots, y_{t-m})^T$, T means transposition, regression coefficients β and the finite variance σ of ϵ_t vary slowly with time.
The expression (3.5) shows that the action a_k increases the gain if

$$-C(y_k - y_t)a_k > c|a_k|.$$

So the action a_k can positively contribute to the gain only if

$$\text{sign}(a_k) = -\text{sign}(y_k - y_t),$$

the contribution is the higher the higher is the value $|y_k - y_t|$ and it is an increasing function of $|a_k|$. So the optimal action a_k^o should be chosen so that s_k reaches one of the bounds s_d or s_u.
From this simple considerations we get very important conclusion: The maximum of the gain over all action sequences coincides with the maximum of the gain over sequences that have non-zero actions at most at turning points at which the commodity price exhibit local extremes. The maximizing actions make the elements of sequence $\{s_t\}_{t\in t^*}$ pendulate between the bounds s_d and s_u.
We denote the turning points t_k, $1 \leq t_{k-1} \leq t_k \leq \hat{t}$, $k \in k^* \equiv \{1, \ldots, \hat{k}\}$, they are defined as follows:

$$t_k = t \text{ if } (y_{t-1} < y_t \wedge y_t > y_{t+1}) \vee (y_{t-1} > y_t \wedge y_t < y_{t+1}).$$

Even if we knew $y(t) \equiv (y_t, y_{t-1}, \ldots, y_0)$ we could not certainly say that t is turning point. We treat the turnings points as random. Without a loss of generality, we can take the first and the last time moments as turning points. Then the other turning points are specified by specifying increments n_k:

$$n_k = t_k - t_{k-1},\ n_k \in \{1, \ldots, \hat{n}\},\ \hat{n} < \infty$$

Let us denote N_k a random quantity determining the increments n_k and $\text{p}(N_k = n)$ the probability that N_k equals to n. The experiments show that the quantities in sequence $\{N_k\}_{n_k\in\{1,\ldots,\hat{n}\}}$ are independent.

The actions in time moments different form $t_k, k \in k^*$ are equaled to zero, $a_\tau = 0$, $\tau \in t^* \setminus \{t_k\}_{k\in k^*}$. With this notation we can rewrite the expression (3.7) of gain g_t, $t \in t^*$ as follows:

$$g_t = -\sum_{k=1}^{\hat{k}} (C(y_{t_{k-1}} - y_{t_k}) s_{t_{k-1}} + c|a_{t_k}|), \tag{4.1}$$

where $t_{\hat{k}}$ is the last turning point, $t_{\hat{k}} = t$.

Our next assumption is that we know all optimal actions $a_t^o, t \in t^*$. The final information state is

$$P_t^o = P_t \cup \{\Theta\} \cup \{a_t^o\}_{t\in t^*}, \tag{4.2}$$

where Θ consists of regression coefficients β, variance σ and the probabilities $\mathrm{p}(N_k = n)$ for all $n \in \{1, 2, \ldots, \hat{n}\}$, $k \in \{1, 2, \ldots, \hat{k}\}$.

Let us denote $\Phi_{t_{k+1}} \equiv (y_{t_{k+1}}, y_{t_{k+1}-1}, \ldots, y_{t_{k+1}-\eta+1}, 1)^T$, where $\eta = max\{m, \hat{n}\}$, and assume that the information content in $\{a_t^o\}_{t \le t_k}$ does not influence the expected value of $\Phi_{t_{k+1}}$. Then for $t_{k+1} = t_k + n$ we have

$$E[\Phi_{t_k+n} \mid y(t_k), a^o(t_k), \Theta, n] \approx E[\Phi_{t_k+n} \mid y(t_k), \Theta, n] = W(\Theta, n)\psi_{t_k} \quad (4.3)$$

where weighting matrix W depends on regression coefficients β and explicitly on n.

Proposition 4.1 (Parametric form of the Bellman function) Let us consider the information state (4.2) and the independence of expected value of Φ_{t_k+n} on $a^o(t_k)$. Then the Bellman function $\nu(P_{t_k}^o)$ corresponding to the maximized gain (3.7) has the fixed functional form

$$\nu(P_{t_k}^o) = K^T(\Theta, s_{t_{k-1}}^o)\Phi_{t_k}. \quad (4.4)$$

The known values $a^o(t_{k-1})$ uniquely determine $s^o(t_{k-1})$. The real $(\eta + 1)$-dimensional vectors $P(\Theta, s)$, $s \in \{s_d, s_u\}$ parameterize the Bellman function.

Proof: We prove this proposition for the recursion (2.6) with maximum:

$$\nu(P_{t_k}^o) = \max_{a_{t_k} \in \{s_d - s_{t_{k-1}}, s_u - s_{t_{k-1}}\}} E[z(d(t_k)) + \nu(P_{t_{k+1}}^o) \mid a_{t_k}, P_{t_k}^o] \quad (4.5)$$

We take $t = \hat{t}$ as the last turning point. At this time the Bellman function is equal to zero and that can be written as $K(\Theta, s) = 0$. To proof a generic induction step we take the form (4.4) and insert it into right-hand side of (4.5). By inserting the parametric form we already performed the maximization, because the optimal decisions are a priori known. On the right-hand side we get

$$\begin{aligned} & -C(y_{t_{k-1}} - y_{t_k})s_{t_{k-1}}^o - c|a_{t_k}^o| \quad (4.6) \\ & +K^T(\Theta, s_{t_k}^o) \sum_n E[\Phi_{t_k+n} \mid y(t_k), \Theta, n]\mathrm{p}(N_{k+1} = n) \\ & \underbrace{=}_{(4.3)} -C(y_{t_{k-1}} - y_{t_k})s_{t_{k-1}}^o - c|a_{t_k}^o| \\ & +K^T(\Theta, s_{t_k}^o) \sum_n W(\Theta, n)\mathrm{p}(N_{k+1} = n)\psi_{t_k} \end{aligned}$$

The last member in the equation $K^T(\Theta, s^o_{t_k}) \sum_n W(\Theta, n)\mathrm{p}(N_{k+1} = n)\psi_{t_k}$ is a linear function of ψ_{t_k}. The maximization over $a^o_{t_k}$ leaves this expression linear function of ψ_{t_k} with weights depending on Θ and $s^o_{t_k}$. The expression of partial gain can be added to the weights to the known positions corresponding to y_{t_k} and $y_{t_{k-1}}$, so we get weighting matrix $\tilde{W}$. Then the equation (4.5) looks as follows:

$$\nu(P^o_{t_k}) = K^T(\Theta, s^o_{t_k}) \sum_n \tilde{W}(\Theta, n)\mathrm{p}(N = n)\psi_{t_k} \tag{4.7}$$

Obviously ψ_{t_k} can be treated as sub-vector of Φ_{t_k}, so by adding zero to appropriate place in Φ_{t_k} we get $\tilde{K}^T(\Theta, s^o_{t_k})\Phi_{t_k}$ as linear function of Φ_{t_k}.
The dependence on $s^o_{t_k}$ can be express as dependence on $s^o_{t_{k-1}}$ because we know that the value of $s^o_{t_{k-1}}$ is on the other boundary of $[s_d, s_u]$ than $s^o_{t_k}$. So finally we get

$$\nu(P^o_{t_k}) = \tilde{K}^T(\Theta, s^o_{t_{k-1}})\Phi_{t_k}. \tag{4.8}$$

The assumptions of prior information about the good guess of past optimal values $s^o(t) \equiv \{s^o_t, s^o_{t-1}, \ldots\}$ as well as about times of turning points t_k allow us to estimate directly the weighting vectors $P(\Theta, s), s \in \{s_d, s_u\}$.
Let us consider fixed $\Theta^o \in \Theta^*$ for which the parametric form of the Bellman function is closest to the correct one. This permits us to suppress Θ as the argument of the weighting vectors $K(s) \equiv K(\Theta^o, s)$.
Now when we know the form of Bellman function, we will pay attention to the methods of approximation of $K(s)$ themselves. Let us present the methods without the turning points. We do this because the way of implementing the methods together with modeling of turning points is not clear yet and formally there is not any big difference (we only suppress second index k).

4.2 Approximation by least squares

This method is fully described in Křivánek (2008). We present here only brief description with one difference from the way it is written in Křivánek (2008), namely we will use our assumption that the optimal decisions are a priori known.
We start with following equation:

$$\nu(P_t) = \max_{s_t} E[z(d(t)) + \nu(P_{t+1}) \mid P_t, s_t]. \tag{4.9}$$

Then we take the parametric form of the Bellman function (4.4) and make the approximation

$$\nu(P_t) = K^T(s_{t-1})\Phi_t \tag{4.10}$$

where $\Phi_t = [y_t, \ldots, y_{t-\eta+1}, 1]^T$ and $K^T(s_{t-1}) = [k_1(s_{t-1}), \ldots, k_\eta(s_{t-1}), k_{\eta+1}(s_{t-1})]$ with k_i as functions of s_{t-1}, only two values of s_{t-1} make sense, $s_{t-1} \in \{s_d, s_u\}$.
We are now in time $\tilde{t}$ and we work with time moments $t \in \{1, 2 \ldots, \tilde{t}\}$, so we have good guesses of optimal actions a_t^o and optimal values s_t^o. We can insert these actions into (4.9), thus we already performed maximization step and there is no need to write expected value, because we already have the exact values.
So it holds:

$$K(s_{t-1}^o)\Phi_t + \varepsilon_t = z(d(t)) + K(s_t^o)\Phi_{t+1}, \tag{4.11}$$

where ε_t denotes the deviation we did by taking parametric form of the Bellman function and by assuming the Bellman function is time-invariant. For each $t \in \{1, 2, \ldots, \tilde{t}\}$ we can express ε_t as follows:

$$\varepsilon_t = z(d(t)) + K(s_t^o)\Phi_{t+1} - K(s_{t-1}^o)\Phi_t. \tag{4.12}$$

We are searching for $K(s_{t-1}^o)$ that would minimize

$$LS = \sum_{t=1}^{\hat{t}} \varepsilon_t^2. \tag{4.13}$$

By minimizing (4.13) we get $K^o(s_{t-1}^o)$ and insert it into (4.10). This gives us the approximation of time-invariant Bellman function which we use for finding the best decision in time $\tilde{t} + 1$.

4.3 Extended iteration spread in time

We will proceed the same way as in previous section, we take the stationary Bellman equation

$$\nu(P_t) + \kappa = \max_{a_t} E[z(d(t)) + \nu(P_{t+1}) \mid a_t, P_t] \tag{4.14}$$

and insert into it the parametric form of the Bellman function (4.4). We get

$$K^T(s^o_{t-1})\Phi_t + \underbrace{\kappa + C(y_{t-1} - y_t)s^o_{t-1} + c|a^o_t|}_{\tilde{\kappa}} + \varepsilon_{t+1} = K^T(s^o_t)\Phi_{t+1} \tag{4.15}$$

where innovations ε_t denotes the deviation we did by taking parametric form of the Bellman function.
The innovations ε_t have finite variances ρ_t varying slowly in time, so from now we will deal with its (approximately) common value ρ. The innovation in time $t < \tilde{t}$ obviously depends on $K, y(t), s^o_t, s^o_{t-1}$ and ρ, but it also depends on previous ε_{t+1}, because that influenced $K(s^o_t)$ and similarly we can go on until $\tilde{t}$. For evaluation the optimal values $s^o_t, t < \tilde{t}$ we needed all prices $y(\tilde{t})$, that gives us the dependency of ε_{t+1} on $y(\tilde{t})$. In our following ideas and operations we will neglect this dependency assuming it is not significant, plus we will assume that they have normal distribution $\mathcal{N}_{\varepsilon_{t+1}}(0, \rho)$. The construction of ε_t gives us its zero expected value:

$$E[\varepsilon_t \mid K, y(t), \rho, s^o(\tilde{t})] \approx E[\varepsilon_t \mid K, y(t), \rho, s^o(t)] \approx E[\varepsilon_t \mid K, y(t), \rho] = 0.$$

Again we can express ε_{t+1}:

$$\varepsilon_{t+1} = K^T(s^o_t)\Phi_{t+1} - K^T(s^o_{t-1})\Phi_t - \tilde{\kappa}. \tag{4.16}$$

We can look at (4.16) as at the regression model: $K^T(s^o_t), K^T(s^o_{t-1})$ and $\tilde{\kappa}$ are coefficients and Φ_{t+1} with Φ_t express regressors. In approximation by least square we were trying to minimize ε_{t+1}, now we will try to find most probably value of ε_{t+1} by finding most probably value of $K(s), s \in \{s_d, s_u\}$. We will use the method of maximum likelihood.
We can split both columns of $K(s)$ as follows

$$K^T(s_d) = [\alpha_d, p_d], K^T(s_u) = [\alpha_u, p_u],$$

where α_d, α_s are scalars. The equation (4.14) imply that if a function $\tilde{\nu}$ solves the equation than even $\tilde{\nu} + arbitrary\ constant$ does. That gives us a freedom

to shift the value of the Bellman function by an arbitrary constant, so we can assume that $\alpha_d > 0, \alpha_u > 0$.
y_{t+1} is the only price that is included in Φ_{t+1} and is not included in Φ_t, so we can transport the distribution of ε_{t+1} on distribution of y_{t+1}. That gives us the model parameterized by the weights K of the Bellman function and by the variance ρ:

$$f(y_{t+1} \mid K, \rho, s_t^o, \Phi_t) \tag{4.17}$$
$$= \sqrt{\frac{|\alpha_u|^{2\delta(s_u,s_t^o)}|\alpha_d|^{2\delta(s_d,s_t^o)}}{2\pi\rho}} \exp\left\{-\frac{([\alpha_u, p_u, \alpha_d, p_d, \tilde{\kappa}]^T\zeta_{t+1})^2}{2\rho}\right\}$$

where
$\zeta_{t+1} \equiv [\delta(s_u, s_t^o)\Phi_{t+1}^T - \delta(s_u, s_{t-1}^o)\Phi_t^T, \delta(s_d, s_t^o)\Phi_{t+1}^T - \delta(s_d, s_{t-1}^o)\Phi_t^T, -1]$,
$\delta(s, \tilde{s}) = \begin{cases} 1 & \text{if } s = \tilde{s} \\ 0 & \text{otherwise} \end{cases}$

Now we construct the likelihood function $L_t(K, \rho)$ for (4.17):

$$L_t(\alpha_u, p_u, \alpha_d, p_d, \tilde{\kappa}, \rho) = |\alpha_u|^{\upsilon_{ut}}|\alpha_d|^{\upsilon_{dt}}\rho^{-\frac{1}{2}(\upsilon_{ut}+\upsilon_{dt})}(2\pi)^{-\frac{1}{2}(\upsilon_{ut}+\upsilon_{dt})}$$
$$\times \exp\left\{-\frac{([\alpha_u, p_u, \alpha_d, p_d, \tilde{\kappa}]^T A_t[\alpha_u, p_u, \alpha_d, p_d, \tilde{\kappa}]}{2\rho}\right\} \tag{4.18}$$

where

$A_t = A_{t-1} + \zeta_t^T\zeta_t$, starting by $A_0 = 0$
$\upsilon_{ut} = \upsilon_{u(t-1)} + \delta(s_u, s_t^o)$, starting by $\upsilon_{u0} = 0$
$\upsilon_{dt} = \upsilon_{d(t-1)} + \delta(s_d, s_t^o)$, starting by $\upsilon_{d0} = 0$.

Let us fix the time moment t and simplify the notation:

$A = A_t, \upsilon_u = \upsilon_{ut}, \upsilon_d = \upsilon_{dt}, \upsilon = \upsilon_u + \upsilon_d$
$\mu \equiv [\alpha_u, p_u, \alpha_d, p_d, \tilde{\kappa}]^T$,
$b_u \equiv [1, 0, \ldots, 0]^T$ is $(2\eta + 3)$-dimensional vector with scalar 1 at the position of α_u,
$b_d \equiv [0, \ldots, 0, 1, 0, \ldots, 0]^T$ is $(2\eta + 3)$-dimensional vector with scalar 1 at the position of α_d.

From the likelihood function (4.18) we make the logarithmical likelihood function and get

$$\hat{\mu}, \hat{\rho} \in \arg\min_{\mu,\rho} \underbrace{\left[\upsilon_u \ln(b_u^T \mu) + \upsilon_d \ln(b_d^T \mu) - \frac{1}{2} \upsilon \ln(\rho) - \frac{\mu^T A \mu}{2\rho} \right]}_{h(\mu,\rho)}$$

We are searching for the minimum of the function $h(\mu, \rho)$, the necessary condition for extremes tells us, that the partial derivation of $h(\mu, \rho)$ are equal to zero. From this condition we get

$$\upsilon_u \frac{b_u}{b_u^T \hat{\mu}} + \upsilon_d \frac{b_d}{b_d^T \hat{\mu}} = \frac{A\hat{\mu}}{\hat{\rho}}, \quad \hat{\rho} = \frac{\hat{\mu}^T A \hat{\mu}}{\upsilon} \tag{4.19}$$

If we try to express one of unknown quantities $\hat{\mu}$ or $\hat{\rho}$ we find out that one equation is a different version of the second one. So we take $\hat{\rho}$ as a parameter and we will work with the first equation from (4.19).
Introducing

$$\begin{aligned} B_u &\equiv A^{-1} b_u, \; B_d \equiv A^{-1} b_d, \\ \omega_u &\equiv B_u^T A B_u = b_u^T A^{-1} b_u, \; \omega_d \equiv B_d^T A B_d = b_d^T A^{-1} b_d, \\ \psi &\equiv B_u^T A B_d = b_u^T A^{-1} b_d, \end{aligned}$$

we search for the solution of (4.19) in following form

$$\hat{\mu} = \beta_u B_u + \beta_d B_d,$$

where β_u, β_d are scalars.
We insert this form into (4.19) and then confront the coefficients of b_d and b_u from both sides of the equation. We get:

$$\frac{\upsilon_u}{\beta_u \omega_u + \beta_d \psi} = \frac{\beta_u}{\hat{\rho}}, \quad \frac{\upsilon_d}{\beta_d \omega_d + \beta_u \psi} = \frac{\beta_d}{\hat{\rho}} \tag{4.20}$$

It has the solution

$$\beta_d = \sqrt{\frac{L + \text{sign}(N)\sqrt{(L)^2 + 4MN}}{2N}} \tag{4.21}$$

$$\beta_u = \frac{1}{\psi}\left(\frac{\hat{\rho}\upsilon_d}{\beta_d} - \beta_d \omega_d\right) \tag{4.22}$$

where

$$\begin{aligned} L &= \hat{\rho}(\frac{\psi^2}{\omega_u}(\upsilon_u - \upsilon_d) + 2\upsilon_d\omega_d) \\ M &= (\hat{\rho}\upsilon_d)^2 \\ N &= \psi^2\frac{\omega_d}{\omega_u} - \omega_d^2 \end{aligned} \tag{4.23}$$

The final solution can be constructed by a simple iterations initialized by counter $j = 1$ and a guess $\hat{\rho}_{(j-1)}$ of $\hat{\rho}$.

- Appoint $\hat{\rho}_0$.
- Evaluate guess β_{dj} inserting $\hat{\rho}_{(j-1)}$ into (4.21) and then insert β_{dj} into (4.22) and evaluate β_{uj}.
- Define $\hat{\rho}_j$ according to (4.19) with the guess of $\hat{\mu}_j = \beta_{dj}B_d + \beta_{uj}B_u$
- Increase counter j and repeat the algorithm till (expected) convergence.

Let us now shortly confront the approximation by least squares and our method. In both cases we used the parametric form of the Bellman function, in first presented method we joined the deviation done by taking time-invariant Bellman function (κ) and the deviation done by inserting the parametric form. In second method we split these two deviations. On the other hand we do not need an assumption about normality in the approximation by least squares. But the biggest difference is, as we said before, that the approximation by least squares is trying to find $K(s)$ that would minimize the deviation, the method extended iteration spread in time is trying the most probably value of $K(s)$.

4.4 Experiments

In this section we first present the results of the approximation by least squares without insertion of optimal values. This method has been tested on real data for quite a long time, so its results are predicated.
The method was tested on 5 markets: with cocoa (CC), petroleum-crude oil light (CL), 5-year U.S. treasury note (FV2), Japanese yen (JY) and wheat (W). The results are in following table.

Market	**Net profit** (USD)	**Trades**	**Winnig trades**	**Losing trades**	**Transaction cost** (USD)
CC	-101510	1147	159	988	-13920
CL	-102310	829	181	648	-10740
FV2	-21961.25	337	57	280	-3680
JY	-305712.5	1563	246	1317	-19600
W	-85355	612	102	510	-7380

There are only few performed experiments of the approximation by least squares with the optimal decisions yet, but in these few experiment the usage of the optimal decisions and of the parametric Bellman function showed good results. We present here the table of results, even though we can not confront them directly with previous method due to small number of realized trades.

Market	**Net profit** (USD)	**Trades**	**Winnig trades**	**Losing trades**	**Transaction cost** (USD)
CC	-6300	19	7	12	-220
CL	-33430	91	25	66	-1180
FV2	12949.375	15	6	9	-160
JY	-72427.5	32	5	27	-340
W	-11175	17	4	13	-200

Based on the results of experiments, even though we are still in minus numbers, we do expect big improvement.

Chapter 5

Conclusions

In this work we were able not only to explain the usage of the theory of dynamic programming and its approximations on the futures dealing, but we were also able to follow with longtime running research by proposing the new approach to solve the optimization task of futures dealing.
We proved that the best approximation of the form of the Bellman function is the parametric one $K(\Theta, s)$ and that we should do the decision only on time moments which are turning points. Next contribution of this work is the proposal of new method *Extended iteration spread in time* even though we were not able to directly confront the results of this method with the results of *The approximation by least squares*. But based on theoretical confrontation we do expect good results of our method.
As we said in the introduction, this work must be treated as small part of the research. We will have to test the method *Extended iteration spread in time* on real data and also we will have to find a way how to model the appearance of the turning points. During the construction of the parametric form of the Bellman function as well as during proposal itself we made a lot of assumptions, in the future we should examine if all assumptions are needed or how to reduce the influence of the assumptions.

Bibliography

Kárný, M. (2006). *Optimized Bayesian dynamic advising : theory and algorithms*. London, New York : Springer.

Kárný, M. and Zeman, J. (2008). *Adaptive optimization of the trade with commodities*. Draft of the report.

Křivánek, O. (2008). *Dynamic decision making based on iterations-spread-in-time strategy*. Czech technical university in Prague.