Dynamic Cash Management Models

Lancaster University Management School

Zimian Zhang

March 2021

Abstract

Classical cash management models concern how an organisation should maintain their (liquid) cash balances in order to meet cash demands over time. In these models the balance can be increased or decreased to offset penalties for not being able to meet a cash demand or the opportunity cost of holding too much cash, respectively. The external source from which this money comes from or is sent to is not explicitly modelled but is assumed to be available at all times. In this thesis we contribute to the cash management problem by discussing three novel cash management models.

To begin with, we include a second asset to the cash management model and assume the cash inflows are generated from this asset. We formulate this problem as a discrete Markov decision process (MDP) and solve it by the classic backward iteration method. We show that the optimal cash policy for this model possesses the two-threshold two-target form. Moreover we observe that the agent should take a 'safer' cash policy when the company has a balanced cash inflows and outflows.

Then we introduce loan opportunities to the model. In this problem, we allow the agent taking loans from financial intermediates. We assume there is one type of unsecured loan with fixed interest rate and the manager can take this loan repeatedly once his previous debt is paid off. We also solve this model via the discrete MDP approach. Moreover we propose a heuristic for this problem based on the policy improvement which is shown to perform strongly in our experiments.

At last, we consider an agent managing a cash account and a number of assets accounts. Hence both cash policies and asset allocation policies are studied simultaneously. Moreover we assume the agent wishes to pursuit the net profits while controlling the risk associated with his management strategies. We solve this model using a separable Piecewise linear approximate dynamic programming

(PWL ADP) approach. We also provide a heuristic based on the myopic greedy algorithm and the discrete MDP approach as benchmarks. The numerical experiments show that the PWL ADP outperforms the heuristic in terms of objective values and takes significantly less solution time comparing with the discrete MDP.

Declaration

This thesis is my own work and it has not been submitted in substantially the same form for the award of a higher degree elsewhere. Any sections of the thesis which have been published, or submitted for a higher degree elsewhere, have been clearly identified. The thesis' word-length conforms to the permitted maximum. I would like to grant the institutional repository a number of permissions and conditions with respect to online access to the work.

Acknowledgements

My most sincere gratitude goes to my supervisor Dr. Christopher Kirkbride. I could not thank him enough for his invaluable support to my research and for his priceless guidance for me to grow as a research scientist. His continuous enthusiasms for research has been and will be a great inspiration in my whole career and my life.

I would like to thank Lancaster University Management Science department for supporting my PhD journey and providing facilities and connections. Specially I would like to thank Professor Kevin Glazebrook, Professor John Boylan, Dr. Dave Worthington, Dr. Peter Jacko and Dr. Zhan Pang. I am very much grateful for their advices and help in the process of my doctoral work. I would also like to thank admin team members Ms. Gay Bentinck and Ms. Lindsay Newby for their help.

In addition, I would like to thank my friends and colleagues in Britain and back in China, Chao Gong, Xin Liu, Qiulin Yang, April Zhou, Yue Yuan, Daniel Waller, Anh Vu, Muhammad Umair, Jung Yejin and Mr. and Ms. Wood for standing by my side in good and bad times.

At last, I owe a big thank you to my family, my father Jian Zhang, my mother Lihua Zhou to whom I am deeply indebted for their love and support all these years.

Contents

List of Figures

List of Tables

List of Symbols

n	Account index
N	Total number of profitable asset accounts
t	Time index
$\mathcal{T}$	Set of discrete points in time horizon
T	End of time horizon
a_t	Action taken by the agent at time t
a^*	Optimal action
$\vec{a}_t^b, \vec{a}_t^s$	N-vectors of buying action or selling action
$\Gamma(a_t)$	Transaction cost associated with action a_t
s, s'	States
s_t	States at time t
$\vec{s}_t$	$(N+1)$-vector of state at time t
$\mathbf{S}, \mathbf{S}_t$	State space, or the set of available states at time t
$\mathbf{A}, \mathbf{A}_t$	Action space, or the set of available actions at time t
$p_t(s_{t+1}\|s_t, a_t)$	Transition probability
π, Π	Policy, and the set of all possible policies
$R_t(s_t, a_t)$	One-step reward
$V_t(s_t)$	Total reward over the planning horizon
$\hat{V}_t(s_t)$	Estimation of total reward
γ	Discounted rate
$x_t, x_t^{a_t}$	Pre-decision or post-decision cash balance
$y_t, y_t^{a_t}$	Pre-decision or post-decision asset level
υ_t	Uncontrolled cash flow
Δx	Change of uncontrolled cash flow over one period
$\Theta(\Delta x, x_t^{a_t})$	Cash shortage penalty
h	Cash holding/shortage cost coefficient
$f(x)$	Probability of density function on x
K^+, K^-, k^+, k^-	Parameters of transaction cost function
I, i	Total iteration number, iteration index
r_f	Risk-free interest rate
$\vec{r}$	Vector of return rates on assets
r_n	Return rate on the n^{th} asset
J	Total number of sample points or realisations of random variables
j	Index number of sample points or realisations of random variables
$P^\pi(\vec{s}_t)$	Probability of a company not going bankrupt under policy π

Z	Loan size
ζ	Each loan repayment
L	Loan age
ι	Loan interest rate
U	Agent's utility function
$U_{t_1 \to t_2}$	Agent's cumulative utility, agent's cumulative utility from period t_1 to t_2
$\omega, \omega_{t_1 \to t_2}$	Profit measure function, cumulative profit measure from period t_1 to t_2
$\phi, \phi_{t_1 \to t_2}$	Risk measure function, cumulative risk measure from period t_1 to t_2
λ	Coefficient of agent's preference towards to risk

Chapter 1

Introduction

The research topic for this thesis is the construction of strategies that optimise a company's cash allocation policies over multiple periods. This research starts at a classic cash management problem where the manager can replenish his cash balance by selling some of his asset, which is assumed to be available all the time. Based on this classic cash management model, we introduce three problems: the one cash account one asset account management problem, the two accounts management problem with loan opportunities and the cash management problem with multiple assets. To introduce this thesis, this chapter presents the research overview, the research significance and contributions, and the thesis outline.

1.1 Overview

Cash management deals with the company's cash holding and allocation strategies. With insufficient cash holdings, the company exposes itself to the risk of cash shortage, which normally results in a great amount of penalty. On the other hand, a high cash holding level may indicate the inefficient use of resources. By holding the financial resources in the form of cash, the company renounces the profit it could have gained if such resources have been invested into other assets. The goal of cash management is obtaining a cash holding strategy so that the manager, which will be referred to as 'the agent' throughout this thesis, could control the risk of cash deficit while accruing the profit by investing extra cash into profitable assets.

Figure 1.1 illustrates the structure of our study. Our work starts with a traditional cash management model where the company's asset is assumed infinite and available at all times. In this model, the company faces external cash inflows and cash outflows/demands in each period. At the beginning of each period, the manager can replenish his cash balance or withdraw cash from the asset account. For each period cash deficit results in shortage costs while positive cash levels causing cash holding costs. The objective is to minimise the total costs over an infinite horizon. We use the cash holding level as the system's state and formulate the problem as a discrete Markov decision process (MDP). Then we solve this MDP using the classic backward recursion method.

Next we develop the traditional cash management model to a two accounts model by the inclusion of a second asset. In this model we consider a company with a cash account and an asset account. We allow the agent to be able to make a transfer between these two accounts at each decision epoch with some transfer fee. A negative cash level causes cash shortage costs while the asset generates incomes. The goal is to maximise the net profit over the planning horizon. Then we formulate this model as a discrete MDP using both the cash level and the asset level as the system's state. The backward recursion method is also adopted to solve this model.

At last we propose two advanced models based on the two accounts cash management model. In the first advanced model we introduce loan opportunities to the two accounts model. That is we allow the agent to take loans from financial intermediates. Once the loan is taken, the cash account receives that loan immediately and the offset continues in following periods till the debt is paid off. We assume the agent can take this loan repeatedly if the company has no previous debt outstanding. Two approaches are proposed to solve this model. The first approach is formulating the model as a three-dimensional discrete MDP and solve it using the backward iteration method. The second approach is a heuristic based on the policy improvement algorithm. In the second advanced model we replace the asset account with multiple assets. To solve this problem we propose a double-pass separable Piecewise linear approximate dynamic programming (PWL ADP) approach, which is a multi-dimensional version of the successive projective

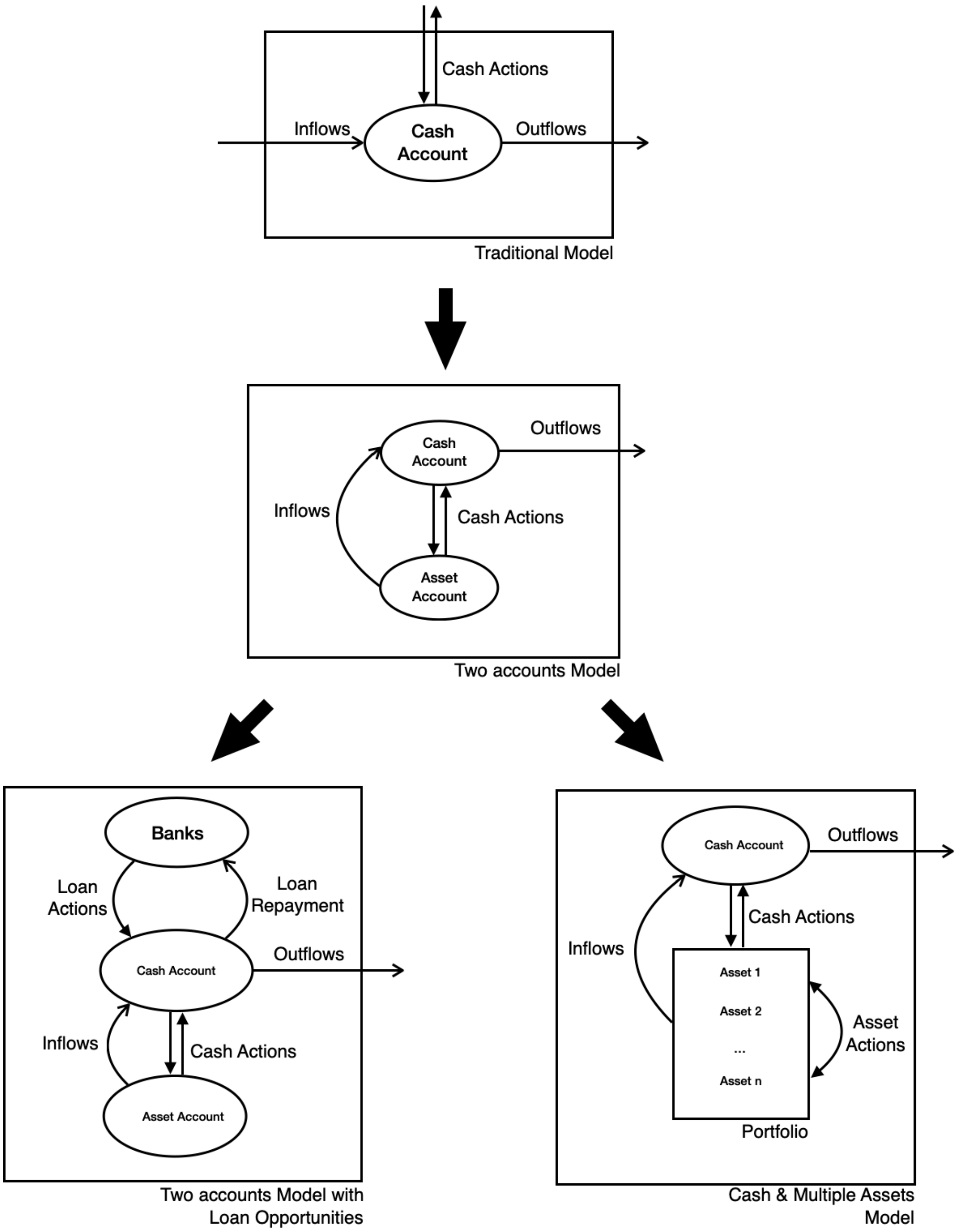


Figure 1.1: Structure of our study

routine (SPAR) proposed by Powell et al. (2004). We also solve this problem via the discrete MDP approach and a heuristic approach which is based on the static stochastic programming approach and use the results as benchmarks.

1.2 Research significance and contributions

This thesis contributes to both the cash management theory and the associated strategic algorithms. In terms of enriching the literature of cash management models, our studies address the source of cash inflows. In traditional cash management studies, cash inflows are considered exogenous to the model and are normally described by a stochastic process. In our research, we discuss the scenario where the cash inflows are equivalent to the incomes generated by company's assets. To find the cash management policy under such condition, we present a two accounts cash management model where the agent adjusts the holdings of a cash account and an asset account to accrue net profits over an infinite planning horizon. Our study shows the optimal policy of the two accounts model can be seen as a multi-dimensional version of the classic two-threshold two-target policy presented by Eppen & Fama (1968).

Our research also contributes to the literature by the inclusion of loan opportunities. In practice, taking loans from financial intermediates is common for companies to finance themselves. However in the literature, very few cash management studies discuss the possibilities of taking loans. To our best knowledge, only Sastry (1970) and Nascimento & Powell (2010) consider loans in the cash management model. In their work, taking short term loans is considered as a compulsory action when the company's cash balance cannot meet the cash demand. The repayment of loan interest is seen as a type of cash shortage penalty. In our research we expand the cash management model to a cash-asset-loan management model where the agent is allowed to replenish his cash balance by selling asset as well as taking short term loans.

At last we present a cash management model with multiple assets i.e. instead of one general asset account, we allow the agent investing his cash surplus to a portfolio. The agent's objective is to maximise the net profit over the planning horizon by adjusting his cash balance as well as the holdings of each asset account. This model can be seen as a combination of a cash management model and a portfolio management model. To our best knowledge, our work is the first work combining a dynamic cash managment model with the dynamic portfolio management theory.

In the algorithmic sense, we propose a novel heuristic for the cash management with loans problem based on the policy improvement method. The basic idea can be described as follows: we first solve the cash management problem where the loan is unavailable to the manager. Then we solve the problem in which a bank offers one loan option to the manager. Regardless of the manager's decision, the loan opportunity expires after this time period. We show that this cash management model with one loan opportunity can be easily solved given the optimal policy from the no loan model. We also show that the cash management model with any number of loan opportunities can be solved given the result from the model with one less loan opportunities. Hence start with the no loan model, we solve the model with one more loan opportunity at each iteration until the state values (i.e. the expected net profits) do not change significantly. We show that after a number of iterations, the result from this model is a good approximation of the model with infinite loan opportunities. We also formulate the cash management model with infinite loan opportunities as a discrete Markov decision process and solve it via the traditional backward recursion method. Compared with this discrete MDP approach, the heuristic is shown to perform strongly in our experiments while requiring significant less solution time.

Moreover in the cash management model with multiple assets we present a double-pass separable PWL ADP approach which is based on the SPAR algorithm proposed by Powell et al. (2004). To our best knowledge, this is the first attempt in the literature using this algorithm to solve a high-dimensional, combined, dynamic cash and asset model. We also solve this model via a heuristic method and a dynamic programming method as benchmarks. The PWL ADP approach is shown to outperform the heuristic method in terms of the objective value while taking less computational cost comparing with the dynamic programming method.

1.3 Thesis outline

This thesis is organised in eight chapters:

Chapter 2 presents the literature of cash management studies. It starts with an introduction to the origin of cash management including the first research addressing the cash management problem (Baumol 1952), the first cash management model introducing stochastic elements into the model (Miller & Orr 1966), and Eppen & Fama's research (Eppen & Fama 1968) which suggests the two-trigger two-target form of cash policies that is widely accepted in following studies. Then we discuss some new perspectives in cash management studies including the modelling of the uncontrolled cash flows, the operation conditions (i.e. the transfer fees, the holding costs and the cash shortage penalties), the incorporation of cash management in asset management, and the methodologies adopted in cash management studies. At last we present some studies on cash policies from a financial perspective.

Chapter 3 provides some basic concepts and tools used throughout this thesis. Here we present the framework of the discrete Markov decision process. Then we explain how the state values are estimated using the tabular method and the min-affine functions. At last we introduce the conditional value-at-risk and some other time consistent risk measures.

In Chapter 4 we discuss the traditional model where the agent is allowed to replenish his cash balance by selling some of his assets which are assumed available all times. Moreover we formulate this problem as a discrete MDP and solve it via the backwards recursion method. At last we conduct a series of numerical experiments under different operational conditions.

Chapter 5 develops the traditional cash management model by the inclusion of a second asset. This asset generates an income which is considered as the source of cash inflows. We also formulate this problem as a two dimensional MDP. In addition, we show that with a small modification, the MDP approach can be used to calculate the company's insolvency probabilities. Furthermore we discuss the scenario where the second asset does not generate a cash income directly. Instead, the asset grows at a fixed rate. This model may applies to the case where the agent holds financial assets.

Chapter 6 introduces loan opportunities to the two accounts model. In this chapter we propose two approaches to solve this cash-asset-loan model. The first approach is formalising the model as a discrete MDP and solve it via the backward recursion method. We also propose a heuristic based on the policy improvement method. At last we conduct numerical experiments to compare the performance of these two approaches and examine the policies under different loan conditions (i.e. the loan interest rate, the loan age, and the loan size).

Chapter 7 discusses the model where the agent holds multiple assets instead of one general asset account. Three approaches are proposed to solve this model: a heuristic approach based on a static model, the discrete MDP approach, and the approximate dynamic programming approach where we use the separable Piecewise linear functions to approximate state values. At last we conduct a set of numerical studies based on the real data and compare the accuracy and the efficiency of these approaches.

Chapter 8 summarises the thesis and points out the research limitations and the plan of future research.

Chapter 2

Literature Review in Cash Management

2.1 Introduction

In financial management, cash is a unique asset which has full liquidity and low profitability. Holding too much cash means the inefficiency of financial resource allocation, while an insufficient cash balance normally exposes companies to the risk of becoming overdrawn. Hence cash holding strategies are of a great interest to the decision makers. In this section we give a literature review in the topic of cash management. To begin with, we briefly introduce the origin of the cash management studies. We highlight three important early studies: Baumol's study (Baumol 1952) which is the first study on the cash management problem, Miller & Orr's model (Miller & Orr 1966) which introduces a stochastic element into cash management models and Eppen & Fama's research (Eppen & Fama 1968) which suggests the two-trigger two-target form of cash policies that is widely accepted in following studies. Then we introduce the new developments on the cash management studies focusing on the following aspects: (a) the discussion of cash flows/demands process, (b) the operational conditions including the transaction costs, the cash holding/shortage costs and the risk consideration, (c) the incorporation of cash management in asset management, (d) the methodologies adopted in cash management studies and (e) the discussion of the determinants on cash policies from the perspective of financial theories.

2.2 Origin of cash management studies

One of the earliest studies on cash management problems is contributed by Baumol (1952) where the cash is considered analogous to other types of inventory and the author consequently makes a parallel between monetary theory with inventory theory. In their study, the economic order quantity (EOQ), which is one of the most common approaches in inventory studies, is adopted to analyse the advantages and the disadvantages of holding cash. They identify two sources of cash management cost, the transaction cost i.e. the cost associated with investment or withdrawal transactions and the opportunity cost i.e. the 'opportunity cost' which represents the profit renounced by the manager when he holds the cash instead of investing into profitable assets. In their model it is assumed that the manager withdraws $£a$ cash from asset each time spaced evenly throughout the year. They also assume the demand for cash over each year is predetermined and denoted by D. Hence for each year D/a transactions are required and if the transfer fee Γ is fixed for each transaction, the total transaction cost for this year is $\Gamma D/a$. If the manager withdraws $£a$ and spends it in a steady stream, the average cash holding will be $a/2$. The 'opportunity cost', namely the profit the manager could have gained if he had invested this cash into the asset, with annual return rate r is equal to $rx/2$. management cost function:

$$TC = \frac{\Gamma D}{a} + \frac{ra}{2}.$$

The optimal withdrawals can be given by taking the first derivatives of the total cash management cost function with respect to a and it can be written as:

$$a = \sqrt{\frac{2\Gamma D}{r}}.$$

Many early studies on the cash management problem are based on this model. For example, Tobin (1956) completes Baumol's model (Baumol 1952) by permitting the number of transactions into cash to take on only positive integer values. Moreover, Tobin's model (Tobin 1956) maximises his earnings of interest net of transaction costs instead of minimising the total cost of cash management and proves one of the Baumol's assumptions (Baumol 1952), namely the cash withdrawals should be equally spaced over time and equal in size. Sastry (1970) introduces the concept of the cash shortage and takes into consideration of the cash shortage penalty. In this study, once the company has cash deficit, the manager must take a loan from financial intermediaries and pay the related interest. They modify the optimal cash withdrawals to

$$a = \sqrt{\frac{2\Gamma D}{r} \cdot \frac{h}{r+h}}$$

where h is the cash shortage penalty coefficient (e.g. loan interest rate). Later Whalen (1966) develops the cash management theory with the consideration of cash deficit and introduces the concept of the cost of cash illiquidity. In addition, Whalen (1966) allows the variability in cash inflows and outflows and shows that the optimal precautionary cash balance should be higher in scenarios with more uncertainty in cash flows.

Miller & Orr (1966) formally identify two accounts in the cash management model: a short term asset account which has low risk and high liquidity and a cash account which can be used to meet the cash demand. At any time, the manager can sell his asset to replenish the cash account or invest his cash into the profitable asset. For each transaction, regardless of the transaction size, a fixed transfer fee must be paid. They also introduce a stochastic element into the model by assuming that the cash flows can be described by Bernoulli process, i.e. for each period the cash level will either increases by $£m$ with probability Δx or decreases by Δx with probability $1-p$. Then they show that the optimal cash management policy poccesses the (L, B, U) form where two limits for the cash holding level, namely the upper limit (U) and the lower limit (L) will be defined. The agent should adjust his cash balance to a target level B only when the firm's cash holding level reaches the upper limit or the lower limit.

Miller & Orr's study (Miller & Orr 1966) is developed in Eppen & Fama's work (Eppen & Fama 1968) where the authors formalise the cash management problem into a Markov decision process and adopt the linear programming method. Through numerical experiments, Eppen & Fama (1968) suggest that for a model with the fixed plus proportional transaction cost function, the optimal cash management policy is of the two-trigger two-target form, i.e. if the cash holding level exceeds the upper trigger level, the agent should buy the asset and reduce his cash level back to the upper target and if the cash level goes below the lower trigger level, the agent should sell his asset and replenish the cash account back to the lower target. Later this two-trigger two-target form of cash management strategy is proved optimal by Constantinides & Richard (1978), Harrison et al. (1983) and Milbourne (1983).

2.3 Cash flows/demands process

Since Eppen & Fama (1968) and Milbourne (1983) proposed the two-trigger two-target policy, the theory of cash management has been developed in many aspects. One of the most important aspect is the consideration of the uncontrolled cash flows process (or equivalently the cash demands process) in the cash management theory. For example, in Bar-Ilan et al.'s study (Bar-Ilan et al. 2004), the process of cash flows is assumed to be a superposition of Brownian Motion and a compound Poisson process with positive and negative jumps. According to Bar-Ilan et al. (2004), this assumption provides a more general model for cash demand as the Brownian motion is a good description in normal times while the compound Poisson process describing the critical losses in financial crisis. Similar studies are also conducted by Benkherouf & Bensoussan (2009) where the cash demand is assumed to be a mixture of a diffusion process and a compound Poisson process, Yamazaki (2016) where the uncontrolled cash flows follow a general Levy process with only negative jumps, and Azcue & Muler (2019) where a compound Poisson process with two-sided jumps and negative drift is used to describe the money stock.

The limitation of the classic independent identical distribution (i.i.d.) assumption on cash flows is pointed out by Hinderer & Waldmann (2001). In their work, they propose a new 'environmental variable' to describe the business environment. The 'environmental variables' are self-related and the random variables of cash flows are realised depending on the 'environmental variables'. The i.i.d. assumption is also relaxed in Gormley & Meade (2007) where the authors use a data set from a large multinational and develop a time series model to forecast the cash flows. In light of the idea of applying forecasting techniques into the cash management problem Salas-Molina et al. (2017) examine the effect of the accuracy of cash flows' forecasts on cash management strategies and provide a model to help the manager to decide whether it is worthy to improve the forecasts accuracy while making cash management strategies. On the other hand, Yao et al. (2006) reckon that the past historical data is unable to provide a forecast for cash demand and hence consider the cash demands based on the fuzzy logic concepts and develop a fuzzy stochastic single-period model. In their work, the cash demands are described as a 'hybrid data' which consists of fuzzy information and random components.

Moreover, the multi-dimensional cash flows/demands are also studied in the literature. Baccarin (2009) studies a general multi-dimensional cash management problem where the cash process is viewed as a diffusion process in a multi-dimensional space. In his work, the existence of the optimal policy is proved and a numerical experiment in two dimensions is calculated as an example. Alvarez & Lippi (2013) studies the problem where the demand for cash comes from two sources: one is frequent and small and the other one is infrequent and large. Nascimento & Powell (2010) studies a mutual fund cash management problem where the institutional and retail cash demands are identified separately. In their model, it is assumed that if the cash balance cannot meet the total demand, the manager must liquidate his portfolio and hence the liquidation cost occurs. If the institutional demand is higher than the cash balance, the manager must take a loan from financial intermediaries and the financial cost is also charged.

2.4 Operational conditions

In the literature there are mainly three types of transaction cost function adopted in the cash management models: the fixed transaction cost where a fixed amount of transfer fee is charged for each transaction (e.g. Baumol (1952) and Tobin (1956)), the proportional transaction cost where the transfer fee is charged depending on the size of the transaction (e.g. Bensoussan et al. (2009) and Nascimento & Powell (2010)) and the fixed plus proportional transaction cost function which is a combination of both (e.g. Hinderer & Waldmann (2001), Baccarin (2002) and Salas-Molina, Pla-Santamaria & Rodriguez-Aguilar (2018)).

However other types of transaction cost are also considered in the literature. For example, Baccarin (2009) imposes polynomial growth conditions on the transaction costs since they reckon that the transaction costs normally increase less than proportionally to the size of the transaction. Sato & Sawaki (2009) consider the scenario where the company is financed via two short term funds, hence two different transaction costs occur in the model. In Baccarin & Sanfelici (2006), the transaction cost function is a mixture of a fixed component and a variable component which is assumed to be sublinear. This assumption is based on the observation of the real world that two separate transfers are usually more expensive than implementing the same with one transaction.

Another essential aspect of cash management models is the holding/shortage costs. The cash holding cost measures the price of keeping too much cash while the shortage cost measures the price of not having sufficient cash balance. Early studies have proved the existence of optimal solutions in the cash management models with the linear holding/shortage costs function (e.g. Constantinides & Richard (1978), Harrison & Taylor (1978) and Harrison et al. (1983)). Penttinen (1991) is one of the earliest studies considering the nonlinear holding/shortage costs functions in the cash management problem. In their model, general convex functions are adopted to describe the holding/shortage costs. Later Baccarin (2002) proves that an optimal control band policy always exists in the model with a quadratic holding/shortage costs function. In light of these studies, Baccarin & Sanfelici (2006) and Baccarin (2009) impose polynomial growth conditions on the holding/shortage costs in their cash management model.

Keeping idle cash increases the 'opportunity cost', but an insufficient cash holding level exposes the company to the risk of overdraft penalties. Most studies in the literature focus on minimising the expected cost of cash management strategies over the planning horizon. However there are scholars who also take into consideration the risk associated with the cash policies. For example Salas-Molina, Pla-Santamaria & Rodriguez-Aguilar (2018) incorporate risk considerations with the cash management problem and propose a multi-objective model that allows the manager adjust his cash policies based his risk preference. In addition a cost-risk space is derived in Salas-Molina, Rodriguez-Aguilar & Díaz-García (2018) for the cash management models and loss curves are constructed to assess the performance of models under different operational conditions (e.g. transaction cost parameters, risk preference). Salas-Molina (2020) studies the robustness of the multi-objective cash management model to the misspecifications in both means and variances of the cash flows process.

2.5 Incorporation of cash management in asset management

Most studies in the literature only focus on the management of cash balance. A recent development on the topic of cash management is considering cash as one of the financial assets and studying the cash policies within the topic of asset management. Bensoussan et al. (2009) consider a cash management model with two types of financial asset: one is the deposits in a bank account and the other one is the investment in stock. The bank deposit is an asset with high liquidity and low profitability and it can be used to meet the cash demand. The stock, on the other hand, cannot fulfil the demand for cash directly but generates two types of profit: the fixed dividends and the uncertain capital gain. Wu & Li (2012) propose a mean-variance portfolio optimisation model with the consideration of cash flows and cash holding strategies. Then they show the existence of optimal solutions in their model and analyse the mean-variance efficient frontier. da Costa Moraes & Nagano (2014) propose a management model for three accounts: apart from the cash account, there are two potential assets: the first asset has the full liquidity but lower profitability and the second asset has higher profitability but requires 0 to 30 days of lead time. Yao et al. (2013) and Yao et al. (2016) incorporate the cash management problem into the asset-liability management model where the cash account is treated as one of many assets affecting the company's liability. In these studies, a mean-variance model over multiple period horizon is proposed and formulated as a stochastic optimal control problem.

2.6 Methodologies

Various methodologies have been adopted in cash management studies. In Baumol (1952), the cash management model is considered as a special case of the inventory model and the economic order quantity (EOQ) method, which is one of the most common approaches in inventory studies, is adopted in their studies. The basic idea of the EOQ approach is analysing both advantages and disadvantages of holding an inventory and looking for the optimal management strategies. Later in Eppen & Fama (1968), the cash management problem is formulated as a Markov decision process and solved via the linear programming approach. Based on the results from the numerical studies, they suggest that the optimal solution of their cash management model possesses the (L, l, u, U) form. Since then, the optimal impulse control technique firstly introduced by Bensoussan (1984) is widely adopted in the studies of cash management (e.g. Feinberg & Lewis (2007), Baccarin (2009) and Baccarin & Marazzina (2014)). In an optimal impulse control problem, actions are taken when the pre-decided cash holding levels are hit. A usual technique in impulse control problems is associating the problem with a quasi-variational inequality. The optimal policy can be derived once a regular solution of this inequality is found (see Constantinides & Richard (1978), Harrison et al. (1983), Eastham & Hastings (1988) and Korn (1997)). A number of machine learning techniques are also used in cash management studies. Liu & Xin (2008) and Schroeder & Kacem (2019) design online algorithms to study the cash management strategies which do not require the knowledge of the distribution of cash flow, only the information on the lower and upper bound of the future cash demand. Relying on machine learning and mathematical programming techniques, Salas-Molina (2019) also relaxes the assumption of knowing the cash flow's distribution and provides a data-driven procedure to fit the cash management models to the data. da Costa Moraes & Nagano (2014) adopt genetic algorithm and particle swarm optimisation algorithms in cash management models with multiple assets. Through numerical experiments they show that both algorithms are applicable in finding the optimal solutions and the particle swarm optimisation algorithms outperforms genetic algorithm. Nascimento & Powell (2010) formulate the cash management problem as a dynamic program and provide an approximate dynamic programming algorithm in finding the optimal cash policy. In their algorithm, Piecewise linear functions are used to estimate the value of states and the optimal solution is derived from not the value of each Piecewise linear function but the gradient of each segment of the functions.

2.7 Cash policies from a financial perspective

Although the modelling of cash management is a popular topic in the field of operations research (O.R.), the cash control policies have also been discussed from the perspective of financial theories. Although they differ from the studies of cash management in O.R. to a large extent, these financial studies provide essential insight on the determinants of cash policies and could be highly valuable to the cash management modelling. Almeida et al. (2004) study the sensitivity of corporate cash holding level to the volatility of the cash flows. Their study shows that the constrained company tends to hold more cash in the scenario with the volatile cash flows. Such relationship is not valid for the unconstrained companies, i.e. the companies that have unrestricted access to the external capital. The sensitivity of cash holding to the cash flow's volatility is also identified in Yan (2006) where they study the cash holding strategies adopted by the mutual fund companies. Khurana et al. (2006) show that the financial developments increase the sensitivity of companies' cash holding level to the cash flows and Kusnadi & Wei (2011) argues that compared to financial developments, the legal protection of investors plays a more essential role in companies' cash controlling policies. Chen et al. (2014) take a survey in China covering $12,400$ firms in 120 cities and show that the high quality of local government leads to lower cash holding levels for the companies. The negative relationship between the cash holding and both the geographic and industrial diversification is pointed out by Fernandes & Gonenc (2016). Kusnadi & Wei (2011), Sasaki & Suzuki (2019) and Cui et al. (2020) study the impact of banks policies on the non-financial companies' cash policies. Their researches show that healthy banks or banks with more liquidity normally induce the growing firms to hold more cash. These determinants on cash holding policies verified by empirical studies are rarely considered by O.R. scholars and could be important reference to the future studies on cash management.

Chapter 3

Basic Concepts and Notations

3.1 Introduction

In this chapter we summarise the basic concepts and the notation that will be used in following chapters. To begin with we introduce the framework of the discrete Markov decision process (Bellman 1957) which will be adopted throughout this study. In Chapters 4, 5, and 6, the cash management problem will be formulated as a discrete Markov decision process and the value for each state will be approximated via the linear interpolation method. In Chapter 7, the state value will be approximated by a set of Piecewise linear functions. Both the linear interpolation method and the Piecewise linear functions approximation method are introduced in Section 3.3. Then in Section 3.4, we give a brief introduction to the conditional value-at-risk (Rockafellar et al. 2000). We also highlight the lack of time consistency of the conditional value-at-risk and introduce some studies on the time consistent risk measures. The risk measure will be adopted in Chapter 7 as a component of the investor's objective.

3.2 Markov decision process

A Markov decision process (MDP) is a discrete-time stochastic control process which models decision making problems in which outcomes are partly stochastic and partly under the control of the manager. According to Puterman (2014), five elements are normally identified in the framework of a Markov decision process

(MDP), namely decision epochs, states, actions, transition probabilities and rewards. At each decision epoch in the planning horizon, the system occupies a state which can be observed by the decision maker. Based on the information of the current state, an action is chosen from a set of feasible actions by the agent. As a result the agent receives an immediate reward (or a penalty which can be seen as a negative reward) and the system transitions to another state determined jointly by the action and some probability functions. The goal is to find the optimal decision rule that maximises the cumulative reward over the planning horizon.

1. Decision epochs

   Assume, in a decision making problem, the agent can take actions over a planning horizon and wishes to find the optimal decision rule. In the MDP framework, the continuous planning horizon is divided into discrete periods. The set of discrete periods can either be finite (i.e. $\mathcal{T} = \{1, 2, ..., T\}$ for some integer $T < \infty$) or countably infinite (i.e. $\mathcal{T} = \{1, 2, ..., \infty\}$). We write

   $$\mathcal{T} = \{1, 2, ..., T\}, T \leq \infty$$

   to include both cases. The decision epoch is the point of time when the decision is made and we let it correspond to the beginning of each period.

2. States

   At the beginning of period $t \in \mathcal{T}$, the system occupies a state s_t. Take the cash management problem as an example, in a problem where only the cash account management is of interest to the agent, we could define the state as the amount of cash holdings. However when the management strategy involves more than one account, a N-dimensional state $\vec{s}_t = (s_t^1, s_t^2, ..., s_t^N)$ where $\vec{s}_t \in \mathbf{S}_t = \mathbf{S}_t^1 \times \mathbf{S}_t^2 \times ... \times \mathbf{S}_t^N$ should be adopted. Note that $\mathbf{S}_t$ represents the set of all possible states at time t and for $n = \{1, 2, ..., N\}$, $\mathbf{S}_t^n$ represents the set of all possible values for the n^{th} state element s_t^n.

3. Actions

   We let $\mathbf{A}$ be the set of all possible actions and $\mathbf{A}_t \subseteq \mathbf{A}$ be the set of all feasible actions at period t. At each decision epoch, the agent observes the system's current state s_t and takes an action $a_t \in \mathbf{A}_t$ based on the state information. We define a decision rule A_t^π as the mapping from the state set to the action set, i.e.

$$A_t^\pi : \mathbf{S}_t \to \mathbf{A}_t.$$

In addition, we define a policy π as the set of all decision rules over the planning horizon, i.e.

$$\pi = (A_1^\pi, A_2^\pi, ..., A_T^\pi), \quad T \leq \infty, \quad \pi \in \Pi$$

where Π represents the set of all possible policies.

4. Transition probabilities

   If at period t, the system occupies the state s_t and the action a_t is taken by the decision maker, the system state at the next decision epoch is determined by the transition probability $p_t(s_{t+1}|s_t, a_t)$. We assume that

$$\sum_{s_{t+1} \in \mathbf{S}_{t+1}} p_t(s_{t+1}|s_t, a_t) = 1.$$

   In the MDP, we also assume the Markov property, i.e. given any current state s_t, the future states are independent of how the system reached s_t.

5. One-step reward

   In our study, we let the agent receive an amount of immediate reward as a result of taking the action a_t in the state s_t. Hence the immediate reward for a state-action pair can be computed via a two-argument function $R_t : \mathbf{S}_t \times \mathbf{A}_t \to \mathbb{R}$. Note that the reward can be interpreted as the value of an income or the negative value of a penalty.

Assume that the future rewards are discounted by the factor γ and the agent's objective is to find the optimal policy that maximises the expected discounted cumulative rewards over the planning horizon $\mathcal{T} = \{1, 2, ..., T\}$.

$$\max_{\pi \in \Pi} \mathbb{E} \left\{ \sum_{\tau=1}^{T} \gamma^{\tau-1} R_\tau(s_\tau, a_\tau) \right\}, \quad T \leq \infty. \tag{3.1}$$

To reduce the complexity of the optimisation problem (3.1), we formulate it as a dynamic program using the classic backward recursion method. We first introduce the concept of a state value $V_t(s_t)$, which is defined as the expected sum of the discounted reward received in following periods given the system visiting the state s_t at time t and the optimal policy is adopted afterwards, i.e.

$$V_t(s_t) = \max_{\pi \in \Pi} \mathbb{E} \left\{ \sum_{\tau=t}^{T} \gamma^{\tau - t} R_\tau(s_\tau, a_\tau) \right\}, \quad T < \infty.$$

In the infinite horizon case, the value of state s can be expressed as:

$$V(s) = \max_{a \in \mathbf{A}} \mathbb{E} \left\{ \sum_{\tau=0}^{\infty} \gamma^{\tau} R(s, a) \right\}.$$

The state value can be expressed by the recursive equation which is known as the Bellman's optimality equation. In the finite horizon case, the recursive equation can be written as:

$$V_t(s_t) = \max_{a_t \in \mathbf{A}_t} \left\{ R_t(s_t, a_t) + \gamma \sum_{s_{t+1} \in \mathbf{S}_{t+1}} p_t(s_{t+1} | s_t, a_t) V_{t+1}(s_{t+1}) \right\}. \tag{3.2}$$

In the infinite horizon case the recursive equation can be written as:

$$V(s) = \max_{a \in \mathbf{A}} \left\{ R(s, a) + \gamma \sum_{s' \in \mathbf{S}} p(s' | s, a) V(s') \right\}. \tag{3.3}$$

In the case of a finite planning horizon, we define the terminal value state $V_T(s_T)$ based on the state information s_T. In the case of an infinite planning horizon, the terminal state information is negligible due to the discount factor γ. Then the state values as well as the optimal policy over the planning horizon can be obtained recursively via the equation (3.2) and (3.3).

3.3 State value estimation

In a cash management problem, the levels of cash balance and other asset-holdings are normally used as the MDP states. However the level of each financial account is continuous while the discrete MDP framework normally requires discrete states. Hence the estimation of state values plays an essential role in our studies. In this section we discuss two estimation methods for the state values. In the problem with a small state space, we use the tabular method where the state space is discretised and the value of each discrete state is recorded in an array of a table. The values for those states among the discrete states are approximated via the linear interpolation method. In the problem with a large state space, we use separable Piecewise linear concave functions to estimate the state values.

Tabular method and linear interpolation

In the task with a small state space, we approximate the value using the tabular method. Assume that we are dealing with a problem with N-dimensional continuous states $\vec{s} = (s^1, s^2, ..., s^N)$ where $\vec{s} \in \mathbf{S} = \mathbf{S}^1 \times \mathbf{S}^2 \times ... \times \mathbf{S}^N$. In the tabular method, we discretise the space of the n^{th} state element into a discrete set $\mathbf{S}^n = \left\{ s^n_{[1]}, s^n_{[2]}, ..., s^n_{[M]} \right\}$ for $n = \{1, ..., N\}$. Then we use arrays or tables to record the discrete state values. When we recursively solve the equation (3.2) and the state s_{t+1} lies in the middle of discrete states, the value $V_{t+1}(s_{t+1})$ can be approximated using the linear interpolation method. In the case of a one dimensional state problem, the linear interpolation method can be described as follows: if the state s lies between the discrete states $s_{[m]}$ and $s_{[m+1]}$ and the values $V(s_{[m]})$ and $V(s_{[m+1]})$ can be obtained from a lookup table, the value of state s can be approximated as:

$$V(s) \approx \frac{s - s_{[m]}}{s_{[m+1]} - s_{[m]}} \left(V(s_{[m+1]}) - V(s_{[m]}) \right). \tag{3.4}$$

In the case of a multi-dimensional state model, the approximation equation (3.4) can be implemented for each dimension. For example, if we want to approximate the value of a N-dimensional state $\vec{s} = (s^1, s^2, ..., s^N)$ and for each state element s^n, the adjacent discrete states are $s^n_{[m^n]}$ and $s^n_{[m^n+1]}$, the value $V(s^1, s^2, ..., s^N)$ can be approximated via the following procedure:

$$
\begin{cases}
V(s^1, s^2, ..., s^N) & \approx \dfrac{s^1 - s^1_{[m^1]}}{s^1_{[m^1+1]} - s^1_{[m^1]}} \Big(V(s^1_{[m^1+1]}, s^2, ..., s^N) \\
& - V(s^1_{[m^1]}, s^2, ..., s^N) \Big) \\
V(s^1_{[m^1]}, s^2, ..., s^N) & \approx \dfrac{s^2 - s^2_{[m^2]}}{s^2_{[m^2+1]} - s^2_{[m^2]}} \Big(V(s^1_{[m^2]}, s^2_{[m^2+1]}, s^3, ..., s^N) \\
& - V(s^1_{[m^1]}, s^2_{[m^2]}, s^3, ..., s^N) \Big) \\
V(s^1_{[m^1+1]}, s^2, ..., s^N) & \approx \dfrac{s^2 - s^2_{[m^2]}}{s^2_{[m^2+1]} - s^2_{[m^2]}} \Big(V(s^1_{[m^1+1]}, s^2_{[m^2+1]}, s^3, ..., s^N) \\
& - V(s^1_{[m^1]}, s^2_{[m^2]}, s^3 ..., s^N) \Big) \\
V(s^1_{[m^1]}, s^2_{[m^2]}, s^3, ..., s^N) & \approx \dfrac{s^3 - s^3_{[m^3]}}{s^3_{[m^3+1]} - s^3_{[m^3]}} \Big(V(s^1_{[m^1]}, s^2_{[m^2]}, s^3_{[m^3+1]}, s^4, ..., s^N) \\
& - V(s^1_{[m^1]}, s^2_{[m^2]}, s^3_{[m^3]}, s^4, ..., s^N) \Big) \\
& ...
\end{cases} . \tag{3.5}
$$

Separable Piecewise linear concave approximation

One limitation of the tabular method is that a large amount of memory is required to approximate state values and policies, especially when the model states involve multiple dimensions. Hence we implement approximation functions to estimate state values when we deal with optimisation problems with multi-dimensional states. To be specific, in Chapter 7 we consider the separable Piecewise linear concave approximations for the values of combinations of cash and risky assets.

In the literature, the separable Piecewise linear approximation approach is wildly adopted in optimisation problems. For example, Godfrey & Powell (2002*a*) and Godfrey & Powell (2002*b*) adopt the separable Piecewise linear approximations for the net profit in fleet management. In Godfrey & Powell (1997), the Piecewise linear approximation is applied to inventory and distribution problems. In He et al. (2012), the separable Piecewise linear function is used to approximate the secrete estradiol levels and the ovary diameters in the dosage control problem in the controlled ovarian hyperstimulation treatment.

The basic idea of the separable Piecewise linear approximation can be explained as follows. Given a N-dimensional state $\vec{s} = (s^1, s^2, ..., s^N)$ we assume that the contribution of each state element to the state value can be described as a min-affine function and the value for the state can be approximated as the sum of all the min-affine functions, i.e.

$$V(s^1, s^2, ..., s^N) \approx \sum_{n=1}^{N} \min_{m=1,...,M} \{b_{m,n}s^n + c_{m,n}\}.$$

One advantage of this approximation is that the optimisation of the separable Piecewise linear concave function can be easily transformed to a linear programming model, i.e. the objective function

$$\max \sum_{n=1}^{N} \min_{m=1,...,M} \{b_{m,n}s^n + c_{m,n}\}$$

is equivalent to

$$\begin{aligned} & \max \sum_{n=1}^{N} z_n \\ \text{s.t.} \quad & z_n \leq b_{m,n}s^n + c_{m,n} \quad \text{for} \quad n = 1, ..., N;\ m = 1, ..., M. \end{aligned} \tag{3.6}$$

3.4 Conditional value-at-risk and time consistent risk measure

In a typical cash management problem, the goal of the investor is usually to maximise net profit or minimise the total cost. However the agent may also be interested in the risk associated with management strategies. Hence in Chapter 7, the policy risk is considered as a component of the investor's objective as well as the net income.

The expected shortfall, also known as the conditional value-at-risk (CVaR) is wildly adopted in financial management studies as a risk measure. CVaR which accounts for the expected return or loss in the worst scenarios is defined as follows: Let $g(\vec{r}, a)$ be the gain function given a stochastic return vector $\vec{r} \in \mathbf{R}$ and an action $a \in \mathbf{A}$. Let $f(\vec{r})$ be the probability density function of $\vec{r}$. The probability of the gain $g(\vec{r}, a)$ at least equal to a threshold ρ is:

$$\Psi(\vec{r}, \rho) = \int_{g(\vec{r},a) \geq \rho} f(\vec{r}) \mathrm{d}\vec{r}.$$

With respect to a specified probability level $\alpha \in (0, 1)$, the values of value-at-risk (VaR_α) is the highest amount ρ such that with probability α, the gain will at least equal to ρ, i.e.

$$\rho_\alpha(\vec{r}) = \max \left\{ \rho \in \mathbb{R} : \Psi(\vec{r}, \rho) \geq \alpha \right\}.$$

The α-CVaR is the conditional expectation of gains lower than or equal to the amount $\mathrm{VaR}_\alpha(\vec{r})$, i.e.

$$\phi_\alpha(\vec{r}) = \frac{1}{1-\alpha} \int_{g(\vec{r},a) \leq \rho_\alpha(\vec{r})} g(\vec{r}, a) f(\vec{r}) \mathrm{d}\vec{r}.$$

The main advantage of adopting CVaR in our study as the risk measure is its coherency, i.e. satisfies properties of monotonicity, sub-additivity, homogeneity and translational invariance (see Artzner et al. (1999) and Pflug (2000)). The monotonicity property states that the investment with higher gains leads to less risk, i.e. if $g_1 \geq g_2$, then we have

$$\phi(g_1) \geq \phi(g_2).$$

The sub-additivity property suggests that two investments together is at least as good as adding two risks separately, i.e.

$$\phi(g_1 + g_2) \geq \phi(g_1) + \phi(g_2).$$

The homogeneity property states that the risk measure is proportional to the size of the investment, i.e. if $c \in \mathbb{R}^+$, then

$$\phi(c g_1) = c \phi(g_1).$$

At last, the translation invariance implies that any cost involved with the actions reduces the risk measure by the same amount, i.e. if $c \in \mathbb{R}$, then

$$\phi(g_1 - c) = \phi(g_1) - c.$$

Another advantage of the CVaR measure is that for discrete scenarios, CVaR can be expressed as a linear program which can be easily incorporated in optimisation problems. According to Rockafellar et al. (2000), the risk measure $\phi_\alpha(\vec{r})$ can be expressed as:

$$\phi_\alpha(\vec{r}) = \sup_{\rho \in \mathbb{R}} \left\{ \rho - \frac{\mathbb{E}\{g(\vec{r}, a) - \rho\}^-}{1 - \alpha} \right\} \tag{3.7}$$

where $g(.)^- = -\min\{g(.), 0\}$. If we generate J simulations of the return vector, the equation (3.7) is equivalent to the following linear program:

$$\begin{aligned} \max_{\rho, a, z_1, \ldots, z_J} \quad & \rho - \frac{1}{(1-\alpha)J} \sum_{j=1}^{J} z_j \\ \text{s.t.} \quad & z_j \geq \rho - g(\vec{r}_j, a) \quad \text{for } j = 1, .., J \\ & z_j \geq 0 \quad \text{for } j = 1, ..., J. \end{aligned}$$

Although CVaR is a very useful risk measure in static optimisation problems, it is difficult to adopt CVaR in multi-period problems due to the lack of time consistency. The concept of time consistent risk measures is introduced by Boda & Filar (2006) and can be defined as follows. Let ϕ_t for $t \in \mathcal{T}$ be a risk measure for period t and let $\phi_{t_1 \to t_2}$ for $t_1, t_2 \in \mathcal{T}$ be the function measuring the risk from period t_1 to period t_2, the risk measure ϕ is called time consistent if it satisfies the following condition:

> The action a_t^* for each stage $t = 1, ..., T$ is optimal in each one-step risk management problem, i.e.
>
> $$a_t^* = \arg\max_{a_t} \phi_{T-t+1}(a_t | a_{t+1}^*, ..., a_T^*) \quad \text{for } t = 1, ..., T,$$
>
> if and only if the policy $\pi^* = (a_1^*, ..., a_T^*)$ is the optimal policy in the multi-stage problem, i.e.
>
> $$\pi^* = \arg\max_{\pi} \phi_{1 \to T}(\pi).$$

A number of time consistent risk measures are discussed in the literature. For example, Boda & Filar (2006) proposes a target-percentile risk measure model in which the agent considers not only the state of the original system but also his target. Cheridito & Kupper (2013) present a time consistent convex monetary risk measure in terms of one-step penalty functions. Rudloff et al. (2014) suggests a time consistent risk measure based on the traditional CVaR measure. In their model, the risk function is recursively defined as

$$\begin{cases} \phi_{t \to T}(\vec{r}_t) = \sup_{\rho \in \mathbb{R}} \left\{ \rho - \dfrac{\mathbb{E}\{\phi_{t+1 \to T}(\vec{r}_t) - \rho\}^-}{1-\alpha} \right\} & \text{for } t = 1, ..., T-1, \\ \quad \phi_T(\vec{r}) = \phi_\alpha^T(\vec{r}_T). & \end{cases}$$

where $\phi_\alpha^T(\vec{r}_T)$ is the expected shortfall value for period T.

In Chapter 7, we will adopt the time consistent risk measure proposed by Meng et al. (2011) which is the sum of CVaR of each period, i.e.

$$\phi_{t_1 \to t_2} = \sum_{t=t_1}^{t_2} \phi_\alpha^t(\vec{r}_t) \quad \forall t_1 < t_2, t_1, t_2 \in \mathcal{T}. \tag{3.8}$$

In their study, this risk measure is shown to be time consistent and the optimality equation is also derived.

Chapter 4

A Cash Management Model with An Infinite Asset

4.1 Introduction

Traditional cash management models in the literature (e.g. Eppen & Fama (1968)) are mainly focusing on the management strategies of the cash account. In these models, the company's asset is assumed to be infinite and can be used to replenish the cash balance at each decision epoch. The impact of cash policies on the company's asset level is normally neglected. In this chapter, we aim to solve this traditional cash management model via the Markov decision process approach. This chapter will be used as a benchmark for the following chapters where the impacts of cash policies on the company's asset will be closely examined. It can also be seen as the connection between the cash management models in the literature and the following cash management models in our thesis.

4.2 Problem description

In this model, we assume the agent manages a company with an infinite asset and he can replenish his cash holdings by selling his asset. This problem can be viewed as a trade-off between the cost of cash holding and the cost caused by insufficient cash balance to meet demands for cash. The cash holding cost can be interpreted as 'an opportunity cost' that represents the profit renounced by the agent when he chooses to hold the resource as cash instead of investing into profitable asset. On the other hand, an insufficient cash balance exposes the company to the risk of cash deficit which will jeopardise the regular business. Moreover, a low cash holding level normally requires more frequent cash replenishments and thus incurs higher transaction costs. The goal of cash management is to find a strategy that minimises the total cash management cost, i.e. the transaction cost, the cash holding cost and the cash shortage cost.

At the beginning of each period, the manager can control his cash holding level by selling or buying the asset. Let a_t be the action taken at time t and $A(x_t)$ be the set of all feasible actions at period t given the current cash level x_t. Note that a_t with a positive value represents selling asset to replenish cash balance while a_t with a negative value represents a buying action. Each action incurs the transfer fee which can be described by the transaction cost function $\Gamma(a_t)$. The partially fixed partially proportional transaction cost function (4.1) originally proposed by Milbourne (1983) is widely used in the literature (e.g. Baccarin (2002), Feng & Muthuraman (2010) and Salas-Molina, Pla-Santamaria & Rodriguez-Aguilar (2018)). We also adopt this transaction cost function in our cash management models.

$$\Gamma(a_t) = \begin{cases} K^- - k^- a_t & \text{if} \quad a_t < 0 \\ 0 & \text{if} \quad a_t = 0 \\ K^+ + k^+ a_t & \text{if} \quad a_t > 0 \end{cases} \tag{4.1}$$

where K^+ and K^- are the fixed transaction costs which should be paid once the agent takes a buying/selling action. Meanwhile the variable part of the transaction costs is proportional to the size of each transfer and k^+ and k^- are the proportional transaction cost coefficients.

Additionally, we introduce the concept of a post-decision cash level $x_t^{a_t}$, i.e. the company's cash holding level at period t immediately after the agent takes action a_t. The transition function can be expressed as:

$$x_t^{a_t} = x_t + a_t - \Gamma(a_t).$$

During this period, the cash level varies stochastically. We denote the uncontrolled cash level at time period t by υ_t and assume that the dynamics of the uncontrolled cash level can be approximated as a Wiener process with a constant drift μ and a fixed standard deviation σ, i.e.

$$\begin{cases} \mathrm{d}\upsilon_t = \mu \mathrm{d}t + \sigma \mathrm{d}W_t \\ \upsilon_0 = x_0 \end{cases} \tag{4.2}$$

where W_t is a standard one-dimensional Wiener process. Let Δx_t denote the change of the uncontrolled cash flow at time period t.

By the end of period t, if a cash shortage occurs (i.e. $x_t^{a_t} + \Delta x_t \leq 0$), the company's business will be jeopardised. We assume this damage to the company can be measured quantitatively and be described by a shortage cost function. On the other hand, if the cash account remains positive (i.e. $x_t^{a_t} + \Delta x_t > 0$), the relevant cost is the profit the agent could have gained if he had invested the cash into profitable asset. We use a proportional function (4.3) to describe the holding/shortage cost:

$$\Theta(x_t^{a_t}, \Delta x_t) = \begin{cases} h^-|x_t^{a_t} + \Delta x_t| & \text{if} \quad x_t^a + \Delta x_t \leq 0 \\ h^+(x_t^{a_t} + \Delta x_t) & \text{if} \quad x_t^a + \Delta x_t > 0 \end{cases}. \tag{4.3}$$

Note that the coefficient of cash holding cost h^+ is equivalent to the return rate on the asset.

The timing of each event in one period is shown in Figure 4.1. For a cash management model in multiple periods, the system transitions to the next state following:

$$x_{t+1} = x_t + a_t - \Gamma(a_t) + \Delta x_t$$

with the one-step cash management cost:

$$\Gamma(a_t) + \Theta(x_t^{a_t}, \Delta x_t).$$

To account for the time value of the total cash management cost, we introduce a discount factor γ. The discount factor indicates the agent's time preference (see Frederick et al. (2002)). Let r_f be the risk-free interest rate for each period, £1 at

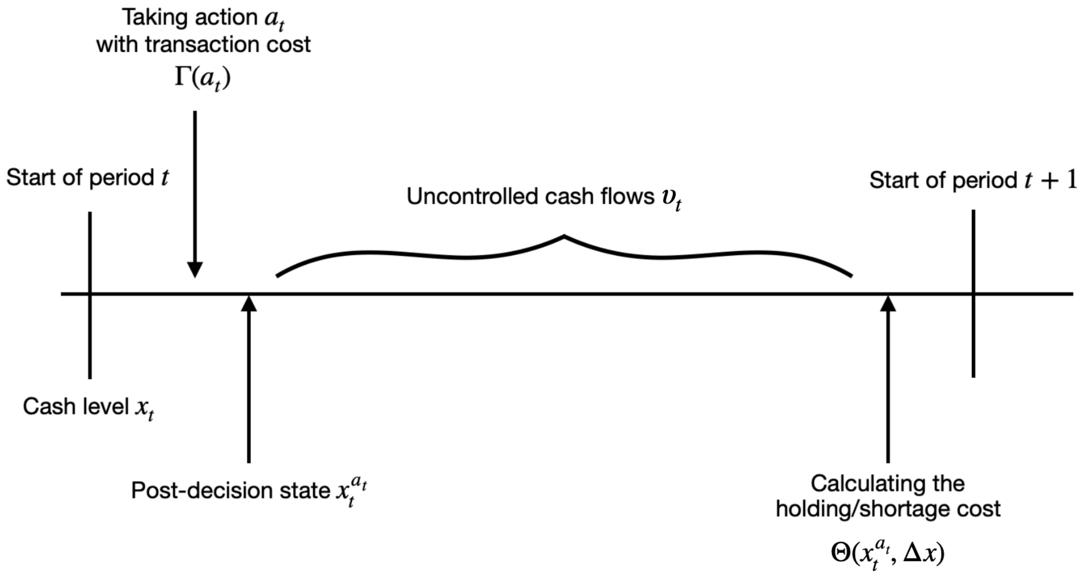


Figure 4.1: Timing of events in the model with an infinite asset

period t is equivalent to £$1/(1+r_f)^{t-1}$ at period 1, i.e.

$$\gamma = \frac{1}{1+r_f}. \tag{4.4}$$

In the cash management model with an infinite horizon, the goal is to find the best policy that minimises the expected discounted total cost:

$$\min_{\pi \in \Pi} \mathbb{E} \left\{ \sum_{t=1}^{\infty} \gamma^{t-1} \left(\Gamma(a_t) + \Theta(x_t^{a_t}, \Delta x_t) \right) \right\}. \tag{4.5}$$

4.3 Formalising the discrete Markov decision process

In this section we formalise the cash management model with an infinite asset as a discrete Markov decision process and solve the model via the classic backward recursion method.

To begin with, we discretise the cash level space and use the discretised cash level as the system's state s_t. We fix the minimum cash level $s_{[1]}$ and the maximum cash level $s_{[M]}$. Let $\mathbf{S}_t$ be the set of all possible states at time t, i.e. $\mathbf{S}_t = \{s_{[1]}, s_{[2]}, ..., s_{[M]}\}$. Note that the increment of any two successive states is fixed (i.e. $\Delta s = s_{[m+1]} - s_{[m]}, \forall m = \{0, 1, ..., M-1\}$). Also note that the state set does not vary with t, i.e. $\mathbf{S}_t = \mathbf{S}, \forall t$. For any cash level x_t, the index number m of its closest state $s_{[m]}$ can be obtained by:

$$m = \lfloor (x_t - s_1)/\Delta s \rceil \tag{4.6}$$

where $\lfloor x \rceil$ represents the closest integer to x.

Similarly we discretise the action space. Let $a_{[1]}$ be the 'minimum action', $a_{[M']}$ be the 'maximum action' and Δa be the increment of two successive actions (i.e. $\Delta a = a_{[m'+1]} - a_{[m']}, \forall m' = \{1, 2, ..., m'-1\}$). Note that in most cases, the 'minimum action' $a_{[1]}$ has a negative value, which represents the maximum cash that can be invested into asset. The 'maximum action' $a_{[M']}$ normally has a positive value which represents the maximum fund that can be used to replenish cash balance. Given $s_{[1]}$, $s_{[M]}$ and the current cash level x_t, $a_{[1]}$ and $a_{[M']}$ are bounded by the following constraints:

$$\begin{cases} a_{[1]} \geq s_{[1]} + \Gamma(a_{[1]}) - x_t \\ a_{[M']} \leq s_{[M]} + \Gamma(a_{[M']}) - x_t \end{cases} . \tag{4.7}$$

These constraints guarantee that the post-decision cash level is contained in the space $[s_{[1]}, s_{[M]}]$. We also need to make sure that the action of 'doing nothing' is contained in the action space, thus the action space can be described as:

$$\mathbf{A}_t(x_t) = \{a_{[1]}, ..., a_{[M']}\} \cup \{0\}.$$

Let the uncontrolled cash flow v_t be the exogenous information available at the end of period t. With the assumption of Wiener process, the change of the uncontrolled cash level (Δx) can be viewed as a random number following a normal distribution. Note that the probability density function of Δx_t is:

$$f\left(\Delta x_t\right) = \frac{1}{\sqrt{2\sigma^2\pi}} \exp\left\{-\frac{(\Delta x - \mu)^2}{2\sigma^2}\right\}$$

where μ and σ are parameters.

As a result of taking action $a_t \in \mathbf{A}_t(x_t)$, the transaction cost $\Gamma(a_t)$ immediately occurs. Moreover, by the end of period t, the holding/shortage cost can be calculated given $x_t^{a_t}$ and Δx_t. The reward of (s_t, a_t) for period t can be viewed as the expected total cost, i.e. the sum of the transaction costs and the expected holding/shortage cost:

$$R(x_t, a_t) = \Gamma(a_t) + \int_{-\infty}^{\infty} f(\Delta x)\Theta(x_t^{a_t}, \Delta x)\mathrm{d}\Delta x. \tag{4.8}$$

We define the value of a state-action pair $V_t(s_t, a_t)$ as the discounted expected total cash management cost over the infinite horizon once the system visits state s_t and the agent takes action a_t. It can be expressed as:

$$\begin{aligned} V_t(s_t, a_t) &= R(s_t, a_t) + \gamma \mathbb{E}\{\hat{V}_{t+1}(x_{t+1}|s_t, a_t)\} \\ &= R(s_t, a_t) + \gamma \int_{-\infty}^{\infty} f(\Delta x)\hat{V}_{t+1}(x_{t+1}|s_t, a_t, \Delta x)\mathrm{d}\Delta x \end{aligned}. \tag{4.9}$$

Combining equation (4.8) and (4.9), we have:

$$V_t(s_t, a_t) = \Gamma(a_t) + \int_{-\infty}^{\infty} \left(\Theta(x_t^{a_t}, \Delta x) + \gamma \hat{V}_{t+1}(x_{t+1}|s_t, a_t, \Delta x)\right) f(\Delta x)\mathrm{d}\Delta x. \tag{4.10}$$

Note that in the backward update process, we first set state values at the end of time horizon to zeros, i.e.

$$\hat{V}_{(0)}(s) = 0, \ \forall s \in \mathbf{S}.$$

Then for each iteration we update the state values in the previous period based on the current state values. Hence for iteration $i = 1, 2, ..., I$, we have

$$V_{(i)}(s_{(i)}, a_{(i)}) = \Gamma(a_{(i)}) + \int_{-\infty}^{\infty} \left(\Theta(x_{(i)}^{a_{(i)}}, \Delta x) + \gamma \hat{V}_{(i-1)}(x_{(i-1)}|s_{(i)}, a_{(i)}, \Delta x)\right) f(\Delta x)\mathrm{d}\Delta x. \tag{4.11}$$

Also note that in practice we use the linear interpolation method to estimate the cash level value $\hat{V}_{(i-1)}(x_{(i-1)})$ based on the values of the adjacent states, i.e.

$$\begin{cases} m^* = \dfrac{x_{(i-1)} - s_0}{\Delta s} \\ m_f = \left\lfloor \dfrac{x_{(i-1)} - s_0}{\Delta s} \right\rfloor \\ \hat{V}_{(i-1)}(x_{(i-1)}) = (m^* - m_f)\left(V_{(i-1)}(s_{[m_f+1]}) - V_{(i-1)}(s_{[m_f]})\right) + V_{(i-1)}(s_{[m_f]}) \end{cases}. \tag{4.12}$$

We define the value of state $s_{(i)}$ as the minimum discounted expected total cost after the system visits state $s_{(i)}$, i.e.

$$V_{(i)}(s_{(i)}) = \min_{a_{(i)} \in A(x_{(i)})} \left\{ R(s_{(i)}, a_{(i)}) + \gamma \mathbb{E}[\hat{V}_{(i-1)}(x_{(i-1)}|s_{(i)}, a_{(i)})] \right\}.$$

Algorithm (1) describes the details of the backward recursion method in MDP. Note that in Step 2.2 and Step 4.2, we need to solve equation (4.11) by discretising the continuous random variable Δx. Now the state-action pair value can be expressed as:

$$V_{(i)}(s_{(i)}, a_{(i)}) = \Gamma(a_{(i)}) + \sum_{\Delta x} p(\Delta x|s_{(i)}, a_{(i)}) \left\{ \Theta(x_{(i)}^{a_{(i)}}, \Delta x) + \gamma \hat{V}_{(i-1)}(x_{(i-1)}|s_{(i)}, a_{(i)}, \Delta x) \right\}$$

where $p(\Delta x|s_{(i)}, a_{(i)})$ is the probability that the uncontrolled cash flows change by Δx given the state $s_{(i)}$ and the action $a_{(i)}$.

In practice, we provide two discretisation methods: the simulation method and the Gauss-Hermite quadrature method (see Steen et al. (1969)). In the simulation method, we generate J realisations of a random variable following a normal distribution $\mathcal{N}(\mu, \sigma)$. Each realisation is viewed as a possible outcome of Δx with the same probability $p = 1/J$. In the Gauss-Hermite quadrature method, we use the Gaussian quadrature to approximate equation (4.11). Let $\Delta x = \sqrt{2}\sigma x_k + \mu$ and $z(\Delta x) = \Theta(x_{(i)}^{a_{(i)}}, \Delta x) + \gamma \hat{V}_{(i-1)}(x_{(i-1)}|s_{(i)}, a_{(i)}, \Delta x)$. Equation (4.11) can be approximated by:

$$\begin{aligned} V(x_{(i)}, a_{(i)}) &= \Gamma(a_{(i)}) + \int_{-\infty}^{\infty} z(\Delta x) f(\Delta x) \mathrm{d}\Delta x \\ &\approx \Gamma(a_{(i)}) + \frac{1}{\sqrt{\pi}} \sum_{j=1}^{J} w_j z\left(\sqrt{2}\sigma x_j + \mu\right) \end{aligned} \tag{4.13}$$

where J is the number of sample points, x_j are the roots of the Hermite polynomial $H_J(x)$, and the corresponding weights w_j are given by:

$$w_j = \frac{2^{J-1} J! \sqrt{\pi}}{J^2 \left[H_{J-1}(x_j)\right]^2}. \tag{4.14}$$

In experiments, we apply these discretization methods and use the same results in all states.

Algorithm 1 The classic backward recursion method in the CM model with an infinite asset

Step 1: Initialisation:

Step 1.1: Set the value of each terminal state to zero, i.e. $V_{(0)}(s_{(0)}) = 0$ for $s_{(0)} \in \mathbf{S}$.

Step 1.2: Define a small number ϵ. Set the iteration index $i := 1$.

Step 2: Update backwards:

Step 2.1: For each state $s_{(i)} \in S$, find the feasible action set $\mathbf{A}_t(s_{(i)})$.

Step 2.2: For each state-action pair $(s_{(i)}, a_{(i)} | s_{(i)} \in \mathbf{S}, a_{(i)} \in \mathbf{A}_t(s_{(i)}))$, get the state-action value $V_{(i)}(s_{(i)}, a_{(i)})$ via equation (4.11).

Step 2.3: For each state $s_{(i)} \in \mathbf{S}$, update the state value via:

$$V_{(i)}(s_{(i)}) = \min_{a_{(i)} \in A(s_{(i)})} V_{(i)}(s_{(i)}, a_{(i)}).$$

Step 3: Value comparison:

Step 3.1: For each state $s_{(i)} \in \mathbf{S}$, record the value difference:

$$\mathrm{diff}(s_{(i)}) = |V_{(i)}(s_{(i)}) - V_{(i-1)}(s_{(i)})|.$$

Step 3.2: If $\max_{s_{(i)} \in S} \mathrm{diff}(s_{(i)}) \geq \epsilon$, set $i := i + 1$ and go to **Step 2**, else go to **Step 4**.

Step 4: Policy output:

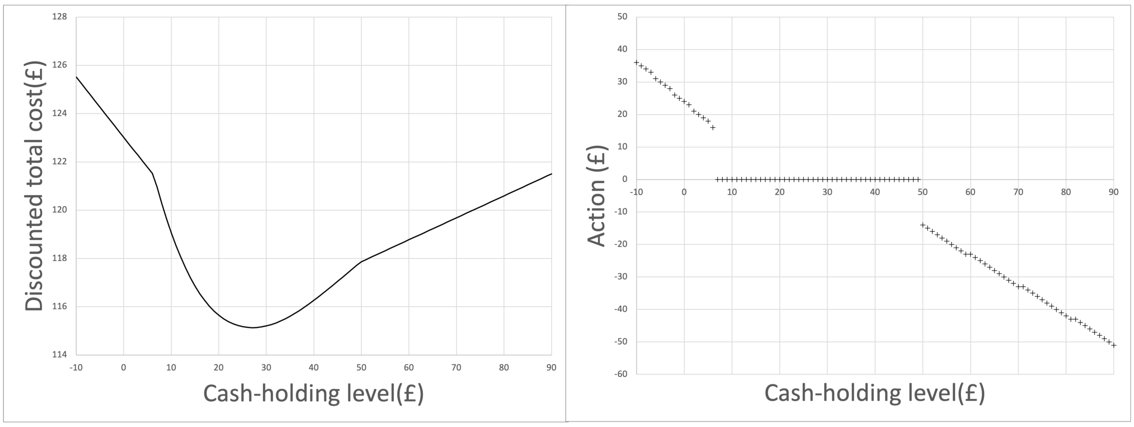
Step 4.1: For each state $s_{(i)} \in S$, find the feasible action set $\mathbf{A}_t(s_{(i)})$.

Step 4.2: For each state-action pair $(s_{(i)}, a_{(i)} | s_{(i)} \in \mathbf{S}, a_{(i)} \in \mathbf{A}_t(s_{(i)}))$, get the state-action value $V_{(i)}(s_{(i)}, a_{(i)})$ via equation (4.11).

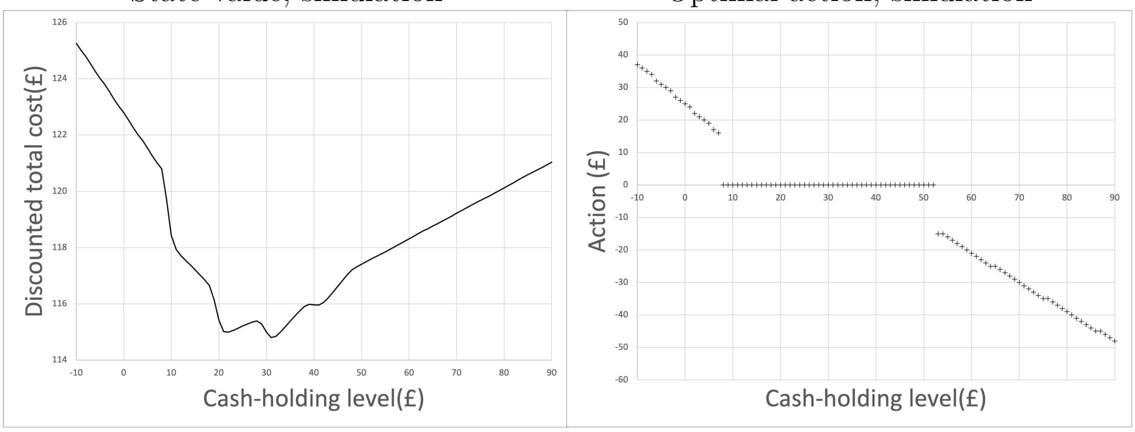
Step 4.3: For each state $s_{(i)} \in S$, output the optimal action via:

$$a^*(s_{(i)}) = \arg\min_{a_{(i)} \in A(s_{(i)})} V_{(i)}(s_{(i)}, a_{(i)}).$$

Figure 4.2 shows the state values and the corresponding policies with these two discretisation methods. In this example, we assume that Δx is following the distribution $\mathcal{N}(\mu = 0, \sigma = 10)$. Moreover, we set parameters $K^+ = 2, K^- = 1, k^+ = 0.2, k^- = 0.1, h^+ = 0.05, h^- = 2$ and $r_f = 0.02$. In the simulation method, we generate 300 samples, each of which is a realisation of Δx with probability $p = 1/300$. In the Gauss-Hermite quadrature method, we use 9 sample points. It is shown that the simulation method returns a smooth state value function while the Gauss-Hermite quadrature indicates a much more fluctuating curve. In terms of the policies, on the other hand, both methods give very similar result. Since the Gauss-Hermite quadrature requires much less sample observations than the simulation method, its solution time is dramatically lower especially when the discretisation level of state space gets finer. Hence we will use the Gauss-Hermite quadrature method with 9 sample points in the following numerical experiments unless specified otherwise.



State value, simulation　　Optimal action, simulation



State value, quadrature　　Optimal action, quadrature

Figure 4.2: The simulation method versus the Gauss-Hermite quadrature method

4.4 Numerical experiments

This cash management model closely resembles Eppen & Fama's work (Eppen & Fama (1968)). In this section, we mainly discuss the impacts of the transaction costs, the shortage/holding cost and the distribution of uncontrolled cash flows on the cash management policies. Some results have been discussed in the literature e.g. Baccarin (2002), Benkherouf & Bensoussan (2009), and Feinberg & Lewis (2005). We present these results as a baseline for models in next section. All the experiments are programmed in C++ 12.0.0 on a PC with 2.5 GHz Quad-Core Intel Core i7 and 8 GB memory.

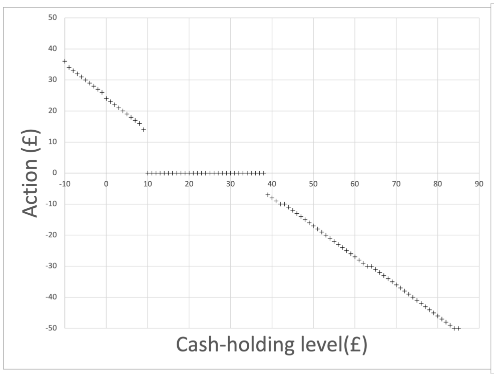
4.4.1 Transaction cost, holding cost and shortage cost

To illustrate the impact of transaction cost on cash management policies, we consider three transaction cost setting: low, medium, and high. The specific parameters are shown in Table 4.1. Figure 4.3 shows the optimal policies under different transaction costs and a simulated path over 100 periods following the corresponding strategy. Note that for each period, we plot both the pre-decision and post-decision states to show the patterns of the optimal policies more clearly. It can be seen that with a low cash balance, the model suggests selling actions (i.e. replenishing the cash account) and with a high cash holding level, it suggests buying actions (i.e. reducing cash holdings). Moreover all the policy figures give a 'do nothing area' where the agent should not take any action once the system visits these states. The simulation figures illustrate that the optimal policy is subject to a 'two-trigger two-target' form (also known as the (L, l, u, U) policy).

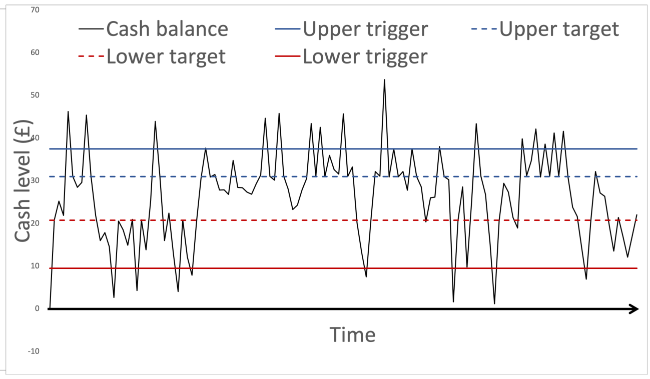
Table 4.1: Parameters for the transaction cost function

	Low	Medium	High
Fixed into cash account K^+ (£)	1	2	3
Fixed from cash account K^- (£)	0.5	1	1.5
Proportional into cash account k^+	0.1	0.2	0.3
Proportional from cash account k^-	0.05	0.1	0.15

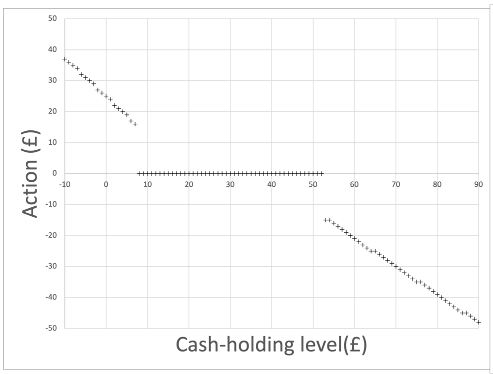
The (L, l, u, U) policy indicates that once the cash balance reaches the trigger level, it is optimal to adjust them back to the corresponding target level. For



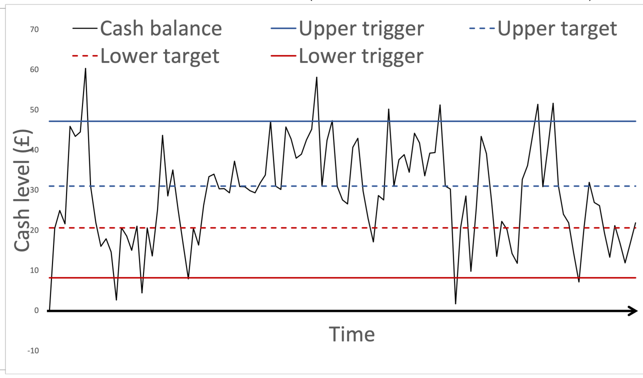
Policy (low transaction cost)



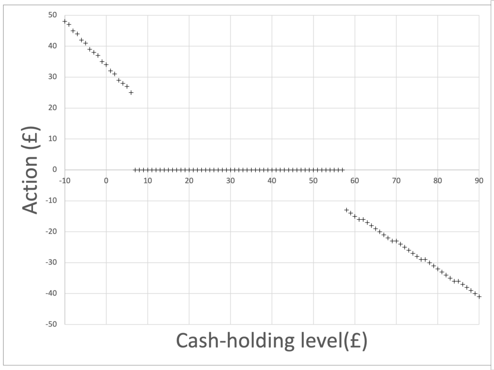
A simulated path (low transaction cost)



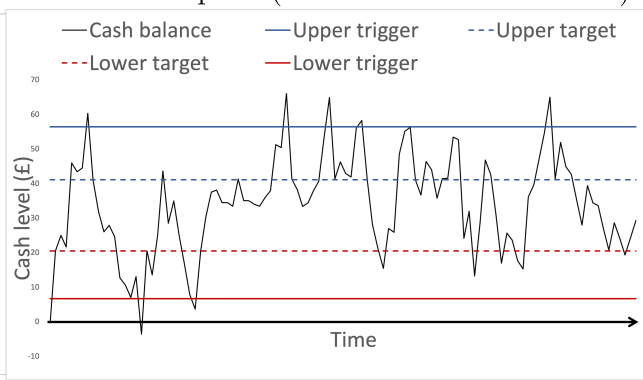
Policy (medium transaction cost)



A simulated path (medium transaction cost)



Policy (high transaction cost)



A simulated path (high transaction cost)

Figure 4.3: The impact of transaction cost on cash management policies

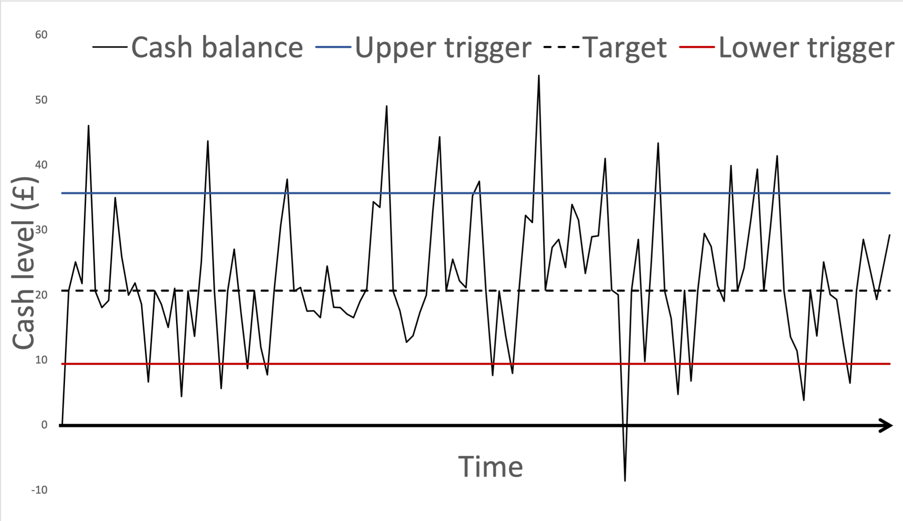
example, under the medium transaction cost, the agent should take the action of 'doing nothing' if his cash balance is between £9 and £47. Once the cash holding level changes to £10 or any lower position, the agent should sell his asset and replenish his cash balance back to £21. Likewise, if his cash level exceeds £47, the agent should keep £31 as cash and invest the rest into asset.

We also want to point out that when the transaction cost are purely fixed, i.e. $k^+ = k^- = 0$, the target levels will merge together ($l = u$). When the transaction cost are purely proportional, i.e. $K^+ = K^- = 0$, the upper/lower target level overlaps the upper/lower trigger level ($L = l$ and $U = u$). Figure 4.4 shows a simulated sample under these transaction cost functions. It is clear that with the fixed transaction cost, the optimal policy is of the (L, B, U) form. In other words, if the cash flow goes above the upper trigger level U or below the lower trigger level L, it should be adjust back to the target level B. With the purely proportional transaction cost, we have a (l, u) policy which suggests the cash account should be adjusted back to the l (or u) level once it is higher (or less) than that level.

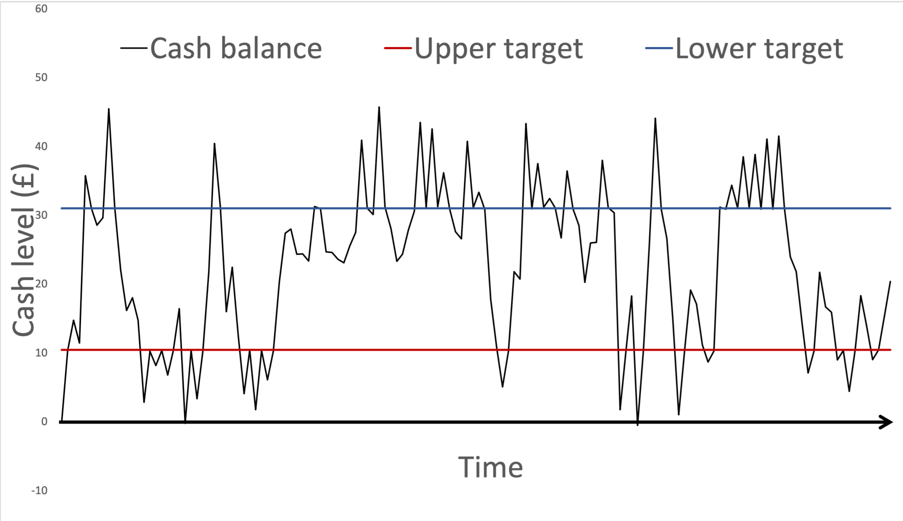
Now we examine the impact of shortage/holding cost on the cash management policies. We use the medium fixed plus proportional transaction cost (as shown in table 4.1) and set the holding cost coefficient as $h^+ = (0.01, 0.05, 0.1)$ and the shortage cost coefficient as $h^- = (0.5, 2, 10)$. Table 4.2 gives the (L, l, u, U) policies for all shortage cost and holding cost combinations. It is clear that with higher holding cost, all target and trigger levels are decreasing, which suggests that there is less motivation to hold cash when the return on investment is high. Likewise the cash balance is of more value with the increase of shortage penalty coefficients.

Table 4.2: Policies under different holding/shortage cost in the one-account model

	$h^+ = 0.01$				$h^+ = 0.05$				$h^+ = 0.1$			
	L	l	u	U	L	l	u	U	L	l	u	U
$h^- = 0.5$	(1	20	62	98)	(-3	10	31	43)	(-5	10	21	27)
$h^- = 2.0$	(10	22	73	105)	(8	21	31	47)	(7	10	21	34)
$h^- = 10$	(19	32	73	108)	(18	21	42	55)	(17	21	31	42)



Fixed transaction cost



Proportional transaction cost

Figure 4.4: Cash management policies with other types of transaction cost

4.4.2 Cash flow

So far we have only used cash level as the state in MDP and the drift of cash flow (μ) is assumed to be zero. Now we consider the cash management model with other drifts of cash flow and add the drift into the state information. In other words, at each decision epoch, the agent has access to the information of both his cash balance and the drift of the uncontrolled cash flow before he takes actions. Currently the cash flow drift (μ) is still viewed as exogenous information (i.e. the agent's action has no impact on μ) and is assumed to be constant over periods. Note that this model can be seen as a connection between the classic cash management model and the models in the following sections, in which the cash flow drift will be considered endogenous to the system.

Figure 4.5 shows the optimal policy obtained from the MDP model under the medium transaction cost (the rest parameters are set to $h^+ = 0.05, h^- = 2, \sigma = 10$ and $i_f = 0.02$). Each vertical line in the figure represents a (L, l, u, U) solution for a separate MDP problem and Figure 4.5 can be seen as a two-dimensional version of the two-trigger two-target cash management policy for companies with different cash drifts in the cash flow. If the system is in the grey area, the agent should adjust his cash balance and move the system to target spots which are marked by red crosses. It is clear that trigger levels and target levels are much lower for those companies with a higher cash flow drifts. In this experiment, once the company has cash flow with drifts higher than £13, the growth of cash flow itself is sufficient enough and the agent has no motivation to replenish his cash balance. Moreover, Figure 4.6 illustrates the impact of the standard deviation σ of the uncontrolled cash flow on the optimal policies. It shows that the trigger levels and the target levels are much higher in the case with larger σ. This suggests that with more volatility in the cash flow, the agent should hold a larger cash balance. Note that this sensitivity of cash holding strategy to the cash flow's volatility is observed empirically by Almeida et al. (2004) and Yan (2006). Our work can be seen as a theoretical support for their studies.

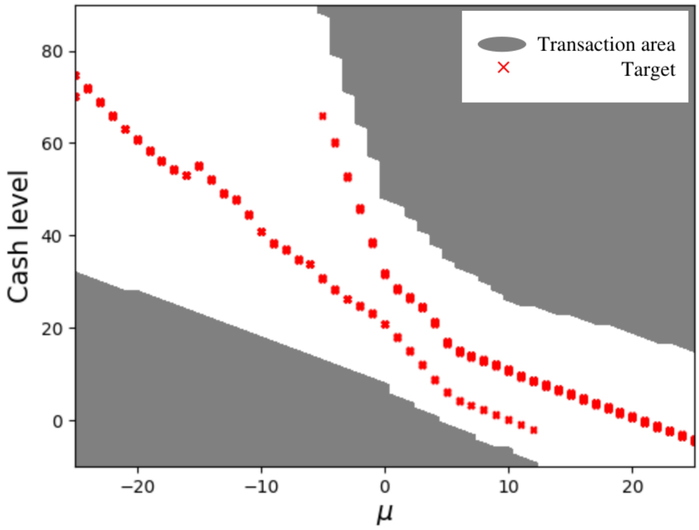
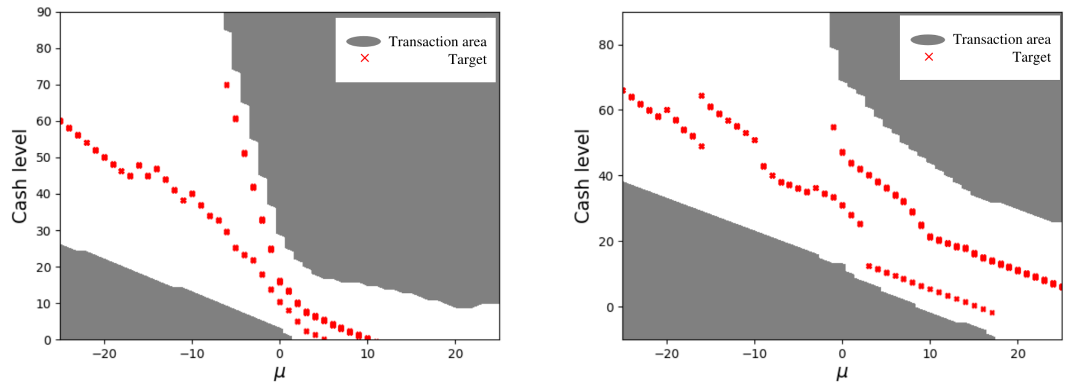


Figure 4.5: Cash management policy in MDP with (x, μ)



$\sigma = 5$ $\sigma = 15$

Figure 4.6: Policies with different standard deviation of cash flow

4.5 Conclusion

In this chapter, we studied the traditional cash management model where the company's asset is assumed to be infinite. This model was formalised as a discrete Markov decision process and was solved via the classic backward recursion method. A series of numerical experiments were also conducted in order to examine the impact of transaction costs, shortage costs and the cash flows on the cash policies. We first showed that with a fixed plus proportional transaction cost function the cash policy is of the two-trigger two-target form. When the transaction cost function is purely proportional to the transfer size, the target levels overlap the trigger levels and the cash policy changes to the (l, u) form. On the other hand, if the transaction cost function is purely fixed, the upper target level overlaps the lower target level which leads to the (L, B, U) cash policies. In addition we numerically showed that the manager should hold a larger cash balance with a higher shortage cost and hold a smaller cash balance with higher holding cost. At last, through the experiments on different standard deviations of the external cash outflows we showed that it is optimal for the manager to hold more cash when the external cash demand has a higher volatility.

Chapter 5

A Cash Management Model with Two Accounts

5.1 Introduction

Traditional cash management models concern how an organisation should maintain its cash balance in order to meet cash demands over a planning horizon. In these models the cash balance can be increased or decreased to offset penalties for not being able to meet a cash demand or the opportunity cost of holding too much cash. The external source from which this money comes from or is sent to is not explicitly modelled but is assumed to be available at all times. In this chapter we aim to contribute to this problem by explicitly modelling this external source by the inclusion of an asset account. This asset will generate an income which we allow to be either deposited directly to the cash account or contributes to the asset account's volume. Then we will model this two dimensional cash management problem, in which credits and debits from/to the cash balance are to/from the asset account and incur transaction costs for these movements, as a Markov decision process and solve the problem via the classic backward recursion method. To our best knowledge, this is the first study in the literature explicitly modelling the external source of cash inflows. We will also show that with a small modification the backward recursion method can be used to calculate the company's insolvency probabilities. Finally the impact of the parameter settings on the optimal policy will be studied in a series of numerical experiments.

5.2 Problem description and assumptions

In this model, we include an asset account with finite volume in the cash management model. Moreover we consider the size of the asset account at each time as an extra dimension of the state space. This model can be described as follows: Consider an agent who manages a cash account (x) and an asset account (y). At each period, the asset generates profit which is viewed as a source of cash inflow. A transfer (a) between these accounts can be made after the company receives this internal cash inflow. Then the company faces an external stochastic cash flow. We assume the external cash outflow normally dominates the external cash inflow hence it can be viewed as a negative value of the external cash demand. Then after the process of the uncontrolled external cash flow, a shortage penalty occurs if there exists cash deficit. Figure 5.1 shows the timing of each event for one time. The objective of this model is to maximise the expected discounted net profit over an infinite horizon.

For the sake of simplicity, a few assumptions will be made before we formalise this model into a Markov decision process (MDP). To begin with, we assume that the return on asset is deterministic and proportional to the size of the asset account. One could argue that this assumption is rather unrealistic considering returns on investment are normally very difficult to predict. But the randomness of this return can be easily integrated with the stochasticity from the external cash flow (υ_t). Similar to the model in Chapter 4, we use Wiener process with parameters (μ, σ) to approximate the uncontrolled external cash flow. Now the external cash flow over the whole period t can be expressed as:

$$\begin{cases} \mathrm{d}\upsilon_t = \mu \mathrm{d}t + \sigma \mathrm{d}W_t \\ \upsilon_0 = x_0 \end{cases} \tag{5.1}$$

where W_t is a standard one-dimensional Wiener process and x_0 is the initial cash balance.

In addition, we adopt the fixed plus proportional function (4.1) to describe the transaction cost. Note that in the real world, the transfer fees are normally paid in the form of cash, hence the agent must sell an extra amount of asset to pay

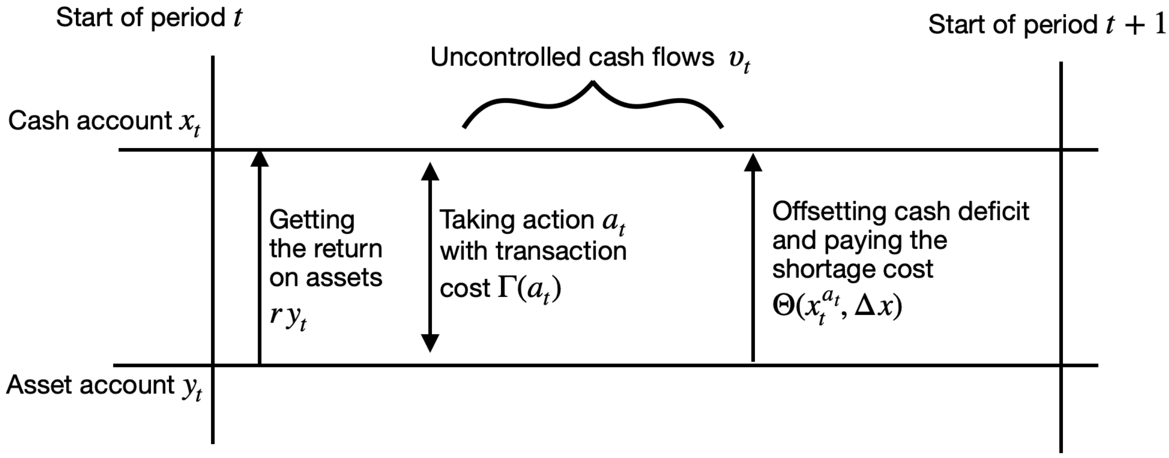


Figure 5.1: Timing of events in the model with two accounts

the transfer fee when he takes a selling action. In our model, we consider this extra amount of asset as the selling fee directly. In other words, it is assumed the selling fee is in the form of asset instead of cash. This can also be interpreted as selling asset at a lower price due to the lack of liquidity. Let $x_t^{a_t}$ and $y_t^{a_t}$ denote the post-decision cash level and the post-decision asset level. The transfer function can be written as:

$$\begin{bmatrix} x_t^{a_t} \\ y_t^{a_t} \end{bmatrix} = \begin{bmatrix} x_t + ry_t + a_t - \Gamma(a_t) \cdot 1_{a_t \leq 0} \\ y_t - a_t - \Gamma(a_t) \cdot 1_{a_t > 0} \end{bmatrix} \tag{5.2}$$

where r is the expected return rate on asset and $1_{a_t \leq 0}$ is an indicator function:

$$1_{a_t \leq 0} = \begin{cases} 1 & \text{if } a_t \leq 0 \\ 0 & \text{otherwise} \end{cases}.$$

Moreover we adopt the proportional shortage cost function (5.3) into this cash management model, but we discard the holding cost as it represents the profit the agent could have gained by investing the cash into asset. This 'opportunity cost' has already been taken into consideration by maximising the return on asset.

$$\Theta(\Delta x_t, x_t^{a_t}) = \begin{cases} h|x_t^{a_t} + \Delta x_t| & \text{if } x_t^{a_t} + \Delta x_t < 0 \\ 0 & \text{otherwise} \end{cases}. \tag{5.3}$$

We also assume the non-negativity of both cash account and asset account.

Without a cash shortage at period t, the system transitions to next time period following:

$$\begin{bmatrix} x_{t+1} \\ y_{t+1} \end{bmatrix} = \begin{bmatrix} x_t^{a_t} + \Delta x_t \\ y_t^{a_t} \end{bmatrix}. \tag{5.4}$$

In the case of a cash shortage, i.e. $x_t^{a_t} + \Delta x_t < 0$,the agent will be forced to sell his asset and offset such deficit with a shortage penalty at the end of each period. Let $d_t = |x_t^{a_t} + \Delta x_t|$ be the cash deficit at the end of period t, the system transitions following:

$$\begin{bmatrix} x_{t+1} \\ y_{t+1} \end{bmatrix} = \begin{bmatrix} 0 \\ y_t^{a_t} - d_t - \Gamma(d_t) - \Theta(x_t^{a_t}, \Delta x) \end{bmatrix}. \tag{5.5}$$

Finally we discount the value in future via the risk-free interest rate r_f (see equation (4.4)) and assume that the goal is to find the optimal policy that maximises the expected discounted net profit over the infinite horizon. For each period, the net profit can be interpreted as the incomes generated from the asset ry_t minus the transaction cost $\Gamma(a_t)$ and the shortage penalty $\Theta(x_t^{a_t}, \Delta x)$. Hence we have:

$$\max_{\pi \in \Pi} \mathbb{E} \left\{ \sum_{t=1}^{\infty} \gamma^{t-1} (ry_t - \Gamma(a_t) - \Theta(x_t^{a_t}, \Delta x_t)) \right\}. \tag{5.6}$$

5.3 A discretised Markov decision process approach

We study the cash management model with two accounts where the cash account is used to fulfil the demand for cash while the asset account generates profit which is pursued by the manager. In this model, we consider the cash flow as a superposition of an endogenous process and an exogenous process. The endogenous cash inflow is explained by the return on assets and the exogenous cash flow is assumed following Wiener process. In this section, we formulate this model as a discretised Markov decision process and present some preliminary results.

5.3.1 The discretised MDP model and the classic backward recursion method

First of all, we discretise the cash level space into the set $\mathbf{S}^x = \left\{s^x_{[1]}, ..., s^x_{[M]}\right\}$ and the asset level space into the set $\mathbf{S}^y = \left\{s^y_{[1]}, ..., s^y_{[M']}\right\}$. The system's state can be described by a two-element tuple, $\vec{s}_t = (s^x_t, s^y_t)$, s.t $s^x_t \in \mathbf{S}^x, s^y_t \in \mathbf{S}^y$. For the cash account, we fix the minimum value $s^x_{[1]}$, the maximum value $s^x_{[M]}$ and the cash state increment $\Delta s^x = s^x_{[m+1]} - s^x_{[m]}$, $\forall m = 1, 2, ..., M-1$. Similarly for the asset account, we choose $s^y_{[1]}, s^y_{[M']}$ and Δs^y as the minimum asset value, the maximum asset value and the asset state increment respectively. For any cash-asset pair (x_t, y_t), the indexes m, m' of its closest states $\left(s^x_{[m]}, s^y_{[m']}\right)$ can be obtained via:

$$m = \left\lfloor \frac{x_t - s^x_{[1]}}{\Delta s^x} \right\rceil, \tag{5.7}$$

$$m' = \left\lfloor \frac{y_t - s^y_{[1]}}{\Delta s^y} \right\rceil. \tag{5.8}$$

Note that for simplicity's sake we set the minimum value for the cash balance and the asset level (i.e. $s^x_{[1]}$ and $s^y_{[1]}$) to zeros in the following experiments unless specified otherwise.

Given the current cash level x_t and the asset level y_t, we have the discretised action set $a_t \in \mathbf{A}_t(x_t, y_t) = \left\{0, a_{[1]}, ..., a_{[M'']}\right\}$. Note that the action $a_t = 0$ implies doing nothing at period t which is always a feasible action regardless of x_t and y_t. We use the value of a_t to represent the size of the transfer between cash and asset and use the sign of a_t to represent the direction of this transfer. An action with a negative value stands for the cash invested into asset account while a positive a_t means selling asset and replenishing cash balance. Now we describe the feasible action set $\mathbf{A}_t(x_t, y_t)$ by adding some constraints on $a_{[1]}$ and $a_{[M'']}$:

$$\begin{cases} a_{[1]} \geq s^x_{[1]} + \Gamma(a_{[1]}) - x_t - r y_t & \text{Constraint I} \\ a_{[1]} \geq y_t - s^y_{[M']} & \text{Constraint II} \\ a_{[M'']} \leq y_t - \Gamma(a_{[M'']}) - s^y_{[1]} & \text{Constraint III} \\ a_{[M'']} \leq s^x_{[M]} - x_t - r y_t & \text{Constraint IV} \end{cases}.$$

Constraint I states that after the buying action with the corresponding transaction cost, the agent should at least hold the minimum cash level $s^x_{[1]}$ in his cash balance. Constraint II states that the current asset and the newly bought asset together should not exceed the asset account capacity $s^y_{[M']}$. Similarly Constraint III and Constraint IV ensure that after the selling action, the company still possesses the asset above the minimum asset value $s^y_{[1]}$ and the cash holdings not exceeding its cash account capacity $s^x_{[M]}$. The feasible action set can be obtained following Algorithm (2).

Algorithm 2 Find the feasible action set $\mathbf{A}_t(x_t, y_t)$

Step 1: Initialisation:

Set the action increment Δa;
Set the feasible action set $\mathbf{A}_t(x_t, y_t) = \{0\}$;
Set the minimum action:

$$a_{[m]} = \max\left\{\frac{s^x_{[1]} - x_t - ry_t + K^-}{1 + k^-}, y_t - s^y_{[M']}\right\} \tag{5.9}$$

Step 2: Action iteration:

If action $a_{[m]}$ doesn't violate **Constraint III** or **Constraint IV**, add action $a_{[m]}$ into the feasible action set $\mathbf{A}_t(x_t, y_t)$, set $a_{[m]} += \Delta a$, go to **Step 2**;
else **return** the feasible action set $\mathbf{A}_t(x_t, y_t)$.

Let the immediate reward of visiting state (x_t, y_t) and taking action a_t be the return on asset minus transaction cost and shortage cost, i.e.

$$R(x_t, y_t, a_t) = ry_t - \Gamma(a_t) - \int_{-\infty}^{\infty} f(\Delta x_t)\Theta\left(x_t^{a_t}, \Delta x_t\right) \mathrm{d}\Delta x_t$$

where $f(\Delta x_t)$ is the probability density function of the change of the exogenous cash flow over period t. Now we can write the Bellman equation for this cash management model:

$$\begin{aligned} V_t\left(\vec{s}_t, a_t\right) &= ry_t - \Gamma\left(a_t\right) + \textstyle\int_{-\infty}^{\infty} \left(-\Theta\left(x_t^{a_t}, \Delta x_t\right)\right. \\ &\quad \left.+\gamma \hat{V}_{t+1}\left(x_{t+1}, y_{t+1} \mid \vec{s}_t, a_t, \Delta x_t\right)\right) f(\Delta x_t)\mathrm{d}\Delta x_t. \end{aligned} \tag{5.10}$$

Similar to the equation (4.11), the Bellman equation (5.10) can be solved via the classic backward recursion method, i.e.

$$V_{(0)}(\vec{s}) = 0,\ \forall \vec{s} \in \mathbf{S}$$

and

$$\begin{aligned} V_{(i)}\left(\vec{s}_{(i)}, a_{(i)}\right) &= ry_{(i)} - \Gamma\left(a_{(i)}\right) + \textstyle\int_{-\infty}^{\infty} \left(-\Theta\left(x_{(i)}^{a_{(i)}}, \Delta x_{(i)}\right)\right. \\ &\quad \left. + \gamma \hat{V}_{(i-1)}\left(x_{(i-1)}, y_{(i-1)} \mid \vec{s}_{(i)}, a_{(i)}, \Delta x_{(i)}\right)\right) f(\Delta x_{(i)}) \mathrm{d}\Delta x_{(i)}. \end{aligned} \tag{5.11}$$

In practice, we use the bilinear interpolation method to estimate the value $V_{(i-1)}(x_{(i-1)}, y_{(i-1)})$ based on the values of its adjacent states.

Given a pair of cash-asset level (x, y), we firstly obtain the adjacent states: $\left(s_{[m]}^x, s_{[m']}^y\right)$, $\left(s_{[m+1]}^x, s_{[m']}^y\right)$, $\left(s_{[m]}^x, s_{[m'+1]}^y\right)$ and $\left(s_{[m+1]}^x, s_{[m'+1]}^y\right)$ via the equation (5.7) and (5.8). Then using the bilinear interpolation, we have:

$$\begin{cases} \theta_x = \dfrac{x - s_{[1]}^x}{\Delta s^x} - m \\ \theta_y = \dfrac{y - s_{[1]}^y}{\Delta s^y} - m' \\ \hat{V}\left(x, s_{[m']}^y\right) = \theta_x V\left(s_{[m+1]}^x, s_{[m']}^y\right) + (1-\theta_x) V\left(s_{[m]}^x, s_{[m']}^y\right) \\ \hat{V}\left(x, s_{[m'+1]}^y\right) = \theta_x V\left(s_{[m+1]}^x, s_{[m'+1]}^y\right) + (1-\theta_x) V\left(s_{[m]}^x, s_{[m'+1]}^y\right) \\ \hat{V}(x, y) = \theta_y \hat{V}\left(x, s_{[m'+1]}^y\right) + (1-\theta_y) \hat{V}\left(x, s_{[m']}^y\right) \end{cases} . \tag{5.12}$$

Finally we use the simulation method and the Gauss-Hermite quadrature method (as discussed in Section 4.3) to discretise the continuous random variable Δx_t. The performance of these two methods will be compared in Section 5.3.2. Now the discounted expected net profit over the infinite horizon for each discretised state can be updated following Algorithm (3).

5.3.2 Preliminary results

In this section, we give some preliminary results from the classic backward recursion algorithm. We assume that for each period the external cash flow can be viewed as a random number following the normal distribution $\mathcal{N}(\mu = -£5, \sigma = 5)$. Note that we use the negative sign to represent the cash outflow from the cash

Algorithm 3 The classic backward recursion method in the CM model with two accounts

Step 1: Initialisation:

Step 1.1: Set the value of each terminal state to zero, i.e. $V_{(0)}(\vec{s}_{(0)}) = 0$ for $\vec{s}_{(0)} \in \mathbf{S}$.

Step 1.2: Define a small number ϵ. Set the iteration index $i := 1$.

Step 2: Update backwards:

Step 2.1: For each state $\vec{s}_{(i)} \in \mathbf{S}$, find the feasible action set $A(\vec{s}_{(i)})$ via **algorithm** (2).

Step 2.2: For each state-action tuple $(\vec{s}_{(i)}, a_{(i)})$ for all $\vec{s}_{(i)} \in \mathbf{S}$ and $a_{(i)} \in \mathbf{A}_t(\vec{s}_{(i)})$, get the state-action value $V_{(i)}(\vec{s}_{(i)}, a_{(i)})$ via equation (5.11).

Step 2.3: For each state $\vec{s}_{(i)} \in \mathbf{S}$, update the state value via:

$$V_{(i)}(\vec{s}_{(i)}) = \min_{a_{(i)} \in \mathbf{A}_t(\vec{s}_{(i)})} V_i(\vec{s}_{(i)}, a_{(i)}).$$

Step 3: Value comparison:

Step 3.1: For each state $\vec{s}_{(i)} \in \mathbf{S}$, record the value difference:

$$\text{diff}(\vec{s}_{(i)}) = |V_{(i)}(\vec{s}_{(i)}) - V_{(i-1)}(\vec{s}_{(i)})|.$$

Step 3.2: If $\max_{\vec{s}_{(i)} \in \mathbf{S}} \text{diff}(\vec{s}_{(i)}) \geq \epsilon$, set $i := i + 1$ and go to **Step 2**, else go to **Step 4**.

Step 4: Policy output:

Step 4.1: For each state $\vec{s}_{(i)}$, find the feasible action set $\mathbf{A}_t(\vec{s}_{(i)})$.

Step 4.2: For each state-action tuple $(\vec{s}_{(i)}, a_{(i)})$ for all $\vec{s}_{(i)} \in \mathbf{S}$, $a_{(i)} \in \mathbf{A}_t(\vec{s}_{(i)})$, get the state-action value $V_{(i)}(\vec{s}_{(i)}, a_{(i)})$ via equation (5.11).

Step 4.3: For each state $\vec{s}_{(i)} \in \mathbf{S}$, output the optimal action via:

$$a^*(\vec{s}_{(i)}) = \arg\min_{a_{(i)} \in \mathbf{A}_t(\vec{s}_{(i)})} V_{(i)}(\vec{s}_{(i)}, a_{(i)}).$$

account. It is equivalent to the company facing a stochastic cash demand which follows the distribution $\mathcal{N}(\mu = £5, \sigma = 5)$. In terms of the rest parameters, we let the return rate on asset be $r = 5\%$, the risk-free interest rate be $r_f = 2\%$ and the shortage penalty coefficient be $h = 2$. We focus on the cash management strategy with cash levels between £0 and £100 and asset levels between £0 and £200.

Figure 5.2(a) reveals the optimal policies when we discretise the random variable of the external cash flow using the simulation method. In this method we generate 300 samples of the external cash flow and assume that each sample $\Delta x_{(i)}^{j}$ is realised with probability $p = 1/300$ for $j = 1, 2, ..., 300$. Now the updating equation (5.11) can be approximated by:

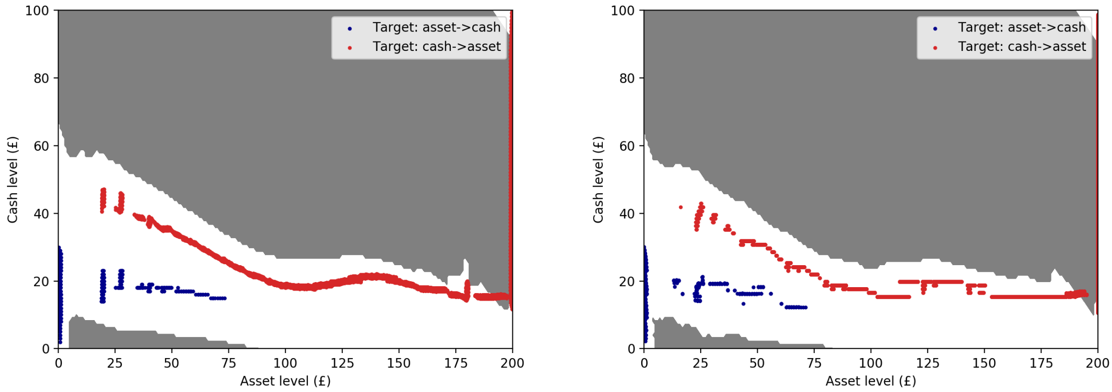
$$\begin{aligned} V_{(i)}\left(\vec{s}_{(i)}, a_{(i)}\right) &\approx r y_{(i)} - \Gamma\left(a_{(i)}\right) + \textstyle\sum_{j=1}^{300} p \left(-\Theta\left(x_{(i)}^{a_{(i)}}, \Delta x_{(i)}^{j}\right)\right. \\ &\quad \left. + \gamma \hat{V}_{(i-1)}\left(x_{(i-1)}, y_{(i-1)} \mid \vec{s}_{(i)}, a_{(i)}, \Delta x_{(i)}^{j}\right)\right). \end{aligned} \tag{5.13}$$

The result shows that the suggested policy of this two accounts cash management model still possesses the two-trigger two-target form. But instead of the trigger/target points in the traditional model, this two accounts cash management model provides trigger/target frontiers related to both the cash levels and the asset levels. As shown in Figure 5.2(a) if the company's state is in the white area, no action should be taken by the manager; if it is in the grey area, on the other hand, the agent should adjust its state to one of the target points.

In terms of the Gauss-Hermite quadrature method, the approximate equation can be expressed as:

$$\begin{cases} V_{(i)}\left(\vec{s}_{(i)}, a_{(i)}\right) & \approx r y_{(i)} - \Gamma\left(a_{(i)}\right) + \frac{1}{\sqrt{\pi}} \sum_{j=1}^{J} w_j y\left(\sqrt{2}\sigma x_j + \mu\right) \\ \Delta x_{(i)} & = \sqrt{2}\sigma x_j + \mu \\ y\left(\Delta x_{(i)}\right) & = -\Theta\left(x_{(i)}^{a_{(i)}}, \Delta x_{(i)}\right) + \gamma \hat{V}_{(i-1)}\left(x_{(i-1)}, y_{(i-1)} \mid \vec{s}_{(i)}, a_{(i)}, \Delta x_{(i)}\right) \end{cases} \tag{5.14}$$

where J is the number of sample points, x_j are the roots of the Hermite polynomial $H_J(x)$ and the corresponding weights w_j can be obtained via equation (4.14). In practice we select 9 sample points and the optimal policies is shown in Figure 5.2(b). Compared to the simulation method, the Gauss-Hermite quadrature method provides much more jagged solutions. But it can be seen that the policy



(a) Simulation method (b) Gauss-Hermite Quadrature method

Figure 5.2: Optimal policies in the two accounts cash management model: Simulation method versus quadrature method

is still of the two-trigger two-target form. Note that for each update process, the approximation equation (5.13) requires 300 observations while the approximation equation (5.14) only requires 9 observations. This means that the Gauss-Hermite Quadrature method reduces the computational cost to a large extent. In this example, the simulation method takes 5.08 hours while the Gauss-Quadrature method only takes 0.42 hours.

It is also worth pointing out the impact of the asset account capacity on the cash management strategies. For example, when the company's cash and asset holdings are close to their maximum levels (i.e. the state is in the upper-right corner), the algorithm suggests taking no action at all since investment causes more transfer fee but cannot push the asset level beyond its maximum boundary. In addition, many cash-to-asset targets are pushed on the maximum asset boundary since the asset level cannot exceed its maximum value (in this case, £200). One way to alleviate the boundary effect is setting a higher maximum asset level than the states we actually are interested in. Figure 5.3 shows the optimal policy in the two accounts cash management model where we set the maximum asset level to £400. It also includes examples of the policy action when buying and selling the asset, represented the South-East pointing arrow and the North-West pointing arrow respectively. Buying an asset incurs a charge and a reduction in the cash holding and a corresponding increase in the asset holding whereas selling the asset results in the converse. For any state in the upper-right shaded area in Figure

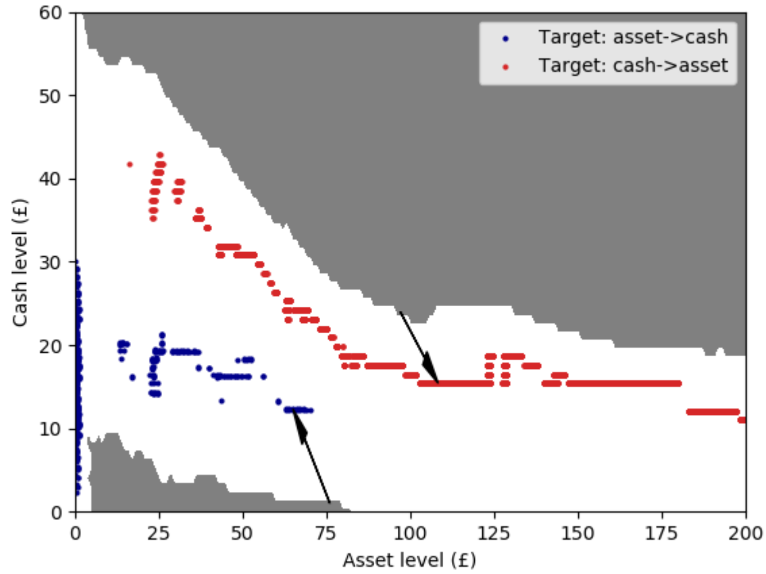


Figure 5.3: Optimal policies in the two accounts cash management model: Quiver graph

5.3 a buying asset action is triggered with the consequent state adjustment to lower cash holding and greater asset holding on the red-highlighted control limit. For any state in the lower-left shaded area, a selling asset action is triggered with the state adjustment to a greater cash holdings and lower asset holding on the blue-highlighted control limit.

This figure illustrates that in general the trigger levels and target levels in terms of cash account drop with the increase of asset size. This is due to the internal cash inflows generated from the asset account. With larger asset size, we expect higher cash inflow at each period and thus the agent wishes to hold lower cash balance. However in Figure 5.3 we spot an exceptional 'bump' on the trigger/target frontier, that is for asset level between £100 and £130, the trigger and target levels increase with the asset size. At last, we notice that once the company's asset account is large enough, the agent does not adopt any selling action to replenish his cash balance regardless the current cash holding level since for each period the internal cash inflow itself is enough to fulfill the external cash demand.

5.4 Insolvency risk

In the previous section, we present a method to find the optimal policy π that maximise company's net profit over the infinite horizon. In this section we will propose a backward recursion method to measure the company's insolvency risk associated with the cash policy π given its initial state $\vec{s}_{t_0}$. Let $P^{\pi}(\vec{s}_t)$ denote the probability of the company not going bankrupt in the whole planning horizon once it visits the state $\vec{s}$ at time t. For the planning horizon $(t_0, ..., T)$, we calculate the company's survival probability backwards. At the last period, the probability of the company not going bankrupt is 1 if it is not insolvent at time T, i.e. for the first iteration $i = 1$:

$$P^{\pi}_{(i)}(\vec{s}_{(i)}) = \begin{cases} 0 & \text{if } \vec{s}_{(i)} = (0,0) \\ 1 & \text{otherwise.} \end{cases}$$

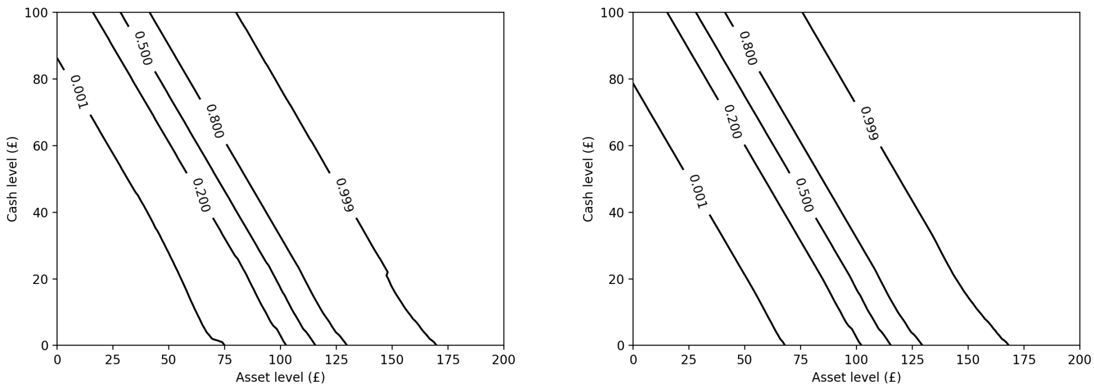
Then at each iteration we calculate the probability one period backwards. The probability of the company not going bankrupt once it visits state (s^x_t, s^y_t) is the sum of the products of the probability of visiting each state at next time period and the company's survival probability once it visits that state, i.e.

$$P^{\pi}_{(i)}(\vec{s}_{(i)}) = \int_{-\infty}^{\infty} P^{\pi}_{(i+1)}\left(x_{(i+1)}, y_{(i+1)} | \vec{s}_{(i)}, \Delta x\right) f(\Delta x) \mathrm{d}\Delta x. \tag{5.15}$$

As discussed in Section 5.3.1, we use the bilinear interpolation method to approximate the company's survival probability $P^{\pi}_{(i+1)}(x_{(i+1)}, y_{(i+1)} | \vec{s}_{(i)}, \Delta x)$ based on its adjacent states' survival probabilities and use the Gauss-Hermite quadrature method to discretise the external cash flow Δx. For a finite horizon problem, we stop the iteration once $i \geq T$. For an infinite horizon problem, we keep the iteration until the survival probability for each state does not change anymore. In practice, we set a small number ϵ and stop the iteration once the update cannot improve the estimations of the survival probabilities at least by ϵ for at least one state, i.e. the iteration stops if

$$\max_{\vec{s} \in \mathbf{S}} \left| P^{\pi}_{(i)}(\vec{s}) - P^{\pi}_{(i+1)}(\vec{s}) \right| < \epsilon.$$

Assume that the external cash flow for each period is subject to the normal



(a) Backward recursion method (b) Simulation method (1000 samples)

Figure 5.4: Survival rate of the company:
Backward method versus Simulation method

distribution $\mathcal{N}(-£5, 5)$, the transaction cost parameters are $(K^-, K^+, k^-, k^+) = (£1, £2, 0.1, 0.2)$, the return rate on asset is $r = 5\%$ and the shortage penalty coefficient is $h = 2$. Figure 5.4(a) reveals the probability of the company not going bankrupt in the infinite horizon given each initial state and the optimal policy π that maximise its net profit. It shows that the survival probability is monotonically increasing with respect to cash level and asset level. Moreover, with small size of initial cash and asset account, the cash demand dominates the cash inflow generated by asset and hence the company's survival probability is close to zero. If the company's asset account can generate cash inflow that is enough to offset the cash demand, the survival probability climbs rapidly. Once the internal cash inflow dominates the external cash demand, the company has a survival probability very close to one.

We also randomly generate 1000 sample path and run simulations for each initial state. In each simulation, we assume the company will never become insolvent once it visits the state $(s^x_{\max}, s^y_{\max})$. The result is shown in Figure 5.4(b). Note that this method gives a similar result to the backward recursion method but requires much more computational time: the simulation method takes 42.7 minutes while the backward method only takes 0.88 minutes for this numerical experiment.

5.5 A cash management model with asset growth

So far we discussed the cash management model where the asset account is considered as a source of cash inflow. Consider a firm that invests its cash into business. With more investment, the manager expect higher return in future. Hence we assume a deterministic amount of cash inflow that is proportional to the size of its asset. In this section we consider another scenario where the company invests its cash into stocks. In this case, the company's asset does not affect the cash flow directly, but its asset (stock price) grows at each period. Similar to the two accounts model with internal cash inflow, we assume the company receives its capital gain at the beginning of each period before the manager taking actions. Hence the post-decision state can be modified as:

$$\begin{bmatrix} x_t^{a_t} \\ y_t^{a_t} \end{bmatrix} = \begin{bmatrix} x_t + a_t - \Gamma(a_t) \cdot 1_{a_t \leq 0} \\ y_t + ry_t - a_t - \Gamma(a_t) \cdot 1_{a_t > 0} \end{bmatrix}. \tag{5.16}$$

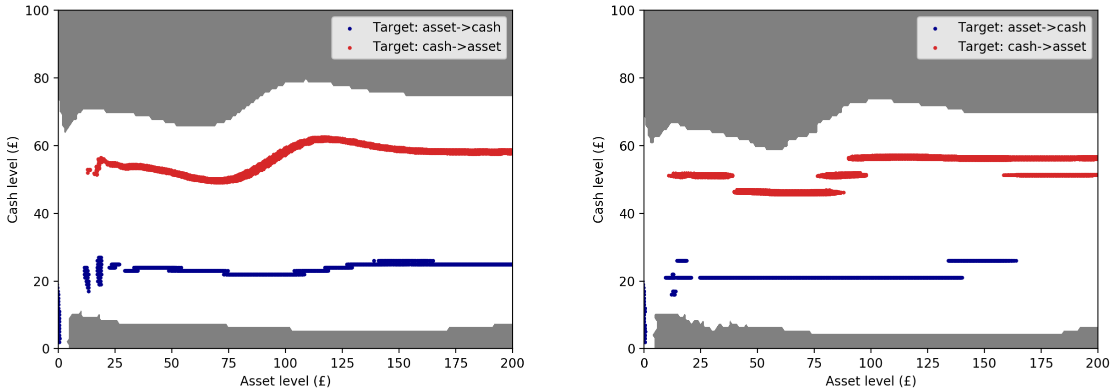
In the case of no cash shortage, the system transitions to next stage following:

$$\begin{bmatrix} x_{t+1} \\ y_{t+1} \end{bmatrix} = \begin{bmatrix} x_t^{a_t} + \Delta x_t \\ y_t^{a_t} \end{bmatrix}. \tag{5.17}$$

If a cash shortage occurs (i.e. $x_t^{a_t} + \Delta x_t < 0$) and the amount of cash deficit is $d_t = |x_t^{a_t} + \Delta x_t|$, the system transitions following:

$$\begin{bmatrix} x_{t+1} \\ y_{t+1} \end{bmatrix} = \begin{bmatrix} 0 \\ y_t^{a_t} - d_t - \Gamma(d_t) - \Theta(x_t^{a_t}) \end{bmatrix}. \tag{5.18}$$

Figure 5.5 shows an example of the optimal policies of the cash management model with asset growth when we discretise the continuous random variable ΔX via the simulation method (300 samples) and the Gauss-Hermite quadrature method. In this example we assume that the external cash demand follows the normal distribution $\mathcal{N}(-£5, 5)$ and set the operational conditions as $K^+ = £1, K^- = £2, k^+ = 0.1, k^- = 0.2, r = 5\%, r_f = 2\%, h = 2$. It shows that the Gauss-Hermite quadrature method compared with the simulation method gives a much more jaggy solution, but both policies of this cash management model still



(a) Simulation method

(b) Gauss-Hermite Quadrature method

Figure 5.5: An example of the optimal policy in the two accounts model with asset growth

possess the two-trigger two-target form, i.e. the agent should not take any action when the company is in the white area. Once the cash holding level reaches the trigger frontiers, the agent should adjust the company's state back to one of the target positions. In addition, similar to the original two accounts model, we spot the 'bump' pattern in the states where the return on asset account is close to the expected cash outflow. In this 'bump area', the cash holding trigger/target level increases with the size of asset account. Note that in the original two accounts model where the returns on asset is in the form of cash, there is no lower trigger frontier (in other words the agent has no motivation to replenish his cash balance) once the cash inflow from asset dominates the cash demand. In this cash management model however, since the asset account does not generate the internal cash inflow directly, the agent should always replenish his cash account.

5.6 Numerical experiments

So far we presented two cash management models: the two accounts model with internal cash inflows and the two accounts model with asset growth. In this section, we undertake numerical experiments to study the cash management policy in different scenarios. To begin with we study the effect of discretisation level of states on the policies. We also discuss the optimal cash management strategies across a range of problem sizes. We then show the influence of the transaction cost parameters. Since the selling and buying transfer fee is assumed to be different and each transaction cost function consists of a fixed part and a proportional part, we propose two sets of experiments. In the first set of experiments, we show the cash management policy in each combination of selling and buying transfer fee and in the next set, we study the policy in different combination of fixed and proportional parameters. Furthermore, we study the impact of the shortage penalty coefficient and the external cash flow on the cash management policy. At last, we report the proportion of selling states and buying states under different combinations of parameter settings. All the experiments are programmed in C++ 12.0.0 on a PC with 2.5 GHz Quad-Core Intel Core i7 and 8 GB memory.

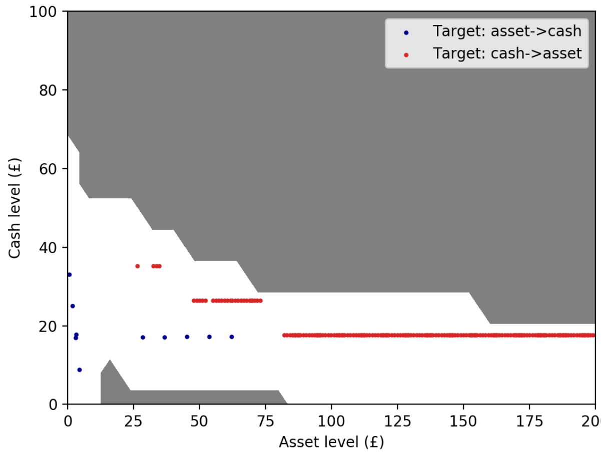
5.6.1 Discretisation level

In our study, we discretise the continuous states (i.e. the cash level and the asset level) as well as the continuous actions and then formulate the cash management model as a discretised Markov decision process. In theory, the policy converges at the optimal solution when states and actions are discretised infinitely. However, with the increase of discretisation level, the computational cost also climbs dramatically. Our goal is to examine the impact of different discretisation level on the policies and find a suitable discretisation solution that gives a relatively accurate policy while requiring reasonable computational time.

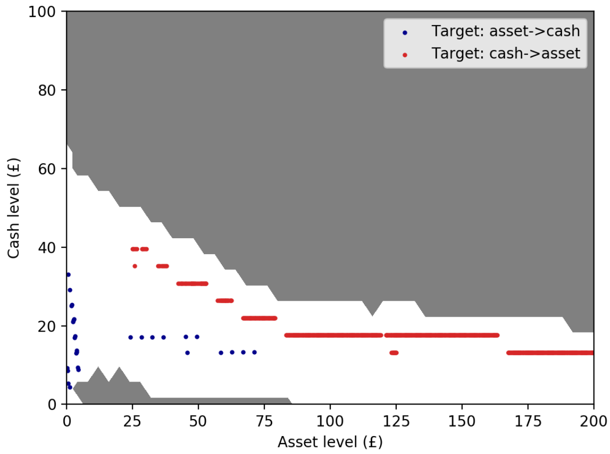
For the sake of simplicity, we let Δ be the common discretisation step for cash states, asset states as well as actions, i.e. $\Delta = \Delta s^x = \Delta s^y = \Delta a$. In this section we adopt 6 discretisation levels, namely $\Delta = (£8, £4, £2, £1, £0.5, £0.25)$, and undertake numerical experiments for the two accounts model with internal cash inflow as well as the model with asset growth. In the numerical experiments, we are interested in the scenarios where cash balance is between £0 and £100 and asset level is between £0 and £200. However as discussed in Section 5.3.2, it requires some extra asset states in these models to alleviate the boundary impact. Hence we set the maximum asset to £400. Moreover, we set other parameters to: $\Delta X \sim \mathcal{N}(-£5, 5), K^+ = £1, K^- = £2, k^+ = 0.1, k^- = 0.2, r = 5\%, h = 2, r_f = 2\%$). Figure 5.6 and Figure 5.7 show the optimal cash management strategies under each discretisation level Δ for the model with internal cash inflow and the model with asset growth. It is clear that all strategies with each discretisation level have a similar pattern but the model with a finer discretisation level gives a less jaggy trigger/target frontier. In addition, it is worth to point out that a part of the jaggedness occurs when we adopt the Gauss-Hermite quadrature method to discretise the variable of the external cash flow, hence the improvement of state space discretisation and action space discretisation cannot remove all aliasing from the policy results.

It is obvious that with a finer discretisation level, the size of state space and the action space grow rapidly and hence the model needs much more computational time. Table 5.1 reveals the total number of states and the corresponding solution time when we adopt each discretisation level. In this table, Model I is the cash management model with internal cash inflow and Model II is the model with asset growth. We also compare the objective function value for each state between every two successive discretisation levels. It can be seen that the maximum value difference drops when we keep halving the discretisation step Δ while the computational time grows rapidly. When we change the discretisation step Δ from £8 to £4, the state value with maximum difference improves by £42.79 (Model I) and £31.72 (Model II). However if we change the discretisation step from £0.5 to £0.25, the state values only change by £3.15 (Model I) and £3.79 (Model II) at most while the computational time increases from 93.62 minutes to 603.55 minutes (Model I) and from 91.68 minutes to 598.63 minutes (Model II). This table shows that $\Delta_4 =$ £1 is a desirable step considering both Model I and Model II with discretisation level Δ_4 requires computational time less than twenty minutes. Halving Δ_4 only improves the state values by £5.09 (Model I) and £5.94 (Model II) at the most but requires more than 1.5 hours.

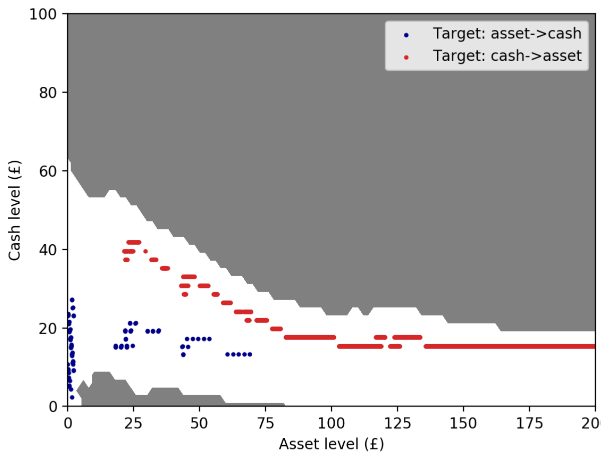
Figure 5.8 illustrates the average state values under each discretisation level. It can be seen that halving the discretisation step Δ from £8 to £4 (namely from Δ_1 to Δ_2) improves the average state value by £8.1 in Model I and by £8.05 in Model II. This improvements decrease rapidly with the increase of discretisation level. Halving Δ_4 only improves the average state values by £0.85 (Model I) and £1.5 (Model II). Thus in the following numerical experiments, we will use $\Delta_4 =$ £1 as the discretisation step.



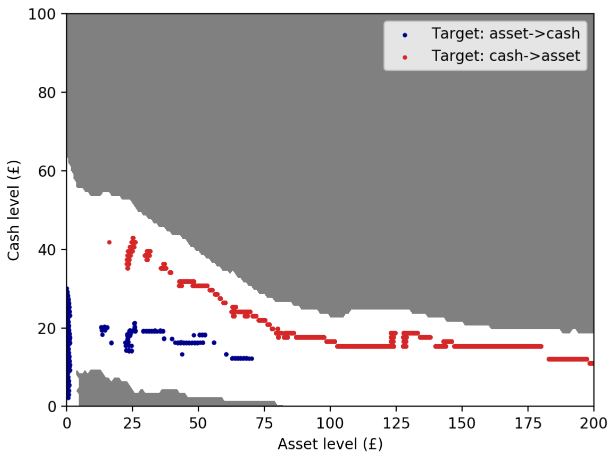
(a) $\Delta = £8$



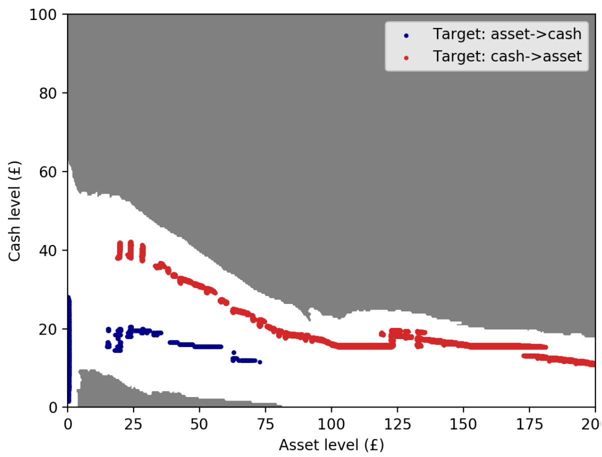
(b) $\Delta = £4$



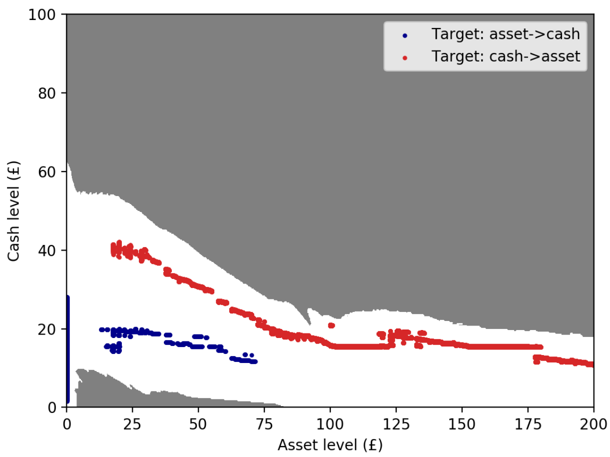
(c) $\Delta = £2$



(d) $\Delta = £1$

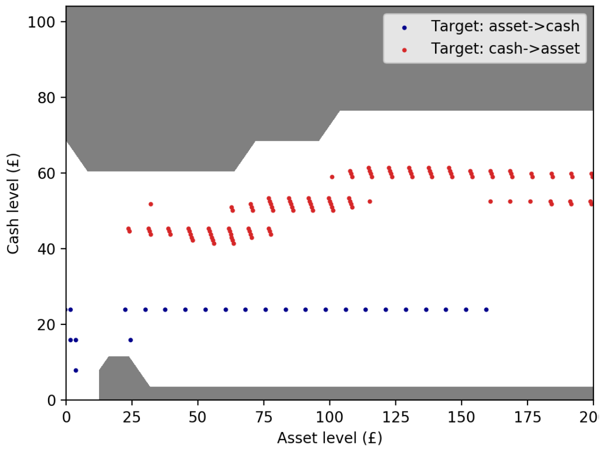


(e) $\Delta = £0.5$

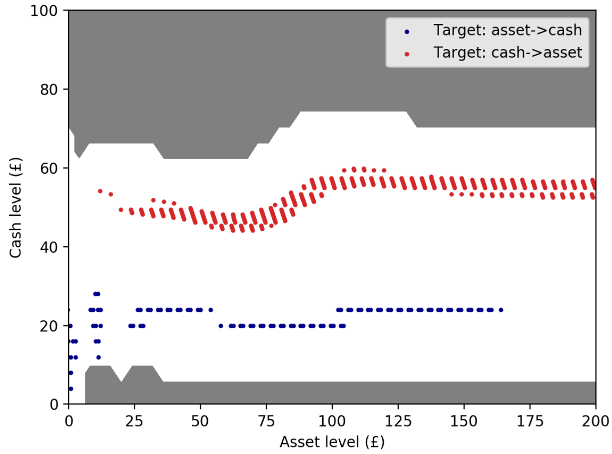


(f) $\Delta = £0.25$

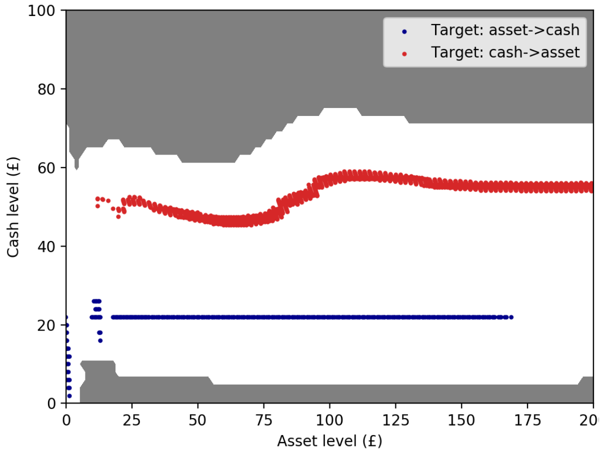
Figure 5.6: Optimal policies in the two accounts model with internal cash flow under each discretisation level



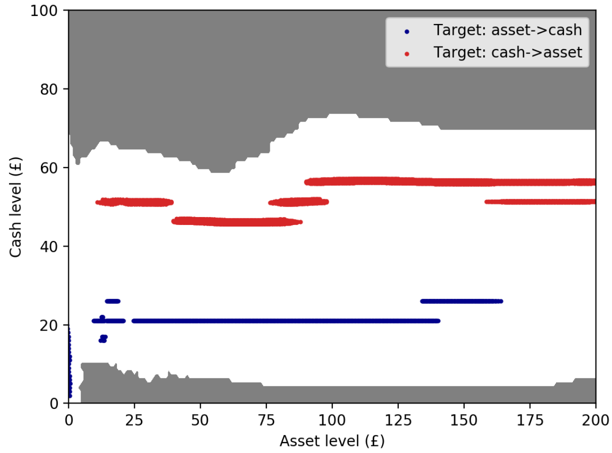
(a) $\Delta = £8$



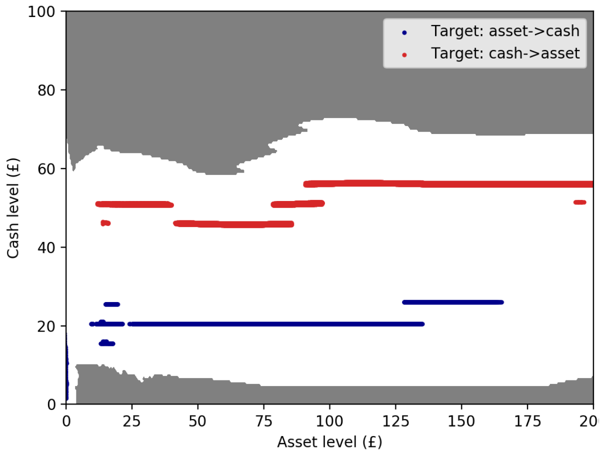
(b) $\Delta = £4$



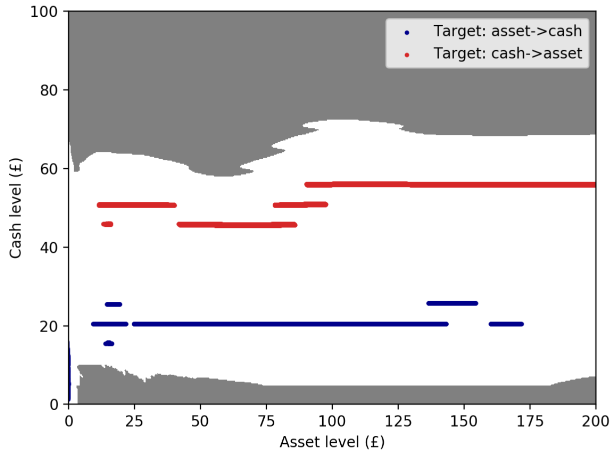
(c) $\Delta = £2$



(d) $\Delta = £1$



(e) $\Delta = £0.5$

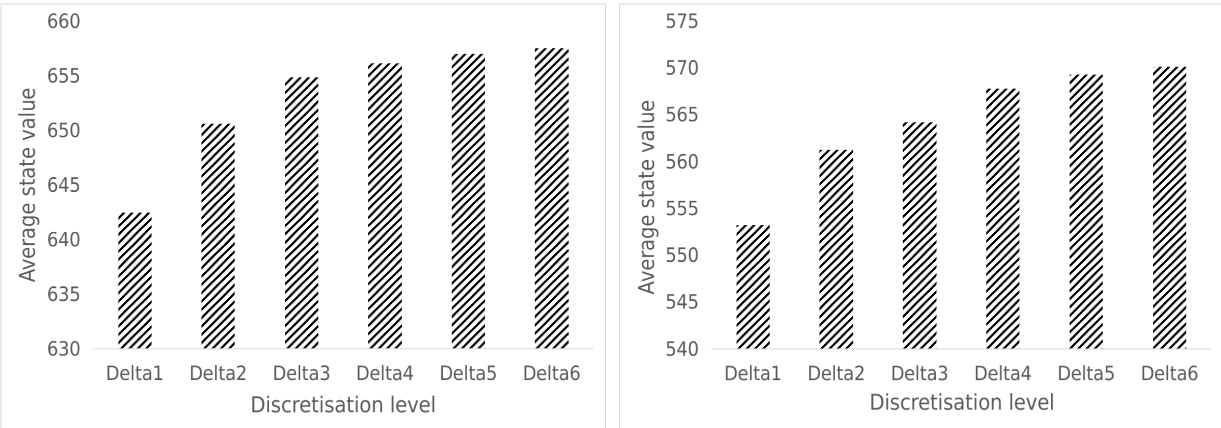


(f) $\Delta = £0.25$

Figure 5.7: Optimal policies in the two accounts model with asset growth under each discretisation level

Table 5.1: Convergence speed of different discretised models

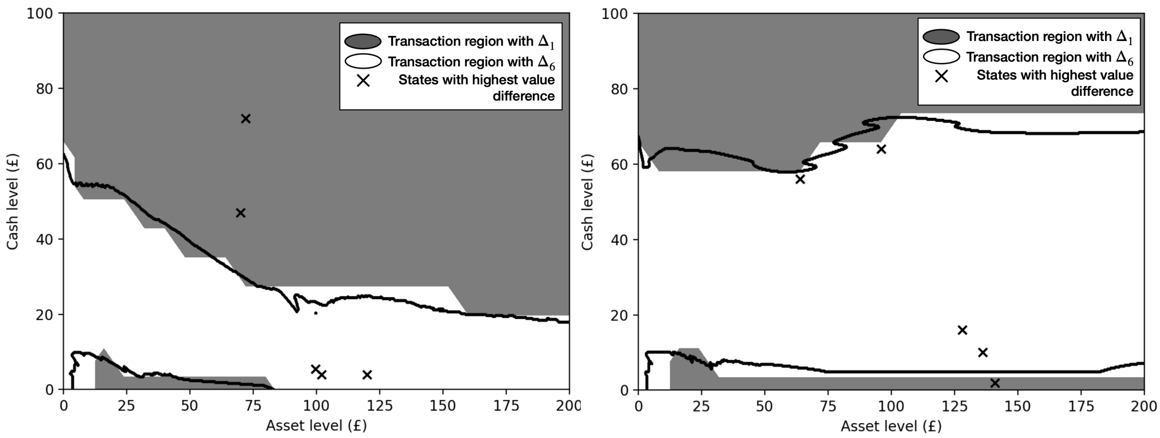
	Number of states	$\max_{\vec{s} \in \mathbf{S}^{\Delta_i}} \left\| V^{\Delta_i}(\vec{s}) - V^{\Delta_{i-1}}(\vec{s}) \right\|$		Computational time	
		Model I (£)	Model II (£)	Model I (minutes)	Model II (minutes)
$\Delta_1 = £8$	714	-	-	0.08	0.08
$\Delta_2 = £4$	2,626	42.79	31.72	0.37	0.36
$\Delta_3 = £2$	10,251	17.50	20.91	2.35	2.26
$\Delta_4 = £1$	40,501	10.73	19.41	16.71	16.21
$\Delta_5 = £0.5$	161,001	5.09	5.94	93.62	91.68
$\Delta_6 = £0.25$	642,001	3.15	3.79	603.55	598.63



(a) Model with internal cash inflow (b) Model with asset growth

Figure 5.8: Average state value under each discretisation level

At last in Figure 5.9 we plot the positions of the states with maximum value difference between each two successive discretisation level. It is worth to point out that after each improvement of the discretisation level, the states with the maximum value difference are in the places where the company has approximately balanced cash outflow and cash inflow/capital gain. For these states, an accurate cash management policy is of essential importance since a good policy may lead the company to the status where the profit dominates the cash demand and the company can keep accumulating its wealth. Otherwise it may lead the company to the status where its profit cannot meet the cash demand and hence the agent needs to keep replenishing the cash balance by selling his asset which will jeopardise its profitability further.



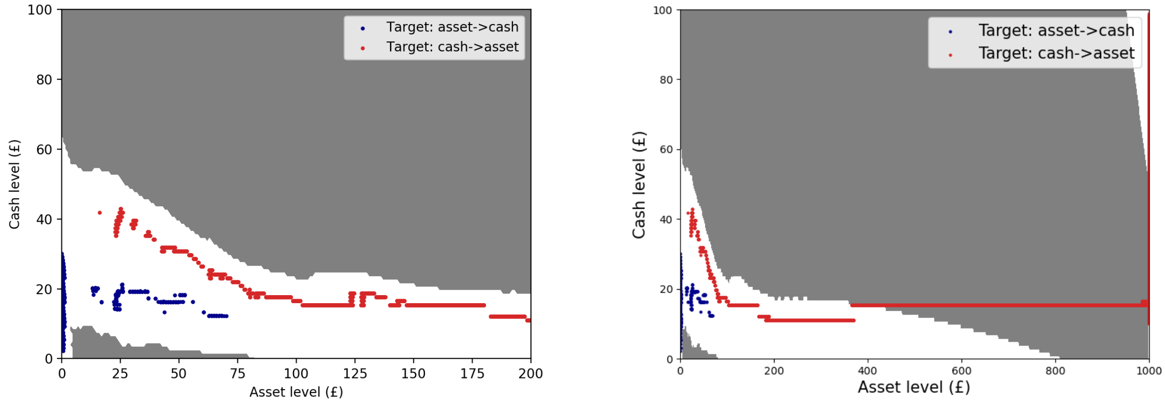
(a) Model with internal cash inflow

(b) Model with asset growth

Figure 5.9: Policy comparison of Δ_1 and Δ_6 and positions of states with maximum value difference

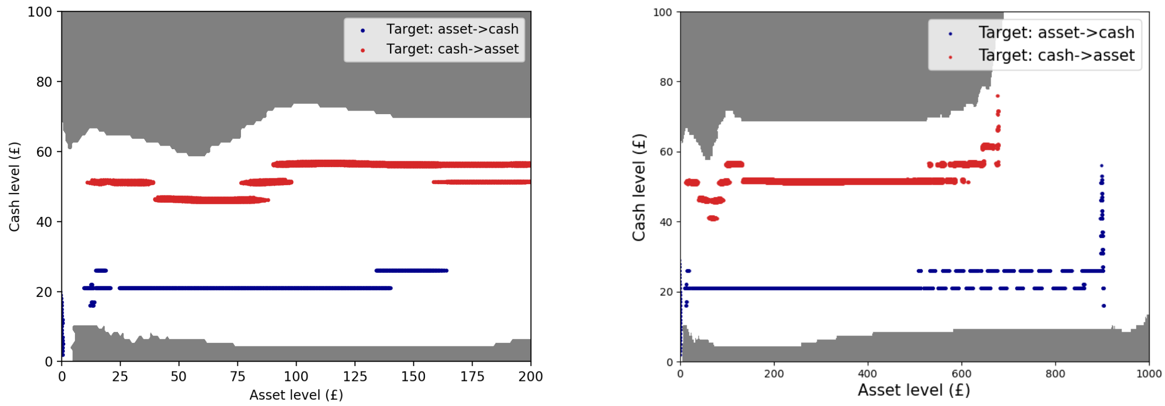
5.6.2 Increasing the scale of the problems

Now we discuss the cash management policy in models with larger state space. We adopt the discretisation level $\Delta = £1$ and increase the state space to $\mathbf{S}^x \times \mathbf{S}^y = [£0, £100] \times [£0, £1000]$. Other parameters are set to: $\Delta X \sim \mathcal{N}(-£5, 5), K^+ = £1, K^- = £2, k^+ = 0.1, k^- = 0.2, r = 5\%, h = 2, r_f = 2\%$. Note that we did not scale up the cash account. The cash policy in any state with cash level higher than the upper trigger level is to transfer the extra cash into the asset account. Hence the states with large cash account is of less interest.



(a) $[£0, £100] \times [£0, £200]$ (b) $[£0, £100] \times [£0, £1000]$

Figure 5.10: Optimal cash policy in the models with internal cash flow in larger state space



(a) $[£0, £100] \times [£0, £200]$ (b) $[£0, £100] \times [£0, £1000]$

Figure 5.11: Optimal cash policy in the models with asset growth in larger state space

Figure 5.10 and Figure 5.11 illustrates the optimal cash policy in the original models and the models with maximum asset level equal to £1000. It can be seen that in the area $\mathbf{S}^x \times \mathbf{S}^y = [£0, £100] \times [£0, £200]$, the optimal cash management policy in the scaled models closely resembles the policy in the original models. Note that slight difference exists because of the boundary effects.

In the model with internal cash inflow as shown in Figure 5.10(b), it can be seen that the upper trigger level decreases with a larger asset. This is because with a larger asset, the company receives more cash inflow in following periods and hence needs less cash balance at the current period. If the asset size is enough, the agent should invest all cash balance into profitable asset because the expected cash inflow itself should be able to meet the cash demand. It is worth to note that the upper target level exceeds the upper trigger level once the asset is large enough. But this does not suggest a selling action. The trigger level represents the states at the beginning of each period when the agent has not received the cash inflow but has to make cash management decisions. The target level reports the states where the system is expected to be after that the cash inflow has been received and that the corresponding action has been taken. Hence a target level higher than the trigger level suggests that the agent should invest all his cash balance at hand and a portion of the cash inflow he is about to receive.

In the model with asset growth as shown in Figure 5.11(b), it can be seen that once the asset level is large enough, all trigger levels and target levels do not change with the size of asset account and the optimal cash management strategy resembles the (L, l, u, U) policy proposed in Eppen & Fama's work (Eppen & Fama (1968)) closely. We also spot significant boundary effect in Figure 5.10(b). Once the asset grows significantly faster than the cash outflow, the asset will keep accumulating and soon reach the capacity of the asset account. During this process, investing cash into the asset account brings little profit while incuring transfer fee. Hence the optimal solution is doing nothing when the asset level is close to the asset account's capacity.

Another way to scale up the model is to change the unit monetary measure in the original model. For instance, in the original model, if we replace £1 with £4, the scaled up model will study the cash policy in the state space $[£0, £400] \times [£0, £800]$ with parameters $\Delta X \sim \mathcal{N}(-£20, 20), K^+ = £4, K^- = £8, k^+ = 0.1, k^- = 0.2, r = 5\%, h = 2, r_f = 2\%$. If we still set the discretisation level to $\Delta = £1$, the shape of the optimal cash policy will be just like the policy from the original model with discretisation level equal to $\Delta = £0.25$. In other words, by studying the cash policy in the models with a finer discretisation level, we also learned the cash policy in a scaled up but less discretised model.

5.6.3 Transaction cost

In this section we examine the impact of transaction cost on the two accounts cash management policy. In our models, we adopt the fixed plus proportional transac-

Table 5.2: Transaction cost parameters

	K^+	K^-	k^+	k^-
Low	£0.5	£0.25	0.05	0.025
Medium	£2	£1	0.2	0.1
High	£5	£2	0.5	0.2

tion cost function and assume the selling transfer fee and the buying transfer fee are different.

As shown in Table 5.2, three scenarios are considered in the following experiments, namely the low transaction cost, the medium transaction cost and the high transaction cost. We propose two sets of numerical experiment: the first set examines each combination of selling transfer fee and the buying transfer fee; the second set examines each combination of fixed transaction parameters and proportional transaction parameters. Figure 5.12 and Figure 5.13 shows the cash management policy with each combinations of selling and buying transfer fee. Note that in some cases, the upper target cash levels are higher than the upper trigger level. This is because in these states the internal cash inflow outweighs cash demand to a large extent. Hence in each period, the post-decision cash level is higher than the initial cash level even when the agent takes a buying action. In Figure 5.12 and Figure 5.13 each column shows the comparison of different selling transfer fee given the same buying transfer fee and each row shows the comparison of different buying transfer fee given each selling transfer fee. It shows that in general the transaction regions (i.e. the states where the agent should take selling or buying actions) shrinks with the increase of transaction cost. Moreover each column of these figures shows that the selling transaction cost mainly affects the selling region (i.e. the states where the agent should sell his asset and replenish his cash balance) but also has a relatively small impacts on the buying region (i.e. the states where the agent should invest his cash into the asset account). Similarly once we fix the selling price, it can be seen that although the buying transaction cost affects both buying region and the selling region, the buying region is much more sensitive to the buying transaction cost than the selling region.

Each column in Figure 5.14 and Figure 5.15 shows the impact of the proportional part of transaction cost function on the cash management policy while each row shows how the fixed part of transaction cost function affects the policy. In Section 4.4.1 we show that in the traditional one account model, the (U, u, l, L) policy will change to the (L, B, U) form (i.e. the agent should adjust his cash balance to one fixed target level once his cash level reaches the upper trigger level or the lower trigger level) if the transaction cost function is purely fixed. Moreover the optimal policy will change to the (l, u) form (i.e. the trigger levels are the same to the target levels) once the transaction cost function is purely proportional. A similar pattern is observed in experiments of the two accounts models. Both Figure 5.14 and Figure 5.15 reveal that when we set the proportional parameters of the transaction cost function to a very small value (e.g. $k^- = 0.025, k^+ = 0.05$), the upper target positions and the lower target positions are very close to each other regardless of the size of the fixed part of the transaction cost function. The gap between the upper target and the lower target becomes wider once we increase the proportional parameters. In addition, the gaps between the trigger frontiers and the target positions widen with the increase of the fixed parameters (K^+ and K^-).

5.6.4 Shortage penalty

In the two accounts cash management models, we assume that the cash shortage penalty is proportional to the size of the cash deficit. In this section we undertake experiments with different penalty coefficient to examine the impact of the penalty coefficient on cash management policy. Let the low, medium and high shortage penalty coefficients be 0.5, 2.0 and 8 respectively. Other parameters are set to $K^+ = £2, K^- = £1, k^+ = 0.2, k^- = 0.1, r = 5\%, r_f = 2\%, \Delta x \sim \mathcal{N}(-£5, 5)$ and the optimal policy is shown in Figure 5.16. We spot that the selling region is amplified with a higher shortage penalty while the buying region shrinks. This can be interpret as that with a higher shortage penalty, the agent tends to adopt a 'safer' policy to avoid the risk of cash deficit, i.e. replenishing his cash balance at a higher trigger cash level and keeping more cash remains when he invests into asset.

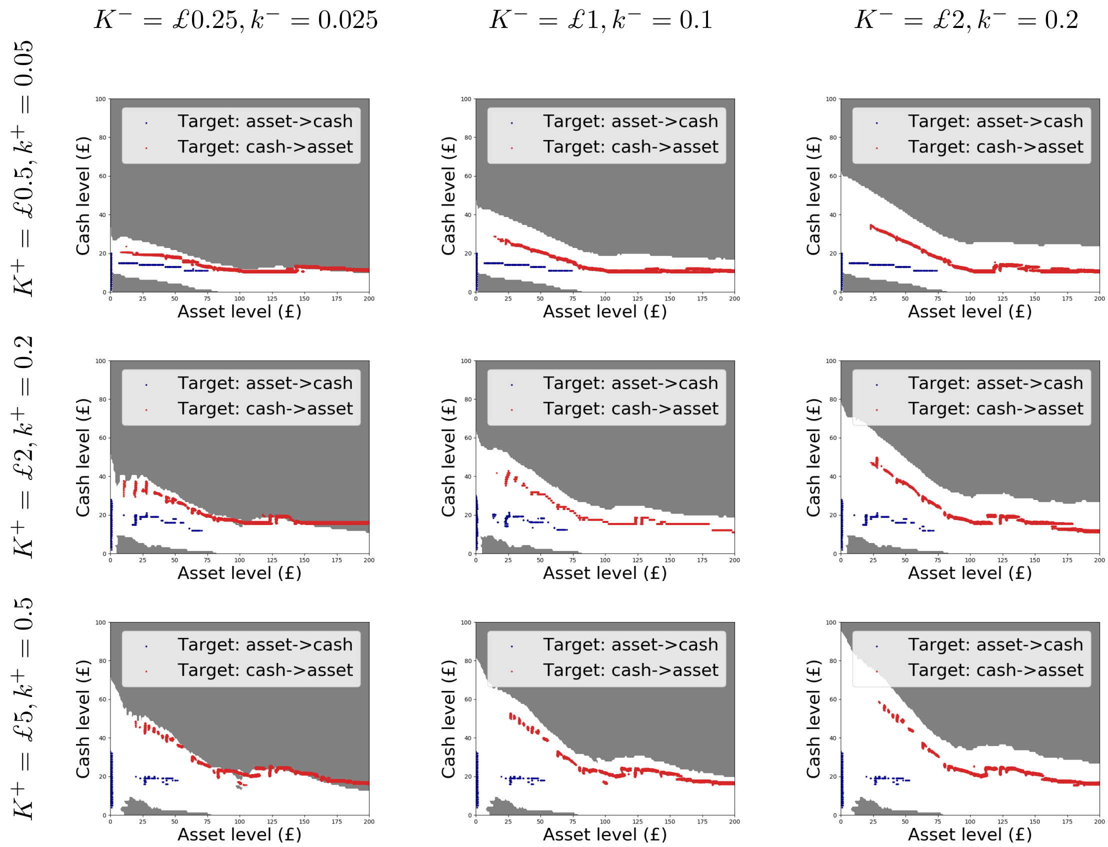


Figure 5.12: Combinations of selling and buying transfer fee in the two accounts model with internal cash inflow

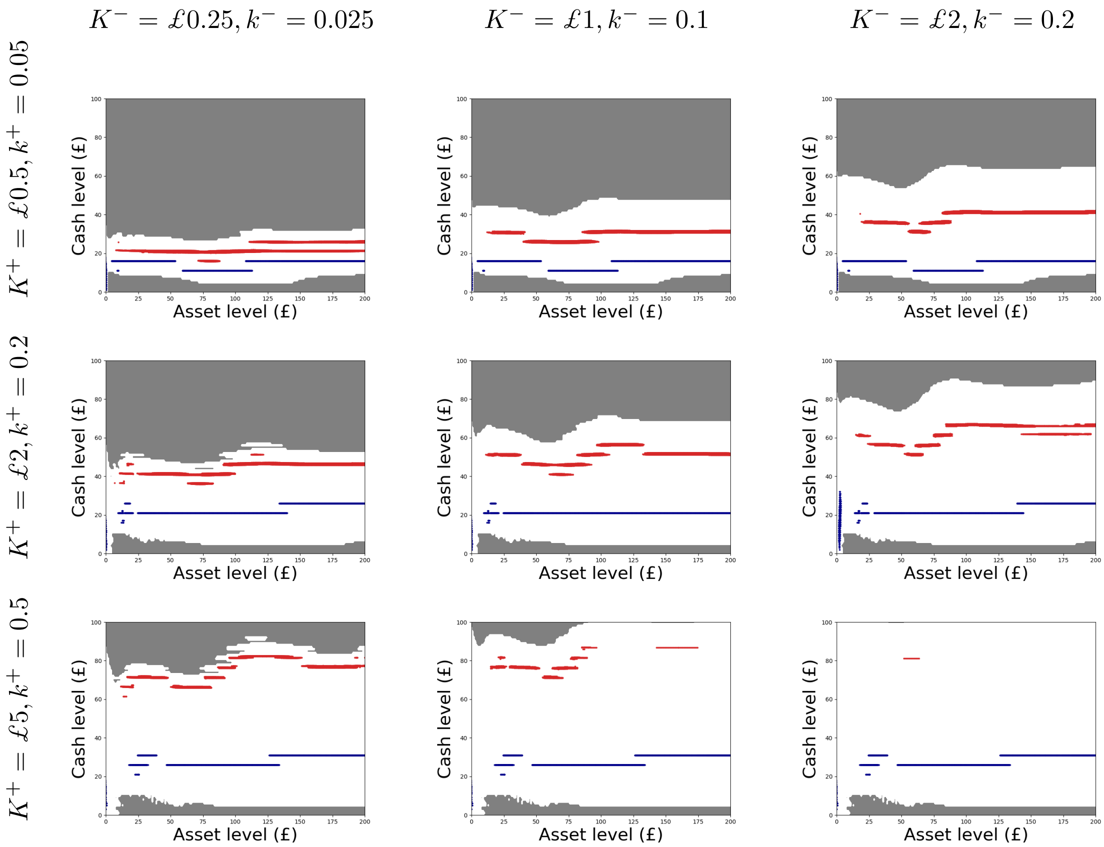


Figure 5.13: Combinations of selling and buying transfer fee in the two accounts model with asset growth

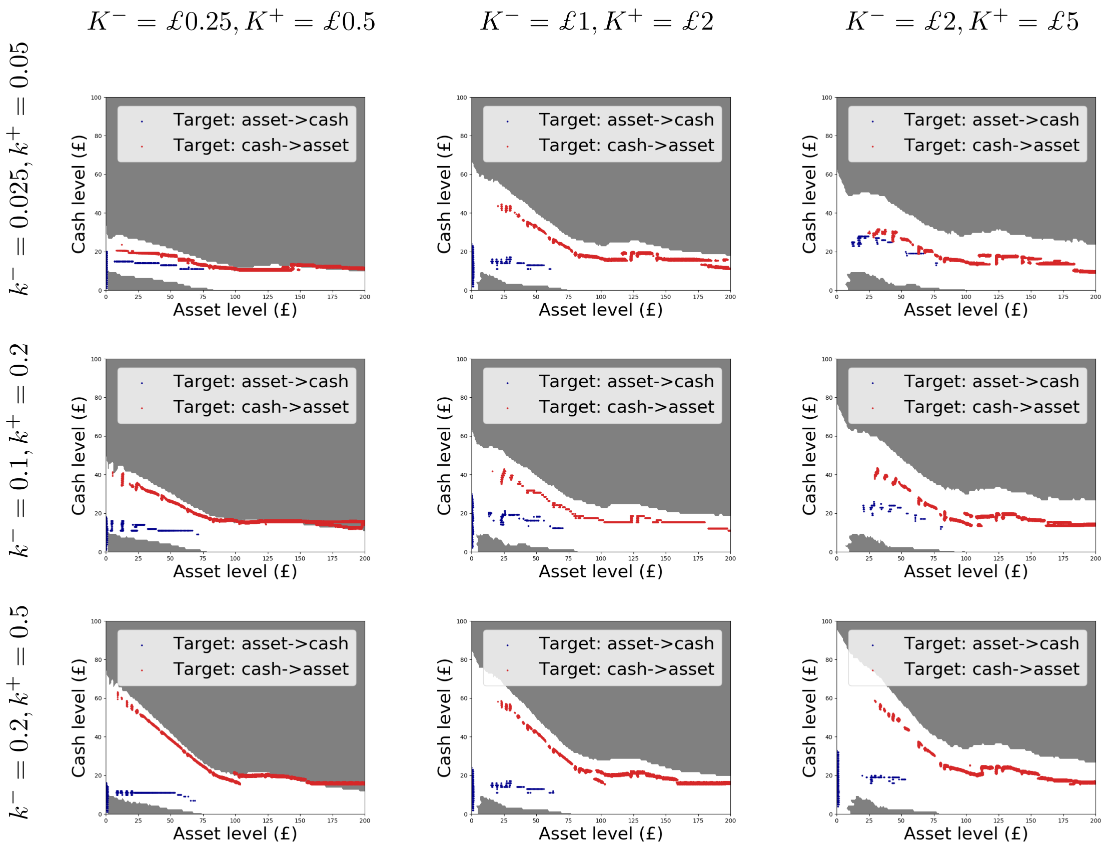


Figure 5.14: Combinations of fixed and proportional transfer parameters in the two accounts model with internal cash inflow

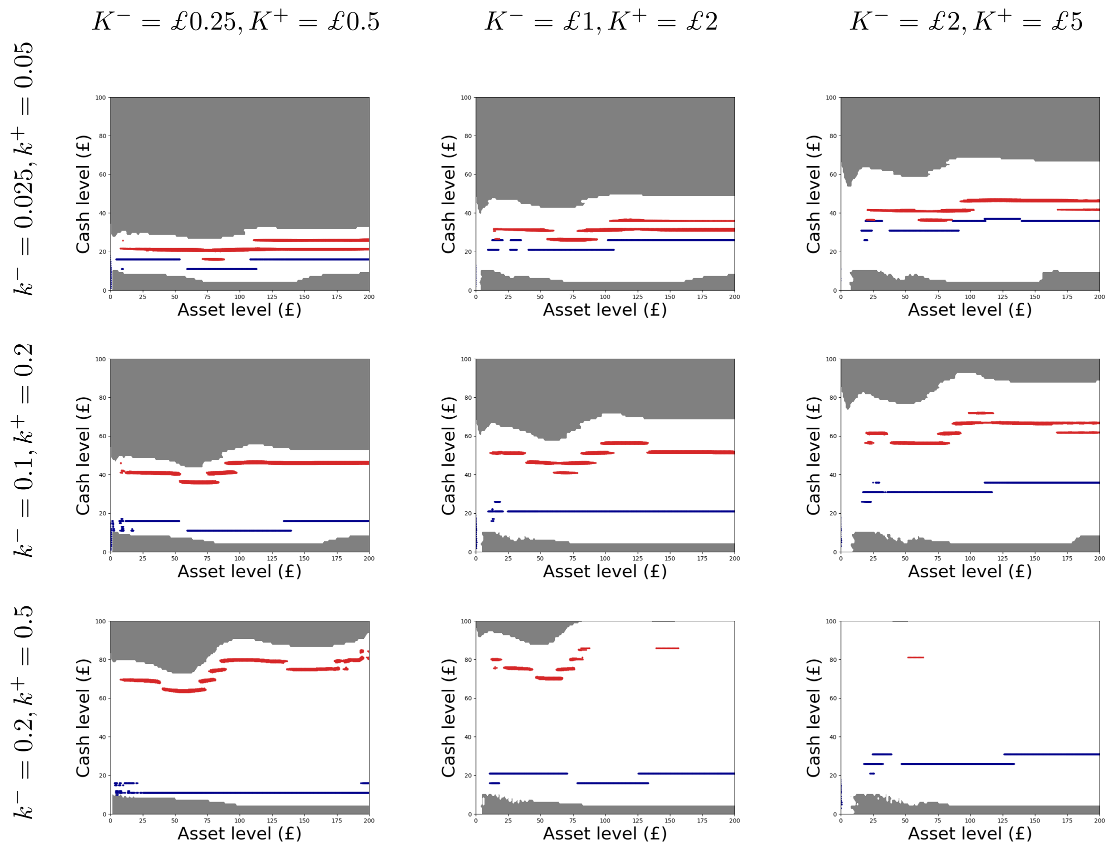
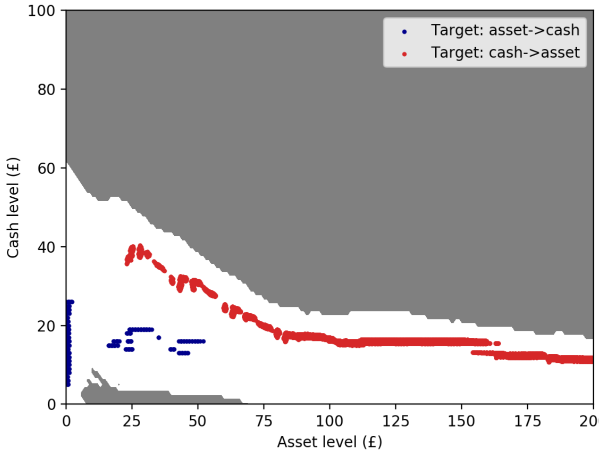
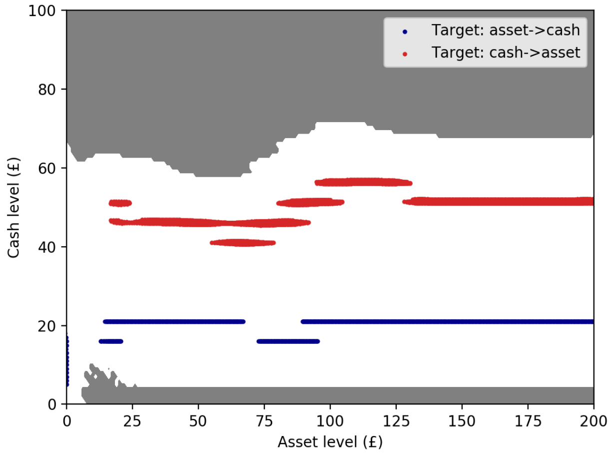


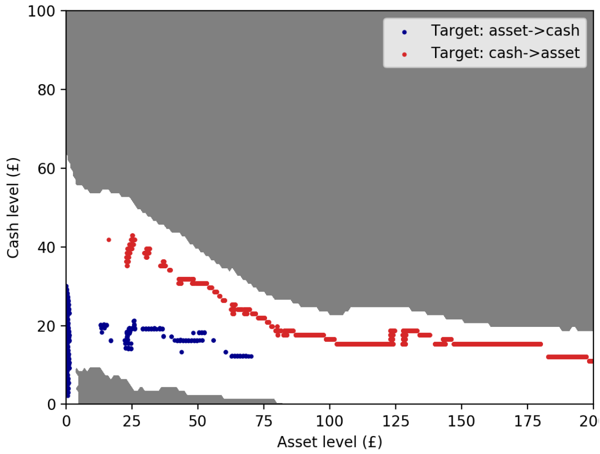
Figure 5.15: Combinations of selling and buying transfer fee in the two accounts model with asset growth



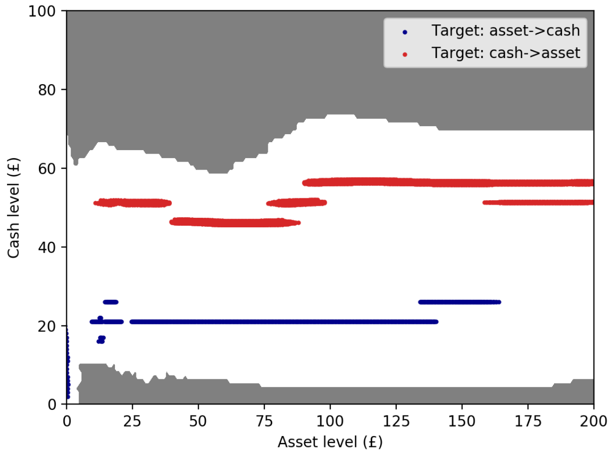
(a) Model with internal cash inflow, $h = 0.5$



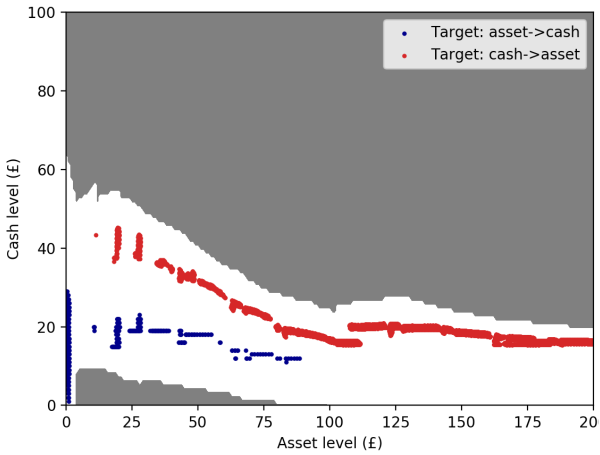
(b) Model with asset growth, $h = 0.5$



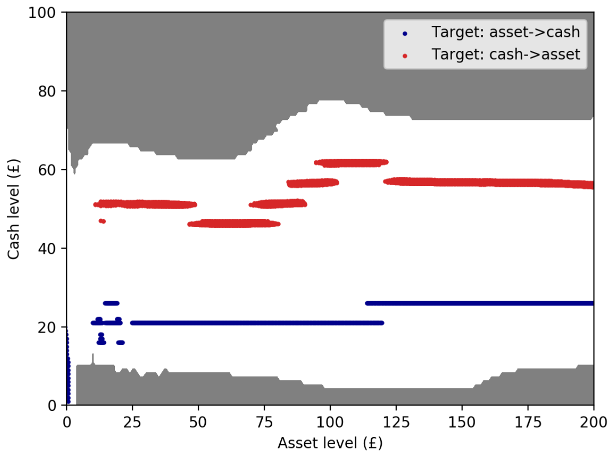
(c) Model with internal cash inflow, $h = 2$



(d) Model with asset growth, $h = 2$



(e) Model with internal cash inflow, $h = 8$



(f) Model with asset growth, $h = 8$

Figure 5.16: Optimal policies in the two accounts model under different shortage penalty coefficient

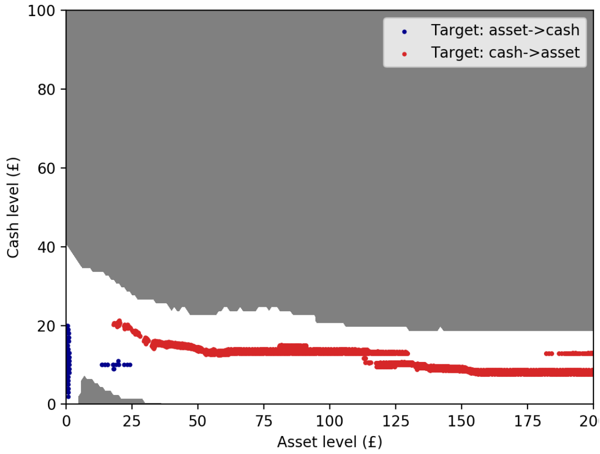
5.6.5 External cash flow

Now we study the cash management policies with different external cash flows. In our models we assume that for each time period, the external cash flow can be described by a normal distributed random variable $\Delta x \sim \mathcal{N}(\mu, \sigma)$. We undertake two sets of numerical experiments to examine the impact of external cash flow's expected value and volatility separately. In the first set of experiments, we let $\mu = (-2.5, -5, -7.5)$ and $\sigma = 5$ and in the second set we let $\mu = -5$ and $\sigma = (1, 5, 10)$. For the parameters we set $K^+ = £2, K^- = £1, k^+ = 0.2, k^- = 0.1, r = 5\%, r_f = 2\%, h = 2$.

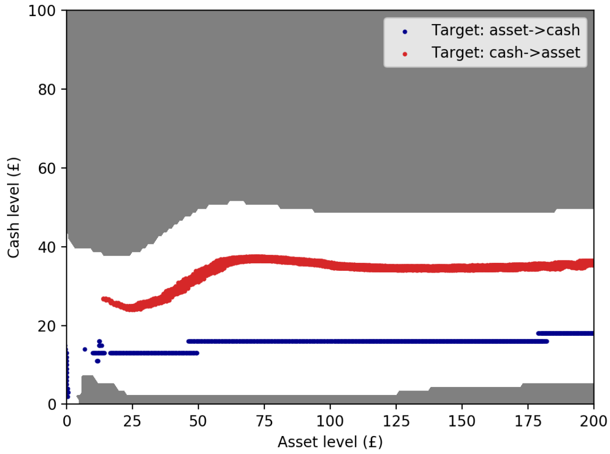
Figure 5.17 shows that both target levels and trigger levels move to a higher position when the expected value of the external cash outflow increases. This means that the agent should hold more cash balance to meet the higher cash demand. Figure 5.18 shows that with more volatility of the external cash outflow, the selling region expands while the buying region shrinks. This can be interpreted as that the agent wishes to adopt a safer policy, i.e. holding more cash and investing less to the risky asset, when the external cash outflow has more stochasticity.

In addition, Figure 5.17 illustrates that in the model with internal cash inflow, generally the agent tends to hold less cash with a larger asset account. This is because large asset generates plenty cash inflow which can be used to offset the cash demand. Note that in these experiments, the agent stops taking selling actions once his asset size is large enough and the cash inflow totally dominates the cash demand. However in these figures we spot a 'bump' of the policy which suggests that when the internal cash inflow approximately matches the external cash outflow, the agent should hold more cash with the increase of asset. Figure 5.17 reveals that the position of the 'bump' changes with the expected value of cash demand and it is always located in the region where the cash inflow approximately matches the demand for cash. Figure 5.18 reveals that the magnitude of this 'bump' increases in the scenarios where the cash outflow has more volatility.

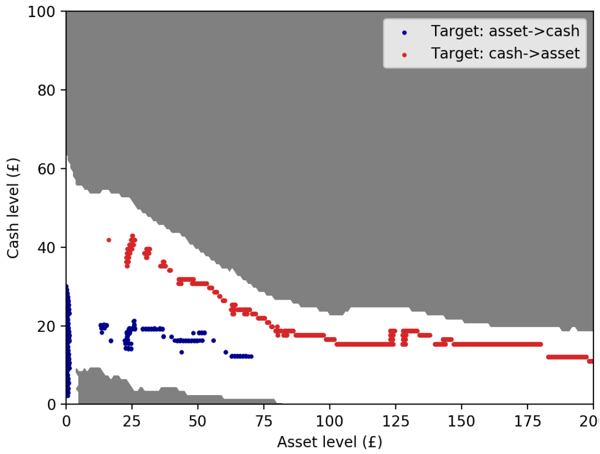
Here we provide one possible explanation for the 'bump' of the cash policy: when the company's asset is insufficient to generate enough cash inflow, the agent needs to replenish his cash balance repeatedly by selling the asset which leads to an even smaller size of the asset account. In this scenario, the company's insolvency



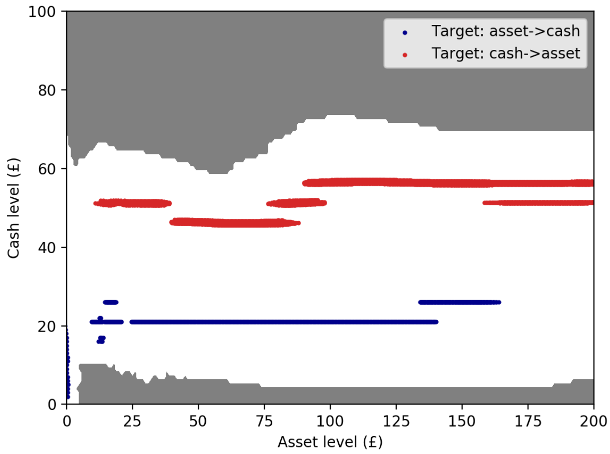
(a) Model with internal cash inflow, $\mu = -2.5$



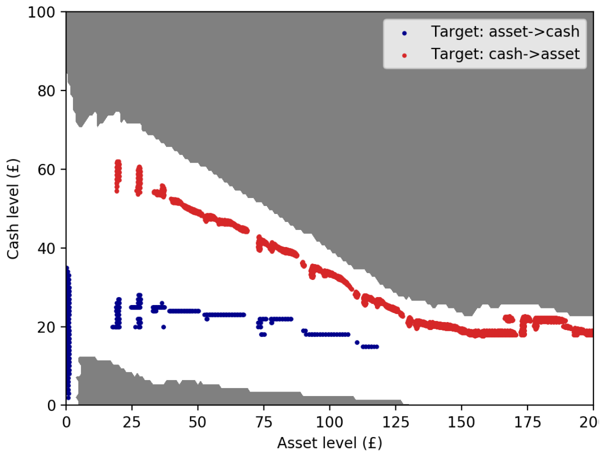
(b) Model with asset growth, $\mu = -2.5$



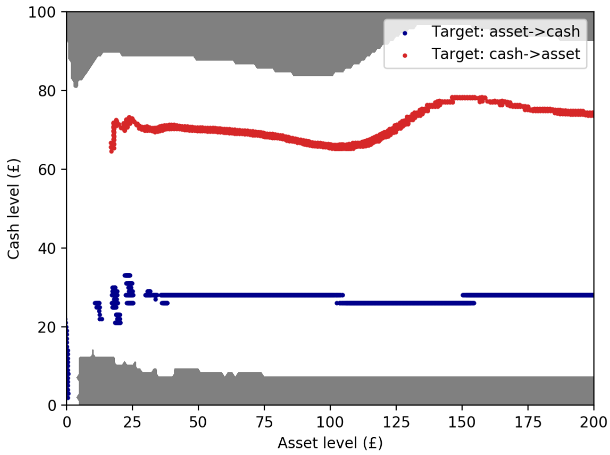
(c) Model with internal cash inflow, $\mu = -5$



(d) Model with asset growth, $\mu = -5$

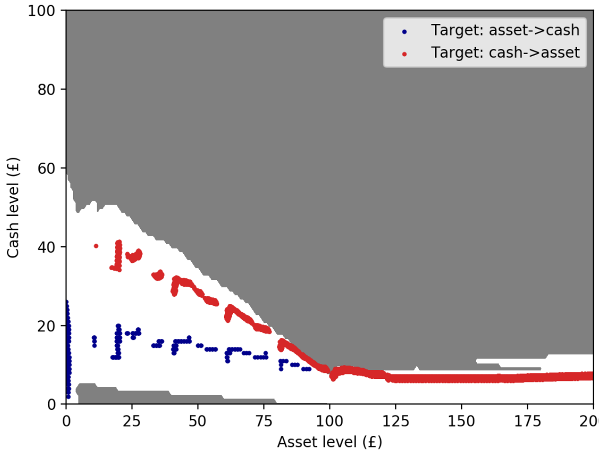


(e) Model with internal cash inflow, $\mu = -7.5$

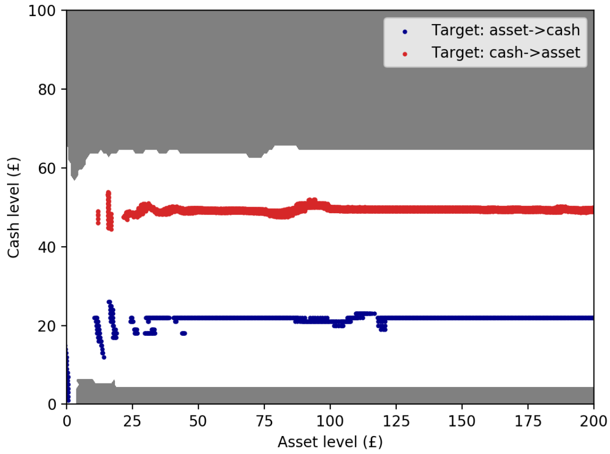


(f) Model with asset growth, $\mu = -7.5$

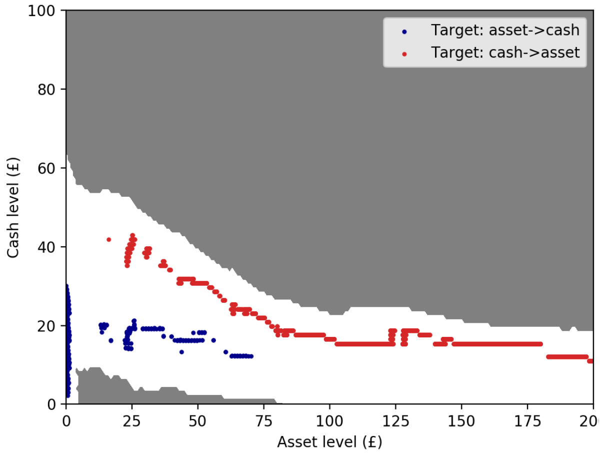
Figure 5.17: Optimal policies in the two accounts model with respect to μ



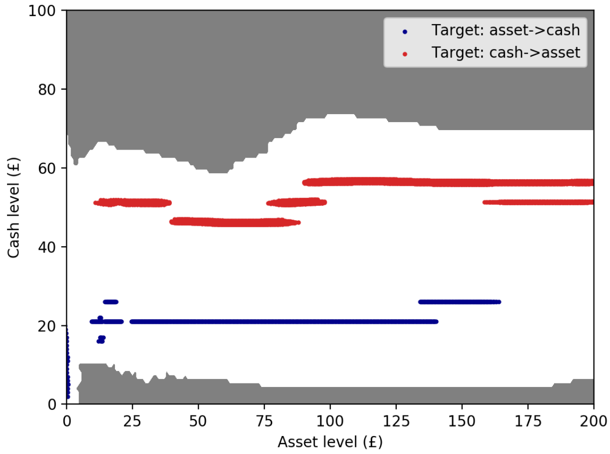
(a) Model with internal cash inflow, $\sigma = 1$



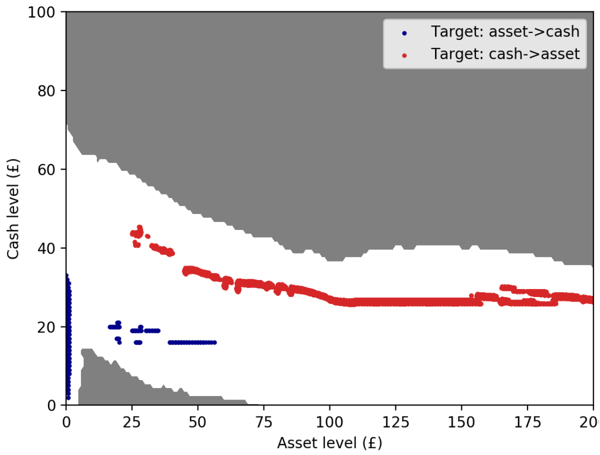
(b) Model with asset growth, $\sigma = 1$



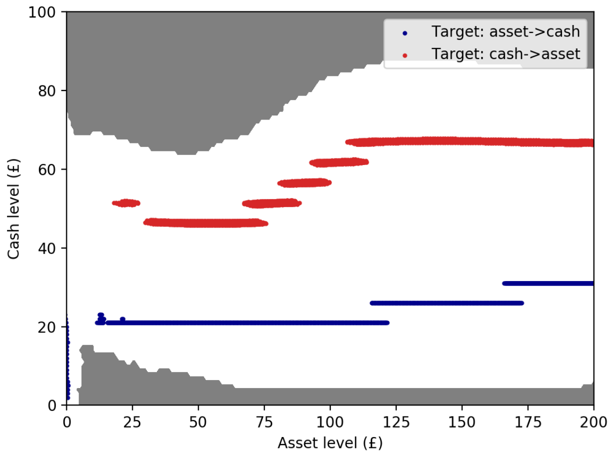
(c) Model with internal cash inflow, $\sigma = 5$



(d) Model with asset growth, $\sigma = 5$



(e) Model with internal cash inflow, $\sigma = 10$



(f) Model with asset growth, $\sigma = 10$

Figure 5.18: Optimal policies in the two accounts model with respect to σ

probability is close to one and this probability can be hardly changed by cash management policies. Considering the company will become bankrupt soon, the agent is inclined to make myopic decisions which is holding just enough cash to meet the current cash demand and investing the rest into the profitable asset. With more asset (but still insufficient to generate the internal cash inflow that dominates the cash demand), the gap between the cash demand and the cash inflow decreases and hence the agent will invest more into assets and harvest the short term gain as much as possible.

However when the asset account is large enough and the cash inflow generated from company's asset approximately matches the cash demand, the company's insolvency probability is highly sensitive to the cash policies. For these states, the agent wishes to adopt a safe policy as the price of cash deficit is not just the shortage penalty paid by the company. It will also jeopardise the company's future profitability and dramatically increase the company's insolvency risk. In these states, with a larger asset account the company's insolvency risk is more sensitive to the cash polices and hence the agent has a higher motivation to adopt a safe policy and hold/replenish more cash balance. This theory is also supported by Figure 5.18 where we show that with more volatility of the cash outflow, the magnitude of the policy 'bump' expands. When the standard deviation of the cash outflow is set to a small value as in Figure 5.18(a), the model has less stochasticity and hence in the states with balanced cash outflow and cash inflow, the agent has low motivation to adopt the safe policy. As a result, the magnitude of the policy 'bump' is rather small. In Figure 5.18(e) where the model has high stochasticity, the policy 'bump' is amplified since the agent wishes to take safe policy to improve his survival probability once the internal cash inflow matches the external cash outflow.

With the increase of company's asset, the cash inflow starts to dominate the cash demand. For each time period, the agent has extra cash to invest and the asset keeps on accumulating. In this scenario, the company's insolvency probability decreases quickly and it is no longer sensitive to the cash policies. Hence the agent gradually loses his motivation to adopt safe policies and start to hold less cash balance. Eventually the company's asset is large enough and for each time period the cash demand is totally dominated by the cash inflow. Once it visits these states, the company has a very small chance to become insolvent. The agent only needs to hold enough cash to avoid the cash deficit for the current time period since the future cash demand can be offset by the future cash inflow.

Figure 5.17 and Figure 5.18 also show the cash management models with asset growth and they have similar patterns with the models with internal cash inflow. That is (a) both cash trigger frontiers and the target cash positions increase with higher expected value or higher standard deviation of the cash outflow, (b) the agent tends to hold/replenish less cash balance with higher asset level when the cash demand outweighs the asset growth or when the asset growth dominates cash demand, (c) the agent tends to hold/replenish more cash balance with higher asset level when the cash demand approximately matches the asset growth and (d) the 'bump' area where the agent tends to hold/replenish more cash balance with higher asset level has a larger magnitude in the scenarios with more volatility.

5.6.6 Transaction Region

We have shown that the optimal cash policy of the two accounts cash management models (including the model with internal cash inflow and the model with asset growth) possess the two-trigger two target form. Hence the company's state space can be divided into three areas: the selling area where the agent should sell his asset and replenish the cash balance, the buying area where the agent should reduce his cash-holding level and invest the extra cash into the asset account, and the doing nothing area where no action should be adopted. In this section, we present the proportion of selling and buying area under different combinations of parameter settings. In terms of transaction cost, we considered three scenarios, namely the low transaction cost, the medium transaction cost and the high transaction cost. The corresponding parameters are reported in Table 5.2. Moreover, we assume the external cash flow can be described by a normal distributed random variable $\mathcal{N}(\mu, \sigma)$. Five settings will be examined in this section, namely $(\mu, \sigma) = (-1, 1)$, $(\mu, \sigma) = (-2.5, 2.5)$, $(\mu, \sigma) = (-5, 5)$, $(\mu, \sigma) = (-7.5, 7.5)$ and $(\mu, \sigma) = (-10, 10)$. We also examine three cash shortage penalty coefficients: $h = 0.2$, $h = 5$ and $h = 8$ and three return rates on the asset account: $r = 0.01$, $r = 0.05$ and $r = 0.1$.

The proportion of the selling area and the buying area under each combination of parameters reported in Table 5.3 and Table 5.4 provides an overall view of the two accounts cash management model. We notice that if other parameters remain the same, with a higher cash shortage penalty coefficient, the proportion of selling area increases while the proportion of buying area decreases. This implies that the agent should hold more cash balance and less asset when the shortage penalty is high. Similarly, the selling area increases and buying area decreases with a lower return rate on asset or a higher cash demand.In other words, when the asset has low profitability or the company faces high cash demand, the agent should keep more cash balance at hand to avoid cash shortage. Moreover we notice that both selling area and buying area are normally smaller in the scenario with a higher transaction cost. This pattern can be interpreted as that the agent should make less transfers between his accounts when the transaction cost is high.

Table 5.3: Proportion of transaction areas: the model with internal cash inflow

			Low transaction cost			Medium transaction cost			High transaction cost		
			h=0.2	h=5	h=8	h=0.2	h=5	h=8	h=0.2	h=5	h=8
$(\mu, \sigma) = (-1, 1)$	$r = 0.01$	Selling area	0.26%	0.49%	0.51%	0.29%	0.41%	0.45%	0.31%	0.37%	0.40%
		Buying area	93.27%	93.10%	93.03%	85.99%	85.76%	85.73%	76.34%	76.26%	76.26%
	$r = 0.05$	Selling area	0.03%	0.04%	0.04%	0.03%	0.04%	0.03%	0.02%	0.03%	0.03%
		Buying area	98.31%	98.21%	98.20%	94.34%	94.36%	94.36%	89.34%	89.30%	89.31%
	$r = 0.1$	Selling area	0.00%	0.01%	0.01%	0.00%	0.01%	0.01%	0.00%	0.00%	0.00%
		Buying area	98.34%	98.31%	98.30%	97.53%	97.55%	97.53%	94.95%	94.94%	94.94%
$(\mu, \sigma) = (-2.5, 2.5)$	$r = 0.01$	Selling area	1.04%	2.15%	2.40%	1.44%	1.98%	1.99%	1.57%	1.87%	1.97%
		Buying area	86.30%	85.33%	85.16%	70.97%	70.44%	70.32%	53.02%	52.59%	52.48%
	$r = 0.05$	Selling area	0.07%	0.19%	0.20%	0.07%	0.16%	0.18%	0.02%	0.07%	0.15%
		Buying area	92.32%	96.67%	96.53%	92.43%	91.78%	91.69%	89.34%	86.89%	86.70%
	$r = 0.1$	Selling area	0.00%	0.05%	0.05%	0.00%	0.05%	0.05%	0.00%	0.03%	0.03%
		Buying area	98.04%	97.94%	97.87%	96.68%	96.51%	96.50%	93.25%	93.07%	93.03%
$(\mu, \sigma) = (-5, 5)$	$r = 0.01$	Selling area	3.13%	5.12%	5.37%	3.33%	4.53%	4.56%	3.48%	4.51%	4.53%
		Buying area	70.50%	68.92%	68.78%	29.83%	28.74%	28.56%	5.88%	5.58%	5.52%
	$r = 0.05$	Selling area	0.21%	0.79%	0.89%	0.23%	0.56%	0.60%	0.19%	0.48%	0.49%
		Buying area	94.52%	98.82%	92.58%	87.86%	86.80%	86.53%	81.05%	80.50%	80.27%
	$r = 0.1$	Selling area	0.02%	0.22%	0.23%	0.00%	0.18%	0.22%	0.00%	0.12%	0.14%
		Buying area	97.42%	96.70%	96.58%	94.64%	93.62%	93.83%	90.05%	89.50%	89.31%
$(\mu, \sigma) = (-7.5, 7.5)$	$r = 0.01$	Selling area	5.25%	8.29%	8.62%	4.82%	7.10%	7.12%	4.66%	7.01%	7.06%
		Buying area	55.35%	53.03%	52.91%	2.54%	2.17%	2.17%	0.00%	0.00%	0.00%
	$r = 0.05$	Selling area	0.45%	1.98%	2.74%	0.37%	1.26%	1.44%	0.28%	1.02%	1.11%
		Buying area	91.82%	88.23%	87.76%	82.24%	80.08%	79.66%	72.73%	71.25%	70.87%
	$r = 0.1$	Selling area	0.079%	0.53%	0.59%	0.01%	0.42%	0.47%	0.00%	0.30%	0.35%
		Buying area	96.66%	94.91%	94.67%	92.37%	90.39%	90.12%	86.40%	85.10%	84.86%
$(\mu, \sigma) = (-10, 10)$	$r = 0.01$	Selling area	7.48%	11.66%	12.01%	6.15%	9.74%	9.81%	5.07%	9.60%	9.70%
		Buying area	41.05%	37.82%	37.73%	0.00%	0.22%	0.27%	0.00%	0.00%	0.00%
	$r = 0.05$	Selling area	0.75%	4.04%	4.69%	0.56%	2.42%	2.81%	0.35%	1.82%	1.93%
		Buying area	88.17%	83.58%	83.02%	75.20%	71.94%	71.39%	61.91%	59.43%	58.84%
	$r = 0.1$	Selling area	0.17%	0.99%	1.11%	0.00%	0.74%	0.84%	0.00%	0.55%	0.63%
		Buying area	95.45%	92.28%	91.91%	89.10%	86.26%	85.95%	82.06%	80.14%	79.76%

Table 5.4: Proportion of transaction areas: the model with asset growth

			Low transaction cost			Medium transaction cost			High transaction cost		
			h=0.2	h=5	h=8	h=0.2	h=5	h=8	h=0.2	h=5	h=8
$(\mu, \sigma) = (-1, 1)$	$r = 0.01$	Selling area	1.95%	1.96%	1.96%	1.95%	1.95%	1.95%	1.93%	1.94%	1.94%
		Buying area	78.26%	78.27%	78.25%	52.35%	52.15%	52.15%	27.77%	27.65%	27.59%
	$r = 0.05$	Selling area	0.99%	1.49%	1.95%	1.03%	1.83%	1.95%	1.77%	1.89%	1.93%
		Buying area	84.23%	84.17%	87.26%	66.11%	66.08%	78.26%	48.73%	48.71%	27.77%
	$r = 0.1$	Selling area	0.59%	1.06%	1.34%	0.96%	1.03%	1.14%	1.04%	1.11%	1.30%
		Buying area	84.96%	84.53%	84.37%	68.03%	67.97%	67.91%	51.33%	51.29%	51.24%
$(\mu, \sigma) = (-2.5, 2.5)$	$r = 0.01$	Selling area	2.84%	3.44%	3.58%	2.91%	3.00%	3.19%	2.93%	3.04%	3.07%
		Buying area	66.49%	65.88%	65.74%	26.45%	25.70%	25.49%	0.00%	0.00%	0.00%
	$r = 0.05$	Selling area	1.52%	2.86%	2.84%	1.96%	2.76%	2.91%	2.03%	2.75%	2.93%
		Buying area	79.29%	78.44%	66.49%	55.28%	54.73%	26.45%	33.29%	33.04%	0.00%
	$r = 0.1$	Selling area	0.61%	2.68%	2.95%	1.75%	2.61%	2.74%	1.92%	2.45%	2.64%
		Buying area	82.06%	81.01%	80.83%	59.77%	59.21%	59.08%	39.12%	38.89%	38.81%
$(\mu, \sigma) = (-5, 5)$	$r = 0.01$	Selling area	4.82%	7.22%	7.44%	5.54%	6.79%	6.82%	5.88%	6.76%	6.83%
		Buying area	50.48%	49.18%	49.05%	0.00%	0.02%	0.02%	0.00%	0.00%	0.00%
	$r = 0.05$	Selling area	1.99%	5.47%	4.82%	2.95%	4.77%	5.54%	2.91%	4.47%	5.88%
		Buying area	74.15%	71.86%	50.48%	43.88%	42.63%	0.00%	19.28%	18.50%	0.00%
	$r = 0.1$	Selling area	0.96%	4.98%	5.58%	1.05%	4.29%	4.85%	2.47%	4.08%	4.38%
		Buying area	78.82%	76.32%	75.99%	52.28%	50.79%	50.50%	29.06%	28.37%	28.24%
$(\mu, \sigma) = (-7.5, 7.5)$	$r = 0.01$	Selling area	7.53%	10.41%	10.72%	7.40%	8.97%	9.04%	7.47%	8.98%	9.08%
		Buying area	37.96%	36.04%	36.01%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%
	$r = 0.05$	Selling area	2.77%	7.86%	7.53%	3.75%	6.45%	7.40%	3.86%	5.81%	7.47%
		Buying area	70.54%	66.66%	37.96%	35.81%	33.63%	0.00%	9.54%	8.08%	0.00%
	$r = 0.1$	Selling area	1.74%	7.39%	8.33%	1.36%	6.32%	6.81%	2.75%	5.76%	6.14%
		Buying area	76.49%	72.92%	72.52%	47.35%	44.87%	44.34%	22.82%	21.53%	21.24%
$(\mu, \sigma) = (-10, 10)$	$r = 0.01$	Selling area	9.83%	13.79%	14.09%	8.85%	11.98%	12.06%	8.67%	12.00%	12.06%
		Buying area	26.92%	23.76%	23.86%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%
	$r = 0.05$	Selling area	3.25%	10.01%	9.83%	3.54%	8.14%	8.85%	5.86%	8.42%	8.67%
		Buying area	66.58%	62.18%	26.92%	28.80%	25.69%	0.00%	3.12%	2.00%	0.00%
	$r = 0.1$	Selling area	2.52%	9.74%	10.83%	1.39%	7.84%	8.64%	1.50%	6.99%	7.51%
		Buying area	73.99%	69.20%	68.70%	42.98%	39.65%	39.08%	17.79%	15.70%	15.32%

5.7 Conclusion

In the two accounts cash management model, we considered an agent who manages a cash account and an asset account and controls his cash balance by selling/buying the assets. We assumed that the profit from the asset is sensitive to the size of the asset account while the external cash demand is stochastically distributed with a constant drift. Moreover we proposed an alternative model where the asset does not generate the profit directly, but its size increases at a certain growth rate. Using the dynamic programming method, we numerically showed that the optimal cash policy for these two dimensional cash management models also possess the two-trigger two-target form. Furthermore, we presented a backward method to calculate the company's insolvency probabilities given its current state. We observed that once the company visits the state where the external cash demand dominates the profit from the asset, the agent has to sell a part of his asset to offset the cash deficit at each period, which will jeopardise the future profitability further. In this scenario, the company goes bankrupt quickly regardless of the manager's cash policies. On the other hand, if the profit significantly outweighs the external cash demand, the manager can reinvest the extra profit after fulfilling the cash demand and hence the asset keeps on accumulating. As a result, the strategies of the cash management are of great importance to the company with a balanced internal cash inflow and external cash outflow since a good cash strategy leads the company to the status with an accumulating asset account. Our study in this chapter showed that the agent should adopt a 'safer' cash policy (i.e. starting the cash replenishment at higher cash holding level and keep more cash at hand) in the balanced states than in other states.

Chapter 6

A Cash Management Model with Loan Opportunities

6.1 Introduction

In Chapter 5 we considered a cash management model with a cash account and an asset account. In this two accounts model, the manager can only replenish his cash holdings by selling a part of his asset. In this chapter we will extend this model to include the opportunity for the agent to take out a loan to supplement his cash balance. Moreover we will present two approaches to solve this cash-asset-loan management problem. In the first approach, we will formulate the problem as a three dimensional Markov decision process where the loan state is considered as an extra dimension in addition to the two accounts model. However the decision of whether to take out a loan or not makes the solution of this extended cash management problem computationally expensive due to the well-known curse of dimensionally. Hence we will also propose an approach (namely the policy improvement heuristic approach) inspired by the policy iteration algorithm proposed by Beranek & Howard (1961). We will show that this heuristic approach reduces the solution time to a large extent while performing strongly in our experiments.

6.2 Problem description and assumptions

In this model, we introduce loan opportunities to the two accounts cash management model with the internal cash inflow. Similar to the two accounts model presented in Chapter 5, we consider an agent who manages a cash account that can be used to fulfil the external cash demand over each period and an asset account that generates the internal cash inflow at the beginning of each period. At the end of each period, a proportional cash shortage penalty occurs with the existence of the cash deficit. In addition we introduce loan opportunities to this model, i.e. when the manager decides to replenish his cash balance, he can sell a part of his asset and/or take a loan from the financial intermediaries. Figure 6.1 shows the cash flows in this model. Once the agent decide to take a loan, the loan expense consisting of both the interest payment and the principal payment will be required in the following periods until the debt is offset. In this model, we consider the loan from the financial intermediaries as a source of the internal cash inflow and the loan expense as a source of the internal cash outflow.

To study this cash/loan management problem, we must first define the structure of the loan opportunity. There are many different types of loan available in the financial market. One of the most common corporate loans provided by banks is the unsecured loan with a fixed interest rate. If the manager takes this loan, the same amount of repayment will be required for each time period until the debt is offset. This loan can be defined by the tuple (Z, L, ι) where Z is the size of the loan, L is the loan age (i.e. how many times of the repayment before the loan is offset) and ι is the loan interest rate. Since the amount of each repayment (ζ) is the same, it can be given by:

$$\zeta = Z\frac{\iota(1+\iota)^L}{(1+\iota)^L - 1}. \tag{6.1}$$

Hence if the agent decides to take this loan opportunity at time t, the internal cash inflow will be increased by Z. And then for the following L periods, the internal cash outflow will be increased by ζ.

For the sake of simplicity, a few assumptions in terms of the loan opportunities will be made before we formulate this problem into a Markov decision process. To

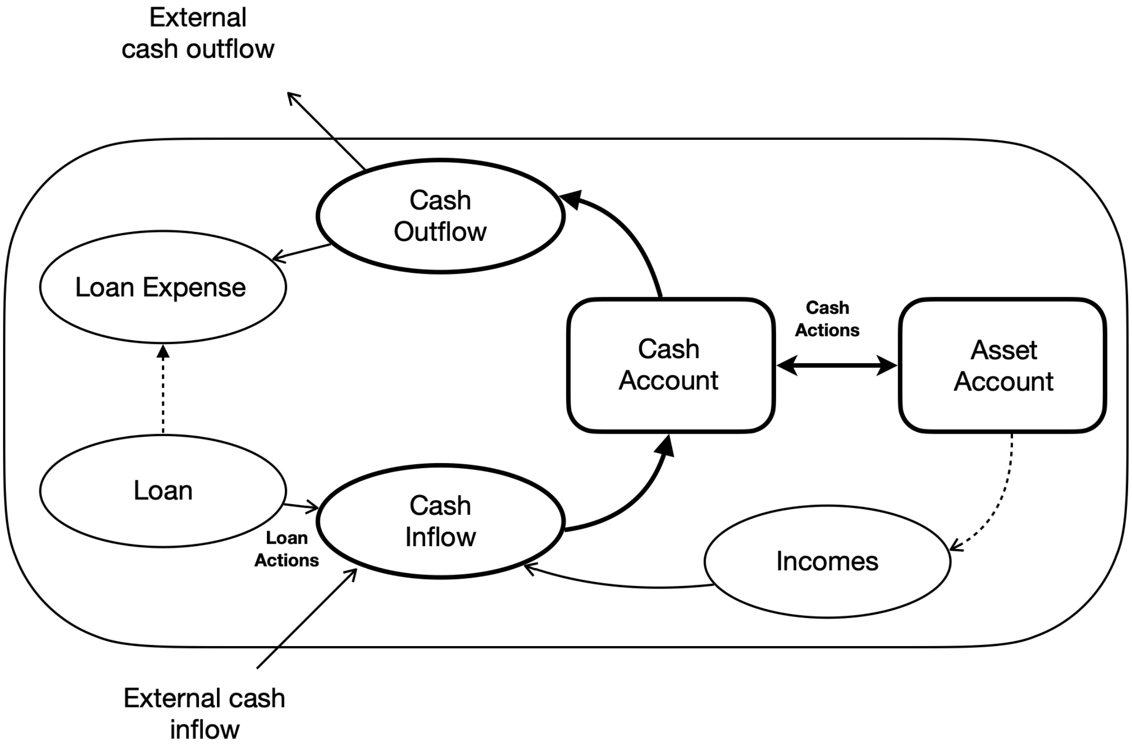


Figure 6.1: Cash flows in the model with loan opportunities

begin with, we assume that there is only one unsecured loan with fixed interest available in the financial market and the company always has access to this loan regardless of its current cash holdings and asset levels. Moreover we assume that after taking the loan from the bank, the agent cannot take another loan until its debt is offset. At last, we discard the consideration of the service charge and the lead time associated with the action of taking a loan.

The timing of each event for one period is shown in Figure 6.2. Consider a company with x_t in the cash account, y_t in the asset account and a debt which remains l periods of repayment. At the beginning of period t, the company receives a mount of internal cash inflow proportional to the size of its asset account. Then if the company has an unpaid loan debt (i.e. $l > 0$), the manager must make a repayment ζ. Otherwise if $l = 0$, the manager can decide whether to take the loan or not. Once the loan is taken, the cash balance increases by Z. The manager can also make a transaction between the cash account and the asset account with the transaction costs which is assumed to be a fixed plus proportional function of the transaction size (see Equation (4.1)). After the external cash flows during the period t, the system transitions to next period if there is no cash shortage.

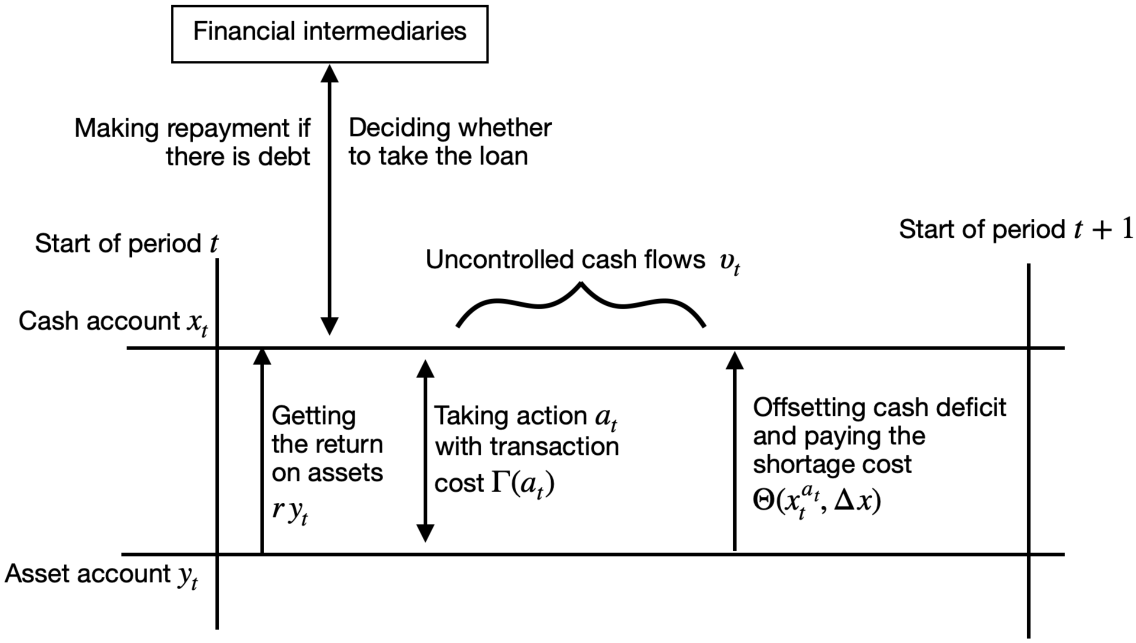


Figure 6.2: Timing of events in the model with loan opportunities

Otherwise, the agent must sell a part of his asset to offset this deficit as well as the cash shortage penalty.

6.3 A discrete Markov decision process approach

We have showed the formulation of the two accounts cash management model into a discrete Markov decision process in Chapter 5. In this section, we develop the model by adding the loan variable into the state space and the loan action into the action space.

Since we assume there is only one type of loan and the loan expense for each period is the same, the variable l is enough to describe the company's debt situation. Note that l denotes the remaining times of the repayment including the current repayment. Similar to Section 5.3, we discretise the cash level space into the set $\mathbf{S}^x = \left\{s^x_{[1]}, ..., s^x_{[M]}\right\}$ and the asset level space into the set $\mathbf{S}^y = \left\{s^y_{[1]}, ..., s^y_{[M']}\right\}$. The loan space can be written as $\mathbf{S}^l = \{0, 1, ..., L\}$. Assuming that at period t, the company's state is $\vec{s}_t = (s^x_t, s^y_t, l_t)$ such that $(s^x_t, s^y_t, l_t) \in \mathbf{S}^x \times \mathbf{S}^y \times \mathbf{S}^l$. The company first receives the internal cash inflow proportional to the size of its asset account and makes a repayment of the loan if it has an unpaid debt. After this repayment, the remaining repayment times reduces by one.

At each decision epoch, we assume that the agent takes a loan action before the cash control action. Hence it is possible for the manager to take the loan and invest a part of the loan into his asset. Let $a^l = 1$ denotes the action of taking the loan and $a^l = 0$ denotes not taking the loan. Since we assume that the company has no access to the loan opportunity if it has an unpaid debt, the space of loan action can be written as:

$$\mathbf{A}^l(l) = \begin{cases} \{0\} & \text{if} \quad l > 0 \\ \{0, 1\} & \text{if} \quad l = 0 \end{cases}. \tag{6.2}$$

After the loan action, the system transitions to the post-loan-decision state:

$$\begin{bmatrix} x_t^{a_t^l} \\ y_t^{a_t^l} \\ l_t^{a_t^l} \end{bmatrix} = \begin{bmatrix} s_t^x + r s_t^y - \zeta \cdot 1_{\{l_t > 0\}} + Z \cdot 1_{\{a_t^l = 1\}} \\ s_t^y \\ (l_t - 1) \cdot 1_{\{l_t > 0\}} + L \cdot 1_{\{a_t^l = 1\}} \end{bmatrix}.$$

Let $\Gamma(a_t)$ be the fixed plus proportional transaction cost function with parameters (K^+, K^-, k^+, k^-) (see Equation (4.1)). With the knowledge of the post-loan-decision state, we can define the feasible cash action space $a_t \in \mathbf{A}_t(x_t^{a_t^l}, y_t^{a_t^l})$. Similar to the previous models, we assume that the post-cash-decision cash/asset levels cannot exceed the maximum cash/asset boundary or fall short of the minimum boundary. The cash action space $A(x_t^{a_t^l}, y_t^{a_t^l})$ can be obtained via Algorithm (2) with the Equation (5.9) replaced by:

$$a_{[m]} = \max \left\{ \frac{s_{[1]}^x - x_t^{a_t^l} + K_b}{1 + k_b}, y_t^L - s_{[K]}^y \right\}.$$

After the cash action a^t, the system transitions to the post-decision state:

$$\begin{bmatrix} x_t^{a_t} \\ y_t^{a_t} \\ l_t^{a_t} \end{bmatrix} = \begin{bmatrix} x_t^{a_t^l} + a_t - \Gamma(a_t) \cdot 1_{\{a_t \leq 0\}} \\ y_t^{a_t^l} - a_t - \Gamma(a_t) \cdot 1_{\{a_t > 0\}} \\ l_t^{a_t^l} \end{bmatrix}.$$

Assume that the external cash flow υ_t can be approximated by Wiener process with parameters (μ, σ) and Δx_t is the change of the uncontrolled cash flow at period t. The cash balance at the end of this period is $x_t^{a_t} + \Delta x_t$ while the asset level and the loan state remain unchanged. If the cash balance is non-negative (i.e. $x_t^{a_t} + \Delta x_t \geq 0$), the system transitions to the period $t+1$ with no cash shortage cost:

$$\begin{bmatrix} x_{t+1} \\ y_{t+1} \\ l_{t+1} \end{bmatrix} = \begin{bmatrix} x_t^{a_t} + \Delta x_t \\ y_t^{a_t} \\ l_t^{a_t} \end{bmatrix}.$$

Otherwise the agent must sell a part of his asset to offset this cash deficit $d_t = |x_t^{a_t} + \Delta x_t|$ as well as the cash shortage penalty $\Theta(x_t^{a_t} \Delta x_t)$. The proportional cash shortage function $\Theta(x_t^a, \Delta x_t)$ is shown in Equation (5.3). With the cash deficit d_t, the system transitions to:

$$\begin{bmatrix} x_{t+1} \\ y_{t+1} \\ l_{t+1} \end{bmatrix} = \begin{bmatrix} 0 \\ y_t^{a_t} - d_t - \Gamma(d_t) - \Theta(x_t^{a_t}) \\ l_t^{a_t} \end{bmatrix}.$$

Considering the loan as a source of the internal cash inflow and the repayment as a source of the internal cash outflow, the expected net income for each period given the current state $\vec{s}_t = (s_t^x, s_t^y, l_t)$ and the action $\vec{a}_t = \left(a_t^l, a_t\right)$ can be written as:

$$\begin{aligned} R_t(\vec{s}_t, \vec{a}_t) &= \mathbb{E}\left\{ rs^y - \zeta \cdot 1_{\{l_t>0\}} + Z \cdot 1_{\{a_t^l=1\}} - \Gamma(a_t) - \Theta(x_t^{a_t}, \Delta x_t) \right\} \\ &= rs^y - \zeta \cdot 1_{\{l_t>0\}} + z \cdot 1_{\{a_t^l=1\}} - \Gamma(a_t) - \textstyle\int_{-\infty}^{\infty} \Theta(x_t^{a_t}, \Delta x_t) \mathrm{d}\Delta x_t \end{aligned}$$

where Δx is the total change of the cash balance under the influence of the external cash flow and $f(\Delta x)$ is the relative probability density function.

Let the r_f be the risk-free interest rate and hence the future income will be discounted by $\gamma = 1/(1+r_f)^{t-1}$. The goal of this model is to maximise the expected discounted total income over an infinite horizon, i.e.

$$\max_{\pi \in \Pi} \mathbb{E}\left\{ \sum_{t=1}^{\infty} \gamma^{t-1} r y_t - \zeta \cdot 1_{\{l_t>0\}} + Z \cdot 1_{\{a_t^l=1\}} - \Gamma(a_t) - \Theta(x_t^{a_t}, \Delta x) \right\}.$$

Now we can write the Bellman equation for this cash management model with loan opportunities. Let $V_t(\vec{s}_t, \vec{a}_t)$ be the value of the system visiting the state

$\vec{s}_t = (s_t^x, s_t^y, l_t)$ at period t and the manager adopts the loan action a_t^l and the cash action a_t. For each period, we have:

$$\begin{aligned} &V_t(\vec{s}_t, \vec{a}_t) = r s_t^y - \zeta \cdot 1_{\{l_t > 0\}} + Z \cdot 1_{\{a_t^l = 1\}} - \Gamma(a_t) \\ &+ \int_{-\infty}^{\infty} \left(-\Theta(x_t^{a_t}, \Delta x_t) + \gamma \hat{V}_{t+1} \left(x_{t+1}, y_{t+1}, l_{t+1} | a_t^l, a_t, \Delta x_t \right) \right) \mathrm{d}\Delta x_t. \end{aligned}$$

This equation can be solved via the classic backward recursion method where the value for each state at period t is calculated after the value at period $t+1$. Similar to the equation (4.11) and (5.11), we have

$$\begin{aligned} &V_{(i)}(\vec{s}_t, \vec{a}_t) = r s_{(i)}^y - \zeta \cdot 1_{\{l_{(i)} > 0\}} + Z \cdot 1_{\{a_{(i)}^l = 1\}} - \Gamma(a_{(i)}) \\ &+ \int_{-\infty}^{\infty} \left(-\Theta(x_{(i)}^{a_{(i)}}, \Delta x_{(i)}) + \gamma \hat{V}_{(i-1)} \left(x_{(i-1)}, y_{(i-1)}, l_{(i-1)} | a_{(i)}^l, a_{(i)}, \Delta x_{(i)} \right) \right) \mathrm{d}\Delta x_{(i)}. \end{aligned} \tag{6.3}$$

At the initial iteration we set $V_{(0)}(\vec{s}_{(0)}) = 0$. Then we keep updating the states value backwards until the maximum change of the values between each iterations is less than a small number ϵ.

Similar to the previous models, we use the bilinear interpolation method to approximate the value $\hat{V}_{(i-1)}(x_{(i-1)}, y_{(i-1)}, l_{(i-1)})$ based on its adjacent states and use the Gauss-Hermite quadrature method to approximate the value of the integral term in Equation (6.3) (see Equation (5.12) and Equation (5.14)).

Due to the limited computational resource we only study the scenario where the company has access to one type of loan. However, this dynamic model can be easily formulated as a cash management model with multiple loans. Consider a company having access to Ξ different fixed-interest loans, i.e. $\boldsymbol{\Xi} = \{(Z_1, L_1, \iota_1), ..., (Z_\xi, L_\xi, \iota_\xi)\}$. At time t, the company's state is $\vec{s}_t = (s_t^x, s_t^y, l_t^1, ..., l_t^\xi)$ where l_t^ξ represents the remaining repayment times of the ξ^{th} loan. The action adopted by the manager at period t can be described as $(a_t^1, ..., a_t^\Xi, a_t)$ where a_t^ξ denotes whether taking the ξ^{th} loan or not and a_t denotes the transfer between the cash balance and the asset account. After the manager taking the action, the cash state transitions to $x_t^{a_t} = s_t^x + r s_t^y + \sum_{\xi=1}^{\Xi} \left(-\zeta^\xi \cdot 1_{\{l_t^\xi > 0\}} + Z_\xi \cdot 1_{a_t^\xi = 1} \right) + a_t - \Gamma(a_t) \cdot 1_{\{a_t \leq 0\}}$, the asset state transitions to $y_t^{a_t} = s_t^y - a_t - \Gamma(a_t) \cdot 1_{\{a_t > 0\}}$ and each loan state transitions to $l_t^{a_t, \xi} = (l_t^\xi - 1) \cdot 1_{l_t^\xi > 0} + L \cdot 1_{a_t^\xi = 1}$ for $\xi = \{1, ..., \Xi\}$. The objective function should be modified as:

$$\max_{\pi\in\Pi}\mathbb{E}\left\{\sum_{t=1}^{\infty}\gamma^{t-1}\left(ry_t+\sum_{\xi=1}^{\Xi}\left(-\zeta_t^{\xi}\cdot 1_{\{l_t^{\xi}>0\}}+Z_{\xi}\cdot 1_{\left\{a_t^{\xi}=1\right\}}\right)-\Gamma(a_t)-\Theta(x_t^{a_t},\Delta x_t)\right)\right\}.$$

In the above equation, ry_t, $\Gamma(a_t)$ and $\Theta(x_t^{a_t}, \Delta x_t)$ represent the incomes generated from the asset, the transaction cost and the cash shortage penalty correspondingly. The term $\sum_{\xi=1}^{\Xi}\left(-\zeta_t^{\xi}\cdot 1_{\{l_t^{\xi}>0\}}+Z_{\xi}\cdot 1_{\left\{a_t^{\xi}=1\right\}}\right)$ represents the expenses on the loan-taking action.

6.4 A policy improvement heuristic approach

In the last section, we developed the two accounts cash management model to a model with loan opportunities by adding an extra loan dimension to the state space. However due to the well-known curse of dimensionality, the computational cost explodes exponentially. In this section we present an iterative approach based on policy improvement (PI) that provides solutions close to the MDP approach within a reasonable solution time.

The main idea of the policy improvement heuristic approach can be described as follows. We first find the optimal cash management policy assuming no loan is available on the financial market. Then we improve the cash-loan policies by adding one more loan offer in the rest of the planning horizon. We determine if the agent should take up the loan offer using a single policy improvement step. We repeat this process until adding one more loan in the horizon does not improve the system's state values.

In this approach, instead of introducing the loan dimension to the state space, we solve a loan-decision problem based on the two accounts cash management model. In the cash-asset-loan model, we assume that at time t, the manager has a cash account s_t^x and an asset account s_t^y and he can decide whether to take the fixed-interest loan (Z, L, ι) or not. If the loan is taken, the internal cash inflow will increase by Z at this time period and the internal cash outflow will increase by ζ for the next L periods. Then at period $t+L+1$ the manager can decide whether to take the loan again. Let $V_t^l(\vec{s}_t)$ be the maximum value of the company

$\vec{s}_t = (s_t^x, s_t^y)$ with l times of repayments left. The objective is to decide whether to take the loan or not when $l = 0$ as well as finding the corresponding cash action. The value of the company visiting state $\vec{s} = (s_t^x, s_t^y)$ with no debt and the agent taking the loan can be expressed as:

$$\begin{aligned}
V_t^0(\vec{s}_t|a_t^l = 1) \quad &= \max_{a_t} \quad rs_t^y + z - \Gamma(a_t) \\
&+ \int_{-\infty}^{\infty} \left(-\Theta(x_t^{a_t,\Delta x_t}) + \gamma V_{t+1}^L(x_{t+1}, y_{t+1})|\Delta x_t\right) \mathrm{d}\Delta x_t \\
V_{t+1}^L(\vec{s}_{t+1}) \quad &= \max_{a_{t+1}} \quad rs_{t+1}^y - \zeta - \Gamma(a_{t+1}) \\
&+ \int_{-\infty}^{\infty} \left(-\Theta(x_{t+1}^{a_{t+1}}, \Delta x_{t+1}) + \gamma V_{t+2}^{L-1}(x_{t+2}, y_{t+2})|\Delta x_{t+1}\right) \mathrm{d}\Delta x_t \\
&\dots \\
V_{t+L}^1(\vec{s}_{t+L}) \quad &= \max_{a_{t+L}} \quad rs_{t+L}^y - \zeta - \Gamma(a_{t+L}) \\
&+ \int_{-\infty}^{\infty} \left(-\Theta(x_{t+L}^{a_{t+L}}, \Delta x_{t+L}) + \gamma V_{t+L+1}(x_{t+L+1}, y_{t+L+1})|\Delta x_{t+L}\right) \mathrm{d}\Delta x_t.
\end{aligned}$$

If the company has no debt outstanding and the manager decides not to take this loan, the value of visiting state $\vec{s}_t = (s_t^x, s_t^y)$ at time t can be expressed as:

$$\begin{aligned}
V_t^0(\vec{s}_t|a_t^l = 0) \quad &= \max_{a_t} \{rs_t^y - \Gamma(a_t) \\
&+ \int_{-\infty}^{\infty} \left(-\Theta(x_t^{a_t}, \Delta x_t) + \gamma V_{t+1}^0(x_{t+1}, y_{t+1})|\Delta x_t\right) \mathrm{d}\Delta x_t \Big\}.
\end{aligned}$$

Now the maximum value of visiting state $\vec{s}_t = (s_t^x, s_t^y)$ with no debt outstanding can be written as:

$$V_t^0(\vec{s}_t) = \max\left\{V_t^0(\vec{s}_t|a_t^l = 1), V_t^0(\vec{s}_t|a_t^l = 0)\right\}. \tag{6.4}$$

To solve Equation (6.4), we introduce an auxiliary problem, namely the cash management problem with limited loan offers. In this problem, we assume that at time t, if the company has no debt outstanding, the bank offers the loan (Z, L, ι) to the company. Moreover we assume that such opportunity will be offered κ times including this offer in the rest of the horizon independent of the manager's loan decision at period t. Let $W_t^{l,\kappa}(\vec{s}_t)$ represents the maximum value of the company visiting state $\vec{s}_t$ with l times of repayments left and the manager having access to this loan for κ times in the rest of the horizon. In the original cash-asset-loan model, we assume that without any unpaid debt, the manager always has access to this loan, i.e. $V_t^0(\vec{s}_t) = W_t^{0,\infty}(\vec{s}_t)$. In addition, the two accounts cash management model presented in Chapter 5 can be interpreted as the model where the bank will offer the loan to the company for 0 times in the rest of the horizon. Hence the value $W_t^{0,0}(\vec{s}_t)$ can be obtained via Algorithm (3). Now the value $W_t^{0,\kappa}(\vec{s}_t)$ for each κ can be calculated backwards:

$$
\begin{cases}
W_t^{0,\kappa}(\vec{s}_t) & = \max\left\{W_t^{0,\kappa}(\vec{s}_t|a_t^l=1), W_t^{0,\kappa}(\vec{s}_t|a_t^l=0)\right\} \\
W_t^{0,\kappa}(\vec{s}_t|a_t^l=0) & = \max_{a_t} \quad rs_t^y - \Gamma(a_t) \\
& + \int_{-\infty}^{\infty} \left(-\Theta(x_t^{a_t}, \Delta x_t) + \gamma W_{t+1}^{0,\kappa-1}(x_{t+1}, y_{t+1})|\Delta x_t\right) \mathrm{d}\Delta x_t \\
W_t^{0,\kappa}(\vec{s}_t|a_t^l=1) & = \max_{a_t} \quad rs_t^y + Z - \Gamma(a_t) \\
& + \int_{-\infty}^{\infty} \left(-\Theta(x_t^{a_t}, \Delta x_t) + \gamma W_{t+1}^{L,\kappa}(x_{t+1}, y_{t+1})|\Delta x_t\right) \mathrm{d}\Delta x_t \\
W_{t+1}^{L,\kappa}(\vec{s}_{t+1}) & = \max_{a_{t+1}} \quad rs_{t+1}^y - \zeta - \Gamma(a_{t+1}) \\
& + \int_{-\infty}^{\infty} \left(-\Theta(x_{t+1}^{a_{t+1}}, \Delta x_{t+1}) + \gamma W_{t+2}^{L-1,\kappa}(x_{t+2}, y_{t+2})|\Delta x_{t+1}\right) \mathrm{d}\Delta x_{t+1} \\
W_{t+2}^{L-1,\kappa}(\vec{s}_{t+2}) & = \max_{a_{t+2}} \quad rs_{t+2}^y - \zeta - \Gamma(a_{t+2}) \\
& + \int_{-\infty}^{\infty} \left(-\Theta(x_{t+2}^{a_{t+2}}, \Delta x_{t+2}) + \gamma W_{t+3}^{L-2,\kappa}(x_{t+3}, y_{t+3})|\Delta x_{t+2}\right) \mathrm{d}\Delta x_{t+2} \\
& \ldots \\
W_{t+L}^{1,\kappa}(\vec{s}_{t+L}) & = \max_{a_{t+L}} \quad rs_{t+L}^y - \zeta - \Gamma(a_{t+L}) \\
& + \int_{-\infty}^{\infty} \left(-\Theta(x_{t+L}^{a_{t+L}}, \Delta x_{t+L}) + \gamma W_{t+L+1}^{0,\kappa-1}(x_{t+L+1}, y_{t+L+1})|\Delta x_{t+L}\right) \mathrm{d}\Delta x_{t+L}
\end{cases}
$$

6.5 Numerical experiments

In this section, we conduct numerical experiments to study the performance of the policy improvement heuristic approach in comparison with the classic Markov decision process approach. Then we examine the impacts of loan conditions including the loan interest rate, the loan age and the loan size on the agent's loan action as well as the state values in the cash-loan-asset model. All the experiments are programmed in C++ 12.0.0 on a PC with 2.5 GHz Quad-Core Intel Core i7 and 8 GB memory.

6.5.1 Performance of the policy improvement heuristic approach

We adopt the settings of the external cash flow and the operational conditions proposed in Section 5.3.2. We let the external cash outflow for each period be a normally distributed random variable $\Delta x_t \sim \mathcal{N}(\mu = -£5, \sigma = 5)$ and adopt the fixed plus proportional transaction function with parameters $(K^+, K^-, k^+, k^-) = (£2, £1, 0.2, 0.1)$. Moreover we set the proportional shortage penalty coefficient to $h = 2$ and the return rate on the asset to $r = 0.05$. We also set the risk-free interest rate to $r_f = 0.02$, hence the discount factor can be expressed as $\gamma = \frac{1}{1+r_f} = 98.04\%$. In terms of the loan opportunities in the financial market, we set the loan interest rate to $\iota = 0.04$, the loan age to $L = 10$ and the loan size to $Z = £20$. According to Equation (6.1), once the manager takes the £20 loan, he will make a repayment of £2.47 for the next 10 periods. In the numerical experiments, we focus on the loan/cash policy in states $\vec{s} \in S^x \times S^y = [0, 100] \times [0, 200]$. However, as discussed in Section 5.3, we set the capacity of the asset account to £400 to alleviate the impact of the asset boundary on the loan/cash policy.

Figure 6.3 shows the cash policy and the loan policy in the MDP approach and the policy improvement heuristic (PIH) approach when the company has no debt outstanding (i.e. $l = 0$). It can be seen in Figure 6.3(a) that the buying policy remains the trigger-target form, i.e. when the cash balance reaches the upper grey area, namely the trigger frontier, the agent should invest his cash into the asset and adjust the cash balance back to a lower level. Moreover when the company visits a state with a low cash balance and a low asset level, the company is going bankrupt in a few periods. In this scenario, it is optimal for the agent to sell all his asset and use the cash to fulfil the cash demand as much as possible. Figure 6.3(b) shows the optimal loan policy when $l = 0$. The manager should take the loan if the company visits states in the grey area and should renounce this loan opportunity if it is in the white area. This figure suggests that when the company has a low cash balance and a high asset level, the agent should replenish his cash account by taking the loan instead of selling his asset. In addition, since the return rate on asset is higher than the loan interest rate, the company can benefit from the action of taking a loan and investing it into the asset immediately if it has both a sufficient cash level and a large asset level.

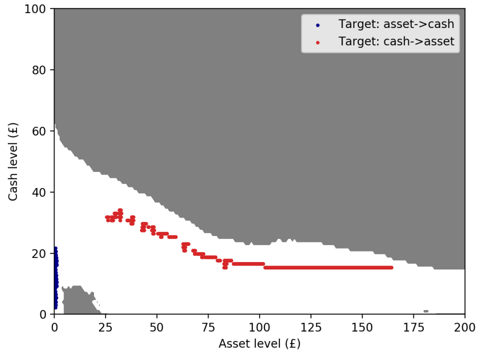
Figures 6.3(c)-(h) show the cash and loan polices via the policy improvement heuristic approach after 1 iterations, 10 iterations and 20 iterations. Note that the cash/loan policy suggested by this heuristic approach after i^{th} iteration can be interpreted as the optimal cash/loan policy adopted by the manager knowing that only i loan offers will be given in the whole planning horizon. With the increase of the iteration number i, the policy from the policy improvement heuristic approach is approximating to the optimal policy given by the MDP method. Figure 6.3 reveals that both the policy from the policy improvement heuristic approach after 10 iterations and the policy after 20 iterations resembles the optimal policy suggested by the MDP approach to a large extent. Note that the MDP approach in this cash-asset-loan model requires an extra loan dimension in addition to the two accounts cash management model and it takes 7.38 hours to solve this numerical experiment. In comparison, the policies from the PIH approach after 10 iterations and 20 iterations only takes 0.48 hours and 0.53 hours respectively (including the solution time for the two accounts cash management model).

In Figure 6.4, the average value of all states in the state space $(0, 100) \times (0, 200)$ is plotted against the iteration number in the PIH approach. The optimal average state values (given by the MDP approach) of the model with and without the loan opportunity are also plotted as benchmarks. It can be seen that the optimal average state value without the loan opportunity is £334.162. Once the company has access to the loan $(Z, L, \iota) = (£20, 10, 0.04)$, the optimal average state value can be improved by 3.45%. Figure 6.4 reveals that using the PIH approach, the average state value can be improved by 2.40% after 10 iterations and 3.05% after 20 iterations. After 50 iterations, the percentage difference of state values between the PIH approach and the MDP approach is around 0.027%.

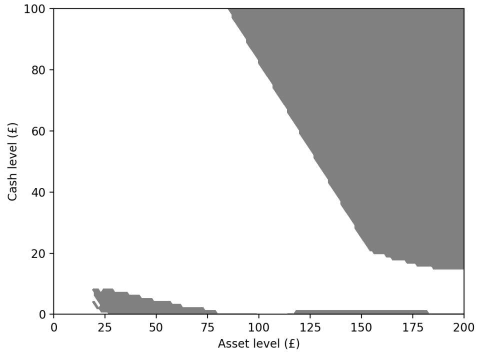
6.5.2 Loan conditions

We now examine the impact of the loan conditions (namely the loan interest rate, the loan age and the loan size) on the loan policy as well as the state values in the cash-asset-loan model.

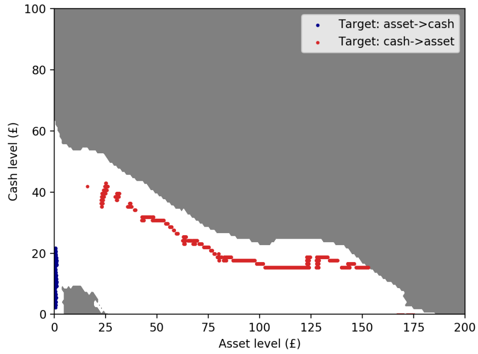
To begin with, we assume the loan provided by the financial intermediaries is £20 and must be repaid in next 10 periods. Figure 6.5 shows the loan policies suggested by the MDP approach and the PIH approach (20 iterations) given three different loan interest rates ($\iota = (0.01, 0.04, 0.08)$). As shown in this figure, if the interest rate is too high (e.g. $\iota = 8\%$), the agent should renounce this loan opportunity regardless to its current state. With a lower interest rate, the loan-



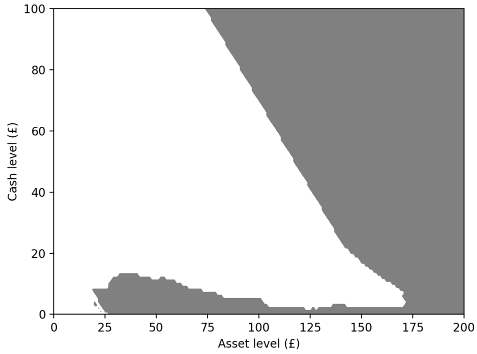
(a) Cash policy (MDP)



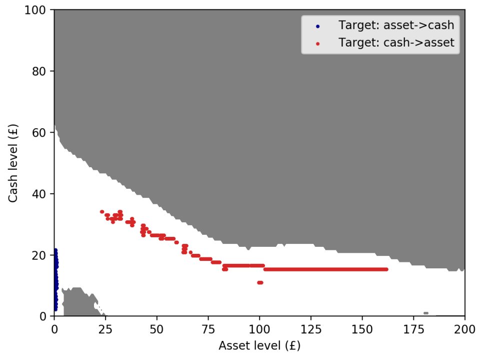
(b) Loan policy (MDP)



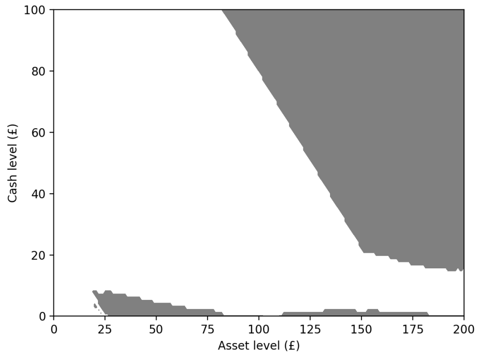
(c) Cash policy (PIH, the 1st iteration)



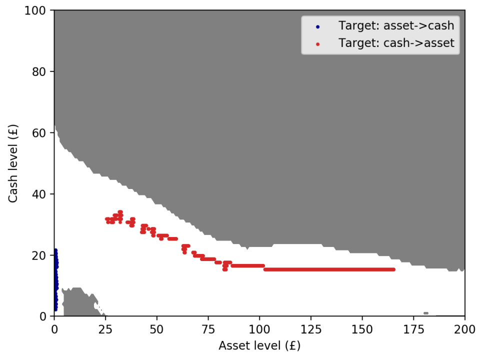
(d) Loan policy (PIH, the 1st iteration)



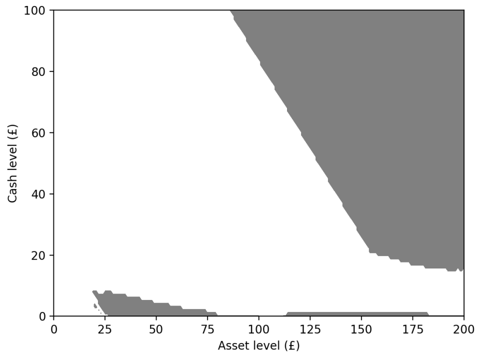
(e) Cash policy (PIH, the 10th iteration)



(f) Loan policy (PIH, the 10th iteration)



(g) Cash policy (PIH, the 20th iteration)



(h) Loan policy (PIH, the 20th iteration)

Figure 6.3: Cash and loan policies from the MDP approach and the policy improvement heuristic approach

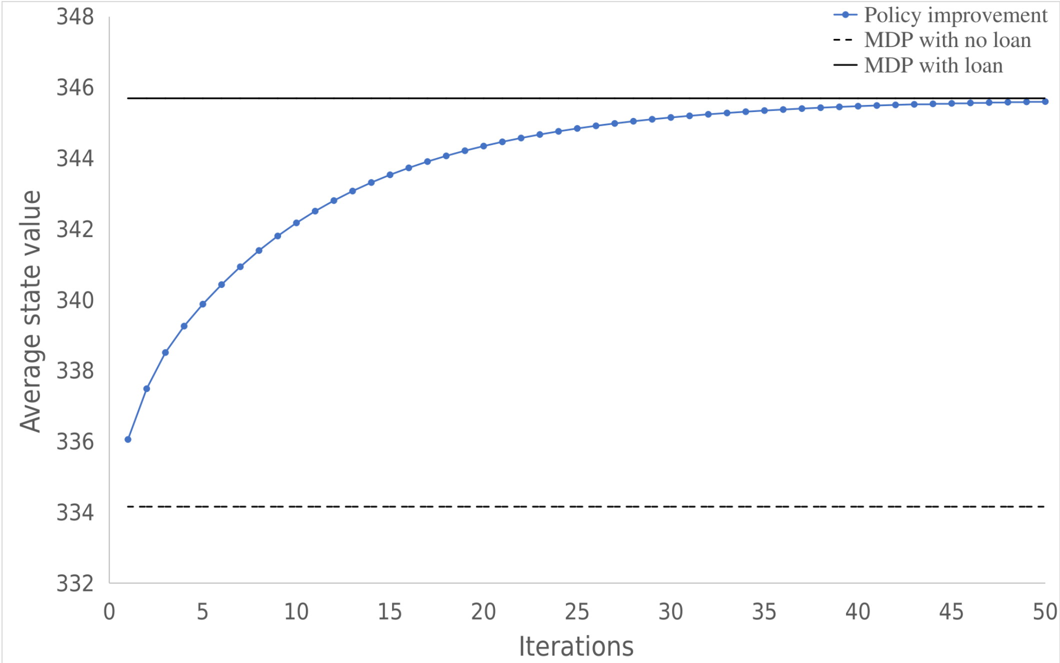


Figure 6.4: The average state value from the MDP approach and the PIH approach

taking area expands. Figure 6.6 reveals the relationship between the average value of all states in the space $(0, 100) \times (0, 200)$ and the loan interest rate. It shows that the average state value drops with the increase of the loan interest rate. If the loan interest rate is higher than 7%, the average value between the model without loan opportunity and the cash-asset-loan model is almost the same since the optimal loan policy in this scenario is not taking the loan in any state. Figure 6.6 also shows that after 20 iterations, the PIH approach is a good approximation to the MDP approach given any loan interest rate.

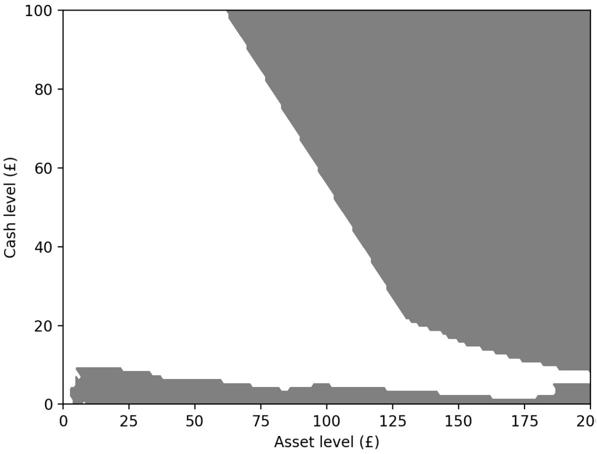
To study the impact of the loan age on the loan policy, we set the loan size to $Z = £20$, the loan interest rate to $\iota = 0.04$ and experiments on different loan ages. Figure 6.7 shows the loan policies suggested by the MDP approach and the PIH approach (20 iterations) when the loan must be paid in $L = 4$, $L = 10$ and $L = 18$ periods. In Figure 6.8, the average values of all states via different approaches (namely the MDP approach, the PIH approach after 10 iterations and the PIH approach after 20 iterations) are plotted against the loan age. As shown in Figure 6.7 and Figure 6.8, with a longer loan age, the loan-taken area expands and the average state value increases monotonically. This is because a longer loan age resulting in a lower loan repayment for each period. For the following L

periods, the company will face a lower total cash outflow and thus need a lower cash balance. As a result, the company can invest more into the profitable asset and has a lower cash shortage risk.

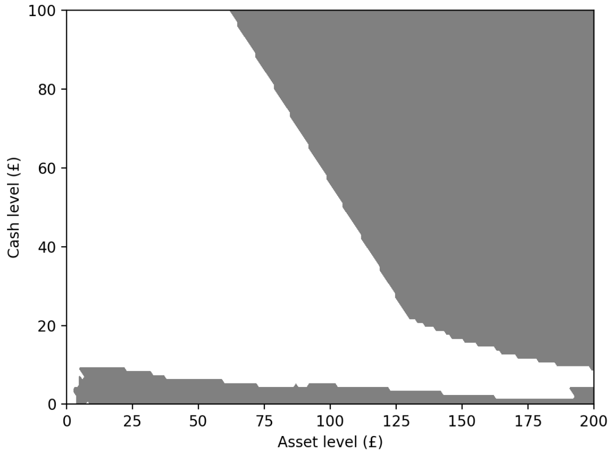
At last we fix the loan interest rate to $\iota = 0.04$, the loan age to $L = 10$ and experiment on different loan sizes. The loan policy obtained from the MDP approach and the PIH approach (20 iterations) is shown in Figure 6.9 and the average state values against the loan age is plotted in Figure 6.10. We observe that the average state value does not always increase with the loan size. If the loan size is small, the cash balance cannot be replenished by the loan. Hence the manager needs to sell a part of his asset. On the other hand, if the manager takes the loan with a large size, he will have too much cash holdings and should invest a part of this loan into his asset. In both scenarios, an extra amount of transaction cost occurs and results in a decrease of the state values. In the numerical studies experimenting on $Z = (8, 12, 16, 20, 24, 28, 32, 36)$, according to the state values obtained via the MDP approach, the average state value reaches maximum (£345.7) when the loan size is set to £16 or £20. Figure 6.10 also shows that for each loan size, the PIH approach after 20 iterations provides a close approximation to the MDP approach.

6.6 Conclusion

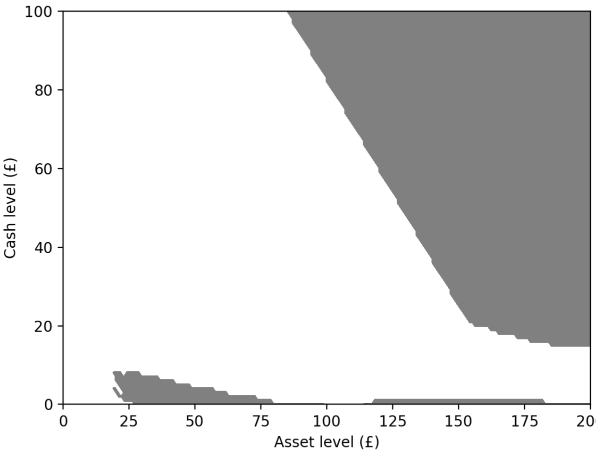
In this chapter we introduced the taking loan option to the cash management model as an alternatively method to supplement the company's cash balance. We also presented two approaches to solve this cash-asset-loan problem. In the first approach, we add the loan state into the state space of the cash management model and formulate the cash-asset-loan problem as a three dimensional Markov decision process. In the second approach (namely the policy improvement heuristic approach) we start with the cash management model without loan opportunity. Then we add one loan offer to the planning horizon and solve the loan-decision problem in addition to the cash holding problem. We repeat this process until adding one more loan offer in the planning horizon does not improve the system's state values. Via numerical studies we showed that the policy improvement heuris-



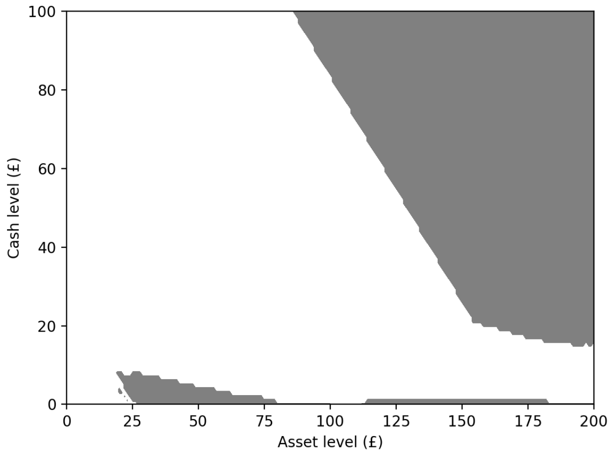
(a) $\iota = 0.01$, MDP



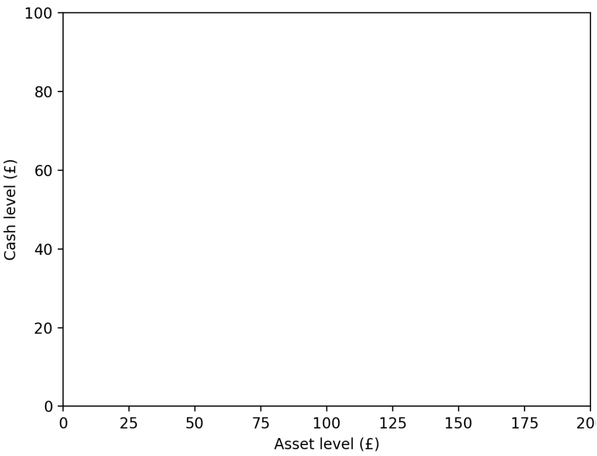
(b) $\iota = 0.01$, PIH



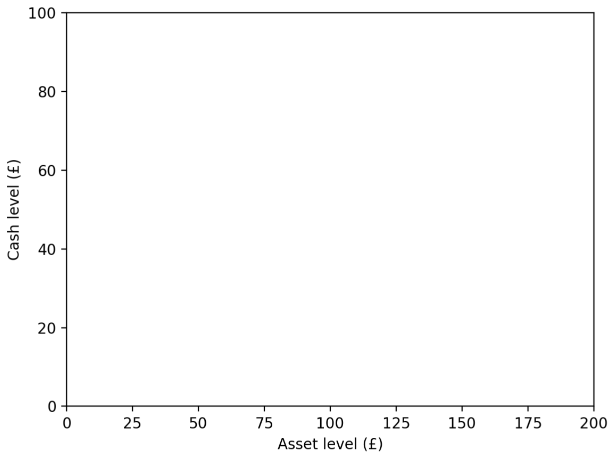
(c) $\iota = 0.04$, MDP



(d) $\iota = 0.04$, PIH



(e) $\iota = 0.08$, MDP



(f) $\iota = 0.08$, PIH

Figure 6.5: Loan policies with different loan interest rate

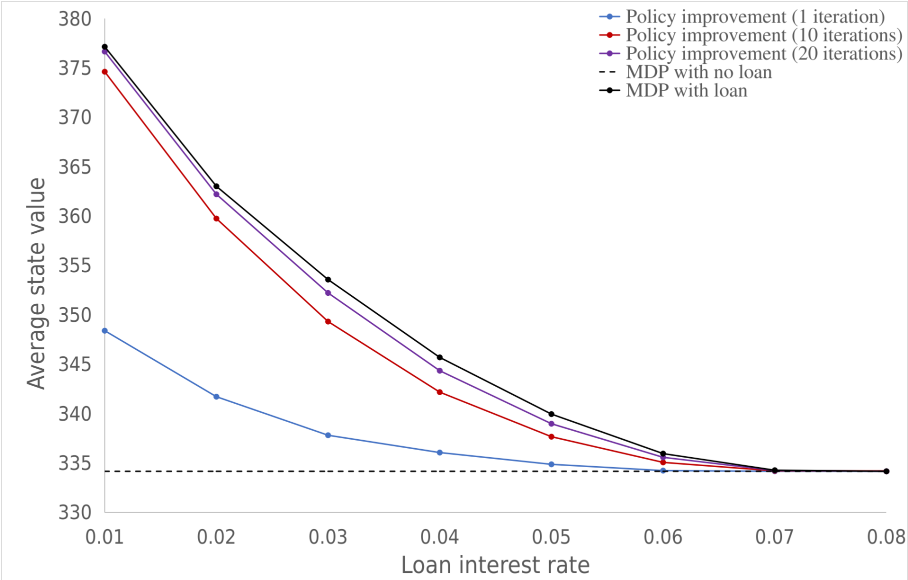
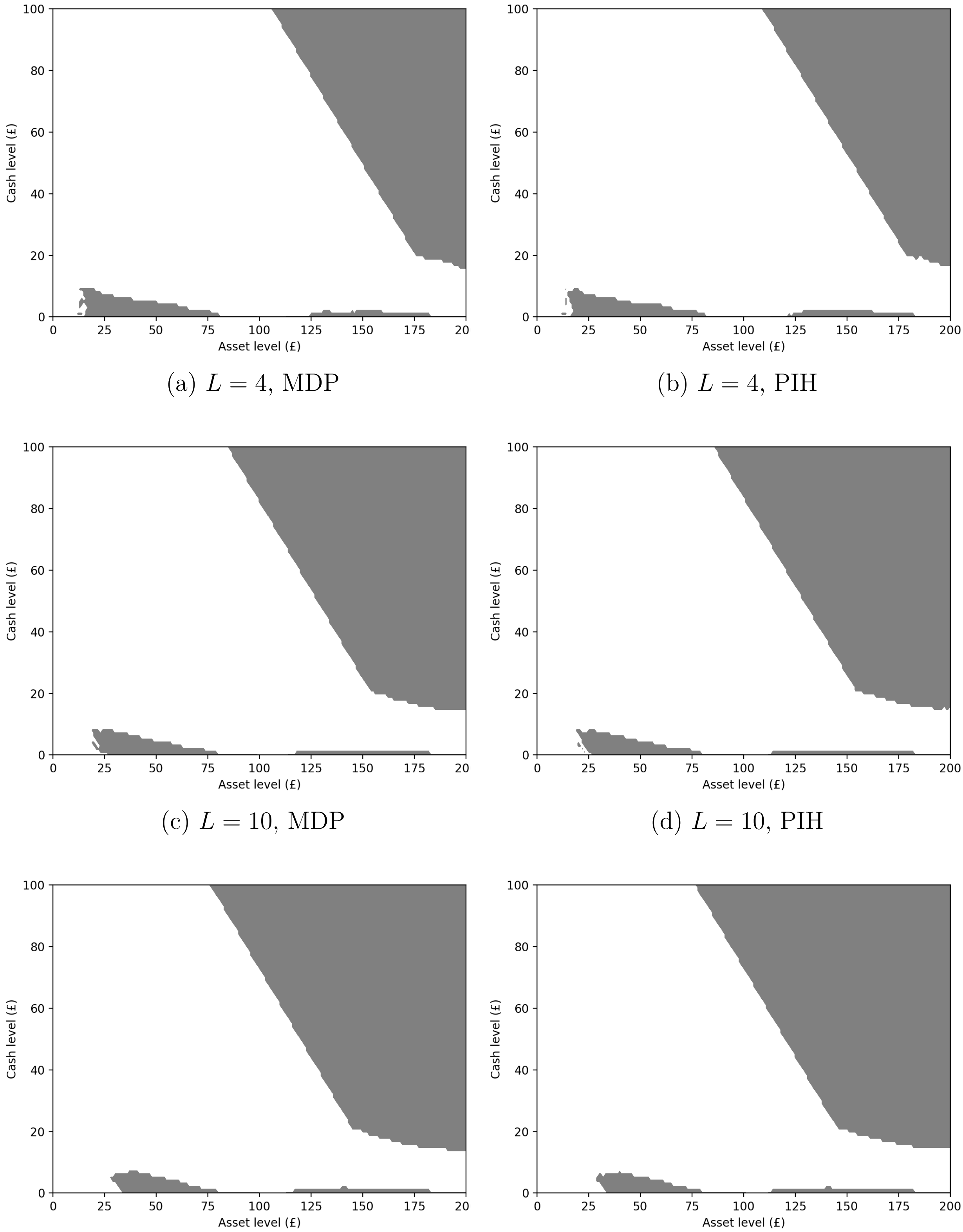


Figure 6.6: The average state value against the loan interest rate

tic approach after 20 iterations provides very similar policies to the MDP approach while reducing the solution time to a large extent (from 7.38 hours to 0.53 hours). At last we examined the impact of loan conditions (i.e. the loan interest rate, the loan age and the loan size) on the loan policies and the system's state values. We observed that the system has a higher state values with a lower loan interest rate or a longer loan age but the company does not always benefit from the loan with a larger size.



(a) $L = 4$, MDP (b) $L = 4$, PIH

(c) $L = 10$, MDP (d) $L = 10$, PIH

(e) $L = 18$, MDP (f) $L = 18$, PIH

Figure 6.7: Loan policies with different loan age

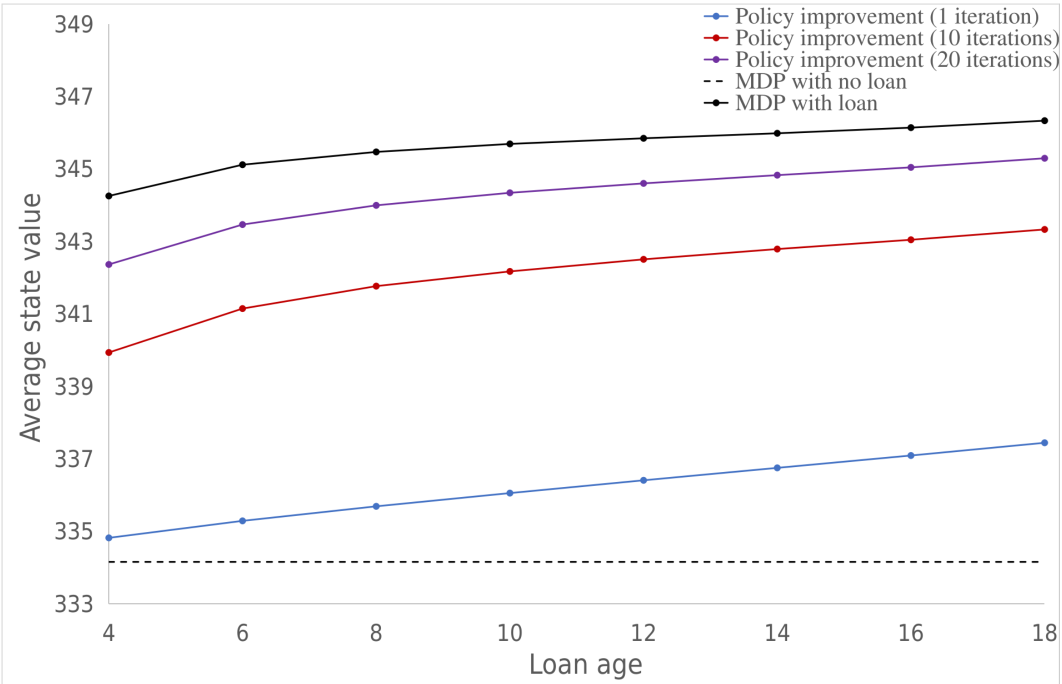
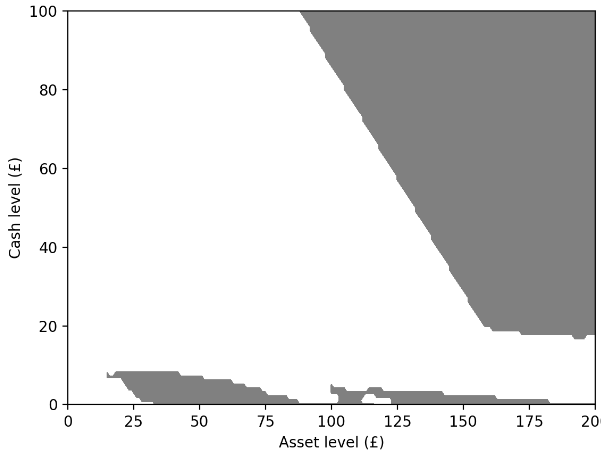
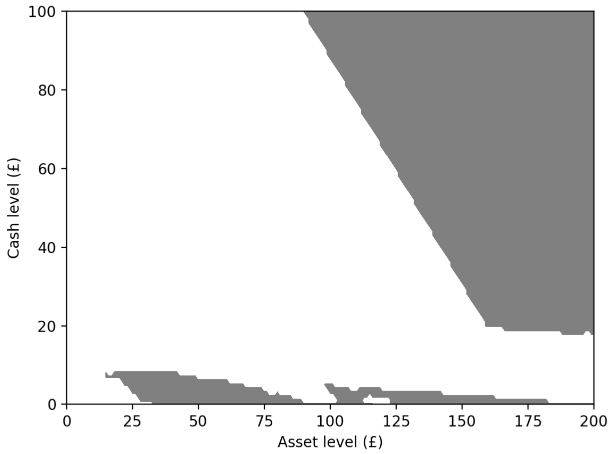


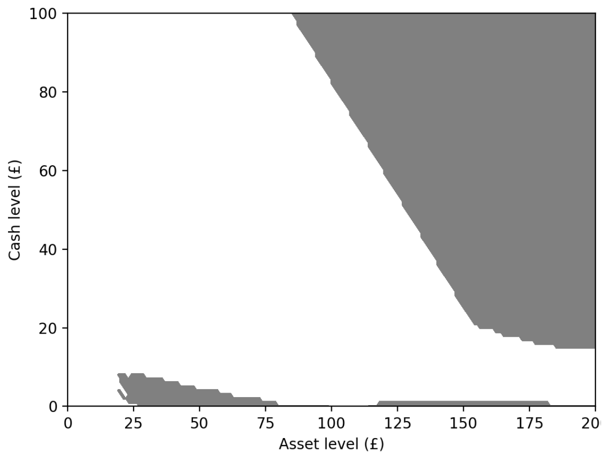
Figure 6.8: The average state value against the loan age



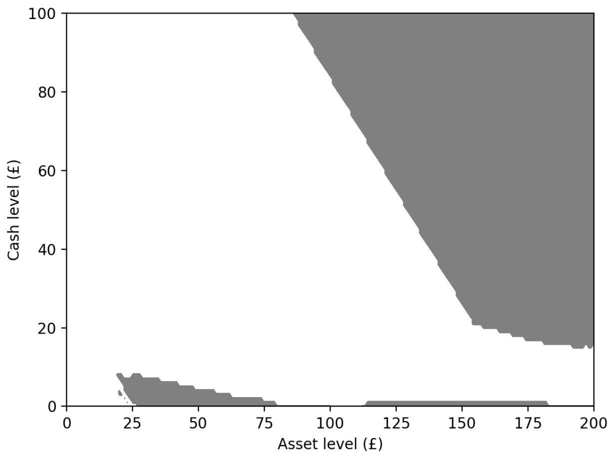
(a) $Z = 8$, MDP



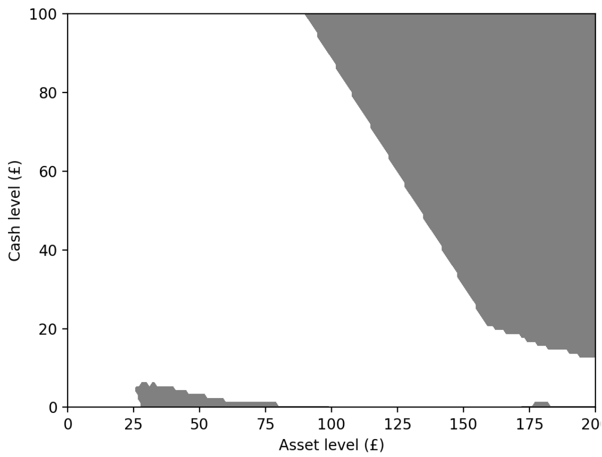
(b) $Z = 8$, PIH



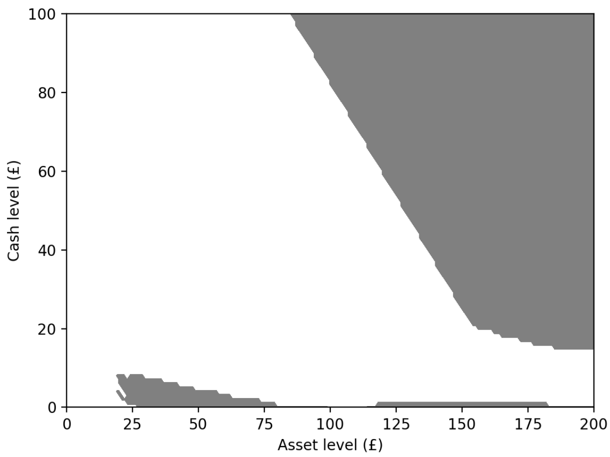
(c) $Z = 20$, MDP



(d) $Z = 20$, PIH



(e) $Z = 36$, MDP



(f) $Z = 36$, PIH

Figure 6.9: Loan policies with different loan size

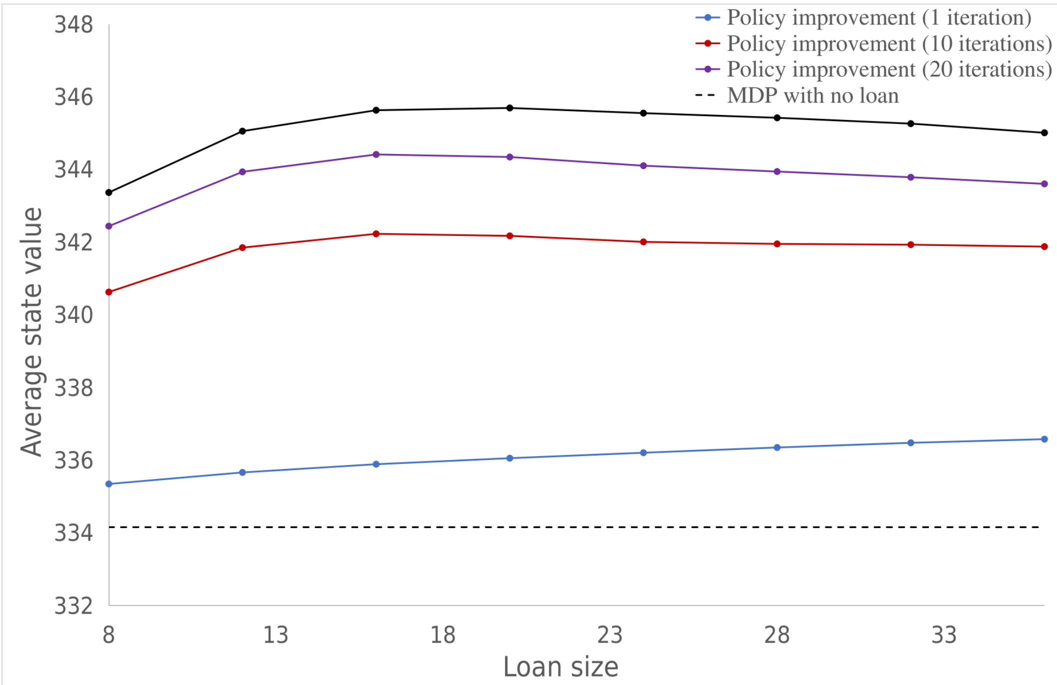


Figure 6.10: The average state value against the loan size

Chapter 7

A Cash Management Model with Multiple Assets

7.1 Introduction

In the previous models discussed in Chapter 5 and Chapter 6 we studied the cash management policy while both the cash account and the asset account being taken into consideration. In this chapter we aim to expand the cash management model into a multiple periods cash-assets management model where the cash is treated as a special type of investment. We will consider an agent who manages one cash account and a number of assets, hence both the cash policies and the asset allocation policies will be studied at the same time. In a typical assets management problem, maximising the expected profit and minimising the policies' risk are both of great interest to the manager. Thus in our cash-assets management model we will introduce the risk measure over multiple periods to the objective function as well as the profit measure. Moreover we will propose three approaches to solve this cash-assets management problem. In the first approach we start with a one-period cash holding/investing model which can be solved via the linear programming method. Based on the static model we will develop a heuristic approach to solve the multi-periods cash-assets management problem. In the second approach we will formulate the problem into a discrete Markov decision process and solve it using the classic back iteration method. At last we will present a double-pass approximate dynamic programming approach which is based on the separable projective approximation routine (SPAR) algorithm proposed by Powell et al. (2004). Based on the real data from four stocks prices (namely AAL, BAC, F and LYG), we will conduct a set of numerical studies and compare the accuracy and the efficiency of these algorithmic approaches. We will also use a synthetic dataset to examine if the double-pass appoximate dynamic programming approach can solve the model with a greater number of assets. To our best knowledge, our work is the first study combining the cash management model with asset mangement theories and the first attempt to adopt the double-pass approximate dynamic programming method in high-dimensional, combined, dynamic cash and asset model.

7.2 Problem description and assumptions

We study a cash management problem with multiple assets where the agent wishes to determine how much financial resource to keep on hand as cash and how much to invest into the multiple assets. The agent wants to strike a balance between having enough cash to control cash shortage risk and pursuing profit on his investments. Both the expected return and the risk of his cash holding/investing strategy will be taken into consideration.

Consider an agent that manages one cash account and several asset accounts over a finite time horizon. At any time the agent can sell and/or buy any amount of assets or make a transfer among these assets. During this horizon, the demand for cash disbursements occurs continuously. Any nonfulfillment of such demand incurs the cash shortage penalty. The objective is to find the best joint cash holding and asset investment strategy in terms of profitability and risk over the whole planning horizon.

We assume that the planning horizon can be discretised into a finite number of time periods. At the beginning of each time period, the agent can take an action such as buying assets, selling assets or transfer among assets and then cannot take an action until the beginning of the next period. The cash demand occurs during each period and must be paid from the cash account. It is also assumed that at the end of each time period, each asset grows with stochastic return rates and the return on cash account is always zero. In the occasion of market decline, the return rate for assets can be negative. Moreover if any action is taken, the transfer fee must be paid. We consider two sets of transfer fees in our model, the buying fee and the selling fee. We assume that the buying/selling transfer fee for different assets are the same and any transfer between two assets must be made via the cash account. One must pay for both buying fee and selling fee to make a transfer between two assets. At last we disregard the consideration of a short market, which means the cash account and all the asset accounts must remain non-negative. If the cash holding level cannot meet cash demand, the agent will be forced to sell his assets to offset this cash deficit and the cash shortage penalty.

7.3 The mathematical model

In this section, we formulate this problem as a stochastic dynamic optimisation model. The goal is to select a best policy in terms of profitability and risk over the whole planning horizon. Hence we construct an agent's utility function as our objective function which is a linear combination of a profit measure and a risk measure. The key elements of the model can be described as follows.

7.3.1 State

Assume that the agent manages one cash account and N asset accounts over T time periods. Let $\vec{s}_t = \{s_{0,t}, s_{1,t}, ..., s_{N,t}\}$ be the state at time t. Let $\vec{x}_t$ be the vector of account levels where $x_{0,t}$ represents the cash holding level while $x_{1,t}, ..., x_{N,t}$ are the levels of n asset accounts at the beginning of this time period. Note that in discrete models, the state vector $\vec{s}$ represents the discretised account levels while in continuous models, $\vec{s}$ and $\vec{x}$ are used interchangeably. For each time period, if the agent takes the action a_t the system transitions into a post-decision state denoted by $\vec{s}_t^{a_t} = \{s_{0,t}^{a_t}, s_{1,t}^{a_t}, ..., s_{N,t}^{a_t}\}$. Since we assume non-negativity for each account, if there is any cash shortage, other assets must be sold to offset such shortage. Similarly, for any deficit in asset accounts, it will be replenished by the cash account. Once the total wealth of cash and asset accounts is non-positive, the system transitions into a boundary state $\vec{s}_o = \{0, ..., 0\}$.

7.3.2 Decision variable

At the beginning of each time period, the manager will examine his current state $\vec{s}_t$ and take an action of selling and/or buying assets. Since all transfers among asset accounts must be made via the cash account, the actions at time period t can be denoted by a two-element tuple $\vec{a}_t = (\vec{a}_t^s, \vec{a}_t^b)$ where $\vec{a}_t^s = \{a_{1,t}^s, a_{2,t}^s, ..., a_{N,t}^s\}$ and $\vec{a}_t^b = \{a_{1,t}^b, a_{2,t}^b, ..., a_{N,t}^b\}$ are the selling amount and buying amount of each asset. The notation $\vec{a}_t$ and $(\vec{a}_t^s, \vec{a}_t^b)$ are used interchangeably in this chapter. Moreover we let $\mathbf{A}_t$ be the set of all feasible actions at time t and $A_t^\pi : \mathbf{S}_t \to \mathbf{A}_t$ be the decision function that determines the action taken on time period t under policy π given state $\vec{s}_t \in \mathbf{S}_t$. We use Π to represent the set of all possible policies, i.e. $\pi \in \Pi$.

7.3.3 Exogenous information process

We let $W_t = (\Delta x_t, \vec{r}_t)$ be the vector of exogenous information available at the end of time t, where Δx_t is the change of uncontrolled cash flow and $\vec{r}_t = \{r_{1,t}, ..., r_{N,t}\}$ is the vector of asset return rates for this time period. Note that the opposite number of Δx_t can be viewed as the cash demand at period t. Since the most usual cash flow probability distribution in the literature is Wiener process (e.g. Miller & Orr (1966), Feng & Muthuraman (2010) and Baccarin (2009)), we use a normal distribution with parameters (μ_t, σ_t) to approximate Δx_t at each time period. We also bound the distribution of Δx_t below zero as negative cash demand is not common in the real world.

For the asset return rates, we use the constant conditional correlation multivariate generalised autoregressive conditional heteroskedasticity (CCC-MV-GARCH) model proposed by Bollerslev (1990) to capture their behaviours. In the CCC-MV-GARCH model, the return rates for assets are described by

$$\begin{aligned}
\vec{r}_t &= \vec{\mu}_t^a + \vec{\epsilon}_t, \\
\vec{\epsilon}_t &= \mathbf{C}_t^{1/2} \vec{e}_t, \\
\mathbf{C}_t &= \mathbf{L}_t^{1/2} \mathbf{P} \mathbf{L}_t^{1/2}.
\end{aligned}$$

In the above equations, $\vec{\mu}_t^a$ is the mean vector for asset return rates, $\mathbf{C}_t^{1/2}$ is the Cholesky decomposition factor of the covariance matrix of $\vec{r}_t$, $\vec{e}_t$ is a random vector with $\mathbb{E}[\vec{e}_t] = 0$ and $\mathrm{Var}[\vec{e}_t] = \mathbf{I}_N$, $\mathbf{I}_N$ is a N by N identity matrix, $\mathbf{L}_t$ is the diagonal matrix of conditional variance and $\mathbf{P}$ is the conditional correlation matrix.

7.3.4 Cost function

There are normally three types of cost in cash management models: transaction cost, cash shortage cost and cash holding cost. In our model, we adopt the first two costs and discard the cash holding cost since it is an opportunity cost representing the profit renounced by the agent when he decides to hold the resource as cash instead of investing into assets. By pursuing return on assets, we have already taken the cash holding cost into consideration implicitly.

Let $\Gamma(\vec{a}_t)$ be the transaction cost associated with action $\vec{a}_t = (\vec{a}^s_t, \vec{a}^b_t)$. In our model, we adopt the proportional transaction cost function with coefficients k^+ and k^-. The total transaction cost given the action a_t is:

$$\Gamma(\vec{a}_t) = k^+ \sum_{n=1}^{N} a^s_{n,t} + k^- \sum_{n=1}^{N} a^b_{n,t}.$$

The cash shortage cost only occurs when the cash demand exceeds the post-decision cash holding level i.e. $x^{a_t}_{0,t} + \Delta x_t < 0$. We assume that the cash shortage cost is proportional to the size of cash deficit with coefficient h. Let $\Theta(x^{a_t}_{0,t}, \Delta x_t)$ be the cash shortage cost function, we have

$$\Theta(x^{a_t}_{0,t}, \Delta x_t) = \max\left\{-h(x^{a_t}_{0,t} + \Delta x_t), 0\right\}.$$

7.3.5 Transition function

Assume that at the beginning of time period t, the agent's holdings for all accounts are $\vec{x}_t = \{x_{0,t}, x_{1,t}, ..., x_{n,t}\}$ and the agent decides to take the action $\vec{a}_t = (\vec{a}^s_t, \vec{a}^b_t)$. The post-decision state can be written as:

$$\vec{s}_t^{\vec{a}_t} = \begin{bmatrix} x_{0,t}^{\vec{a}_t} \\ x_{1,t}^{\vec{a}_t} \\ \cdots \\ x_{N,t}^{\vec{a}_t} \end{bmatrix} = \begin{bmatrix} x_{0,t} + \sum_{n=1}^{N} a_{n,t}^s - (1+k^-)\sum_{n=1}^{N} a_{n,t}^b \\ x_{1,t} - (1+k^+)a_{1,t}^s + a_{1,t}^b \\ \cdots \\ x_{N,t} - (1+k^+)a_{N,t}^s + a_{N,t}^b \end{bmatrix}.$$

During this time period, the uncontrolled cash flow changes by Δx_t and the asset accounts grow with rate vector $\vec{r}_t$. If the post-decision cash holding level is sufficient to fulfil the cash deficit, the state transitions into next time period. Otherwise, the agent must sell his assets to offset the cash deficit along with the corresponding cash shortage cost. For the sake of simplicity, we assume that the agent is not allowed to choose which asset to sell. The asset will be sold in a pre-fixed order in the case of a cash shortage.

7.3.6 Objective function

In a cash management problem, the agent wishes to choose the best joint cash holding and assets investment policy in terms of both his profitability and risk. Hence we introduce an agent's utility function $U(\vec{s}_t, \vec{a}_t)$ which is a linear combination of a profit measure and a risk measure. At time period t, given the state $\vec{s}_t$ and the action $\vec{a}_t$ the agent's one-step utility can be written as:

$$U(\vec{s}_t, \vec{a}_t|\Delta x_t, \vec{r}_t) = (1-\lambda)\omega(\vec{s}_t, \vec{a}_t|\Delta x_t, \vec{r}_t) + \lambda\phi(\vec{s}_t, \vec{a}_t|\Delta x_t, \vec{r}_t).$$

In the above function, λ indicates the agent's preference towards to risk, $\omega(.)$ is a function to measure profit while $\phi(.)$ is a function to measure risk. We use the expected value of net profit to measure the profitability of this action, i.e.

$$\omega(\vec{s}_t, \vec{a}_t|\Delta x_t, \vec{r}_t) = \mathbb{E}\left\{\sum_{n=1}^{N} r_{n,t}x_{n,t} - \Gamma(\vec{a}_t) - \Theta(x_{0,t}^{\vec{a}_t}, \Delta x_t)\right\}.$$

Moreover we use the negative of the CVaR proposed by Rockafellar et al. (2000) to measure the relative risk. The negative CVaR value at probability level $\alpha \in (0,1)$ is defined as

$$\phi(\vec{s}_t, \vec{a}_t | \Delta x_t, \vec{r}_t) = \sup_{\rho \in \mathbb{R}} \left\{ \rho - \frac{\mathbb{E}\left\{ \sum_{n=1}^{N} r_{n,t} x_{n,t} - \Gamma(\vec{a}_t) - \Theta(x_{0,t}^{\vec{a}_t}, \Delta x_t) - \rho \right\}^{-}}{1-\alpha} \right\}$$

where $f(.)^{-} = -\min\{f(.), 0\}$.

Although the CVaR is not time consistent (see Boda & Filar (2006) and Rudloff et al. (2014)), Meng et al. (2011) shows that the sum of CVaR of each period provides a good risk measure in multi-period portfolio optimisation models. Since the cumulative utility function can be written as a linear combination of the cumulative net profit and the sum of CVaR of each period, we adopt this cumulative utility function as the objective in our stochastic dynamic optimisation model, i.e.

$$\max_{\pi \in \Pi} \mathbb{E} \left\{ \sum_{t=0}^{T} U_t(\vec{s}_t, \vec{a}_t) \right\}. \tag{7.1}$$

7.4 Algorithmic strategies

In this section, we present three approaches to solve the cash-assets management problem: the first approach is a heuristic algorithm based on linear programming method. The main idea of this approach is that at each decision epoch, the decision maker solves a static model assuming that no other action can be taken in future. The second approach is formulating the problem into a discrete Markov decision process (MDP) and solving it via the classic backward recursion method. The last approach is a multi-dimensional version of the SPAR algorithm proposed by Powell et al. (2004). In this approach, we use Piecewise linear functions to approximate the values of holding cash or investing in assets. The SPAR algorithm, instead of predicting the value of each account, updates the gradient of each segment of each Piecewise linear function. At each epoch, the agent make decisions based on these gradients instead of the Piecewise linear function values.

7.4.1 A static model and a heuristic approach

Assume that at the current period t, the agent's holdings for all accounts are $\vec{s}_t = (x_{0,t}, x_{1,t}, ..., x_{N,t})$ and the agent wishes to find the decision that maximises his utility for the whole planning horizon $\mathcal{T} = \{t, t+1, ..., T\}$. In a static model, the agent must decide his policy for the whole planning horizon and cannot change his policy afterwards.

Hence the objective function at time t can be written as

$$\max_{\vec{a}_t, \vec{a}_{t+1}, ..., \vec{a}_T} U_{t \to T}\{\vec{s}_t, \vec{a}_t, \vec{a}_{t+1}, ..., \vec{a}_T\}$$

where $U_{t\to T}(.)$ is the agent's cumulative utility function over periods from t to T. Rockafellar et al. (2000) has shown that the optimisation of CVaR value can be formulated as a linear programming model. Based on their method, we add the transaction cost function and the cash shortage cost function into the model and formulate the static cash holding/investing problem as a linear programming model.

In this model, the objective is to maximise the agent's utility from the current time period t to the terminal period T, which is a combination of a profit measure function $\omega_{t\to T}$ and a risk measure function $\phi_{t\to T}$ on his assets. We define the profit measure as the expected net income function, i.e. the expected total income generated from the assets minus the sum of the total transaction cost and the cash shortage cost from period t to T. The profit measure function can be written as:

$$\omega_{t\to T} = \mathbb{E}\left\{\sum_{n=1}^{N} x_{n,t}^{\vec{a}_t}\left(\prod_{\tau=t}^{T}(1+r_{n,\tau})-1\right) - \sum_{\tau=t}^{T}\left(\Gamma(\vec{a}_\tau) + \Theta(x_{0,\tau}^{\vec{a}_\tau}, \Delta x_\tau)\right)\right\}$$

Then we let the expected shortfall from time t to T (i.e. the Conditional Value at Risk) be the risk measure function. According to Rockafellar et al. (2000), this risk measure can be obtained via solving:

$$\begin{aligned}
\max_{\rho\in\mathbb{R}, \vec{a}_t, \vec{a}_{t+1}, ..., \vec{a}_T} \quad & \phi_{t\to T}(\vec{s}_t, \vec{a}_t, \vec{a}_{t+1}..., \vec{a}_T) \\
\text{s.t.} \quad & \phi_{t\to T}(\vec{s}_t, \vec{a}_t, \vec{a}_{t+1}, ..., \vec{a}_T) \geq \rho - \frac{\mathbb{E}\left\{\omega_{t\to T}(\vec{s}_t, \vec{a}_t, \vec{a}_{t+1}..., \vec{a}_T) - \rho\right\}^-}{1-\alpha}
\end{aligned}$$

Note that with the increasing of time periods, this model soon becomes impractical to solve. For the sake of simplicity, we assume the agent can only take an action at time t, i.e. we add the constraint $\vec{a}_{t+1} = ... = \vec{a}_T = 0$ to this model. Now we can write the linear programming model as:

$$
\begin{aligned}
\max_{\rho\in\mathbb{R},\vec{a}_t\in\mathbf{A}_t} \quad & (1-\lambda)\mathbb{E}\left\{\sum_{n=1}^{N} x_{n,t}^{\vec{a}_t}\left(\prod_{\tau=t}^{T}(1+r_{n,\tau})-1\right)-\sum_{\tau=t}^{T}\left(\Gamma(\vec{a}_\tau)+\Theta(x_{0,\tau}^{\vec{a}_\tau},\Delta x_\tau)\right)\right\} \\
& +\lambda\phi_{t\to T}(\vec{s}_t,\vec{a}_t,\vec{a}_{t+1}...,\vec{a}_T) \\
\text{s.t.} \quad & \phi_{t\to T}(\vec{s}_t,\vec{a}_t,\vec{a}_{t+1},...,\vec{a}_T)\geq\rho-\frac{\mathbb{E}\left\{\omega_{t\to T}(\vec{s}_t,\vec{a}_t,\vec{a}_{t+1}...,\vec{a}_T)-\rho\right\}^{-}}{1-\alpha} && \text{I} \\
& x_{0,t}^{\vec{a}_t}=x_{0,t}+\sum_{n=1}^{N}a_{n,t}^{s}-(1+k^{-})\sum_{n=1}^{N}a_{n,t}^{b} && \text{II} \\
& x_{n,t}^{\vec{a}_t}=x_{n,t}-(1+k^{+})a_{n,t}^{s}+a_{n,t}^{b}, \qquad n=1,...,N && \text{III} \\
& x_{n,t}^{\vec{a}_t}\geq 0, a_{n,t}^{s}\geq 0, a_{n,t}^{b}\geq 0, \qquad n=0,...,N && \text{IV} \\
& \vec{a}_{t+1}=...=\vec{a}_T=0 && \text{V}
\end{aligned}
\tag{7.2}
$$

In model (7.2), Constraint (I) is the linear reformulation of CVaR (Rockafellar et al. 2000). Constraint (II) and Constraint (III) specify the post-decision state vector $\vec{x}^{\vec{a}_t}$. Constraint (IV) guarantees the non-negativity of the post-decision state vector and the action vector. Constraint (V) represents the simplicity assumption that the agent can only take actions at time t.

To solve this model numerically, we generate J simulations of Δx_τ^j and $\vec{r}_\tau^j$ for $j=1,...,J,\tau=t,t+1,...,T$. We also need to replace the cash shortage cost function

$$
\Theta(x_{0,\tau}^{a_\tau},\Delta x_\tau^j)=\max\left\{-h(x_{0,\tau}^{\vec{a}_\tau}+\Delta x_\tau^j),0\right\} \qquad \text{for } \tau=t,t+1,...,T
$$

with linear functions. Hence we introduce the new parameter d_τ^j to represent the cash deficit at time τ of the j^{th} simulation. Note that at time t, $\Theta(x_{0,t}^{\vec{a}_t},\Delta x_t^j)$ is equivalent to hd_t^j subject to $d_t^j\geq 0$ and $d_t^j\geq-(x_{0,t}^{\vec{a}_t}+\Delta x_t^j)$. For periods from $t+1$ to T, given that $\vec{a}_{t+1}=...=\vec{a}_T=0$, the cost function can be replaced with hd_τ^j along with constraints $d_\tau^j\geq 0$ and $d_\tau^j\geq-\sum_{l=t}^{\tau}\left\{x_{0,t}^{\vec{a}_t}+\Delta x_l^j\right\}-h\sum_{l=t}^{\tau-1}d_l^j$ for $\tau=t+1,t+2,...,T$. Now the model (7.2) can be equivalently formulated as (7.3).

$$
\begin{aligned}
\max_{\rho,\vec{a}_t,z_j,d^j_\tau,\forall j,\forall\tau} \quad & (1-\lambda)\frac{1}{J}\sum_{j=1}^{J}\left\{\sum_{n=1}^{N} x^{\vec{a}_t}_{n,t}\left(\prod_{\tau=t}^{T}(1+r_{n,\tau})-1\right)-\sum_{\tau=t}^{T} hd^j_\tau\right\} \\
& + \lambda\left(\rho-\frac{1}{(1-\alpha)J}\sum_{j=1}^{J} z_j\right)-\Gamma(\vec{a}_t) \\
\text{s.t.} \quad & z_j \geq \rho-\left(\sum_{n=1}^{N} x^{\vec{a}_t}_{n,t}\left(\prod_{\tau=t}^{T}(1+r_{n,\tau})-1\right)-\sum_{\tau=t}^{T} hd^j_\tau\right) \\
& \qquad \text{for } j=1,...,J \qquad \text{I} \\
& x^{\vec{a}_t}_{0,t} = x_{0,t}+\sum_{n=1}^{N} a^s_{n,t}-(1+k^-)\sum_{n=1}^{N} a^b_{n,t} \qquad \text{II} \\
& x^{\vec{a}_t}_{n,t} = x_{n,t}-(1+k^+)a^s_{n,t}+a^b_{n,t}, \text{ for } n=1,...,N \qquad \text{III} \\
& x^{\vec{a}_t}_{n,t} \geq 0, a^s_{n,t} \geq 0, a^b_{n,t} \geq 0, \text{ for } n=0,...,N \qquad \text{IV} \\
& d^j_t \geq -(x^{\vec{a}_t}_{0,t}+\Delta x^j_t), \text{ for } j=1,...,J \qquad \text{V} \\
& d^j_\tau \geq \sum_{l=t}^{\tau} -(x^{\vec{a}_t}_{0,t}+\Delta x^j_l)-h\sum_{l=t}^{\tau-1} d^j_l, \\
& \qquad \text{for } j=1,...,J;\ \tau=t+1,...,T \qquad \text{VI} \\
& z_j \geq 0, d^j_\tau \geq 0 \\
& \qquad \text{for } j=1,...,J;\ \tau=t,t+1,..,T \qquad \text{VII}
\end{aligned}
\tag{7.3}
$$

In Constraint (I), we consider the profit measure function as the gains from assets minus the cumulative cash shortage penalties from time t to T. We estimate the expected profit by calculating the net profit in each simulation. Similar to model (7.2), Constraint (II), (III) and (IV) specify the post-decision states and ensure the non-negativity of states as well as actions. Constraint (V) and (VI) describe the cash deficit at time t and the cumulative cash deficit from time $t+1$ to T correspondingly. At last, Constraint (VII) ensures the non-negativity of the auxiliary variables.

In a static model where the agent can only take actions at time t (and no other action can be taken afterwards), the optimal decision a^*_t can be found by solving model (7.3). Now we propose a heuristic approach that provides a dynamic solution allowing the agent to take actions at any time period. The main idea of this approach as shown in algorithm (4) is repeatedly solving model (7.3) at each period in the whole planning horizon assuming that no action can be taken in future. This approach will be used in the next section as a baseline.

Algorithm 4 The heuristic approach

Step 1: Generate J random numbers/vectors as the simulated cash changes and asset return rates for the whole planning horizon: $\Delta x_t^j, \vec{r}_t^j$ for $t = 1, ..., T, j = 1, ..., J$.

Step 2: Observe the current state $\vec{s}_t$, obtain the static solution by solving:

$$\vec{a}_t^* = \arg \max_{\vec{a}_t \in \mathbf{A}_t} U_{t \to T} \left\{ \vec{s}_t, \vec{a}_t | \vec{a}_{t+1} = ... = \vec{a}_T = 0, \Delta x_\tau^j, \vec{r}_\tau^j, \forall j = 1, ..., J, \tau = t, ..., T \right\}$$

Step 3: Calculate the post-decision state $\vec{s}_t^{\vec{a}_t^*}$.

Step 4: Observe the cash changes and asset return rate $\Delta x_t, \vec{r}_t$, calculate the state for next time period $\vec{s}_{t+1}$.

Step 5: If $t <= T$, update $t := t + 1, \vec{s}_t := \vec{s}_{t+1}$ and go to **Step 2** else **Return** $\vec{a}_t^*, \forall t = 1, ..., T$.

7.4.2 The discrete Markov decision process approach

If we discretise the state space and the decision space, the cash management problem can be formulated as a discrete Markov decision process. Consider an agent managing one cash account and n asset accounts and each account is discretised into m states. Such discretisation requires m^{n+1} states in total. Now we use the backward dynamic programming method to find the policy that maximises the cumulative utility function (7.1) over the finite planning horizon by solving the Bellman equation:

$$\begin{aligned} V_t(\vec{s}_t) &= \max_{\vec{a}_t \in \mathbf{A}_t} \left\{ U_t(\vec{s}_t, \vec{a}_t) + \mathbb{E}\left[\hat{V}_{t+1}(\vec{x}_{t+1}) | \vec{s}_t, \vec{a}_t \right] \right\} \\ &= \max_{\vec{a}_t \in \mathbf{A}_t} \left\{ U_t(\vec{s}_t, \vec{a}_t) + \sum_{\vec{s}_{t+1} \in \mathbf{S}_{t+1}} p(\vec{s}_{t+1} | \vec{s}_t, \vec{a}_t) \hat{V}_{t+1}(\vec{s}_{t+1}) \right\} \end{aligned} \tag{7.4}$$

where $\mathbf{S}_{t+1}$ is the set of all possible discretised state space at time $t + 1$.

In this approach, we create two look-up tables: the pre-decision state table and the post-decision state table.

Then we recursively update the state value for each table for each period. For each post-decision state, we sample 300 paths to represent the company's different financial performance. Based on these samples, we calculate the expected return, the expected shortfall (i.e. CVaR) and the agent's utility for the current time period. The post-decision state value is updated via the following equation:

$$\begin{aligned} V_t(\vec{s}_t^{\vec{a}_t}) &= U_t(\vec{s}_t^{\vec{a}_t}) + \mathbb{E}\left\{\hat{V}_{t+1}(\vec{x}_{t+1}|\vec{s}_t^{\vec{a}_t})\right\} \\ &\approx U_t(\vec{s}_t^{\vec{a}_t}) + \frac{1}{300}\sum_{j=1}^{300}\hat{V}_{t+1}(\vec{s}_{t+1,j}) \end{aligned} \tag{7.5}$$

where $\vec{x}_{t+1,j}$ is company's accounts holding-levels at time $t+1$ in the j^{th} path. For the sake of simplicity, we round it to the nearest discretised state $\vec{s}_{t+1,j}$.

Once the table for the post-decision state value at time t is updated, it is easy to get the pre-decision state value table at the same time period by solving

$$V_t(\vec{s}_t) = \max_{\vec{a}_t \in \mathbf{A}_t}\left\{-\Gamma(\vec{a}_t) + V_t(\vec{s}_t^{\vec{a}_t})\right\}. \tag{7.6}$$

7.4.3 The approximate dynamic programming approach

Since the objective of our model is to maximise the cumulative utility function over the planning horizon, at each time period, the optimal action can be written as:

$$\vec{a^*}_t(\vec{s}_t) = \arg\max_{\vec{a}_t \in \mathbf{A}_t} U_t(\vec{s}_t, \vec{a}_t) + \mathbb{E}\{\hat{V}_{t+1}(\vec{s}_{t+1})\}, t = 0, ..., T.$$

Although the the MDP method can be used to solve this model, due to the well-known curses of dimensionality, it quickly becomes impractical when the number of accounts increases or the discretisation level of state space gets finer.

In this section, we present a double-pass separable Piecewise linear approximate dynamic programming approach, which is a multi-dimensional version of the SPAR algorithm proposed by Powell et al. (2004). The main idea of this approach is to construct an approximating function, learning the gradients of each cash/asset holdings, i.e. the marginal utility gains for each investment over iterations.

Post-decision state

A typical Q learning approach requires the estimation of each state-action pair, which is a form of post-decision state. Note that the agent's utility function, namely the objective function, is a linear combination of the expected returns on assets accounts $\omega(s_t)$ and the expected shortfall $\phi(s_t)$, i.e. the CVaR value. Since both $\omega(s_t)$ and $\phi(s_t)$ have the property of translation-equivariant, i.e. with respect to a random function $f^r(.)$ and a deterministic function $f^d(.)$ we have

$$\omega(f^r + f^d) = \omega(f^r) + f^d,$$

$$\phi(f^r + f^d) = \phi(f^r) + f^d,$$

the utility function can be rewritten as a function of the post-decision state:

$$\begin{aligned} U_t(\vec{s}_t, \vec{a}_t) &= (1-\lambda)\omega(\vec{s}_t, \vec{a}_t) + \lambda\phi(\vec{s}_t, \vec{a}_t) \\ &= (1-\lambda)\{\omega(\vec{s}^{\vec{a}_t}) - \Gamma(\vec{a}_t)\} + \lambda\{\phi(\vec{s}_t^{\vec{a}_t}) - \Gamma(\vec{a}_t)\} \\ &= -\Gamma(\vec{a}_t) + U_t(\vec{s}_t^{\vec{a}_t}). \end{aligned}$$

Now the optimal action can be written as a function of the post-decision state:

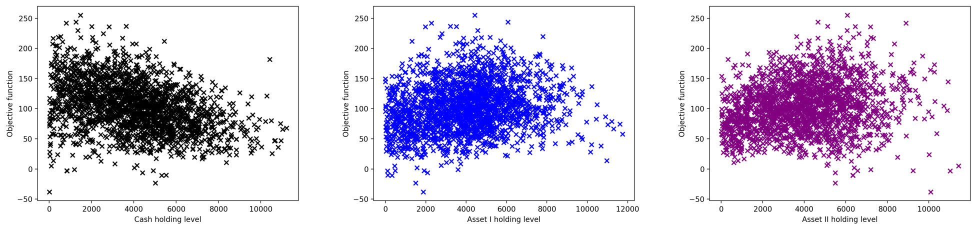
$$\vec{a^*}_t = \arg\max_{\vec{a}_t \in \mathbf{A}_t} \left\{-\Gamma(\vec{a}_t) + \mathbb{E}(V_t(\vec{s}_t^{\vec{a}_t}))\right\} \tag{7.7}$$

where

$$V_t(\vec{s}_t^{\vec{a}_t}) = U_t(\vec{s}_t^{\vec{a}_t}) + \max_{\vec{a}_t \in \mathbf{A}_t} \left\{-\Gamma(\vec{a}_{t+1}) + \mathbb{E}(\hat{V}_{t+1}(\vec{s}_{t+1}^{\vec{a}_{t+1}}))\right\}.$$

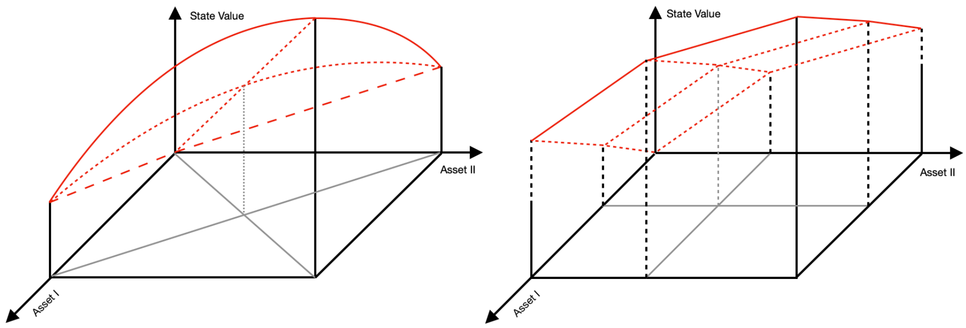
Separable PWL function approximation

To solve function (7.7), it is necessary to approximate the post-decision state value $V_t(\vec{s}_t^{\vec{a}_t})$. It is apparent that $V_t(\vec{s}_t^{\vec{a}_t})$ is a concave function at each accounts holding-level due to the concavity of CVaR measure (see Rockafellar et al. (2000)). In Figure 7.1, we plot the post-decision state value at t_1 with regard to each account size in a two time period model with one cash account and two asset accounts. In this experiment, the agent starts at a fixed initial position $\vec{s}_{t_0} = [10000, 1000, 1000]$ and takes a random action at t_1 and the optimal action at t_2. Since a_{t_1} is obtained randomly, the system visits random post-decision states at the first period. Then, with only one time period left, it can be considered as a static model and the optimal action taken by the agent can be obtained via solving the model (7.3). Figure 7.1 illustrates the concavity of the post-decision state values with respect



(a) $V_t(S_t^{a_t})$ by x_0 level (b) $V_t(S_t^{a_t})$ by x_1 level (c) $V_t(S_t^{a_t})$ by x_2 level

Figure 7.1: Post-decision state value at t_1 in a two-period model



(a) True state value (b) Separable PWL functions approximation

Figure 7.2: Separable PWL functions approximate the true state value

to each account holding level.

In the light of this, we approximate $V_t(\vec{s}_t^{\vec{a}_t})$ using separable and additive Piecewise linear functions, each of which is a min-affine function of the corresponding account holding-level. To be more specific, we use the min-affine function $\min_{m=1,\ldots,M}\left\{b_{m,n}^t x_{n,t}^{\vec{a}_t} + c_{m,n}^t\right\}$ to represent the 'contribution' of the n^{th} account to the post-decision state value and we use the sum of each 'accounts contribution' to approximate the post-decision state value. Figure 7.2 illustrates the basic idea of this approximation. This approximate function can be written as

$$\mathbb{E}\{V_t(\vec{s}_t^{\vec{a}_t})\} \approx \sum_{n=1}^{N} \min_{m=1,\ldots,M}\{c_{m,n}^t + b_{m,n}^t x_{n,t}^{\vec{a}_t}\}.$$

Substituting this approximate function into equation (7.7), the decision function at each iterations can be written as

$$\vec{a}^{*}_t = \arg\max_{\vec{a}_t \in \mathbf{A}_t}\left\{-\Gamma(\vec{a}_t) + \sum_{n=0}^{N} \min_{m=1,\ldots,M}\left\{b_{m,n}^t x_{n,t}^{\vec{a}_t} + c_{m,n}^t\right\}\right\}. \tag{7.8}$$

The optimal action vector can be obtained by solving the linear programming model (7.9).

$$
\begin{aligned}
\max_{\vec{a}_t \in \mathbf{A}_t} \quad & -\sum_{n=1}^{N} \left(k^+ a^s_{n,t} + k^- a^b_{n,t}\right) + \sum_{n=0}^{N} y_{n,t} \\
\text{s.t.} \quad & x_{0,t} + \sum_{n=1}^{N} (a^s_{n,t} - (1+k^-)a^b_n) \geq 0 && \text{I} \\
& x_{n,t} - (1+k^+)a^s_{n,t} + a^b_{n,t} \geq 0, \\
& \quad \text{for } \ n = 1, ..., N && \text{II} \\
& c^t_{m,0} + \left(x_{0,t} + \sum_{n=1}^{N} (a^s_{n,t} - (1+k^-)a^b_n)\right) b^t_{m,0} \geq y_{0,t}, \\
& \quad \text{for } \ m = 1, ..., M && \text{III} \\
& c^t_{m,n} + \left(x_{n,t} - (1+k^+)a^s_{n,t} + a^b_{n,t}\right) b^t_{m,n} \geq y_{n,t}, \\
& \quad \text{for } \ m = 1, ..., M; n = 1, ..., N && \text{IV} \\
& a^s_{n,t}, a^b_{n,t} \geq 0, \ n = 1, ..., N && \text{V}
\end{aligned}
\tag{7.9}
$$

Constraint (I) and (II) specify that the post-decision states (cash holding levels and assets levels) must remain non-negative. Moreover the second term in equation (7.8) (i.e. $\max_{\vec{a}_t} \left\{ \sum_{n=0}^{N} \min_m b^t_{m,n} x^{\vec{a}_t}_{n,t} + c^t_{m,n} \right\}$) can be reformulated as $\max_{\vec{a}_t} \sum_{n=0}^{N} y_{n,t}$ under Constraint (III) and Constraint (IV). Constraint (V) ensures the non-negativity of action vectors.

Note that the parameters $c^t_{m,n}$ for $m = 1, ..., M; n = 0, ..., N$ do not affect the action vectors $\vec{a}_t$ and hence can be dropped from the model. In other words, we do not need the value of the approximate function, but only the gradient $b^t_{m,n}$ in each min-affine function.

Double-pass separable PWL ADP algorithm

Algorithm (5) describes the double-pass separable PWL ADP algorithm. In Step 1, we discretise each accounts holding-level into M segments with the same increment Δm. (In practice, we set $M = 100$ unless specified otherwise). Because of the concavity of the approximate function, we know that $b^t_{n,m} \geq b^t_{n,m+1}$ for $m = 1, ..., M-1$. In other words, all the gradients must be decreasing in each account dimension. In the initialisation step, we set all gradient parameters to zeros. We also fixed the total iteration number I and the total time period number T for the planning horizon in this step.

Algorithm 5 The double-pass separable PWL ADP algorithm

Step 1: Initialisation:
  Step 1.1: Set the total iteration number I, the total segment number M for each accounts and the total number of time periods T.
  Step 1.2: Set initial estimates of the gradients to zeros for each segment m of the post-decision cash/assets holding level $b^t_{m,n} = 0, \forall n, m, t$.
  Step 1.3: Set the iteration index to 1, i.e. $i = 1$.
Step 2: Path sampling
  Step 2.1: Generate a starting state: $\vec{s}_{t_0}$
  Step 2.2: Sample/Observe $\Delta x_{t,(i)}, \vec{r}_{t,(i)}, \forall t$.
Step 3: The forward pass: set $t = 1$, $\vec{s}_t = \vec{s}_{t_0}$.
  Step 3.1: Solve $a^*_t = \arg\max_{\vec{a}_t \in \mathbf{A}_t} \left\{ -\Gamma(\vec{a}_t) + \min_{m=1,...,M} \left\{ b^t_{m,n} x^{\vec{a}_t}_{n,t} + c^t_{m,n} \right\} \right\}$;
  Step 3.2: Calculate and record the post-decision state $\vec{s}^{\vec{a}_t}_t$;
  Step 3.3: Observe the next pre-decision state $\vec{s}_{t+1}$ given $\Delta x_{t,(i)}, \vec{r}_{t,(i)}$;
  Step 3.4: Update the state and time index: $\vec{s}_t := \vec{s}_{t+1}, t := t + 1$.
  If $t \leq T$ go to **Step 3.1**, else go to **Step 4**.
Step 4: The backward pass: set $t = T, \hat{V}_{T+1}(\vec{s}^{\vec{a}_{t+1}}_{T+1}) = 0, \forall \vec{s}^{\vec{a}_{T+1}}_{T+1}$.
  Step 4.1: Retrieve the post-decision state $\vec{s}^{\vec{a}_t}_t$.
  Step 4.2: Get $U_t(\vec{s}^{\vec{a}_t}_t)$ using the simulation method and calculate the new observed post-decision state value:
  $V_t(\vec{s}^{\vec{a}_t}_t) = U_t(\vec{s}^{\vec{a}_t}_t) + \max_{\vec{a}_{t+1} \in \mathbf{A}_{t+1}} \{ -\Gamma(\vec{a}_{t+1}) + \hat{V}_{t+1}(\vec{s}^{\vec{a}_{t+1}}_{t+1}) \}$
  Step 4.3: For $n = 0, ..., N$, do
    Step 4.3.1: Generate state $\vec{s}^{\vec{a}_t}_{t,n+}$ by replacing $x^{\vec{a}_t}_{t,n}$ with $\vec{x}^{\vec{a}_t}_{t,n+} \Delta m$;
    Generate state $\vec{s}^{\vec{a}_t}_{t,n-}$by replacing$x^{\vec{a}_t}_{t,n}$ with$\vec{x}^{\vec{a}_t}_{t,n-} \Delta m$;
    Step 4.3.2: Calculate the post-decision state value $V_t(\vec{s}^{\vec{a}_t}_{t,n+})$, $V_t(\vec{s}^{a_t}_{t,n-})$ and the corresponding segment index m^*.
    Step 4.3.3: Calculate the new gradients $\tilde{b}^t_{m^*+1,n}$ and $\tilde{b}^t_{m^*,n}$.
    $\tilde{b}^t_{m^*+1,n} = (V_t(\vec{s}^{\vec{a}_t}_{t,n+}) - V_t(\vec{s}^{\vec{a}_t}_t))/\Delta m$
    $\tilde{b}^t_{m^*,n} = (V_t(\vec{s}^{\vec{a}_t}_t) - V_t(\vec{s}^{\vec{a}_t}_{t,n-}))/\Delta m$
    Step 4.3.4: Calculate $\hat{b}^t_{m^*,n} = (1 - \theta^t_{m^*,n}) b^t_{m^*,n} + \theta^t_{m^*,n} \tilde{b}^t_{m^*,n}$ and
    $\hat{b}^t_{m^*+1,n} = (1 - \theta^t_{m^*+1,n}) b^t_{m^*+1,n} + \theta^t_{m^*+1,n} \tilde{b}^t_{m^*+1,n}$
    Step 4.3.5: Update the gradients using the projection function.
    i.e. $b^t_{m,n} = \Omega(b^t_{m,n}, \hat{b}^t_{m^*,n}, \hat{b}^t_{m^*+1,n}), \forall m$.
  Step 4.4: Update time index: $t := t - 1$.
  If $t \geq 1$ go to **Step 4.1**, else go to **Step 5**.
Step 5: Update the iteration index: $i := i + 1$.
If $i <= I$ go to **Step 2**, else **Return** $b^t_{m,n}, \forall t, \forall n, \forall m$.

In Step 2, we generate the path for the whole planning horizon. To begin with, we fix a singe initial state $\vec{s}_{t_0}$. Then we generate the exogenous information for the path i, i.e. the stochastic cash flows $\Delta x_{t,(i)}$ and the return rate vector $\vec{r}_{t,(i)}$ for $t = 1, ..., T$. Note that at time t, the decision maker only has access to the previous information $(\Delta x_{1,(i)}, \vec{r}_{1,(i)}), ..., (\Delta x_{t-1,(i)}, \vec{r}_{t-1,(i)})$.

Now we update the gradients for each account holding level using both the forward pass and the backward pass. For the forward pass, as described in Step 3, at the beginning of each time period, the agent observes his current holding level for each account $\vec{s}_t = (x_{0,t}, ..., x_{N,t})$. Then a decision is made by solving model (7.9) given the current gradient estimation and then we record the post-decision state. Next, given the exogenous information $(\Delta x_{t,(i)}, \vec{r}_{t,(i)})$, the system transitions into the pre-decision state for next time period.

For the backward pass, we first set the value for all states after the end of planning horizon to zero. Then we update the gradient parameters backwards. Specifically, we retrieve the post-decision state $\vec{s}_t^{\vec{a}_t}$ and calculate the new observed state value using:

$$V_t(\vec{s}_t^{\vec{a}_t}) = U_t(\vec{s}_t^{\vec{a}_t}) + \max_{a_{t+1} \in \mathbf{A}_{t+1}} \{-\Gamma(\vec{a}_{t+1}) + \hat{V}_{t+1}(\vec{s}_{t+1}^{\vec{a}_{t+1}})\}$$

where $U_t(\vec{s}_t^{\vec{a}_t})$ is the estimation on agent's utility value for time t obtained via Monte Carlo method. Since our goal is to update the gradient of the state value with respect to each account holding level instead of the state value itself, we also need to observe the values of adjacent states. For each account dimension, there are two adjacent states whose value we need to observe. For example, the post-decision state $\vec{s}_t^{\vec{a}_t}$ on the n^{th} dimension has two adjacent states $\vec{s}_{t,n+}^{\vec{a}_t}$ and $\vec{s}_{t,n-}^{\vec{a}_t}$. If the state $\vec{s}_t^{\vec{a}_t}$ is in the m^{th} segment on the n^{th} dimension, we can obtain $\vec{s}_{t,n+}^{\vec{a}_t}$ and $\vec{s}_{t,n-}^{\vec{a}_t}$ by replacing $x_t^{\vec{a}_t}$ with $x_t^{\vec{a}_t} + \Delta m$ and $x_t^{\vec{a}_t} - \Delta m$ respectively. The new observed gradients $\tilde{b}_{m,n}^t$ and $\tilde{b}_{m+1,n}^t$ become

$$\begin{aligned} \tilde{b}_{m+1,n}^t &= \frac{V_t(\vec{s}_{t,n+}^{\vec{a}_t}) - V_t(\vec{s}_t^{\vec{a}_t})}{\Delta m}, \\ \tilde{b}_{m,n}^t &= \frac{V_t(\vec{s}_t^{\vec{a}_t}) - V_t(\vec{s}_{t,n-}^{\vec{a}_t})}{\Delta m}. \end{aligned}$$

With these new observed gradients, we update the gradient parameters via

$$\hat{b}_{m,n}^t = (1 - \theta_{m,n}^t) b_{m,n}^t + \theta_{m,n}^t \tilde{b}_{n,t}^m, \quad \forall m, \ \forall n$$

where $\theta_{m,n}^t$ is the corresponding stepsize. In this process, due to the stochasticity of the data, the estimating function might lose the concavity, thus we must perform a projection operation to ensure the function's concavity. In our study, we obtain the projection function suggested by Nascimento & Powell (2010). The projection ensures concavity by forcing the newly updated gradients $b^t_{m^*,n}$ to be greater than or equal to $b^t_{m^*+1,n}$ and forcing other gradients that violating the function's concavity to be equal to $b^t_{m^*,n}$ or $b^t_{m^*+1,n}$. This projection operation can be described as:

$$\Omega(b^t_{m,n}, \hat{b}^t_{m^*,n}, \hat{b}^t_{m^*+1,n}) = \begin{cases} \frac{\hat{b}^t_{m^*,n}+\hat{b}^t_{m^*+1,n}}{2} & \text{if} \quad m = (m^* \text{ or } m^*+1) \\ & \quad \text{and} \quad \hat{b}^t_{m^*,n} < \hat{b}^t_{m^*+1,n} \\ \hat{b}^t_{m^*,n} & \text{if} \quad m < m^* \\ & \quad \text{and} \quad b^t_{m,n} < \hat{b}^t_{m^*,n} \\ \hat{b}^t_{m^*+1,n} & \text{if} \quad m > m^*+1 \\ & \quad \text{and} \quad b^t_{m,n} > \hat{b}^t_{m^*+1,n} \\ b^t_{m,n} & \text{otherwise} \end{cases} .$$

Stepsize rules

The stepsize indicates how much we adjust our estimate of gradients after each new observation. The choice of stepsize rule affects the convergence behaviour of an ADP algorithm to a large extent. In our study, three different stepsize rules will be evaluated. First of all, we evaluate the constant stepsize rule, i.e. at each iteration we use a fixed stepsize regardless of the iteration index or the state space. We also experiment with the harmonic stepsize rule

$$\theta_i = \frac{o}{o+i-1} \tag{7.10}$$

where o is a constant number and i is the current iteration number. This stepsize rule suggests high values at the first few iterations which will drop very quickly later.

Both the constant stepsize rule and the harmonic stepsize rule give one global stepsize value for all gradients. Since the estimate of gradient with different convergence rate requires different stepsize, one global stepsize value might not suit all gradients $b^t_{m,n}$. Thus we also adopt the stochastic stepsize rule proposed by George & Powell (2006). It is given by

$$\theta_i^{\vec{s}} = 1 - \frac{\left(\bar{\sigma}_i^{\vec{s}}\right)^2}{\bar{\delta}_i^{\vec{s}}}$$

where i is the number of visits to state $\vec{s}$, $\left(\bar{\sigma}_i^{\vec{s}}\right)^2$ is the estimate of the variance of the observation error and $\bar{\delta}_i^{\vec{s}}$ is an estimate of the total squared variation between the observation and the estimate. These are obtained via the following procedure:

$$\begin{aligned}
\nu_n^{\vec{s}} &= \frac{\nu_{i-1}^{\vec{s}}}{1 + \nu_{i-1}^{\vec{s}} - \bar{\nu}} \\
\bar{\beta}_i^{\vec{s}} &= \left(1 - \nu_i^{\vec{s}}\right) \bar{\beta}_{i-1}^{\vec{s}} + \nu_i^{\vec{s}} \left(\hat{X}^n - \bar{\theta}^{n-1}\right), \\
\bar{\delta}_i^{\vec{s}} &= \left(1 - \nu_i^{\vec{s}}\right) \bar{\delta}_{i-1}^{\vec{s}} + \nu_i^{\vec{s}} \left(\hat{X}^n - \bar{\theta}^{n-1}\right)^2, \\
\bar{\lambda}_i^{\vec{s}} &= \left(1 - \theta_i^{\vec{s}}\right)^2 \bar{\lambda}_{i-1}^{\vec{s}} + \left(\theta_i^{\vec{s}}\right)^2, \\
\left(\bar{\sigma}_i^{\vec{s}}\right)^2 &= \frac{\bar{\delta}_i^{\vec{s}} - \left(\bar{\beta}_i^{\vec{s}}\right)^2}{1 + \bar{\lambda}_{i-1}^{\vec{s}}}.
\end{aligned}$$

This rule gives each gradient a stepsize value based on the visits of the corresponding states, the estimate variance of the observation error as well as the total squared variation between the observation and the estimate.

7.5 Numerical experiments

In this section, we conduct numerical experiments to study the performance of the double pass PWL ADP approach (we will refer it as the ADP in the rest of this chapter) in the cash management problem described in Section 7.2. Firstly we describe the instances considered and the data we obtained. Then we study the convergence behaviours of the ADP algorithm, mainly focusing on the impact of stepsize rules and discretisation levels. After that, we compare the ADP algorithm with two alternative algorithms, the heuristic approach where the agent makes the decision at each epoch assuming that no other actions will be taken afterwards, and the discrete Markov decision approach, where we discretise the state space and the action space and then solve the maximisation problem using the classic backward dynamic programming method. All the algorithms are programmed in python 3.7.6 on a PC with 2.5 GHz Quad-Core Intel Core i7 and 8 GB memory.

7.5.1 Problem instances

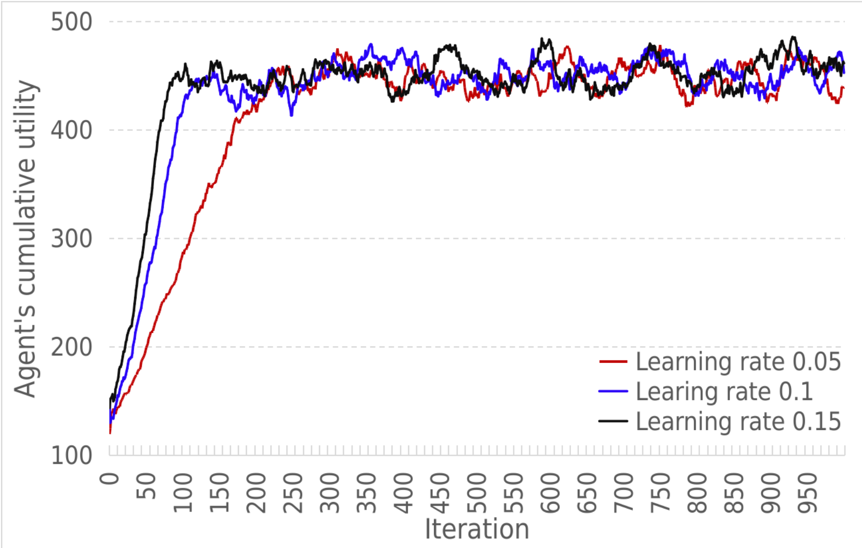
In the numerical experiments, we propose the following problem instance: the agent manages one cash account and four asset accounts. The cash account has zero profitability while the asset price fluctuates at each working day. For each time period, the agent cares about both his profitability and the risk, in other words, his utility function for each time period consists of the expected return and the CVaR value. The goal is to find the best strategy to maximise his cumulative utility over 5 working days.

We assume that at each working day, the agent has to meet the cash demand which is normally distributed with parameters ($\mu_t^d = 100, \sigma_t^d = 50$) and is bounded above zero. Apart from holding as cash, the agent can also invest into four stocks: AAL, BAC, F and LYG. We obtain the weekly return rates of these four stocks (from 4$^{\text{th}}$-Jan-2010 to 3$^{\text{th}}$-Jan-2020) from Yahoo! Finance (https://finance.yahoo.com/). Then we assume the CCC-MV-GARCH model can capture the return rates behaviours and calculate the parameters μ_t and $H_t^{1/2}$ using the historical data. After that, we use this model to generate training data set as well as the policy evaluation data set. At last we set the buying cost k_b and the selling cost k_s to 0.3% and 0.6% respectively. We also choose $S_{t_0} = [10000, 1000, 1000, 1000, 1000]$ as the initial position and $h = 50\%$ as the cash shortage penalty coefficient.

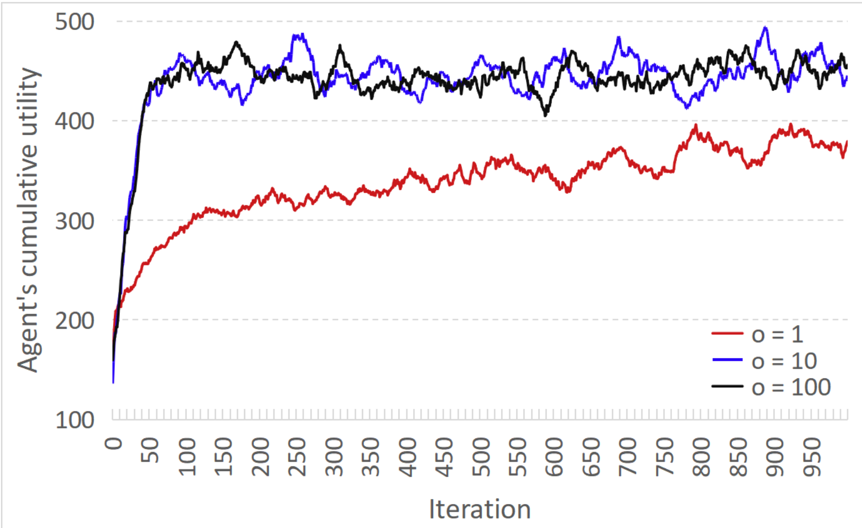
7.5.2 Convergence behaviour of the ADP algorithm

Now we study the convergence rate of the ADP algorithm in terms of stepsize rules and discretisation levels. To begin with, we use the CCC-MV-GARCH model to generate a training data set and a policy evaluation data set. Then for each experiment we train the gradients for $1,000$ iterations using the training data set. After every 5 iterations, we retrieve the policy and examine its performance using the evaluation data set. The performance of a policy is measured by the average simulated objective value of 20 sample paths.

Figure 7.3(a) shows the convergence speed of the ADP algorithm with constant



(a) Constant learning rate



(b) Harmonic learning rate

Figure 7.3: ADP algorithm with deterministic stepsize rule

learning rates. In our experiments, we choose three different constant learning rates $\theta = 0.05, \theta = 0.1$ and $\theta = 0.15$ and examine policies' performance every 5 iterations. It can be seen that the algorithm with $\theta = 0.05$ has the lowest convergence rate and the one with $\theta = 0.15$ outperforms others in terms of convergence speed. Moreover the performance of all three learning rates are quite similar once they converge.

Instead of a fixed stepsize value, the harmonic stepsize rule gives a stepsize based on the number of iterations. The rationale behind the harmonic rule is that in the early iterations, the observations differ from the estimate to a large extent and thus the algorithm requires a large stepsize. In the later iterations,

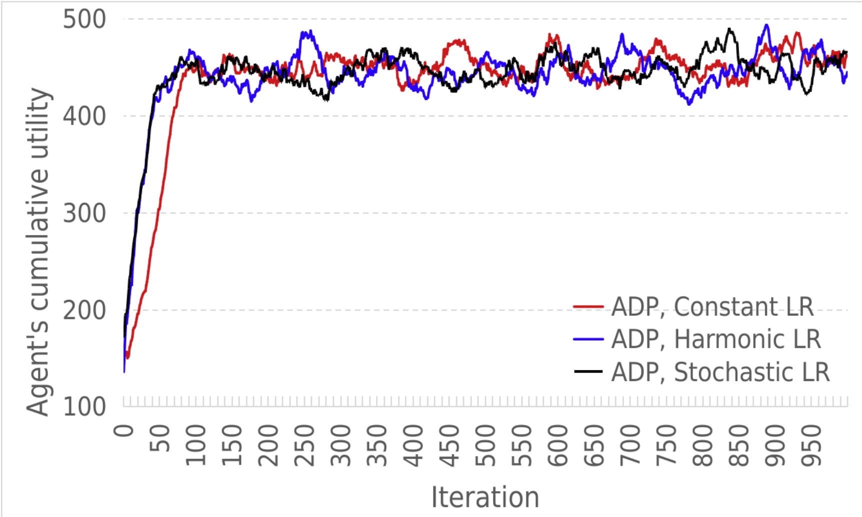


Figure 7.4: Convergence behaviour of ADP with different stepsize rules

the estimate gets closer to the true value and the algorithm needs a small stepsize to prevent over-sensitivity to new observations. However the harmonic stepsize rule needs the designer to decide how fast the stepsize value diminishes by tuning the parameter a in equation (7.10). In Figure 7.3(b), we experiment with three different parameters: $o = 1, o = 10$ and $o = 100$. It can be seen that with parameter $o = 1$, the stepsize diminishes quickly and it converges much slower than other two lines. The harmonic stepsize rules with parameter $o = 10$ and $o = 100$ have very similar performance.

Both the constant stepsize rule and the harmonic stepsize rule requires the designer to predict parameters a priori. Moreover they give one global stepsize for all gradient estimates that need to be updated. The stochastic stepsize, on the other hand, can avoid these drawbacks by assigning a stepsize value to each estimate based on its performance. We adopt this stepsize rule and compare it with other stepsize rules with the best performance (for constant learning rate, we set θ equal to 0.15 and for harmonic learning rate we set $o = 100$). The result is shown in Figure 7.4. It can be seen that the ADP approach with stochastic stepsize rule has a very similar performance with the ADP approach with the harmonic stepsize rule ($o = 100$). Both of them outperform the ADP algorithm with the constant learning rate in terms of convergence rate. All three stepsize rules gives similar values once they converged.

In the ADP algorithm, we discretise each dimension of the state space into

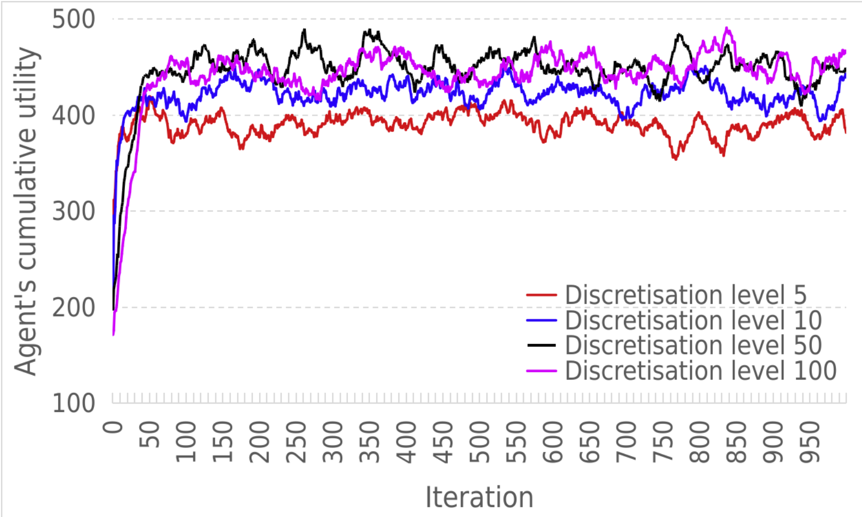


Figure 7.5: Convergence behaviour of ADP with different discretisation level

M segments and then estimate the gradient of the state value in each segment. The discretisation level is also a vital factor of the convergence behaviour and the algorithm's accuracy. We now examine the ADP algorithm with different discretisation levels ($M = [5, 10, 50, 100]$) as in Figure 7.5. We observe that the coarse discretisation levels ($M = 5, M = 10$) converge quickly but reach a lower value eventually than the fine discretisation levels ($M = 50, M = 100$).

7.5.3 Comparison of algorithms

In this section, we compare the ADP algorithm with two alternative algorithms in terms of the cumulative utility as well as the cumulative net profits for the whole planning horizon. The heuristic approach, as discussed in Section 7.4.1, solves a static model via linear programming method at each decision epoch assuming that no other action can be taken in future. This algorithm takes around 8 seconds to solve a model with five time periods.

In the MDP approach, we discretise the state space along with the action space and then solve the dynamic model using the classic backward method. The discretisation level is one of the major factors that influences the MDP algorithm's solution time and the accuracy. To determine the discretisation level in the MDP method, which will be used as one of the benchmarks for the ADP algorithm, we experiment on 5 different discretisation levels, i.e. M is set to be equal to $(4, 5, 6, 7, 8)$. For each discretisation level, we use the same set of training data

Table 7.1: Comparison on discretisation levels in MDP

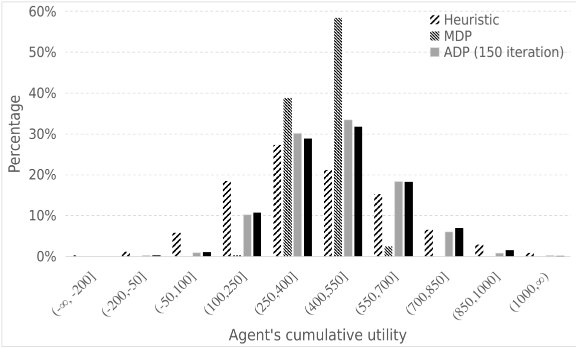
Discretisation level	Number of states	Solution time (m)	Cumulative utility	
			Mean	Std. dev.
4	10,240	1.9	348.5	66.9
5	31,250	13.4	382.2	61.5
6	77,760	47.6	389.5	61.8
7	168,070	148.3	409.1	69.1
8	327,680	491.4	421.1	67.5

Table 7.2: Comparison on approaches

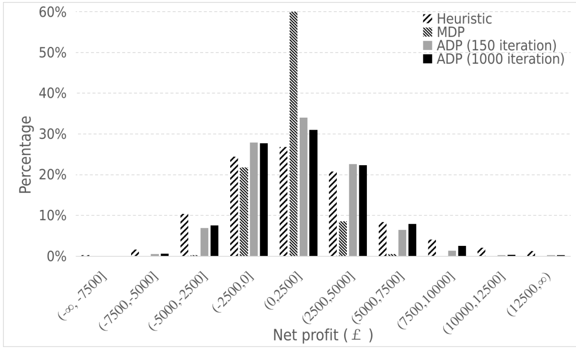
Algorithm	Solution time (m)	Cumulative utility		Net profits (£)	
		Mean	Std. dev.	Mean	Std. dev.
Heuristic approach	0.14	403.1	231.1	1644.9	4125.7
MDP (discretisation level 8)	491.4	421.1	67.5	881.8	1189.2
ADP (150 iterations)	12.7	444.5	164.5	1224.6	2824.6
ADP (1,000 iterations)	79.3	447.8	177.6	1344.1	3048.0

to calculate the post-decision state value $V_t(\vec{s}_t^{\vec{a}_t})$ for $t = 1, ..., 5$. After that, we examine their performance using the same set of evaluation data. The performance is measured by the average simulated utility value of 1000 sample paths. Table 7.1 shows the experiment result for each discretisation level in the MDP approach. When the discretisation level (M) changes from 4 to 8, the number of states vary from $10,240$ to $327,680$ and the solution time increases from 1.9 minutes to around 8 hours. The objective value also rises as the discretisation level gets finer. When the discretisation level increases from 4 to 8, the agent's utility value improves by 20.8%.

Now we compare the performance of four algorithms: the heuristic approach, the MDP approach at discretisation level 8, the ADP algorithm after 150 iterations and the ADP algorithm after 1000 iterations. For each ADP approach, we use the stochastic stepsize rule and set the discretisation level to $M = 100$. Table 7.2 shows the average agent's utility value along with the average net profit of 1000 samples following these four approaches. Figure 7.6 shows the distribution of these samples in terms of both agent's utility and net profit.



(a) Cumulative utility distributions



(b) Net profits distributions

Figure 7.6: Policy evaluation on different algorithms

We observe that both the ADP approach after 150 iterations and the ADP after 1000 iterations have better performance than the alternative approaches in terms of agent's utility. The former one outperforms the heuristic approach and the MDP approach (discretisation level 8) by 10.3% and 5.6% respectively while the latter one outperforms the alternatives by 11.1% and 6.3%.

In terms of the net profit, the heuristic approach gives a policy with the highest expected value but also the highest volatility. The MDP approach, on the other hand, has the lowest expected profit but with the highest stability. The ADP approaches give more balanced results: a policy with expected profit higher than the MDP approach and lower volatility than the heuristic approach. Moreover, as shown in Figure 7.6(b), most samples (690 out of 1000) following the policy in the MDP approach generate profits that lie in the interval $(0, 2500]$. Meanwhile the heuristic approach, the ADP after 150 iterations and the ADP after 1000 iterations have 268, 340 and 310 samples with profit contained in that interval. We also identify that 18 out of 1000 samples following the policy in the heuristic approach lose more than £5000 comparing to 5 out 1000 in ADP approach after 150 iterations, 6 out of 1000 in ADP approach after 1000 iterations and 0 out of 1000 in the MDP approach.

7.6 Extension to a greater number of assets

In order to test the ADP algorithm, we synthetically generate 20 assets. The return rate of each asset is assumed to be normally distributed with parameters (μ_t^n, σ_t^n)

where n is the index of asset. To simulate different assets, we generate μ_t^n and σ_t^n for $n = 1, 2, ..., 20$ from a uniform distribution over the interval $(0.001, 0.01)$. The average weekly returns on the stocks AAL, BAC, F and LYG vary from 0.00103 to 0.00811. Hence we believe the interval $(0.001, 0.01)$ is a good approximation of the real world. Other parameters (namely the cash demand distribution, the transaction cost parameters and the shortage penalty coefficient) remain unchanged. The goal is to determine if the ADP algorithm can solve problems with a greater number of assets.

We conduct four series of experiments where the agent is assumed to manage 5 asset accounts, 10 asset accounts, 15 asset accounts and 20 asset accounts respectively while managing the cash account. All the assets available to the agent are randomly chosen from the 20 synthetic assets. For each setting, we repeat 300 times and report the average solution time and the corresponding results in terms of the objective function (i.e. agent's cumulative utility). The result is shown in Figure 7.3. We also report the results from the heuristic method as a benchmark. Note that the MDP method is not practical in these settings due to the curse of dimensionality.

Table 7.3: Comparison on problem sizes

Number of assets	Algorithm	Solution time (m)	Cumulative utility	
			Mean	Std. dev.
5 assets	Heuristic approach	0.18	643.55	71.43
	ADP (150 iterations)	16.60	715.27	31.93
	ADP (1,000 iterations)	136.63	743.57	34.31
10 assets	Heuristic approach	0.44	670.09	66.10
	ADP (150 iterations)	45.69	729.44	25.92
	ADP (1,000 iterations)	364.59	734.88	27.68
15 assets	Heuristic approach	0.93	687.69	67.21
	ADP (150 iterations)	91.07	739.13	23.63
	ADP (1,000 iterations)	738.56	738.25	24.37
20 assets	Heuristic approach	1.56	685.33	63.47
	ADP (150 iterations)	154.69	740.76	22.14
	ADP (1,000 iterations)	1220.51	741.83	22.45

It can be seen that with the increase of the asset account number, the solution time of the ADP algorithm grows fast. Specifically, our experiments suggest the solution time increases super-linearly with the number of assets, but less than exponentially. Our experiments show that in the case of 20 assets, the ADP algorithm can still solve the problem within reasonable time. The ADP algorithm with 1000 iterations can solve the problem with 20 assets within 21 hours while the ADP algorithm with 150 iterations only takes less than 3 hours. However these two methods provide very similar results in terms of the objective function, both outperform the heuristic approach to a large extent(8.09% and 8.24% respectively in terms of the average value of the objective function). Moreover, the ADP algorithms provide strategies with more stability (i.e. less standard deviations) compared with the heuristic approach. This suggests that in this experimental settings, the agent should use the ADP algorithm with 150 iterations to obtain his cash-asset management strategy. Due to the limited time and computational resources, we only experiment the ADP algorithm on the problem with 20 assets. The future work includes testing the algorithm on a more-assets model and examining the accounts limitation given a certain solution time. It would also be interesting to explore the methods to accelerate the ADP algorithm in dynamic cash-asset management models.

7.7 Conclusion

In this chapter we proposed a cash-assets allocation problem where the agent manages one cash account and a number of asset accounts over a finite planning horizon. We assumed that the agent wishes to maximise his net profit and minimise his policy's risk. In this model, we used the expected profit value as the profit measure and the sum of the conditional value-at-risk for each period as the risk measure. Then we proposed three approaches to solve this cash-assets management problem, namely the heuristic approach in which the agent makes decisions at each epoch assuming that no other action will be taken in future, the discrete Markov decision process approach and the approximate dynamic programming approach. Through the numerical studies, we observed that the heuristic approach requires the least computation time but returns the lowest objective value (i.e. agent's cumulative utility). This approach gives a policy with the highest expected value of the net profit, but also takes the highest risk. The MDP approach at discretisation level 8 gives a policy with the most stable performance and the lowest expected profit. Moreover it takes significantly more solution time (around 8 hours) than other approaches. It will soon be impractical once the discretisation level gets finer. The ADP algorithm gives a policy with better performance than other approaches in terms of the objective value within reasonable solution time. At last, we examined if the ADP algorithm can solve problems with high dimensions. We generated a synthetic dataset describing 20 different assets. Then we used both the heuristic method and the ADP method to solve the problem where the agent is assumed to manage 5 assets, 10 assets, 15 asses and 20 assets respectively while managing one cash account. The result shows that although the solution time increases quickly with the increase of model's dimensionality, the model with 20 assets still remains solvable. The heuristic method can solve the model rapidly, but provides worse results in terms of the objective function.

Chapter 8

Conclusion and Critical Reflections

8.1 Conclusion of the thesis

This thesis discusses three novel cash management models: (a) the model where the agent manages one cash account and one asset account, (b) the model where the agent manages two accounts and is allowed to take loans from financial intermediates, and (c) the model where the agent manages one cash account and multiple asset accounts.

In the study of the two accounts cash management model, we discuss the scenario where the asset generates an income that is the source of cash inflows and the scenario where the volume of the asset account grows at a fixed rate instead of generating the cash income. We formulate both problems as discrete MDP models and solve them via the classic backward recursion method. In addition, we show that the insolvency probabilities can also be obtained using the backward recursion method. Through a series of numerical experiments, we observe that the optimal cash policies of this two accounts model possess the two-threshold two-target form and can be seen as the two dimensional versions of the classic (L, l, u, U) policy. Our study shows that generally the agent tends to hold less cash with a larger asset account. This is because a larger asset account generates higher cash incomes which can be used to offset cash demand. Hence the agent has less motivation to hold cash. However when the system occupies balanced states where the income generated from the asset account approximately equals the expected cash demand, the agent tends to adopt a 'safer' cash policy (i.e. starting to replenish cash balance at higher cash trigger level and increasing the target cash level). We believe this is because in balanced states the insolvency risk of the company is sensitive to the cash policy. As a result, the agent in these states wishes to hold more cash to improve the company's survival probability. In the unbalanced states where the cash demand overweights the income or the income dominates the cash demand, cash policies have litter impact on the insolvency risk and the agent wishes to hold less cash to harvest the profit as much as possible.

In the cash-asset-loan model, we introduce loan opportunities to the cash management model. We assume that only one type of loan is available on the financial market and the agent can only take the loan when previous debt is paid off. We propose two approaches to solve this model. The first approach is to formulate the model as a high dimensional discrete MDP and solve the model via the backward recursion method. We also show that with small modification it can be used to solve the problem with multiple types of loan. However the computational cost (i.e. solution time and the computational memory) increases dramatically due to the well known curse of dimensionality. The second approach is a heuristic based on the policy improvement. In this approach, we start with a model where the agent has no access to loans. Note that this model is equivalent to the previous two accounts model that can be easily solved via the backward recursion method. Then we add one loan opportunity to the model, that is we assume a bank offering one loan option to the agent. This offer expires after that time period regardless of the agent's loan action. We show that this one loan model can be easily solved given the results from the no loan model. We also show that the model with any number of loans can be solved based on the results from the model with one less loans. Then we keep adding one loan option into the model until the state values or policies does not change significantly. Conducting a number of numerical experiments, we show that the second approach performs strongly while requiring much less computational cost. We also study the loan policies under different loan conditions (i.e. loan interest rate, loan age and loan size). The results reveal that lower loan interest rate or longer loan age result in higher profits but the company does not always benefit from the loan with a larger size.

At last we present a model where the agent holds a cash account and a portfolio (i.e. a combination of multiple assets). In this model we assume the agent wishes to accrue profits while controlling the risk of his policies. We use the expected net profit as the profit measure and use the conditional value at risk (CVaR) as the risk measure. Although the CVaR is not time consistent, the cumulative CVaR over multiple periods also provides a good risk measure (see Meng et al. (2011)). We then propose three approaches to solve this model: (a) a heuristic based on the static model, (b) the discrete MDP approach, and (c) the approximate dynamic programming (ADP) approach with Piecewise linear approximations. We find that the heuristic method requires lowest solution time and provides the lowest objective value. The discrete MDP approach returns a higher objective value but takes much more computational costs. The performance of the discrete MDP approach can be improved by a finer discretisation level but the computational costs also climb dramatically due to the curse of dimensionality. Compared with these approaches, the ADP approach performs strongly in our experiments within reasonable solution time. We also examine if the ADP algorithm can be used to deal with models with high dimensionality. The result shows that although the solution time of the ADP algorithm climbs quickly with the increase of problem's dimensionality, the cash management model with 20 assets still can be solved within reasonable time.

8.2 Research limitations and future research

In this thesis, we proposed three novel cash management models. However there are many aspects in these models that can be improved upon due to time limitations, limited computational resources, and restricted access to data.

In the two accounts cash management model, we extend the traditional model by the inclusion of a second asset. This asset generates an income which is considered as the source of cash inflows. However it is assumed that this income is deterministic and proportional to the asset. We believe this study could be extended by the inclusion of a more detailed analyses on the asset's profitability. For example, one could adopt the forecasting techniques to model the profitability of the asset account and obtain a more accurate estimation on cash inflows. Lead time is also essential in the study of cash management strategies. In our work, we assume that if the agent decide to sell his asset, the cash balance can be replenished immediately. It is also assumed that the newly invested asset starts to generate incomes in next time period. However this is not always the case in reality. Hence future research should address the lead time of each transfers between the cash and asset accounts. Furthermore, we assumed cash shortage penalties are always proportional to the cash deficit. However, in practice the influence of cash deficit is not always quantifiable. For instance, frequent cash shortages may jeopardise the company's credit and hence damage its future profitability. To our best knowledge, this influence of cash shortage has not been studied in the literature. Future research on cash management should closely examine the impacts of cash shortages.

In Chapter 6, we introduce loan opportunities to the two accounts cash management model. In this research, it is assumed that there is only one type of loan on the financial market. Future research can explore the scenarios with multiple loans. Section 6.3 reveals that the model with multiple loans can be easily formulated as MDP. However it is impractical to solve such model due to the well known curse of dimensionality. It would be interesting to see how other approaches such as evolutionary algorithms, reinforcement learning algorithms, and deep learning algorithms perform in this model. Moreover, the availability of loans can be clarified in future research. In our study, it is assumed that the company has access to the loans once its previous debt is paid off. With the analysis of the loan availability, this model can be expanded to a comprehensive model providing both loan-taking decisions to the manager and loan-releasing decisions to the financial intermediates, i.e. the company's manager only wishes to take the loan if the company benefits from this loan and the bank manager will release the loan to this company if he believes that the loan incomes overweight the default risk. The company only receives this loan successfully if this loan is beneficial to both parties.

The cash management model with multiple assets explores the scenario where the agent holds one cash account and a number of asset accounts. In this research, we assume the same liquidity for each asset. To be more specific, we assume the transfer fee of selling each asset is the same and the cash account is replenished immediately once the selling actions are taken. This assumption normally holds true with regard to short term securities. However if the agent manages non-financial assets, different transfer fees and lead time for selling assets should be considered. In addition, to solve the cash and multiple assets management model we adopt the separable Piecewise linear approximate dynamic programming approach which assumes that the objective value of a high dimensional state can be expressed as the sum of each state element's contribution and each contribution can be approximated by a Piecewise linear function of the corresponding state element. However, once the objective value includes CVaR, this additive assumption incurs inaccuracy in the model. Hence it would be very interesting to consider other approximations. For example, future research could assume the objective value is a non-linear combination of each Piecewise linear function. Future research could also use the deep reinforcement learning techniques to approximate state values. Besides, in our study the agent makes his decisions based on the state information, which is the holding levels of all accounts. Further improvements of the model can be carried out by adding more state information such as the variance of returns on each asset, the historical price behaviour of each asset, and other financial indexes. At last, we test if the ADP algorithm can be used to solve the problem with a large number of asset accounts. Due to the time limitations, we only examine the problem with 5, 10, 15 and 20 asset accounts. It would be interesting to study a model with even higher dimensions and to check the accounts limitation for the ADP algorithm given a certain solution time. It would also be interesting to explore acceleration methods for the ADP algorithm in high-dimensional models.

Bibliography

Almeida, H., Campello, M. & Weisbach, M. S. (2004), 'The cash flow sensitivity of cash', *The Journal of Finance* **59**(4), 1777–1804.

Alvarez, F. & Lippi, F. (2013), 'The demand of liquid assets with uncertain lumpy expenditures', *Journal of Monetary Economics* **60**(7), 753–770.

Artzner, P., Delbaen, F., Eber, J.-M. & Heath, D. (1999), 'Coherent measures of risk', *Mathematical finance* **9**(3), 203–228.

Azcue, P. & Muler, N. (2019), 'Optimal cash management problem for compound Poisson processes with two-sided jumps', *Applied Mathematics & Optimization* **80**(2), 331–368.

Baccarin, S. (2002), 'Optimal impulse control for cash management with quadratic holding-penalty costs', *Decisions in Economics and Finance* **25**(1), 19–32.

Baccarin, S. (2009), 'Optimal impulse control for a multidimensional cash management system with generalized cost functions', *European Journal of Operational Research* **196**(1), 198–206.

Baccarin, S. & Marazzina, D. (2014), 'Optimal impulse control of a portfolio with a fixed transaction cost', *Central European Journal of Operations Research* **22**(2), 355–372.

Baccarin, S. & Sanfelici, S. (2006), 'Optimal impulse control on an unbounded domain with nonlinear cost functions', *Computational Management Science* **3**(1), 81–100.

Bar-Ilan, A., Perry, D. & Stadje, W. (2004), 'A generalized impulse control model of cash management', *Journal of Economic Dynamics and Control* **28**(6), 1013–1033.

Baumol, W. J. (1952), 'The transactions demand for cash: An inventory theoretic approach', *The Quarterly Journal of Economics* pp. 545–556.

Bellman, R. (1957), 'A Markovian decision process', *Journal of Mathematics and Mechanics* **6**, 679–684.

Benkherouf, L. & Bensoussan, A. (2009), 'Optimality of an (s, S) policy with compound Poisson and diffusion demands: A quasi-variational inequalities approach', *SIAM Journal on Control and Optimization* **48**(2), 756–762.

Bensoussan, A. (1984), 'Impulse control and quasi-variational inequalities'. `https://www.researchgate.net/publication/240042682_Impulse_Control_and_Quasi-Variational_Inequalities`, Last accessed on 2021-12-02.

Bensoussan, A., Chutani, A. & Sethi, S. P. (2009), 'Optimal cash management under uncertainty', *Operations Research Letters* **37**(6), 425–429.

Beranek, W. & Howard, R. (1961), 'Dynamic programming and markov processes', *Technometrics* **3**, 120.

Boda, K. & Filar, J. A. (2006), 'Time consistent dynamic risk measures', *Mathematical Methods of Operations Research* **63**(1), 169–186.

Bollerslev, T. (1990), 'Modelling the coherence in short-run nominal exchange rates: a multivariate generalized arch model', *The Review of Economics and Statistics* **73**(3), 498–505.

Chen, D., Li, S., Xiao, J. Z. & Zou, H. (2014), 'The effect of government quality on corporate cash holdings', *Journal of Corporate Finance* **27**, 384–400.

Cheridito, P. & Kupper, M. (2013), Composition of time-consistent dynamic monetary risk measures in discrete time, *in* 'Finance at Fields', World Scientific, pp. 181–206.

Constantinides, G. M. & Richard, S. F. (1978), 'Existence of optimal simple policies for discounted-cost inventory and cash management in continuous time', *Operations Research* **26**(4), 620–636.

Cui, W., Cuong, L. & Shimizu, K. (2020), 'Cash policy and the bank-firm relationship', *Economic Modelling* **91**.

da Costa Moraes, M. B. & Nagano, M. S. (2014), 'Evolutionary models in cash management policies with multiple assets', *Economic Modelling* **39**, 1–7.

Eastham, J. F. & Hastings, K. J. (1988), 'Optimal impulse control of portfolios', *Mathematics of Operations Research* **13**(4), 588–605.

Eppen, G. D. & Fama, E. F. (1968), 'Solutions for cash-balance and simple dynamic-portfolio problems', *The Journal of Business* **41**(1), 94–112.

Feinberg, E. A. & Lewis, M. E. (2005), 'Optimality of four-threshold policies in inventory systems with customer returns and borrowing/storage options', *Probability in the Engineering and Informational Sciences* **19**(01), 45–71.

Feinberg, E. A. & Lewis, M. E. (2007), 'Optimality inequalities for average cost Markov decision processes and the stochastic cash balance problem', *Mathematics of Operations Research* **32**(4), 769–783.

Feng, H. & Muthuraman, K. (2010), 'A computational method for stochastic impulse control problems', *Mathematics of Operations Research* **35**(4), 830–850.

Fernandes, N. & Gonenc, H. (2016), 'Multinationals and cash holdings', *Journal of Corporate Finance* **39**, 139–154.

Frederick, S., Loewenstein, G. & O'donoghue, T. (2002), 'Time discounting and time preference: A critical review', *Journal of Economic Literature* **40**(2), 351–401.

George, A. P. & Powell, W. B. (2006), 'Adaptive stepsizes for recursive estimation with applications in approximate dynamic programming', *Machine Learning* **65**(1), 167–198.

Godfrey, G. A. & Powell, W. B. (1997), An adaptive approximation method for stochastic, dynamic programs, with applications to inventory and distribution problems, Technical report, Citeseer.

Godfrey, G. A. & Powell, W. B. (2002*a*), 'An adaptive dynamic programming algorithm for dynamic fleet management, i: Single period travel times', *Transportation Science* **36**(1), 21–39.

Godfrey, G. A. & Powell, W. B. (2002*b*), 'An adaptive dynamic programming algorithm for dynamic fleet management, ii: Multiperiod travel times', *Transportation Science* **36**(1), 40–54.

Gormley, F. M. & Meade, N. (2007), 'The utility of cash flow forecasts in the management of corporate cash balances', *European Journal of Operational Research* **182**(2), 923–935.

Harrison, J. M., Sellke, T. M. & Taylor, A. J. (1983), 'Impulse control of Brownian motion', *Mathematics of Operations Research* **8**(3), 454–466.

Harrison, J. M. & Taylor, A. J. (1978), 'Optimal control of a Brownian storage system', *Stochastic Processes and Their Applications* **6**(2), 179–194.

He, M., Zhao, L. & Powell, W. B. (2012), 'Approximate dynamic programming algorithms for optimal dosage decisions in controlled ovarian hyperstimulation', *European Journal of Operational Research* **222**(2), 328–340.

Hinderer, K. & Waldmann, K.-H. (2001), 'Cash management in a randomly varying environment', *European Journal of Operational Research* **130**(3), 468–485.

Khurana, I. K., Martin, X. & Pereira, R. (2006), 'Financial development and the cash flow sensitivity of cash', *Journal of Financial and Quantitative Analysis* **41**(4), 787–808.

Korn, R. (1997), 'Optimal impulse control when control actions have random consequences', *Mathematics of Operations Research* **22**(3), 639–667.

Kusnadi, Y. & Wei, K. J. (2011), 'The determinants of corporate cash management policies: Evidence from around the world', *Journal of Corporate Finance* **17**(3), 725–740.

Liu, B. & Xin, C. (2008), An online model for managing cash: an alternative approach to the Miller-Orr model, *in* '2008 Fourth International Conference on Natural Computation', Vol. 5, IEEE, pp. 314–317.

Meng, Z., Jiang, M. & Hu, Q. (2011), 'Dynamic CVaR with multi-period risk problems', *Journal of Systems Science and Complexity* **24**(5), 907.

Milbourne, R. (1983), 'Optimal money holding under uncertainty', *International Economic Review* pp. 685–698.

Miller, M. & Orr, D. (1966), 'A model of the demand for money by firms', *The Quarterly Journal of Economics* **80**(3), 413–435.

Nascimento, J. & Powell, W. (2010), 'Dynamic programming models and algorithms for the mutual fund cash balance problem', *Management Science* **56**(5), 801–815.

Penttinen, M. J. (1991), 'Myopic and stationary solutions for stochastic cash balance problems', *European Journal of Operational Research* **52**(2), 155–166.

Pflug, G. C. (2000), Some remarks on the value-at-risk and the conditional value-at-risk, *in* 'Probabilistic constrained optimization', Springer, pp. 272–281.

Powell, W., Ruszczyński, A. & Topaloglu, H. (2004), 'Learning algorithms for separable approximations of discrete stochastic optimization problems', *Mathematics of Operations Research* **29**(4), 814–836.

Puterman, M. L. (2014), *Markov Decision Processes: Discrete Stochastic Dynamic Programming*, John Wiley & Sons.

Rockafellar, R. T., Uryasev, S. et al. (2000), 'Optimization of conditional value-at-risk', *Journal of Risk* **2**, 21–42.

Rudloff, B., Street, A. & Valladão, D. M. (2014), 'Time consistency and risk averse dynamic decision models: Definition, interpretation and practical consequences', *European Journal of Operational Research* **234**(3), 743–750.

Salas-Molina, F. (2019), 'Fitting random cash management models to data', *Computers & Operations Research* **106**, 298–306.

Salas-Molina, F. (2020), 'Risk-sensitive control of cash management systems', *Operational Research* **20**(2), 1159–1176.

Salas-Molina, F., Martin, F. J., Rodriguez-Aguilar, J. A., Serra, J. & Arcos, J. L. (2017), 'Empowering cash managers to achieve cost savings by improving predictive accuracy', *International Journal of Forecasting* **33**(2), 403–415.

Salas-Molina, F., Pla-Santamaria, D. & Rodriguez-Aguilar, J. A. (2018), 'A multi-objective approach to the cash management problem', *Annals of Operations Research* **267**(1-2), 515–529.

Salas-Molina, F., Rodriguez-Aguilar, J. A. & Díaz-García, P. (2018), 'Selecting cash management models from a multiobjective perspective', *Annals of Operations Research* **261**(1-2), 275–288.

Sasaki, T. & Suzuki, K. (2019), 'Bank health and cash holdings: Evidence from a bank-centered financial market', *Pacific-Basin Finance Journal* **57**, 101195.

Sastry, A. (1970), 'The effect of credit on transactions demand for cash', *The Journal of Finance* **25**(4), 777–781.

Sato, K. & Sawaki, K. (2009), Optimal impulse control for cash management with two sources of short-term funds, *in* 'The Eighth International Symposium on Operations Research and Its Applications', Citeseer, pp. 323–331.

Schroeder, P. & Kacem, I. (2019), 'Optimal cash management with uncertain, interrelated and bounded demands', *Computers & Industrial Engineering* **133**, 195–206.

Steen, N., Byrne, G. & Gelbard, E. (1969), 'Gaussian quadratures for the integrals $\int_0^\infty \exp(-x^2)f(x)\mathrm{d}x$ and $\int_0^b \exp(-x^2)f(x)\mathrm{d}x$', *Mathematics of Computation* **23**, 661–671.

Tobin, J. (1956), 'The interest-elasticity of transactions demand for cash', *The review of Economics and Statistics* **38**(3), 241–247.

Whalen, E. L. (1966), 'A rationalization of the precautionary demand for cash', *The Quarterly Journal of Economics* **80**(2), 314–324.

Wu, H. & Li, Z. (2012), 'Multi-period mean–variance portfolio selection with regime switching and a stochastic cash flow', *Insurance: Mathematics and Economics* **50**(3), 371–384.

Yamazaki, K. (2016), Cash management and control band policies for spectrally one-sided Lévy processes, *in* 'Recent Advances in Financial Engineering 2014: Proceedings of the TMU Finance Workshop 2014', World Scientific, pp. 199–215.

Yan, X. (2006), 'The determinants and implications of mutual fund cash holdings: Theory and evidence', *Financial Management* **35**(2), 67–91.

Yao, H., Li, X., Hao, Z. & Li, Y. (2016), 'Dynamic asset–liability management in a Markov market with stochastic cash flows', *Quantitative Finance* **16**(10), 1575–1597.

Yao, H., Zeng, Y. & Chen, S. (2013), 'Multi-period mean–variance asset–liability management with uncontrolled cash flow and uncertain time-horizon', *Economic Modelling* **30**, 492–500.

Yao, J.-S., Chen, M.-S. & Lu, H.-F. (2006), 'A fuzzy stochastic single-period model for cash management', *European Journal of Operational Research* **170**(1), 72–90.