Working Paper presented at the

Peer-to-Peer Financial Systems 2023 Workshop

July, 2023

Yield Farming for Liquidity Provision

T. N. Li
New York University

S. Naik
Independent

A. Papanicolaou
North Carolina State University

L. Schönleber
University of Turin

Powered by

P2P Financial Systems

Exponential Science

Yield Farming for Liquidity Provision*

T. N. Lǐ,† S. Naik,‡ A. Papanicolaou,§ L. Schönleber¶

27th July 2023

Abstract

Yield farming is a decentralized finance strategy in which liquidity is provided in exchange for rewards in the form of transaction fees paid by liquidity takers. This article analyzes transaction costs, returns, and risks using on-chain data from major decentralized exchanges. To understand the economic mechanisms of yield farming, we present a mathematical model that incorporates stochastic returns, impermanent loss as a source of risk, and transaction costs. By calibrating the model to the data, we gain insights into the trade-off between future benefits and transaction costs, offering a valuable understanding of yield farming's economic dynamics.

Keywords: Yield Farming, Liquidity Provision, Impermanent Loss, Staking, Stochastic Control, Decentralized Exchanges, Decentralized Finance, Portfolio Optimization, Cryptocurrency.
JEL Classification Codes: G10, G11, G20

*We received helpful comments and suggestions from Agostino Capponi, Bernard Dumas, Diego Garcia, Co-Pierre Georg (discussant), Kay Giesecke, Hanna Halaburda, Haibo Jiang (discussant), Stefan Kassberger, Elisa Luciano, Roberto Marfè, Fabio Moneta (discussant), Giovanna Nicodano, Stanislava Nikolova, Davide Pettenuzzo, Christine Parlour, Julien Pénasse, Stefan Scharnowski (discussant), Bertram Steininger (discussant), and Andrea Tamoni. We thank Mariia Aksenova for her help as a research assistant. We thank Roman Lewandrowski for his help accessing on-chain data. We also thank participants of the Long-Term Investors Seminar at the Università degli Studi di Torino, the Algorand Fintech Lab Kick-Off at Università Commerciale Luigi Bocconi, the 5th Asset Pricing Conference by LTI at Collegio Carlo Alberto, the International Fintech Research Conference at the fintechlab.it, the 5th University of Western Australia Blockchain and Cryptocurrency Conference, the internal seminar at ESOMAS, the ToDeFi 2023, the SIAM Conference on Financial Mathematics and Engineering, the FMA European Conference 2023, and the Economics of Financial Technology Conference 2023. The authors thank the FinTech & Digital Finance Chair of Université Paris Dauphine - PSL in partnership with Mazars and Crédit Agricole CIB, and The Graph Foundation for financial support. The paper previously circulated under the title "Maneuvering and Investing in Yield Farms."

†Courant Institute of Mathematical Sciences, New York University, 251 Mercer Street, New York City, New York 10012, United States. *thomli@cims.nyu.edu*.

‡Independent portfolio manager, *siddharth@equalto.io*.

§Department of Mathematics, North Carolina State University, 2311 Stinson Drive, Raleigh, North Carolina 27695, United States. *apapani@ncsu.edu*.

¶Collegio Carlo Alberto and University of Turin, Piazza Vincenzo Arbarello 8, 10122 Turin, Italy. *lorenzo.schoenleber@carloalberto.org*.

1 Introduction

Yield farming provides access to many who need financial services but whom traditional finance leaves behind. It provides users with inflationary or contract-funded rewards for staking capital or using a protocol, which are then payable in the same underlying asset the user holds. Any user can participate, staking an amount of any size — regardless of how small — and receiving a proportional reward.

— Harvey et al. (2021)

Decentralized finance (DeFi), is a fast-growing emerging financial technology that aims to open new avenues for investment by removing intermediaries in security transactions. DeFi uses smart contracts to create protocols that replicate existing financial services in a more open, interoperable, and transparent way.[1] Yield farming is a *passive* DeFi investment strategy, and is currently one of the most significant growth drivers of the DeFi domain, with a market capital expansion from 500 million United States dollars (USD) to 15 billion USD in 2020 alone, peaking in November 2021 with over 170 billion USD *total value locked (TVL)*.[2]

While there are many forms of yield farming (see Appendix A.1), this article focuses on *yield farming for liquidity provision*, which is the practice of depositing tokens into a *liquidity pool* on a *decentralized exchange (DEX)*. As a reward, liquidity providers receive a share of the trading fees paid by traders who use the pool. In contrast to traditional finance, yield farming is different from traditional liquidity provision because it occurs in the absence of a central market maker (intermediary). Figure 1.1 provides a high-level overview: A liquidity provider (yield farmer) deposits tokens, which is called *staking*, into a liquidity pool. Liquidity pool transactions are recorded and maintained on a blockchain, and the liquidity provider's deposits adhere to the terms of a smart contract. If a liquidity taker (trader) enters the liquidity pool, he can buy and sell tokens by swapping them in the pool according to a known pricing rule (pricing rules will be discussed in detail later in this paper). Swapping tokens incurs a trading fee, and in exchange for providing liquidity, the yield farmer earns a share of these trading fees. Thus, cryptocurrency savings can be used to "passively" generate yield, and hence, liquidity provision is somewhat similar to earning interest in a bank account, with the main difference being that yield farming is a high-risk, high-reward venture in which double-digit percentage annual return rates are not unusual. In addition, yield farming can be viewed as a way to increase the returns on existing cryptocurrency holdings. It is important to emphasize that yield farming is different from simply holding cryptocurrencies, and admittedly, it is more complicated than simply earning interest from a bank account.

Our main contributions in this article are as follows: i) We differentiate yield farming by liquidity provision from traditional finance and explain its essential concepts in simple language for the general audience. ii) Empirically, we quantify returns, risks, and transaction costs using on-chain data from the two major decentralized exchanges, Curve and Uniswap. iii) We investigate yield farming by incorporating all its relevant and particular features into a tractable mathematical framework. iv) To shed light on the economic mechanism and trade-offs, we characterize optimal liquidity-provider

[1] A smart contract is a code that lives on a blockchain. It can both control and define interactions between assets, the blockchain, and network participants.

[2] Source: `https://www.defillama.com/`.

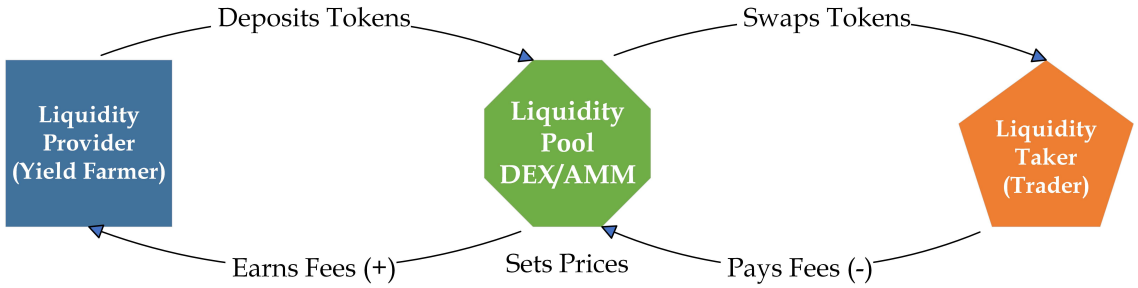

Figure 1.1: Simplified Mechanics of Yield Farming by Liquidity Provision: Liquidity Providers, Traders, and DEX/AMM.

behavior through the lens of our model after calibrating to the data.

We start with the discussion of the different functioning of liquidity provision within DeFi and traditional financial markets, building on Härdle et al. (2020), Harvey et al. (2021), Cong et al. (2021), Makarov and Schoar (2022), and Aquilina et al. (2023). In general, yield farming can be seen as an *over-the-counter (OTC)* market but with a known counterparty on a *decentralized exchange (DEX)* (Lehar and Parlour (2022)), that is governed by an *automated market maker* (AMM) (Capponi and Jia (2022) and Mohan (2022)), behaving according to some known pricing rules.[3] In yield farming, tokens are lent (to a protocol) that resembles security lending; see D'Avolio (2002) and Duffie et al. (2002). Once the tokens are lent (staked) to the liquidity pool, the lender or liquidity provider receives a variable *coupon*, which is governed by the *annual percentage rate (APR)*, similar to a bond, with the additional option to *compound* the *coupon* in order to increase its nominal and therefore future *coupon* payments. The main difference to security lending being that at the end of the term, the protocol most likely will not return the same quantities of individual tokens to the lender, due to the so called *impermanent loss*, which occurs when the price of the deposited tokens changes from when one deposited them. It therefore defines the opportunity-cost dynamic between offering tokens for liquidity provision and simply holding the tokens.[4]

The option to compound the coupon in order to increase future payments seems trivial in traditional finance because when depositing money into a savings account, interest is credited continuously (e.g., monthly), resulting in the well-known concept of interest-on-interest (compounded interest). To contrast, in DeFi, every *action* must be verified by the decentralized consensus mechanism, which incurs transaction costs each time it is invoked; see Donmez and Karaivanov (2022). The interest-on-interest mechanism does not apply automatically, meaning that the yield farming agent (hereafter referred to as "the agent") needs to manually *recompound*, and therefore has to *claim rewards* (interest payments) and deposit them back into the account in order to increase the nominal upon which interest is paid. Hence, the agent faces a trade-off between increasing the nominal and paying the transaction costs for this increase. Among the growing community of yield farming practitioners, the optimal recom-

[3]Building on these concepts, in order to understand the complexity of yield farming, we provide a deeper investigation and description of surrounding concepts, such as *Liquidity Provider token (LP) staking*, as explored in Augustin et al. (2022), as well as *reward tokens*, *boosting*, and *concentrated liquidity*, as studied in Heimbach et al. (2022) in Appendix A.2.

[4]Example: Consider two tokens, A and B, each initially worth 1 USD. The AMM holds identical quantities of 100 of each asset and offers both at a fixed exchange rate of 1:1. Suppose asset B's price appreciates to 2 USD in the wider market. Arbitrageurs exchange all of asset B in the pool for asset A because asset B is more valuable. The pool then holds 200 of asset A worth 200 USD. If, however, the LP did not participate in liquidity provision and held only A and B, his portfolio value would be 300 USD. Therefore, the impermanent loss equals 100 USD (the difference between 300 USD and 200 USD).

pounding is continually debated on various online platforms, where conventional wisdom on optimal recompounding remains fragmented. As this debate has unfolded in the past three to five years, there has been no rigorous academic research on optimal recompounding.[5]

Next, we analyze on-chain data (which refers to data that is stored on a blockchain) to quantify transaction costs for providing liquidity (gas fees), the relevant risks such as the *impermanent loss*, and returns (APRs). We focus on *Curve* and *Uniswap*, which are major (decentralized) exchanges on the Ethereum blockchain that were designed for efficient token trading.[6] The stablecoin pools on Curve show an average transaction cost when providing liquidity of 25 USD, although there have been instances of it surpassing a few hundred USD. In addition, when comparing exchanges specializing in stablecoins to those specializing in Ethereum tokens, we see differences in their respective returns and risks. In our sample, the liquidity provision of stablecoins appears to have lower risk (impermanent loss) but also lower average rewards (3%, with the occasional possibility of reaching 20%). In contrast, for non-stablecoin pairs, the risk of liquidity provision can exceed the returns from the earned trading fees, resulting in a negative net APR. On average, the liquidity pools on Uniswap for non-stablecoin pairs exhibit considerable impermanent loss with extreme values near -30% and reduced rewards of less than 10% when accounting for net APRs. Furthermore, the large cross-section of available tokens on Uniswap allows us to analyze the diversification benefits among the different liquidity pools. Not surprisingly, the cross-correlation between the APRs of pairs of pools is, on average, positive but displays some extreme values in the tails of the distribution.

In the second part of the paper we build a continuous-time stochastic model of token prices in the liquidity pool and formulate a control problem for optimal yield farming. Our framework reveals the fundamental mechanics for cost-effective liquidity provision in DeFi. Specifically, high transaction costs motivate the investigation of the agent's optimal policy for *recompounding* (and *withdrawing*), which can be reduced to the trade-off between i) increasing the nominal and earning more APR in the future and ii) the high gas fees associated with this action. Overall, our results show that transaction costs are of major consideration to DeFi market participants and in line with Grossman and Miller (1988): They lower the costs to liquidity providers.

The construction of our model starts with a mathematical formulation of the AMM's constant product rule for governing prices in the pool. We then derive and characterize the impermanent loss as the major source of risk for the agent who has invested in the pool. For a single pool with a single pair of tokens, our calculations show that the impermanent loss is a function of the respective token return volatilities and the correlation between them. We build further on this model by incorporating a time-varying stochastic APR for a single liquidity pool, a feature that mimics the real-life stochasticity of rewards in yield farming. The presence of fixed transaction costs and manual recompounding faced in DeFi (frictions) motivate us to extend the existing models of Davis and Norman (1990) and Dumas and Luciano (1991), including fixed transaction costs tailored to the yield-farmer's optimal investment problem, and to optimize recompounding so that the portfolio always maintains closeness to an ideal proportion of staked and un-staked tokens; this ideal proportion is similar to the aim portfolio in the transaction costs framework of Garleanu and Pedersen (2013). The fact that the agent faces fixed

[5]See, for example, `https://www.reddit.com/r/UniSwap/comments/sv5uu7/v3_compounding_and_reinvest/`.

[6]Launched in January 2020, Curve's low-fee algorithm was designed specifically for stablecoins. Stablecoins are cryptocurrencies whose value is pegged, or tied, to that of another currency, for example, the USD, a commodity, or a financial instrument; see Arner et al. (2020). Uniswap was created in November 2018 and is estimated to be the largest decentralized exchange allowing users to trade any Ethereum-based tokens.

transaction costs regardless of deposit amount makes the model in-line with actual fees for blockchain verification. Next, we generalize this framework for multiple pools to show how investment can change with additional diversification choices. To our knowledge, this is the first article to propose a normative model with the purpose of characterizing yield farming by liquidity provision in terms of its fundamental elements for risk and return.

Our framework reveals that the optimal recompounding (withdrawal) amount follows a *bang-bang* type of control, which means recompounding (withdrawal) either nothing or everything. Moreover, the results associated with our stochastic control problem point to different *recompounding (withdrawal) regions*, where it is optimal to recompound or not, depending on the region. These regions are the analog of the buy-sell-hold regions identified in the optimal investment problem of Davis and Norman (1990), where here the regions depend on the staked amount, the APR, the impermanent loss, and the transaction costs. Hence, the model is capable of precisely describing the previously discussed trade-off.

We demonstrate the flexibility of our proposed model for different token types. We use data from Curve to create the model to resemble yield farming for stablecoins, and Uniswap data to resemble yield farming for non-stablecoins, with the underlying exogenous stochastic APR process modeled as an exponential Ornstein–Uhlenbeck (OU) process. In order to consider the different magnitudes of risk, the parameters for the impermanent loss are estimated from their respective exchange's token data. In general, we observe that the recompounding regions are increasing in the APR and decreasing in the transaction costs. At the initial time, the recompounding (withdrawal) regions are large (small), and if time passes, they become smaller (larger). A direct comparison of the two token types reveals that a larger impermanent loss leads to a smaller recompounding area and a larger withdrawal area. To better understand how the agent's decision-making is affected by a changing market environment, we conduct a sensitivity analysis by adjusting the transaction cost and the risk tolerance parameters. As expected, lower transaction costs and a larger risk tolerance increase the recompounding region and vice versa.

To obtain additional insight on diversification benefits and the interaction among optimal recompounding controls when facing multiple liquidity pools, we apply our multi-pool framework to a selection of two Uniswap pools. Inspecting the optimal actions for the individual pool in isolation reveals a similar and consistent behavior as that of the previous single pool setting: The recompounding areas are increasing in the APR and the agent recompounds more often (less often) at the initial (terminal) time. In the next step, we analyze the optimal controls for both pools simultaneously and applying a sensitivity analysis. In line with prior intuition, the agent prefers to recompound more frequently in the liquidity pool with the greater APR. When the APRs of the two pools move in the opposite direction, the investor uses the diversification benefits by moving resources from the pool with the lower APR to the pool with the higher APR.

The rest of this article is organized as follows: Section 2 discusses the literature. Section 3 explains the general concepts of yield farming, Section 4 discusses the data, Section 5 introduces the stochastic model, Section 6 presents the numerical output, Section 7 contains parameter sensitivity analysis, and Section 8 concludes.

2 Literature Review

The literature on DeFi is growing and divides itself into various parts: One strand of the literature is represented by the literature on *tokens*, such as platform adaption, token valuation (pricing), and token financing (Prat et al. (2019), Gryglewicz et al. (2021), Goldstein et al. (2022) and Sockin and Xiong (2023)). In addition, yield farming encourages the early adoption of productive platforms (Cong et al. (2021)). Cong et al. (2022) provide informative insights into the economics of "staking" by building a continuous-time model of a token-based economy to study various utility-based functions tokens provide to users. We contribute to this literature by investigating the main quantities (risks, returns, and transaction costs) and the optimal agent's behavior on one particular application of such a platform, that is, yield farming by liquidity provision.

The literature on DeFi describes the differences between decentralized and centralized exchanges in terms of market quality (Barbon and Ranaldo (2021)), and arbitrage rents and order-processing mechanisms (Capponi and Jia (2022) and Lehar and Parlour (2022)). Krishnamachari et al. (2021) and Xu et al. (2023) provide a thorough review on the different AMM protocols and their functioning. To date, the literature on yield farming is relatively undeveloped, with a few exceptions: Cousaert et al. (2022) study the general framework of yield farming by focusing on the protocols and tokens used by aggregators. Heimbach et al. (2022) analyze in detail the risks and returns of Uniswap V3 liquidity providers. In continuous-time framework prices following geometric Brownian motion, Milionis et al. (2022) have identified loss-verses-rebalancing as the primary risk for DeFi liquidity providers, and further decompose this risk into adverse selection cost and an information cost. Cartea et al. (2022) introduce a new comprehensive metric of predictable loss for liquidity providers and derive an optimal liquidity provision strategy. Augustin et al. (2022) study LP token staking and the return chasing behavior of investors on PancakeSwap. We contribute to this literature by developing the general mathematical framework in continuous time that models returns, transaction costs, and risks and that reveals the explicit recompounding mechanism and the optimal (re)allocation within and among different liquidity pools.

There have been many papers that focus on transaction costs models for portfolio management and derivatives pricing. The extension of the Merton optimal investment problem to include proportional transaction costs was first formulated in Magill and Constantinides (1976) and Constantinides (1986), where it was shown that there would be a no-sell region and that small transaction costs would remain as a higher-order effect. Davis and Norman (1990) showed the no-trade region to be wedge-shaped, expressed the optimal strategy as a bang-bang control, and demonstrated that the Hamilton-Jacobi-Bellman equation could be reduced to an equation with a free boundary. Egriboyun and Soner (2010) extended the study of Davis and Norman (1990) with an interpretation of bounded controls as drawdown and reallocation constraints. Dumas and Luciano (1991) considered a version of the problem with no consumption, which led to an exact solution. Duffie and sheng Sun (1990) considered fixed, proportional transaction costs in discrete time lump-sum amounts. A general reference is Kabanov and Safarian (2010), who include an exposition of modified delta hedging strategies for options under proportional transaction costs. Mean reversion toward an aim portfolio is the prototypical behavior that is identified in Garleanu and Pedersen (2013). This phenomenon stems from transaction costs, which restrict the investor's ability to immediately modify their portfolio. As a result, the investor must carefully assess his optimal portfolio not only in the present but also in the future. The contribution

that we make is the extension of these transaction cost techniques to yield farming, but rather than proportional fees, we specifically address how to manage the constant fees that occur when writing to the blockchain.

We also contribute to the literature on liquidity provision in decentralized markets by outlining their differences to both centralized cryptocurrency and traditional financial markets where the assumption of a central market maker or intermediary is crucial (Logue (1975), Glosten and Milgrom (1985), O'Hara and Oldfield (1986), Allen and Gale (2004), and Huang and Wang (2010)). Bianchi et al. (2022) provide empirical evidence that the returns from liquidity provision on centralized cryptocurrency exchanges are primarily concentrated in trading pairs with lower levels of market activity.

3 General Concepts of Yield Farming

Yield farming via liquidity provision is the depositing of tokens into a liquidity pool. Traders who need liquidity can swap tokens within these liquidity pools – paying explicit fees to do so. Liquidity providers earn these fees by facilitating two-way liquidity, but they also bear the risk of capital losses if the fundamental exchange rate changes (impermanent loss). In the following, we explain in more detail the necessary concepts and vocabulary that are involved in yield farming via liquidity provision.

3.1 Decentralized Exchanges, Automated Market-Making, and Liquidity Pools

The liquidity provision occurs through smart contracts without any middleman or intermediator on a DEX, which is a peer-to-peer token trading and swapping platform running on a blockchain that allows participants to exchange tokens in a decentralized manner without a centralized limit order book (unlike a centralized exchange). A DEX permits the self-custody of the assets, namely, the "digital wallet" (wallet) of the participant that is recorded on the assets. The fees for lending and borrowing are transparent and do not have to be shared with any broker. The liquidity provider can freely choose the assets on which liquidity is provided (as opposed to on a CEX). In addition, the liquidity provision is done permissionless, without superfluous or time-consuming application processes.

The decentralized trading mechanism is typically executed by an AMM mechanism, which denotes the underlying protocol that powers a DEX by enabling assets to be traded by using liquidity pools as counterparties rather than a traditional market of buyers and sellers. This is done via matching liquidity providers and liquidity takers where the price of the assets in the pool is determined by a mathematical relationship, for example, the *constant product rule*, which commands that the total amount of liquidity in the pool remains constant. For two tokens, this rule can be described as $L = \sqrt{N_1 N_2}$, where L denotes the total liquidity and N_1 and N_2 denote the amounts of the respective tokens 1 and 2. Hence, as the amount N_1 increases, the amount N_2 must decrease, and vice versa.[7] The major benefits of an AMM are that it is always available and that a traditional counterparty is not necessary to execute a trade.[8]

Liquidity pools are an essential part of the DeFi ecosystem and make up a collection of tokens or

[7] It seems natural that the constant product rule can be extended to more than two tokens; in addition, some platforms also apply other AMM rules, such as the constant sum rule or an *80/20* rule (Balancer Protocol).

[8] Curve uses a special StableSwap invariant that represents a combination of the constant product rule and the constant sum rule; see `https://classic.curve.fi/files/stableswap-paper.pdf`. Uniswap uses a modified product AMM. See `https://uniswap.org/whitepaper-v3.pdf`.

digital assets stored in a smart contract that help to facilitate decentralized trading and that enable essentially anyone with funds to become a liquidity provider and earn trading fees.[9] Users depositing their tokens into a liquidity pool are called *liquidity providers (LPs)*. Typically, to pool liquidity, the amounts that a user supplies must be equally divided between the two tokens. In exchange for providing their liquidity (funds), they earn trading fees from the trades that occur in the liquidity pool according to their share of the liquidity pool. In addition, liquidity pools can solve the problem of new token-based projects with a low user base, by providing a reason for people to hold these tokens – to provide liquidity for a fee.

The estimated returns in the yield farming process are quoted in terms of APRs, which denotes the annualized fraction of earning fees occurred from trading activity ($fees_t$) in the respective pool (over some period) divided by the total volume of the pool (measured in TVL),

$$APR_t = \frac{fees_t}{TVL_t}.$$

In the following, we explain and delve into the various sources of returns and transaction costs one receives or faces when participating in yield farming.

3.2 LP Tokens, Reward Tokens, Boosting, and Concentrated Liquidity

Understanding yield farming in all its facets is a complicated matter. The liquidity provider can increase the returns from yield farming by a significant amount when additionally considering i) LP token staking, ii) the collection of reward tokens, iii) boosting, and iv) managing the impermanent loss through the concentrated liquidity feature. In Appendix A.2, we discuss the concepts in more detail, especially for Curve and Uniswap.

3.3 Quantifying Transactions Costs

An important part and determinant of yield farming are transaction costs, which occur for each action that causes a write transaction on the respective blockchain. In the following, we explain in more detail the fee schedule of the Ethereum blockchain.[10] The simplest transactions, in the context of yield farming, are depositing and withdrawing the tokens to and from the liquidity pool. The major difference between the transaction costs occurring in yield farming and transaction costs occurring in the classical financial system is the fact that these do not increase in volume. Hence, the fee of a transaction on the Ethereum blockchain is not related to the size of the transferred or traded amount. All that matters are i) how much computation a transaction needs (units of gas) and ii) how much the sender is willing to pay for each unit of computation (gas price). In our sample, the average gas fee for a standard Ethereum transfer was 3.35 USD, with extreme values of more than 17.5 USD.[11] In the empirical analysis presented later, we report the transaction costs for adding and removing liquidity for various liquidity pools that, due to their computational complexity, greatly exceed the transaction costs of a simple Ethereum payment.[12]

[9] On Curve, the standard trading fee on all pools is 0.04%. Uniswap charges 0.05%, 0.3%, or 1.0% fee for each trade.

[10] Ethereum's ability to host smart contracts designates it as a smart contract platform, with more than 50% of the DeFi market's TVL; see the "State of Crypto Report" Matsuoka et al. (2022).

[11] In Appendix A.3, we discuss in more detail the calculation of a standard Ethereum transfer.

[12] There are other types of actions involved in yield farming that cause transaction costs: i) depositing (transfer of funds to the liquidity pool), ii) withdrawing (transfer of funds from the liquidity pool to its token pair), iii) claiming

3.4 Risks

Yield farming is a relatively new concept in DeFi, and while yield farming can be a profitable endeavor, it is also risky. In the following, we introduce and provide an overview of some of the key risk concepts to consider when engaging in yield farming.

3.4.1 DeFi Smart Contract Risk, Bugs in the Code, and Risks of Scam

Yield farming and DeFi are a collection of smart contracts, and, hence, one bug in the software can be exploited by a malicious actor to manipulate the project, causing yield farmers to experience a loss.[13] To mitigate this risk, the investor should confirm that the smart contract has been audited by a well-recognized blockchain audit firm.[14] In addition, investors should ensure that they are not using an application that is not properly vetted and whose owners are known.[15] Trust is of importance since one of the most popular types of scam in DeFi is called *a rug pull* or *being rugged*, which refers to a scam in which the creators of a project suddenly and without warning withdraw the funds from their smart contract, leaving investors with worthless tokens.[16]

3.4.2 Impermanent Loss

Yield farming also comes with risks due to the liquidity provision channel of the AMM. The impermanent loss is the difference between the value the crypto assets would have been held and the value of assets in a liquidity pool instead. It occurs when the price of the deposited assets changes compared to when one deposited them. The bigger this change is, the more one is exposed to the impermanent loss. Hence, LPs are subject to the impermanent loss when the prices of tokens in a pool diverge (no matter in which direction the price changes), causing them to underperform a standard buy-and-hold strategy. It is important to mention that the impermanent loss does not necessarily mean that the LP experiences a negative return on the investment: It simply means that the gains from a buy-and-hold strategy outperformed the returns obtained for the liquidity provision.[17] It seems plausible that the more volatile and less positively correlated the assets in the pool are, the more likely it is that one can be exposed to the impermanent loss. In the empirical analysis, we calculate the impermanent loss by comparing the portfolio value of the tokens locked in the pool (V_{Pool}), assuming a constant product pricing rule, to the portfolio value when simply holding the tokens (V_{Hold}), as follows:

$$IL\left(t\right) := \frac{V_{Pool}\left(t\right) - V_{Hold}\left(t\right)}{V_{Hold}\left(t\right)}. \tag{3.1}$$

(requesting the rewards obtained from liquidity provision and LP token staking), iv) staking (depositing the LP token into a staking pool).

[13] Source: `https://decrypt.co/82499/compound-exploit-drains-21m-from-lending-protocol`.

[14] The failure of FTX and Genesis shows that some blockchain auditing companies registered in the *Metaverse* did not properly audited the smart contracts that they were reviewing; see `https://medium.com/@observer1/ftx-auditor-has-an-office-in-the-metaverse-e3273710738b`.

[15] A comprehensive overview of past *rekts* describing severe financial losses within DeFi can be found here `https://rekt.news/`

[16] Source: `https://cointelegraph.com/news/certik-identifies-arbix-finance-as-a-rug-pull-warns-users-to-steer-clear`.

[17] We provide more details about the impermanent loss in Appendix A.4.

3.4.3 Price Risks, (Failed) Stablecoins, and APR Instability

As long as investors hold cryptocurrencies, they expose themselves to the encapsulated price risk of them. In addition, the yield farmer's rewards are paid in tokens, and, as a consequence, if the value of the token declines, so does the overall return on the yield farm investments.

As outlined, a crucial shortcoming of many cryptocurrencies is their excessive volatility. To avoid this, an entire class of cryptocurrencies called stablecoins has emerged, intending to maintain price parity with some target assets such as the USD or the euro (EUR); see Arner et al. (2020). The mechanism by which the stablecoin maintains its *peg* varies by implementation.[18] The failure of TerraUSD (UST) proves that even stablecoins cannot be considered safe assets either; see Liu et al. (2023).

APRs represent the share of earned transaction fees with respect to the TVL. They are therefore dependent on *demand* (earned transaction fees) and *supply* (measured in TVL). As demonstrated later in the empirical section, APRs are volatile, with extreme maximal and minimal values. In addition, as a liquidity pool gains popularity, the APR decreases. This occurs because if a pool offers high returns, more tokens will be added to it, causing a rise in supply, which, in turn, leads to a decrease in the APR.

3.5 The Investment Procedure

This subsection describes the individual actions on a generic yield farming platform. In addition to the generation of the potential yield, we also emphasize the role of the transaction costs in each step.

Initial Step ($t = 0$). Initially, the agent transfers a particular amount of the *Assets Under Management (AUM)*, from the wallet to a given liquidity pool (i.e., locks tokens into a smart contract), which generates returns (earned trading fees) for the provided service. This initial depositing operation results in transaction costs.

Rebalancing Step ($t > 0$). In this step, the agent selects from the following set of choices: i) The earnings from the liquidity pool are reinvested into the liquidity pool (earnings are credited in the wallet and paid in the platform token, which we can reinvest directly). It is important to mention that earnings are not reinvested automatically. We call the operation of reinvesting earnings into the liquidity pool *recompounding*. This depositing operation results in transaction costs. ii) The agent undertakes no action: This neither increases the interest-bearing nominal nor results in any transaction costs.

Figure 3.1 provides visual support for the trade-off. Let us assume an agent with an initial capital of 500 USD invests in a liquidity pool with a yearly fixed APR of 15% (which corresponds to a daily APR of 0.00038). The solid line displays the well-known traditional concept of interest on interest. The

[18] The three primary mechanisms are fiat-collateralized, crypto-collateralized, and non-collateralized stablecoins. Fiat-collateralized stablecoins, the largest category, are backed by off-chain reserves of a specific asset like USD. Off-chain reserves have faced intense scrutiny due to their lack of adherence to the same reserve ratios required of traditional financial institutions. The Commodity Futures Trading Commission (CFTC) and U.S. Securities and Exchange Commission (SEC) have resolved their jurisdiction disputes and focus on more rigorous auditing of these reserves, making them more meaningful than they were from 2016 to 2022. See, `https://www.strausstroy.com/articles/crypto-sec-vs-cftc/`. The second-largest category is crypto-collateralized stablecoins, which are backed by an over-collateralized amount of another cryptocurrency like Bitcoin or Ethereum. Non-collateralized (algorithmic) stablecoins are not backed by any underlying asset and use algorithmic expansion and supply contraction (mint-burn mechanisms) to shift the price to the peg.

dotted line corresponds to the situation the agent is facing when yield farming, that is, interest is not compounded automatically and therefore the wealth only grows arithmetically (namely, by 15% × 500 USD = 75 USD per year). The dashed line corresponds to the wealth of an agent who recompounds at $t = 5$. As visible, the portfolio value drops by the transaction costs (here, 50 USD) but then grows faster due to the fact that the APR is now calculated on the increased base value. Hence, the optimal recompounding area and frequency are dependent on the stacked amount and the transaction costs and are therefore difficult to determine. In the mathematical framework outlined in the later sections, we analyze exactly the described trade-off between increasing the interest-bearing nominal and paying the associated transaction costs considering optimality criteria.

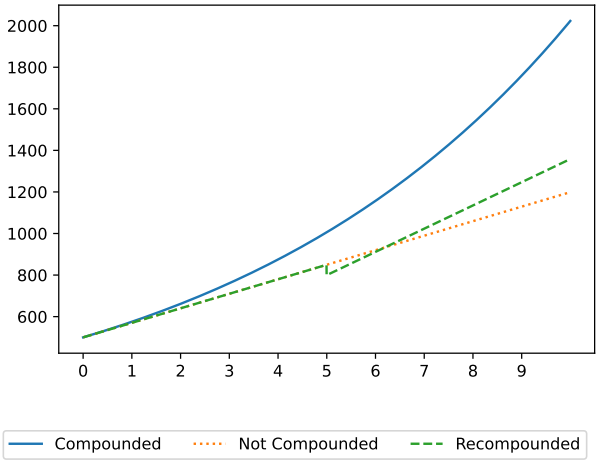

Figure 3.1: Motivation – Recompounding. The figure shows the wealth for an agent in different scenarios. The invested initial amount is set to 500 USD, and the APR is 15%. Thereby, "Compounded" ("Not Compounded") represents the geometric (arithmetic) accumulation of interests. "Recompounded" represents the scenario where the investor recompounds at $t = 5$, paying the transaction costs of 50 USD.

4 Empirical Analysis

In this section, we carry out the empirical analysis to reveal and display the previously discussed quantities of Section 3, and we pay careful attention to transaction costs, risk from the impermanent loss, and stochasticity of APR. We first analyze the liquidity pools and their respective token pairs for stablecoins on Curve. In the next step, we infer additional insight by inspecting a large cross-section of liquidity pools (of any Ethereum-based tokens) on Uniswap.

4.1 Curve Stablecoins Pools

We obtain daily historical data on six liquidity pools for USD stablecoins directly from the Curve website.[19] Table 4.1 reports the pools and their respective token pairs, including the total pool market capitalization. The deepest pool is by far the 3pool, where liquidity is provided for the triple DAI, USDC, and USDT.[20]

[19] Source: `https://classic.curve.fi/dailystats`.

[20] To provide a better understanding of the tokens within a liquidity pool, a more detailed description of them can be found in Table I.1. It is important to mention that the tokens within the pools differ from their collateralization (crypto versus fiat).

Liquidity Pool	Pairs	Market Capitalization
compound	DAI, USDC	252.679m
y	DAI, USDC, USDT, TUSD	171.667m
busd	DAI, USDC, USDT, BUSD	19.618m
susdv2	DAI, USDC, USDT, sUSD	122.298m
pax	DAI, USDC, USDT, USDPAX	11.092m
3pool	DAI, USDC, USDT	3.984b
All Curve	—	17.169b

Table 4.1: Overview – Curve Pools. The table reports the depositable token pair and the overall market capitalization of the six pools from Curve (as of May 12, 2022). Details about the token pairs are provided in Table I.1.

Furthermore, Table 4.2 presents the summary statistics of the respective tokens. The token prices are denoted in USD. It is immediately visible that the respective mean of the token prices circles around 1, with a small standard deviation. Nevertheless, some tokens deviated downward from their fundamental, for example, sUSD.[21]

Token	Mean	Standard Deviation	Minimum	Maximum
DAI	1.004	0.007	0.993	1.034
USDC	1.000	0.000	0.999	1.001
USDT	1.000	0.001	0.995	1.004
TUSD	1.000	0.000	0.997	1.002
BUSD	1.000	0.000	0.998	1.002
sUSD	1.001	0.017	0.807	1.057
USDPAX	1.000	0.002	0.961	1.005

Table 4.2: Summary Statistics – Tokens on Curve Pools. The table reports the summary statistics (mean, standard deviation, and the minimum and maximum observations) of token prices of tokens in the six liquidity pools from Curve. The tokens are quoted in USD. The data is sampled daily, and the sample period is from 04-2021 to 04-2022.

After describing the pools and used tokens on the Curve platform the transaction costs, we discuss risks and returns (APRs). For the sake of exposition, we create an artificial liquidity pool, which we denote as $\bar{\text{C}}$urve pool, where we simply average the quantities (transaction costs, risk measures, and APRs) of the six individual pools on Curve.

4.1.1 Transaction Costs

As outlined previously, transaction costs are relevant determinants of yield farming. To obtain realistic quantities, we analyze the transactions for each of the six Curve pools on etherscan.io, using the respective pool address.[22] We then subset the transaction for the methods *add liquidity* and *remove liquidity* to estimate the transaction costs for depositing and withdrawing liquidity to the respective pool. Although the average transaction cost for removing liquidity is 23 USD, it is slightly more expensive to deposit liquidity amount (34 USD).[23] Transaction costs are highly volatile, displaying volatility in the range of the transaction costs themselves. The peak reveals that paid transaction costs can be easily more than five times as high as the average, exceeding a hundred USD. A graphical

[21] Time-series plots for the individual tokens for each pool are displayed in Figure I.1, which shows a clear deviation from the underlying.

[22] For example, the 3pool transactions can be inferred here: `https://etherscan.io/address/0xbEbc44782C7dB0a1A60Cb6fe97d0b483032FF1C7`.

[23] The transaction costs summary statistics are presented in Table I.2.

representation of the transaction costs is given in Figure 4.1, which displays the average transaction costs for the C̄urve pool.

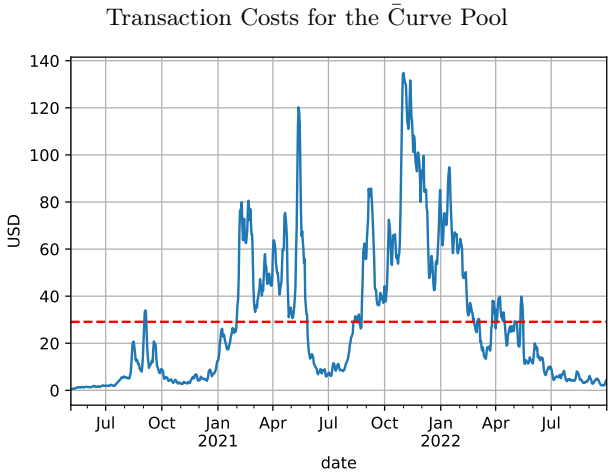

Figure 4.1: Transaction Costs – C̄urve Pool. The figure shows the time-series plot of average (withdrawal and remove liquidity) transaction costs (in USD) of the C̄urve pool (calculated as the average transaction costs from the six Curve pools, as described in Table 4.1). The data is obtained from etherscan.io and sampled daily. The sample period is from 05-2020 to 04-2022. The data is winzorized at the 1% quantile. In the plots, the five-day moving average is depicted.

4.1.2 Risks

In order to shed light on the risks that an agent faces when yield farming, we calculate the impermanent loss the investor faces following equation (3.1).[24] The daily annualized impermanent loss among stablecoins is on average small and ranges from 2 basis points (bps) to 45bps. Inspecting the minimum, it turns out that the impermanent loss can nevertheless reach extreme values of about 80bps.[25] The average impermanent loss of the C̄urve pool is around 10bps, as displayed in Figure 4.2.

4.2 Rewards (APRs)

Next, we discuss the dynamics of the rewards of the yield farmers within the six Curve pools. Average APRs range from almost 0.1% to 5%, with volatility exhibiting a similar magnitude. Particularly interesting are the maximum APRs obtainable, often displaying more than 20% per year, where some pools show a range from 7% to more than 35%.[26] The C̄urve pool, which can be seen as an equally weighted portfolio across the six pools, would have generated an average of 3.1%, with a maximum APR of 24%. The APR dynamics for the C̄urve pool are displayed in Figure 4.3.[27]

To investigate potential *diversification benefits*, we analyze the APR correlation dynamics across the pools. It turns out that the unconditional correlation among all pool APRs is positive. The correlation dynamics are similar, with the lowest average correlation for pax (0.54) and the 3pool (0.54) and a

[24] We calculate the average correlation of the stablecoin pairs returns within the respective pools to provide further insights and to explain the potential impermanent loss. A high correlation among the tokens is favorable to avoid impermanent loss. As seen in Figure I.2, the unconditional average correlation among the token pairs is always positive and ranges between 0.3 and 0.7. In contrast, the conditional average correlation among the token pairs calculated over a rolling window of 30 days ranges from 0.2 to almost 0.9.

[25] Summary statistics of the impermanent loss are provided in Table I.3.

[26] The summary statistics for the historical APRs are presented in Table I.4.

[27] The APR behavior of the six pools over time can be inferred from Figure B.2.

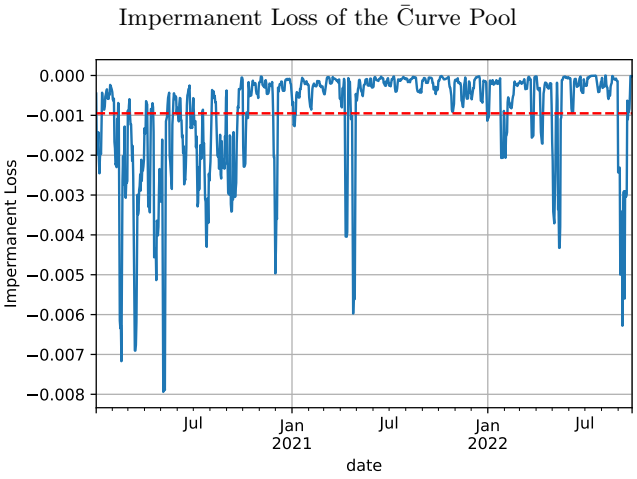

Figure 4.2: Impermanent Loss – C̄urve Pool. The figure shows the time-series plot of the impermanent loss (annualized) of the C̄urve pool as the average impermanent loss from the six Curve pools, as described in Table 4.1. The data is sampled daily, and the sample period is from 02-2020 to 05-2022. The data is winzorized at the 1% quantile. In the plots, the five-day moving average is depicted.

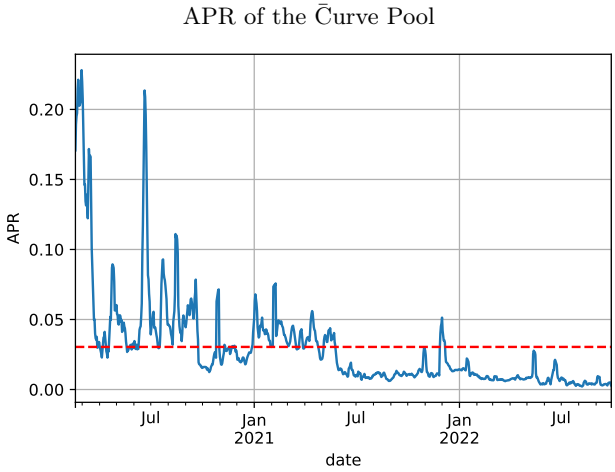

Figure 4.3: APR – C̄urve Pool. The figure shows the time-series plot of the APR of the C̄urve pool as the average APRs from the six Curve pools, as described in Table 4.1. The dotted line represents the average APR. The data is sampled daily, and the sample period ranges from 02-2020 to 10-2022. The data is winzorized at the 1% quantile. In the plots, the five-day moving average is depicted.

larger average correlation for busd (0.61). Hence, one can conclude that the diversification benefits are limited across the analyzed stablecoin pools.[28]

Analyzing the stablecoin pools on Curve reveals various insights about transaction costs (on average 25 USD, with extreme historical values exceeding a few hundred USD), risks (10bps on average but with a minimum of almost –8%), and rewards (on average 3% but easily exceeding 20% on good days). To complement the insights (not only covering stablecoins), in the next section we extend our analysis to pools where any Ethereum-based token can be swapped.

4.3 Uniswap

To reveal additional yield farming insights that do not only provide liquidity for stablecoin pairs, we analyze data from Uniswap V3, which is is the third version of the Uniswap decentralized exchange

[28]The correlation dynamics for the historical APRs are presented in Table I.4.

protocol. We source data from the 100 Uniswap pools with the largest TVLs, using the methodology provided by The Graph. The Graph is a protocol that helps in accessing information on the Ethereum blockchain by allowing users to use a query language called GraphQL. We then calculate the transaction costs, the impermanent loss, and historical APRs for each pool for the period of 5-2021 to 1-2023. The data is aggregated daily.

4.3.1 Transaction Costs

Transaction costs are relevant determinants of yield farming. We therefore analyze the transactions for the largest Uniswap pools. We then subset the transaction for the *Mints* and *Burns* methods to estimate the transaction costs for depositing and withdrawing liquidity to the respective pool. The average transaction cost for removing liquidity is almost 64 USD; it is slightly more expensive to deposit liquidity (83 USD). Transaction costs are highly volatile, displaying volatility in the range of the transaction costs themselves. A graphical representation of the transaction costs is given in Figure 4.4, which displays the average transaction costs for the Ūniswap pools. As visible, the transaction costs are decreasing over time.

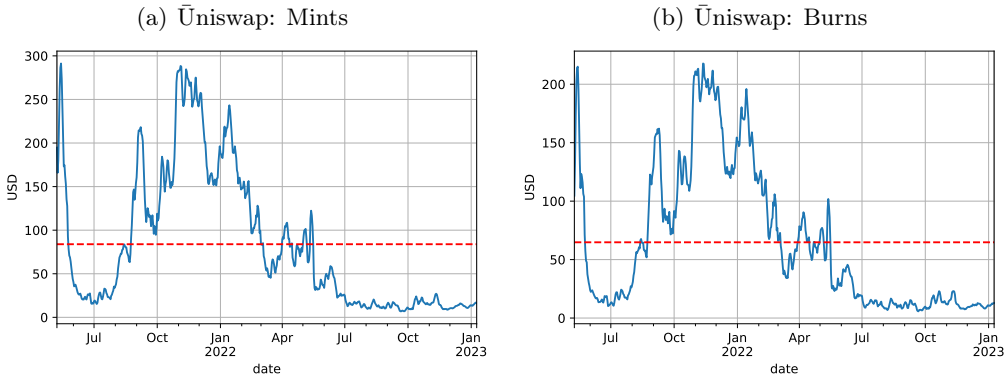

Figure 4.4: Transaction Costs – Ūniswap Pools. The figure shows the time-series plot of average (withdrawal and remove liquidity) transaction costs (in USD) of the Ūniswap pools (calculated as the average transaction costs from the largest Uniswap pools. The data is obtained from The Graph resampled daily. The sample is from 05-2021 to 01-2023. The data is winzorized at the 5% quantile. In the plots, the five-day moving average is depicted.

4.3.2 Risks

The risks for the liquidity pools on Uniswap differ tremendously from those of Curve. Figure 4.5 displays the histogram of the impermanent loss time-series means for individual Uniswap pools. The annualized impermanent loss across the pools is on average –8.3%, with extreme values up to –30%. Therefore, in contrast to the stablecoin liquidity provision, the impermanent loss can be treated as a nonnegligible risk.

4.3.3 Rewards (APRs)

Next, we report the average APRs for the Uniswap pools. Figure 4.6 displays the histogram of the time-series average gross (Panel a) and net APRs (gross APR - impermanent loss) (Panel b). As shown, the mean gross APR is around 13%, with extreme values up to almost 50%. Due to the large

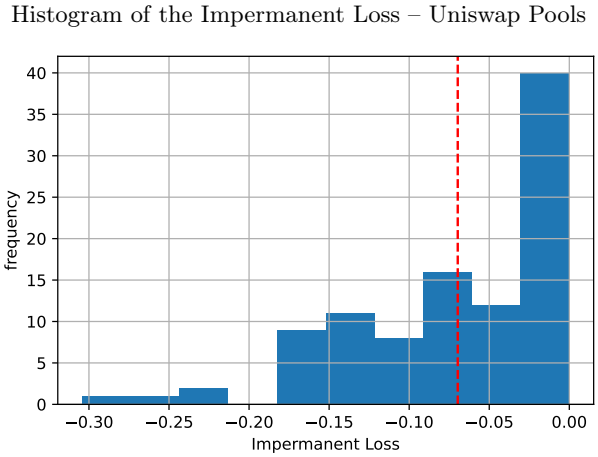

Figure 4.5: Histogram: Impermanent Loss – Uniswap Pools. The figure shows the histogram of the (average) impermanent loss (annualized) of the Uniswap pools. The dotted line represents the average impermanent loss. The data is sampled daily, and the sample period ranges from 05-2021 to 12-2022. The data is winzorized at the 5% quantile.

impermanent loss, the average net APR decreases to only 5% and displays some extremely negative net APRs up to –30%.

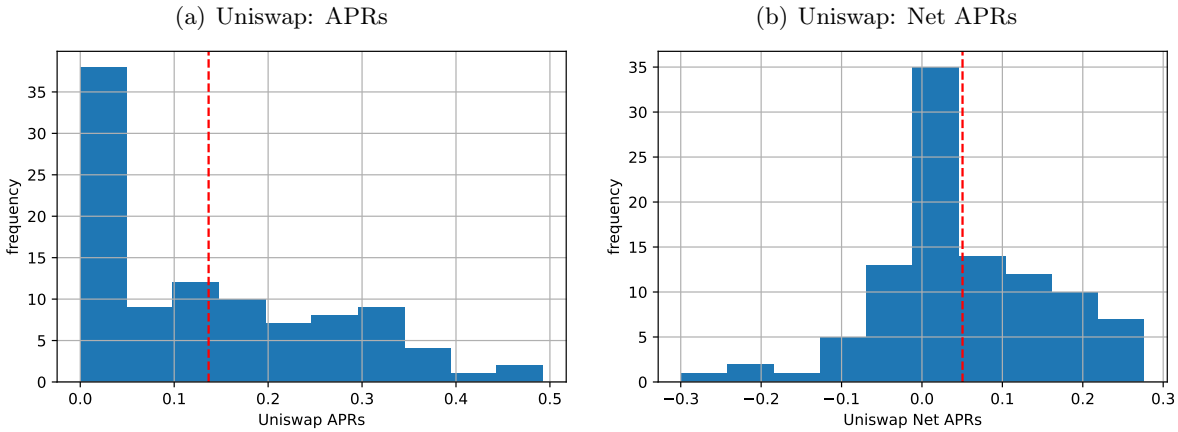

Figure 4.6: Histogram: Uniswap Pool APRs. The figure shows the histogram of the (average) APRs of the Uniswap pools. In Panel a) the gross APRs are displayed, and in Panel b), the net APRs (gross APR - impermanent loss). The dotted line represents the average. The data is sampled daily, and the sample period is from 05-2021 to 12-2022. The data is winzorized at the 1% quantile.

Figure 4.7 displays the gross and net APRs for the $\bar{\text{U}}$niswap pool (as the cross-sectional average across the 100 individual pools) over time. As shown, both the gross and the net APRs are volatile. While the gross APR decreased steadily, the net APR has been moving at approximately 5% since 06-2022.

Next, we analyze the correlation dynamics among the different pool APRs on Uniswap. To do this, we calculate the unconditional pool APR correlations. We then display the 2.5%, the median, and the 97.5% quantile for each pool (the row of the correlation matrix) in Figure 4.8 Panel a). The median correlation is centered at approximately 15%, while the 97.5% reaches values up to almost 80%. Different from the stablecoin pools on Curve, the 2.5% quantile deflects negative values for some of the pools centered at around –20%. Therefore, an agent could potentially use the negative correlation and the accompanying diversification benefits. Figure 4.8 Panel b) plots the quantiles of the

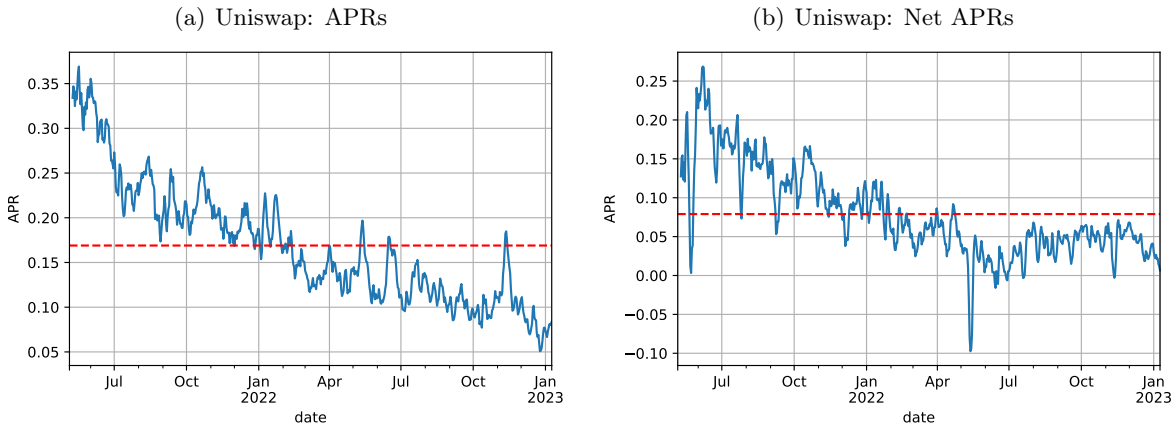

Figure 4.7: Uniswap Pool APRs. The figure shows the cross-sectional average of the APRs of the Uniswap pools ($\bar{\text{U}}$niswap). In Panel a), the gross APRs are displayed and in Panel b) the net APRs (gross APR - impermanent loss). The dotted line represents the average. The data is sampled daily, and the sample period is from 05-2021 to 12-2022. The data is winzorized at the 1% quantile. In the plots, the five-day moving average is depicted.

APR correlation over time, where the correlation is calculated over a rolling window of two weeks. As visible, the quintiles are fairly stable. Nevertheless, they display some increases, especially in months where returns on cryptocurrencies suffered: For example, Bitcoin had its worst months in 05-2021, 06-2012, 05-2022, 06-2022, and November 2022.

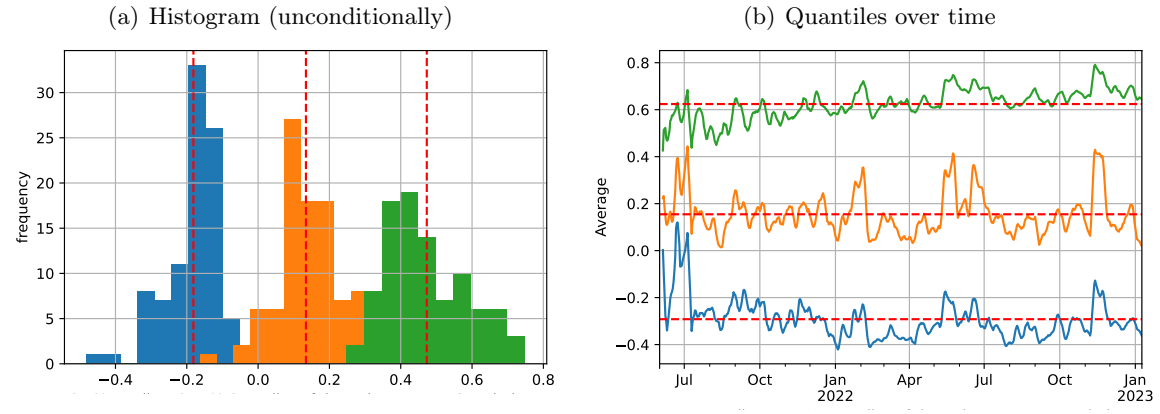

Figure 4.8: Quantiles of the Uniswap Pool APRs Correlation. The figure shows the 2.5% (blue), the median (orange), and the 97.5% (green) quantiles of the correlation among the Uniswap pool APRs: In Panel a), the histogram of the unconditional correlation is displayed, while in Panel b) the quantiles of the correlation are shown over time (calculated over a rolling window of two weeks). The dotted lines represent the respective averages. The data is sampled daily, and the sample period ranges from 05-2021 to 12-2022. The data is winzorized at the 5% quantile.

4.3.4 Risk Return Trade-off

As with any investment, yield farming involves a risk-return trade-off. Higher returns often come with higher risks, and lower risks generally come with lower returns. In the case of yield farming, the risks includes the APR instability and the impermanent loss. In the following, we report the Sharp Ratios (SR) for the Uniswap pools. Figure 4.9 displays the histogram of the gross Sharpe Ratios (Panel a) and the net Sharpe Ratios (Panel b). We calculate the SR for the individual pools as the arithmetic mean return (gross or net APRs) divided by the volatility of the respective APR time-series.

As shown, the mean gross SR is around 87%, with extreme values up to almost 225%. Due to the large impermanent loss, the average net SR decreases to only 30% and displays some extremely negative net SRs up to –50%.

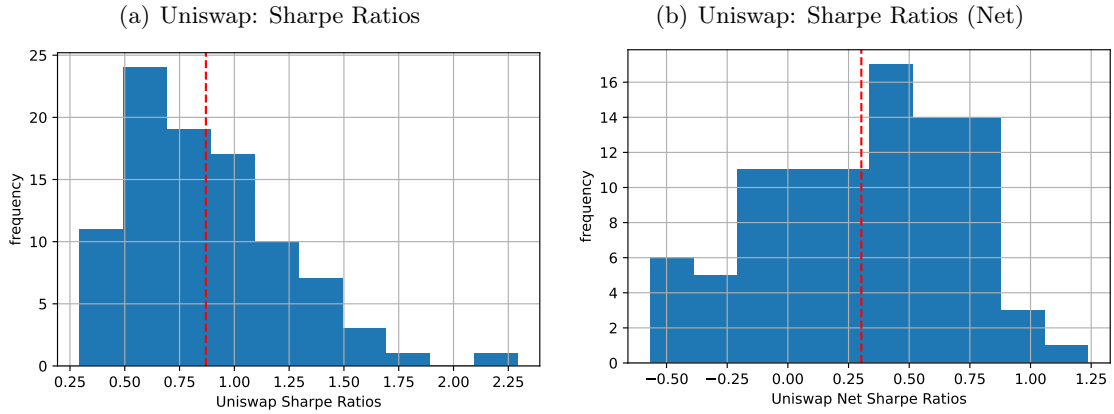

Figure 4.9: Histogram: Sharpe Ratios of the Uniswap Pools. The figure shows the histogram of the Sharpe Ratios (SRs) for the Uniswap pools. In Panel a), the gross SRs are displayed, and in Panel b), the net SRs (gross APR - impermanent loss). The dotted line represents the average. The data is sampled daily, and the sample period is from 05-2021 to 12-2022. The data is winzorized at the 1% quantile.

Analyzing the liquidity pools on Uniswap reveals interesting additional insights about risks (–8.3% on average, with extreme values of almost –30%) and rewards (on average 16% gross and easily exceeding 40% but reduced to less than 10% considering net APRs). As a consequence, yield farming does not necessarily displays an advantageous risk return trade-off represented by the SR. In sharp contrast to the stablecoin pools on Curve, there exist large diversification benefits across liquidity pools on Uniswap, where correlations can reach up to –40%.

5 Stochastic Model for Yield Farming

In this section we formulate the mathematical model of an agent. We start by assuming a pool with a constant product AMM rule,[29] and also that the underlying token prices follow a geometric Brownian motion. These assumptions lead to a characterization of the impermanent loss in terms of the volatility of the tokens' relative price. We then formulate a stochastic control problem to determine an optimal recompounding policy for the agent. To solve this optimization, we take a dynamic programming approach; that is, we derive a Hamilton-Jacobi-Bellman (HJB) partial differential equation (PDE), and then from the solution, we obtain a policy that optimizes the trade-off between adjusting the reward-bearing staked amount and paying the associated transaction costs. As examples, we show the cases of optimal compounding when there are one and two liquidity pools available for staking.

[29]For the sake of simplicity, we do not consider pools with concentrated liquidity, a concept that is explained in Appendix A.2.

5.1 Dynamics under a Constant Product Rule

As to what we do in Section 3.1, let N_t^1 and N_t^2 denote the amounts of two respective tokens that are paired for trading in a liquidity pool. The constant product rule for this liquidity pool is as follows:

$$L \coloneqq \sqrt{N_t^1 N_t^2}, \tag{5.1}$$

where $L > 0$ is a constant. From constant product rule (5.1) a relative price of token emerges,

$$R_t = \frac{N_t^2}{N_t^1} = \text{token 2 per token 1}, \tag{5.2}$$

from which we can deduce the amount of each token in the liquidity pool,

$$N_t^1 = \frac{L}{\sqrt{R_t}}, \qquad N_t^2 = L\sqrt{R_t}. \tag{5.3}$$

We make the following assumption about the market prices and the relative price for tokens in the liquidity pool.

Assumption 5.1. *The relative token price in a liquidity pool is equal to the ratio of the market prices of tokens,*

$$R_t = \frac{P_t^1}{P_t^2} = \frac{\textit{dollar per token 1}}{\textit{dollar per token 2}} = \textit{token 2 per token 1}, \tag{5.4}$$

where P_t^i for $i = 1, 2$ are the market prices of tokens outside of the liquidity pool.

The rationale for equation (5.4) is that if it were not true, then arbitrageurs would enter the liquidity pool and exploit the price discrepancy until it would correct itself. Thus, arbitrageurs will ensure that the ratio of N_t^1 to N_t^2 in the pool maintains a relative price equal to the relative price of the greater market external of the pool.

Our model takes token prices to be given by geometric Brownian motions (GBMs),

$$dP_t^i = \mu_i P_t^i dt + \sigma_i P_t^i dB_t^i, \quad i = 1, 2, \tag{5.5}$$

where B_t^i for $i = 1, 2$ are two correlated standard Brownian motions that are defined on a filtered probability space $\left(\Omega, \mathcal{F}, (\mathcal{F}_t)_{t\geqslant 0}, \mathbb{P}\right)$, $dB_t^1 dB_t^2 = \rho dt$, $\rho \in [-1, 1]$ is the correlation between them, and $\sigma_i > 0$. We apply the Itô-Doeblin formula to the ratio of P_t^1 and P_t^2 to get the stochastic differential equation (SDE) for the relative price R_t,

$$\begin{aligned} dR_t = d\left(\frac{P_t^1}{P_t^2}\right) &= \left(\mu_1 - \mu_2 + \sigma_2^2 - \sigma_1\sigma_2\rho\right) R_t dt + \sigma_1 R_t dB_t^1 - \sigma_2 R_t dB_t^2 \\ &= \tilde{\mu} R_t dt + \tilde{\sigma} R_t d\tilde{B}_t, \end{aligned} \tag{5.6}$$

where $\tilde{\mu} = \mu_1 - \mu_2 + \sigma_2^2 - \sigma_1\sigma_2\rho$, $\tilde{\sigma}^2 = \sigma_1^2 - 2\rho\sigma_1\sigma_2 + \sigma_2^2$, and $\tilde{B}_t = \frac{\sigma_1}{\tilde{\sigma}} B_t^1 - \frac{\sigma_2}{\tilde{\sigma}} B_t^2$. By applying the Itô-Doeblin formula to the ansatz $\ln R_t$, we can solve equation (5.6) and get the expression for R_t, as

follows:

$$R_t = R_0 \exp\left(\tilde{\mu}t - \frac{1}{2}\tilde{\sigma}^2 t + \tilde{\sigma}\tilde{B}_t\right). \tag{5.7}$$

Next, we can straightforwardly derive the expressions for the amounts of the tokens N_t^1 and N_t^2 by using equation (5.3) and equation (5.7), as follows:

$$\begin{aligned} N_t^1 &= \frac{L}{\sqrt{R_t}} = \frac{L}{\sqrt{R_0}} \exp\left(-\frac{1}{2}\tilde{\mu}t + \frac{1}{4}\tilde{\sigma}^2 t - \frac{\tilde{\sigma}^2}{2}\tilde{B}_t\right), \\ N_t^2 &= L\sqrt{R_t} = L\sqrt{R_0} \exp\left(\frac{1}{2}\tilde{\mu}t - \frac{1}{4}\tilde{\sigma}^2 t + \frac{\tilde{\sigma}^2}{2}\tilde{B}_t\right). \end{aligned} \tag{5.8}$$

Then, by applying the Itô-Doeblin formula again to equation (5.8), we can get the following SDEs for the amount of each token in the liquidity pool,

$$\begin{aligned} dN_t^1 &= d\left(\frac{L}{\sqrt{R_t}}\right) = \left(-\frac{\tilde{\mu}}{2} + \frac{3\tilde{\sigma}^2}{8}\right) N_t^1 dt - \frac{\tilde{\sigma}}{2} N_t^1 d\tilde{B}_t, \\ dN_t^2 &= d\left(L\sqrt{R_t}\right) = \left(\frac{\tilde{\mu}}{2} - \frac{\tilde{\sigma}^2}{8}\right) N_t^2 dt + \frac{\tilde{\sigma}}{2} N_t^2 d\tilde{B}_t. \end{aligned} \tag{5.9}$$

5.2 The Impermanent Loss

The fundamental risk faced by yield farmers is the impermanent loss. As described in Section 3.4.2 and Appendix A.4, the impermanent loss is the value of a position in staked tokens minus the value of an unstaked position that initially had the equivalent value as the staked position, and then the difference divided by the value of the unstaked position. At time t, let $\mathcal{V}_{t,s}^{\text{staked}}$ denote the dollar value staked in the pool at time $s \geqslant t$, and let $\mathcal{V}_{t,s}^{\text{held}}$ denote the value of an un-staked position that has an equal dollar amount as time t; that is $\mathcal{V}_{t,t}^{\text{staked}} = \mathcal{V}_{t,t}^{\text{held}}$. For a time increment $\Delta t > 0$, at time $t + \Delta t$ the impermanent loss is defined as follows.

Definition 5.1. *For a time increment $\Delta t > 0$, the impermanent loss from time t to time $t + \Delta t$ is the staked value minus the held value, divided by the held value, as follows:*

$$\Delta I_t^{\Delta t} := \frac{\mathcal{V}_{t,t+\Delta t}^{staked} - \mathcal{V}_{t,t+\Delta t}^{held}}{\mathcal{V}_{t,t+\Delta t}^{held}} = \frac{\left(N_{t+\Delta t}^1 P_{t+\Delta t}^1 + N_{t+\Delta t}^2 P_{t+\Delta t}^2\right) - \left(N_t^1 P_{t+\Delta t}^1 + N_t^2 P_{t+\Delta t}^2\right)}{N_t^1 P_{t+\Delta t}^1 + N_t^2 P_{t+\Delta t}^2}, \tag{5.10}$$

where N_t^i and P_t^i for $i = 1, 2$ are the amounts of tokens and the market prices of tokens, respectively, as defined in Section 5.1.

In terms of $\Delta I_t^{\Delta t}$, which is defined by equation (5.10), the total impermanent loss up to time t is the summation to time t,

$$I_t^{\Delta t} := \sum_{t_i < t} \Delta I_{t_i}^{\Delta t},$$

where $t_i = i\Delta t$ for $i = 0, 1, 2, 3, \cdots, \lfloor t/\Delta t \rfloor - 1$. Total impermanent loss in continuous time is obtained

by taking the limit as Δt goes to zero,

$$I_t \coloneqq \lim_{\Delta t \to 0} I_t^{\Delta t}. \tag{5.11}$$

In discrete time it is straightforward to show that $\Delta I_t^{\Delta t} \leqslant 0$ (see Heimbach et al. (2022)), and in continuous time it is also true that impermanent loss is always nonpositive.

Proposition 5.1. *In continuous time, impermanent loss is the differential of total impermanent loss I_t, which is defined in equation (5.11), and which is equal to $-\frac{1}{8}$ times the variance of the relative price times the length of time increment,*

$$dI_t = -\frac{\tilde{\sigma}^2}{8} dt, \tag{5.12}$$

where $\tilde{\sigma} = \sqrt{\sigma_1^2 - 2\rho\sigma_1\sigma_2 + \sigma_2^2}$ is the volatility of the relative price R_t given by equation (5.6). From equation (5.12) it is clear that $dI_t \leqslant 0$ for all $t \geqslant 0$.

Proof. By the relative price that is defined in equation (5.2) and Assumption 5.1, we have $\frac{N_t^2}{N_t^1} = R_t = \frac{P_t^1}{P_t^2}$. From equation (5.6), we can calculate $d\tilde{B}_t dB_t^1 = \frac{1}{\tilde{\sigma}}(\sigma_1 - \sigma_2\rho)\,dt$ and $d\tilde{B}_t dB_t^2 = \frac{1}{\tilde{\sigma}}(\sigma_1\rho - \sigma_2)\,dt$. Based on equation (5.5) and equation (5.9), by using the Itô-Doeblin formula, the limit as $\Delta t \to 0$ of the numerator in equation (5.10) is

$$\begin{aligned}
&d\left(N_t^1 P_t^1 + N_t^2 P_t^2\right) - \left(N_t^1 dP_t^1 + N_t^2 dP_t^2\right) \\
&= N_t^1 dP_t^1 + P_t^1 dN_t^1 + dN_t^1 dP_t^1 - N_t^1 dP_t^1 + N_t^2 dP_t^2 + P_t^2 dN_t^2 + dN_t^2 dP_t^2 - N_t^2 dP_t^2 \\
&= -\frac{\tilde{\sigma}^2}{4} N_t^1 P_t^1 dt.
\end{aligned}$$

Because this numerator has only a dt term, we can forgo application of Ito-Doeblin for the reciprocal term in equation (5.10) and instead simply take the limit of denominator as Δt goes to zero, giving us

$$dI_t = \frac{d\left(N_t^1 P_t^1 + N_t^2 P_t^2\right) - \left(N_t^1 dP_t^1 + N_t^2 dP_t^2\right)}{N_t^1 P_t^1 + N_t^2 P_t^2} = \frac{-\frac{\tilde{\sigma}^2}{4} N_t^1 P_t^1 dt}{N_t^1 P_t^1 + N_t^2 P_t^2}. \tag{5.13}$$

Then, by applying equation (5.2) and Assumption 5.1, equation (5.13) is reduced to equation (5.12). □

Remark 5.1. *Equation (5.12) is the continuous-time equivalent to the definition of the impermanent loss given in Heimbach et al. (2022) that calculated returns in terms of relative price only. More details on the equivalence of our definition with theirs are given in Appendix C.*

Remark 5.2. *Our continuous-time definition for impermanent loss has some similarities to loss-verses-rebalancing (LVR) introduced in Milionis et al. (2022). LVR quantifies the liquidity provider's inability to keep an ideal ratio of token 1 to token 2. Using the notations of this article and the constant product rule of equation (5.1), LVR is*

$$LVR_t = \int_0^t \left(N_s^1 dP_s^1 + N_s^2 dP_s^2\right) - \left(N_t^1 P_t^1 + N_t^2 P_t^2\right),$$

and from calculations similar to those done in Proposition 5.1 we get $dLVR_t = \frac{\sigma^2}{8}\left(N_t^1 P_t^1 + N_t^2 P_t^2\right) dt$.

LVR can also be used to express the difference of staked and held values in Definition 5.1,

$$\mathcal{V}^{staked}_{t,t+\Delta t} - \mathcal{V}^{held}_{t,t+\Delta t} = -\int_t^{t+\Delta t} dLVR_s + \int_t^{t+\Delta t} \left(N^1_s - N^1_t\right) dP^1_s + \int_t^{t+\Delta t} \left(N^2_s - N^2_t\right) dP^2_s,$$

where we adhere to convention $\mathcal{V}^{staked}_{t,t} = \mathcal{V}^{held}_{t,t} = N^1_t P^1_t + N^2_t P^2_t$. *In the limit as* Δt *tends to zero, it can be shown that* $\sum_{t_i<t} \frac{1}{\mathcal{V}^{held}_{t_i,t_i}} \int_{t_i}^{t_{i+1}} \left(N^j_s - N^j_{t_i}\right) dP^j_s \to 0$ *in mean square for* $j = 1, 2$, *and thus the total impermanent loss in equation* (5.11) *is* $I_t = -\int_0^t \frac{dLVR_s}{N^1_s P^1_s + N^2_s P^2_s} = -\frac{\sigma^2 t}{8}$.

5.3 Optimal Liquidity Provision

As explained in Section 3.5, yield farming means staking tokens and receiving earnings in the form of trading fees. The earned trading fees are credited to the (digital) wallet (and not automatically to the staked account). Hence, to increase the staked amount for (eventual) increased rewards, the yield farmer must first manually recompound the interest by transferring it from the wallet to the pool, an action that incurs transaction costs. Below, we build on the mathematical framework laid out in Section 5.2 and Proposition 5.2 to formulate a stochastic control problem for the optimal yield farming investment.

5.3.1 Portfolio Value with Zero Transaction Costs

In the setting of a single liquidity pool, a portfolio for liquidity provision consists of tokens staked and tokens held in a wallet. Let us define the holdings in our portfolio as follows:

$$\begin{aligned} h_t &= \text{fraction of tokens of the liquidity pool that the agent stakes,} \\ n^1_t &= \text{number of tokens held un-staked in coin 1,} \\ n^2_t &= \text{number of tokens held un-staked in coin 2.} \end{aligned}$$

Let Π_t denote the portfolio value for the yield farmer, and consider its increment of change,

$$\begin{aligned} d\Pi_t = &\underbrace{a_t h_t \left(N^1_t P^1_t + N^2_t P^2_t\right) dt}_{\text{reward for staking tokens}} + \underbrace{h_t d\left(N^1_t P^1_t + N^2_t P^2_t\right)}_{\text{change in staked tokens}} + \underbrace{n^1_t dP^1_t + n^2_t dP^2_t}_{\text{change in un-staked tokens}} \\ &+ \underbrace{r\left(\Pi_t - \left(h_t N^1_t + n^1_t\right) P^1_t - \left(h_t N^2_t + n^2_t\right) P^2_t\right) dt}_{\text{financing costs}}, \end{aligned} \tag{5.14}$$

where $a_t \geqslant 0$ is the APR for staking tokens back into the liquidity pool, and $r \geqslant 0$ is the rate for borrowing/lending in these funds. As we can observe, the unit for each of the four parts in equation (5.14) is dollar value. We next show that the return for yielding farming is a function of the impermanent loss.

Proposition 5.2. *The increment of change for the portfolio* Π_t *is the following differential:*

$$d\Pi_t = \underbrace{h_t \left(a_t - \frac{\tilde{\sigma}^2}{8}\right) \left(N^1_t P^1_t + N^2_t P^2_t\right) dt}_{\textit{net return from staking tokens}} + \underbrace{\left(h_t N^1_t + n^1_t\right) dP^1_t + \left(h_t N^2_t + n^2_t\right) dP^2_t}_{\textit{profit and loss in tokens}} \tag{5.15}$$

$$+\underbrace{r\left(\Pi_t-\left(h_tN_t^1+n_t^1\right)P_t^1-\left(h_tN_t^2+n_t^2\right)P_t^2\right)dt}_{\textit{financing costs}},$$

where $-\frac{\tilde{\sigma}^2}{8}dt$ *is the impermanent loss given by equation* (5.12).

Remark 5.3. *Equation* (5.15) *is a fairly typical equation for returns, containing three different terms: a net rate* $a_t-\frac{\tilde{\sigma}^2}{8}$ *on the value of staked tokens, a return on tokens both staked and un-staked, and the costs for financing.*

Proof. Rearranging the terms of equation (5.13)*, we have*

$$d\left(N_t^1P_t^1+N_t^2P_t^2\right)=\left(N_t^1P_t^1+N_t^2P_t^2\right)dI_t+\left(N_t^1dP_t^1+N_t^2dP_t^2\right). \tag{5.16}$$

Using equation (5.12) *of Proposition 5.1, we plug formula* (5.16) *into equation* (5.14) *and get*

$$\begin{aligned}
d\Pi_t &= h_t\left(N_t^1P_t^1+N_t^2P_t^2\right)\left(a_tdt+dI_t\right)+\left(h_tN_t^1+n_t^1\right)dP_t^1+\left(h_tN_t^2+n_t^2\right)dP_t^2\\
&\quad+r\left(\Pi_t-\left(h_tN_t^1+n_t^1\right)P_t^1-\left(h_tN_t^2+n_t^2\right)P_t^2\right)dt,\\
&= h_t\left(a_t-\frac{\tilde{\sigma}^2}{8}\right)\left(N_t^1P_t^1+N_t^2P_t^2\right)dt+\left(h_tN_t^1+n_t^1\right)dP_t^1+\left(h_tN_t^2+n_t^2\right)dP_t^2\\
&\quad+r\left(\Pi_t-\left(h_tN_t^1+n_t^1\right)P_t^1-\left(h_tN_t^2+n_t^2\right)P_t^2\right)dt.
\end{aligned}$$

Therefore, in terms of the impermanent loss $-\frac{\tilde{\sigma}^2}{8}dt$*, the increment of change for portfolio* Π_t *is given by equation* (5.15). □

For simplicity and elegance of exposition and without losing too much generality, we assume that $\mu_1=\mu_2=0$ in equation (5.5) and $r=n_t^1=n_t^2=0$ in equation (5.15). We also assume that the APR is stochastic and driven by a one-dimensional Ornstein-Uhlenbeck process X_t,

$$\begin{aligned}
a_t &= a\left(X_t\right)\\
dX_t &= -\kappa X_tdt+\sigma dW_t,
\end{aligned} \tag{5.17}$$

where W_t is an independent $\mathcal{F}_t$-measurable one-dimensional standard Brownian motion, $\kappa>0$, and $\sigma>0$. The APR function $a\left(X_t\right)$ will be specified according to different types of liquidity pools; the choice of $a(X_t)$ will be discussed further in Section 6.[30]

By the constant product rule (5.1), the relative price equation (5.2), Assumption 5.1, the various results from Section 5.2 for token prices and their relative price, and plugging in equation (5.5), the increment of change for the portfolio value Π_t given by equation (5.15) becomes

$$\begin{aligned}
d\Pi_t &= h_t\left(a\left(X_t\right)-\frac{\tilde{\sigma}^2}{8}\right)\left(N_t^1P_t^1+N_t^2P_t^2\right)dt+h_tN_t^1dP_t^1+h_tN_t^2dP_t^2\\
&= \left(a\left(X_t\right)-\frac{\tilde{\sigma}^2}{8}\right)S_tdt+S_t\left(\sigma_1dB_t^1+\sigma_2dB_t^2\right)\\
&= \left(a\left(X_t\right)-\frac{\tilde{\sigma}^2}{8}\right)S_tdt+\hat{\sigma}S_td\hat{B}_t,
\end{aligned} \tag{5.18}$$

[30] In contrast to Cong et al. (2022), we assume that our APR is given as exogenous and hence, the rate of staking rewards that an agent earns is not influenced by other agents' behavior.

where $\hat{\sigma}^2 = \sigma_1^2 + 2\rho\sigma_1\sigma_2 + \sigma_2^2$, and $\hat{B} = \frac{\sigma_1}{\hat{\sigma}} B_t^1 + \frac{\sigma_2}{\hat{\sigma}} B_t^2$.

5.3.2 Single-Pool Optimization with Stochastic APR and Transaction Costs

Let K_t denote the dollar value held un-staked in a wallet, and S_t the dollar value staked in the pool. The portfolio value is expressed as

$$\Pi_t = K_t + S_t,$$

where K_t collects the rewards from staking and $S_t = h_t \left(N_t^1 P_t^1 + N_t^2 P_t^2\right)$ takes any impermanent loss. Equation (5.18) is the change in portfolio value if a liquidity provider pays no transaction costs. If there are transactions costs then the movement of tokens between the wallet and the liquidity pool involves controls ω_t and ν_t, with the following pair of differentials for changes in K_t and S_t,

$$\begin{aligned} dK_t &= \left(a\left(X_t\right) S_t - \omega_t + \nu_t\right) dt, \\ dS_t &= \left(\omega_t - \nu_t - c\mathbf{1}_{\{\omega_t + \nu_t > 0\}} - \frac{\tilde{\sigma}^2}{8} S_t\right) dt + \hat{\sigma} S_t d\hat{B}_t, \end{aligned} \tag{5.19}$$

where $\hat{B}_t$ is the Brownian motion defined below equation (5.18) and is independent from W_t in equation (5.17), $\hat{\sigma}$ is given by equation (5.18), $\omega_t \in [0,\, \bar{\omega}]$, $\nu_t \in [0,\, \bar{\nu}]$, $c > 0$ is the constant transaction cost, and $\mathbf{1}_{\{.\}}$ is an indicator function. The control variable ω_t represents at each time t the amount of the rewarded token that is recompounded back into the liquidity pool, and the control variable ν_t represents the amount of the token that is taken out of the liquidity pool.

We can see from equation (5.19), the wallet K_t changes due to three terms: Credited rewards $a\left(X_t\right) S_t$, the amount deposited into the liquidity pool ω_t, and the amount withdrawn from the liquidity pool ν_t. The dynamics of the staked amount S_t intertwines with the just-discussed dynamics of the wallet: The ω_t, which is withdrawn from K_t, is deposited into S_t, the ν_t, which is withdrawn from S_t, is deposited into K_t, and these actions cause a fixed transaction cost. Therefore, the system (5.19) are the dynamics for the yield farming process that was described in Section 3.5.

The control processes ω_t and ν_t are progressively measurable processes with respect to the filtration $\left(\mathcal{F}_t\right)_{t \geqslant 0}$ and are sought to maximize the expectation of a utility function $U\left(\cdot\right)$ with respect to the wallet K_t and the staking account S_t at terminal time $t = T$. This maximization is posed as the following stochastic control problem:

$$Q\left(t,\, x,\, k,\, s\right) = \sup_{(\omega,\, \nu) \in \mathcal{A}} \mathbb{E}\left[U\left(K_T,\, S_T\right) \mid X_t = x,\, K_t = k,\, S_t = s\right], \tag{5.20}$$

where $\left(X_t,\, K_t,\, S_t\right)$ follow equations (5.17) and (5.19), $t \in [0,\, T]$ is the time variable, $(x,\, k,\, s) \in \mathbb{R} \times \mathbb{R}^+ \times \mathbb{R}^+$, and $\mathcal{A}$ is the following set of admissible controls:

$$\mathcal{A} = \left\{\left(\omega_t,\, \nu_t\right)_{0 \leqslant t \leqslant T} \;\middle|\; \left(\omega_t,\, \nu_t\right) \text{is } \mathcal{F}_t\text{-measurable with } \omega_t \in [0,\, \bar{\omega}] \text{ and } \nu_t \in [0,\, \bar{\nu}]\right\},$$

where the control bounds $\bar{\omega}$ and $\bar{\nu}$ are parameters such that $0 < \bar{\omega} < \infty$ and $0 < \bar{\nu} < \infty$.

We take the utility function to be a mixture of power utilities of the wallet K_t and the staked

amount S_t,

$$U\left(K_t,\, S_t\right) = \frac{1}{\gamma_k} K_t^{\gamma_k} + \frac{\eta}{\gamma_s} S_t^{\gamma_s}, \tag{5.21}$$

where $\gamma_k \in (0,\, 1]$ and $\gamma_s \in (0,\, 1)$ are the risk tolerance coefficients, and the parameter $\eta > 0$ weighs the utility of S_t against the utility of K_t. For the yield farming problem under consideration, it makes sense to consider $\gamma_s < \gamma_k$ because a staked position in the pool is riskier than an equal number of tokens in the wallet, and at time T, we cannot instantaneously liquidate S_t to the wallet because of transaction costs. In fact, in the utility function (5.21), we shall allow $\gamma_k = 1$ because it will simplify the stochastic control problem for numerical simulations, thereby allowing us to focus on the risky behavior associated with the decision to recompound claimed rewards. Theoretically, $\gamma_k = 1$ is justifiable because the staked position S_t is considerably more risky than the K_t position. Moreover, K_t can only be driven up very quickly by a large position in S_t; thus, because there are transaction costs and we cannot instantaneously exit a staked position, an overly aggressive appetite for risk in the wallet will be tempered by concave risk tolerance on staking.

For the optimization problem posed in equation (5.20), we can apply the dynamic programming principle to obtain an HJB equation for the optimal value function

$$Q_t + \mathcal{L}Q + \sup_{\substack{\omega \in [0,\, \bar{\omega}] \\ \nu \in [0,\, \bar{\nu}]}} \left[\left(a\left(x\right) s - \left(\omega - \nu\right)\right) Q_k + \left(\left(\omega - \nu\right) - c\mathbf{1}_{\{\omega + \nu > 0\}} - \frac{\tilde{\sigma}^2}{8} s \right) Q_s \right] = 0, \tag{5.22}$$

where $\mathcal{L}Q = \frac{\sigma^2}{2} Q_{xx} - \kappa x Q_x$, and the terminal condition is $Q\left(T,\, x,\, k,\, s\right) = \frac{1}{\gamma_k} k^{\gamma_k} + \frac{\eta}{\gamma_s} s^{\gamma_s}$. The Hamiltonian in equation (5.22) is linear with respect to ω and ν, which leads to the optimal controls being of a bang-bang type, as follows:

$$\omega = \begin{cases} \bar{\omega}, & \text{if } \left(\bar{\omega} - c\right) Q_s > \bar{\omega} Q_k, \\ 0, & \text{otherwise.} \end{cases} \quad \text{and} \quad \nu = \begin{cases} \bar{\nu}, & \text{if } -\left(\bar{\nu} + c\right) Q_s > -\bar{\nu} Q_k, \\ 0, & \text{otherwise.} \end{cases}$$

The fact that the controls are of a bang-bang type represents the first important results of this section and is in line with the intuition: If the agent faces fixed transaction costs c and it is optimal to either recompound into S_t by taking out from K_t, which is $\omega > 0$, or to un-stake from S_t and deposit into K_t, which is $\nu > 0$, then it is optimal to recompound or un-stake the largest amounts possible, respectively.

Case for $\gamma_k = 1$.

When $\gamma_k = 1$ in the utility function (5.21) for the single liquidity pool case, the variable k can be factored out of the solution by using the following ansatz,

$$Q\left(t,\, x,\, k,\, s\right) = k + V\left(t,\, x,\, s\right).$$

We define parameters $\lambda_\omega \in (0,\, 1)$ and $\lambda_\nu \in (0,\, 1)$ to be transaction cost coefficients such that $c = \lambda_\omega \bar{\omega}$ and $c = \lambda_\nu \bar{\nu}$. Then equation for the staked amount defined in formula (5.19) becomes $dS_t = \left(\left(1 - \lambda_\omega\right) \omega_t - \left(1 + \lambda_\nu\right) \nu_t - \frac{\tilde{\sigma}^2}{8} S_t \right) dt + \hat{\sigma} S_t d\hat{B}_t$. Consequently, equation (5.22) can be written

succinctly as

$$V_t + \mathcal{L}V + \frac{\hat{\sigma}^2 s^2}{2} V_{ss} \tag{5.23}$$
$$+ \sup_{\substack{\omega \in \{0,\, \bar{\omega}\} \\ \nu \in \{0,\, \bar{\nu}\}}} \left[\left(a\left(x\right) s - \left(\omega - \nu\right) \right) + \left(\left(1 - \lambda_\omega\right) \omega - \left(1 + \lambda_\nu\right) \nu - \frac{\tilde{\sigma}^2}{8} s \right) V_s \right] = 0,$$

where the operator $\mathcal{L}$ is defined as it is in equation (5.22), and the terminal condition is $V\left(T,\, x,\, s\right) = \frac{\eta}{\gamma} s^\gamma$. From the Hamiltonian in equation (5.23) we obtain the following optimal bang-bang controls:

$$\omega = \begin{cases} \bar{\omega}, & \text{if } \left(1 - \lambda_\omega\right) \bar{\omega} V_s > \bar{\omega}, \\ 0, & \text{otherwise.} \end{cases} \quad \text{and} \quad \nu = \begin{cases} \bar{\nu}, & \text{if } -\left(1 + \lambda_\nu\right) \bar{\nu} V_s > -\bar{\nu}, \\ 0, & \text{otherwise.} \end{cases} \tag{5.24}$$

5.3.3 Two-Pool Optimization with Stochastic APR and Transaction Costs

We now generalize the single liquidity pool optimization of Section 5.3.2 to two liquidity pools. However, please note that this case of two liquidity pools can be easily extended for the case of three or more liquidity pools. Comparing this with the single liquidity pool, there are now four tokens P^i with four Brownian motions B^i for $i = 1,\, 2,\, 3,\, 4$, and four sets of GBM parameters, etc., but the only difference for the optimal yield farming problem is that the credited earning fees are represented by the summation of the two earnings, which are generated from the APRs for of separate liquidity pools. The stochastic system for the liquidity pools is a modification of that in equation (5.17) and equation (5.19),

$$\begin{aligned} dK_t &= \sum_{i=1}^{2} \left(a_i \left(X_t^i \right) S_t^i - w_t^i + \nu_t^i \right) dt, \\ dS_t^i &= \left(w_t^i - \nu_t^i - c_i \mathbf{1}_{\{w_t^i > 0\}} - \frac{\tilde{\sigma}_i^2}{8} S_t^i \right) dt + \hat{\sigma}_i S_t^i d\hat{B}_t^i, \text{ for } i = 1,\, 2 \\ dX_t^j &= -\kappa_j X_t^j dt + \sigma_j dW_t^j, \text{ for } j = 1,\, 2 \end{aligned} \tag{5.25}$$

where $\hat{B}_t^i$ and W_t^j for $i,\, j = 1,\, 2$ are four mutually independent one-dimensional Brownian motions that are defined on a filtered probability space $\left(\Omega,\, \mathcal{F},\, \left(\mathcal{F}_t\right)_{t \geqslant 0},\, \mathbb{P}\right)$, $\omega_t^i \in [0,\, \bar{\omega}_i]$, $\kappa_j > 0$, $\sigma_j > 0$, $c_i > 0$ are the constant transaction costs for the two different liquidity pools, and $\mathbf{1}_{\{\cdot\}}$ is an indicator function. The control variables ω_t^i represent at each time t the amounts of the un-staked i^{th} token that are recompounded back into the i^{th} liquidity pool. For simplicity and elegance of exposition, we assume that investors do not take tokens out of the staking accounts for the case of two liquidity pools; in other words $\nu_t^i \equiv 0\ \forall t$. Functions $a_i\left(X_t^i\right)$ are the APRs for the two liquidity pools, which will be specified in Section 6, when we implement numerical solutions of the HJB equations.

In reality, the correlation between APRs might not be zero. In other words, $dW_t^1 dW_t^2 = \rho_x dt$, where $\rho_x \in [-1,\, 1]$. Our reason for assuming $dW_t^1 \perp dW_t^2$ is that numerical methods for solving these HJB equations are considerably more complicated if $\rho_x \neq 0$. However, we still can model the correlation between the APRs under the assumption $dW_t^1 \perp dW_t^2$. Suppose $\kappa_1 = \kappa_2 = \kappa$, in which case the summation of the two OU processes from equation (5.25) is also an OU process. Thus, to model the correlation between the APRs of the two liquidity pools, we modify the equation for wallet

K_t in equation (5.25), as follows:

$$dK_t = \sum_{i=1}^{2} \left(a_i \left(Y_t^i \right) S_t^i - w_t^i + \nu_t^i \right) dt,$$

where $Y_t^1 = X_t^1$ and $Y_t^2 = \rho_x X_t^1 + \sqrt{1-\rho_x^2} X_t^2$, and thus, Y_t^1 and Y_t^2 are correlated OU processes. Later in Section 6, when we calibrate the OU processes to data, we shall see that mean-reversion rates are comparable, and so it will be reasonable for us to assume that $\kappa_1 = \kappa_2$.

The control process ω_t^i for $i = 1, 2$ of each liquidity pool is a progressively measurable process with respect to the filtration $(\mathcal{F}_t)_{t \geqslant 0}$, which is sought to maximize the expectation of a utility function $U(\cdot)$ that is a function of K_t and S_t^i at terminal time $t = T$. The value function for this stochastic control problem is

$$\begin{aligned} &Q(t,\, x_1,\, x_2,\, k,\, s_1,\, s_2) = \\ &\quad \sup_{(\omega_t^1, \omega_t^2) \in \mathcal{A}} \mathbb{E}\left[U\left(K_T,\, S_T^1,\, S_T^2 \right) \,\middle|\, X_t^1 = x_1,\, X_t^2 = x_2,\, K_t = k,\, S_t^1 = s_1,\, S_t^2 = s_2 \right], \end{aligned} \tag{5.26}$$

where $\left(X_t^1,\, X_t^2,\, K_t,\, S_t^1,\, S_t^2 \right)$ follow equation (5.25), $t \in [0,\, T]$ is the time variable, $(x_1,\, x_2,\, k,\, s_1,\, s_2) \in \mathbb{R} \times \mathbb{R} \times \mathbb{R}^+ \times \mathbb{R}^+ \times \mathbb{R}^+$, and $\mathcal{A}$ is the set of admissible controls,

$$\mathcal{A} = \left\{ \left(\omega_t^1,\, \omega_t^2 \right)_{0 \leqslant t \leqslant T} \;\middle|\; \left(\omega_t^1,\, \omega_t^2 \right) \text{is } \mathcal{F}_t\text{-measurable with } \omega_t^i \in [0,\, \bar{\omega}_i] \text{ for } i = 1,\, 2 \right\},$$

where the bounds of the control variables $\bar{\omega}_i$ for $i = 1, 2$ are parameters such that $0 < \bar{\omega}_i < \infty$.

Similar to the utility function (5.21) in the single liquidity pool model of Section 5.21, for two liquidity pools we still choose the utility function to be a mixture of power utilities of the wallet K_t and the staking amounts S_t^1 and S_t^2,

$$U\left(K_t,\, S_t^1,\, S_t^2 \right) = \frac{1}{\gamma_k} K_t^{\gamma_k} + \frac{\eta}{\gamma_s} \left(S_t^1 + S_t^2 \right)^{\gamma_s}, \tag{5.27}$$

where $\gamma_k \in (0,\, 1]$ and $\gamma_s \in (0,\, 1)$ are the risk tolerance coefficients, and $\eta > 0$ is the scaling parameter that weighs the utility of $S_t^1 + S_t^2$ against K_t. As in the single liquidity pool case, we assume that $\gamma_k = 1$.

The value function $Q\left(t,\, x_1,\, x_2,\, k,\, s_1,\, s_2\right)$ in equation (5.26) has the following HJB equation:

$$\begin{aligned} &Q_t + \frac{1}{2} \sum_{i=1}^{2} \left(\sigma_i^2 Q_{x_i x_i} + \hat{\sigma}_i^2 s_i^2 Q_{s_i s_i} - 2\kappa_i x_i Q_{x_i} \right) \\ &\quad + \sup_{\substack{\omega_1 \in [0, \bar{\omega}_1] \\ \omega_2 \in [0, \bar{\omega}_2]}} \left[\sum_{i=1}^{2} \left(a_i\left(x_t \right) s_i - \omega_i \right) Q_k + \sum_{i=1}^{2} \left(\omega_i - c_i \mathbf{1}_{\{\omega_i > 0\}} - \frac{\tilde{\sigma}_i^2}{8} s_i \right) Q_{s_i} \right] = 0, \end{aligned} \tag{5.28}$$

where the terminal condition is $Q\left(T,\, x_1,\, x_2,\, k,\, s_1,\, s_2\right) = \frac{1}{\gamma_k} k^{\gamma_k} + \frac{\eta}{\gamma_s} \left(s_1 + s_2 \right)^{\gamma_s}$. The Hamiltonian in (5.28) is linear with respect to ω_i for $i = 1, 2$, which leads to the optimal controls being of a bang-bang type

$$\omega_i = \begin{cases} \bar{w}_i, & \text{if } \left(\bar{\omega}_i - c_i \right) Q_{s_i} > \bar{\omega}_i Q_k, \\ 0, & \text{otherwise.} \end{cases}$$

For $\gamma_k = 1$, in equation (5.27) for the two liquidity pools case, the variable k can be factored out of the HJB solution by using the following ansatz:

$$Q\left(t,\, x_1,\, x_2,\, k,\, s_1,\, s_2\right) = k + V\left(t,\, x_1,\, x_2,\, s_1,\, s_2\right).$$

Let $\lambda_i \in (0,\, 1)$ for $i = 1, 2$ be transaction cost parameters such that $c_i = \lambda_i \bar{\omega}_i$. The SDEs for the staked amount S_t^i defined in the stochastic system (5.25) become $dS_t^i = \left(\left(1-\lambda_i\right) w_t^i - \frac{\tilde{\sigma}_i^2}{8} S_t^i\right) dt + \hat{\sigma}_i S_t^i d\hat{B}_t^i$ with $w_t^i \in \{0,\, \bar{w}_i\}$. Then equation (5.28) can be written succinctly as

$$\begin{aligned} V_t + \frac{1}{2}\sum_{i=1}^{2}\left(\sigma_i^2 V_{x_i x_i} + \hat{\sigma}_i^2 s_i^2 V_{s_i s_i} - 2\kappa_i x_i V_{x_i}\right) \\ + \sup_{\substack{\omega_1 \in \{0,\, \bar{\omega}_1\} \\ \omega_2 \in \{0,\, \bar{\omega}_2\}}} \left[\sum_{i=1}^{2}\left(a_i\left(x_t\right) s_i - \omega_i\right) + \sum_{i=1}^{2}\left(\left(1-\lambda_i\right)\omega_i - \frac{\tilde{\sigma}_i^2}{8} s_i\right) V_{s_i}\right] = 0, \end{aligned} \tag{5.29}$$

where the terminal condition is $V\left(T,\, x_1,\, x_2,\, s_1,\, s_2\right) = \frac{\eta}{\gamma_s}\left(s_1 + s_2\right)^{\gamma_s}$, and with the optimal bang-bang control is expressed as

$$\omega_i = \begin{cases} \bar{\omega}_i, & \text{if } \left(1-\lambda_i\right)\bar{\omega}_i V_{s_i} > \bar{\omega}_i, \\ 0, & \text{otherwise.} \end{cases} \tag{5.30}$$

6 Numerical Experiments

Similar to the bang-bang control regions shown in Davis and Norman (1990), the optimal controls derived in Section 5 separate the state-space into regions of *recompounding*, *no recompounding*, and *withdrawing*. HJB equations (5.23) and (5.29) are solved numerically with finite-difference methods (see Appendix D) and from the solution, we obtain a bang-bang policy function that we plot in 2-d as an illustration. These numerical solutions use model parameters estimated from the C̄urve and Ūniswap data on APRs (recall Section 4). The examples in this section will demonstrate the optimal yield farming policy for different DEXs, along with sensitivity analyses to show how the optimal controls change in different market conditions.

6.1 Curve – Single Liquidity Pool with Stochastic APR

We assume the pool's APR is of the form $a\left(X_t\right) := \beta e^{(X_t)}$, where X_t is the OU process defined in equation (5.17). We can observe that $\ln a(X_t) = \ln\beta + X_t$ is also an OU process but with long-term mean $\ln\beta$. We estimate parameters β, κ, and σ from C̄urve Pool time-series data, as presented in Figure 4.3. Our method for estimating the parameters of the OU process is described in Appendix E. From the Curve APR data, we obtain a volatility σ of 1.541, a speed of mean-reversion κ of 1.580, and a β estimate of 0.022. However, to expand the range of the APR, we amplify the values of $a\left(x\right)$ by increasing the β to 0.6. We estimate the volatility parameters $\tilde{\sigma}^2$ given in equation (5.6) and $\hat{\sigma}^2$ given in equation (5.18) from the variance of returns of the respective token time-series. We obtain $\tilde{\sigma} = 0.069$ and $\hat{\sigma} = 0.070$ as an annualized estimate for the six pools on Curve (as the average across all token pairs across all six pools).

To solve the HJB equation (5.23) using the finite-difference scheme (D.10) in Appendix D, we set

$t_{\min} = 0$, $t_{\max} = 3$, $x_{\min} = -2\sigma/\sqrt{2\kappa}$, $x_{\max} = 2\sigma/\sqrt{2\kappa}$, $s_{\min} = 3000$, $s_{\max} = 9000$, $N = t_{\max} \times 365 \times 10$, $I = 100$ and $J = 300$. We also set the parameter values for the utility function (5.21) to $\gamma_k = 1$, $\gamma_s = 0.8$, and $\eta = 4.95$; the values for the upper bounds of the control variables to $\bar{\omega} = 300$, $\bar{\nu} = 300$; and the value for the transaction cost coefficients such that $c = \lambda_\omega \bar{\omega} = \lambda_\nu \bar{\nu} = 10$. However, the finite-difference scheme requires us to implement a numerical boundary condition (see Appendix D), which distorts the solution near to the boundary, so we therefore discard the numerical solution for s between 8000 and 9000; in other words we solve numerically for s up to 9000, but when displaying, we crop the plots at $s = 8000$. Figure 6.1 displays the solution $V(t, x, s)$ for different values of t. As we can see from the plots, the solution is smooth, stable, and concave in s for any time $t \in [0, T]$.

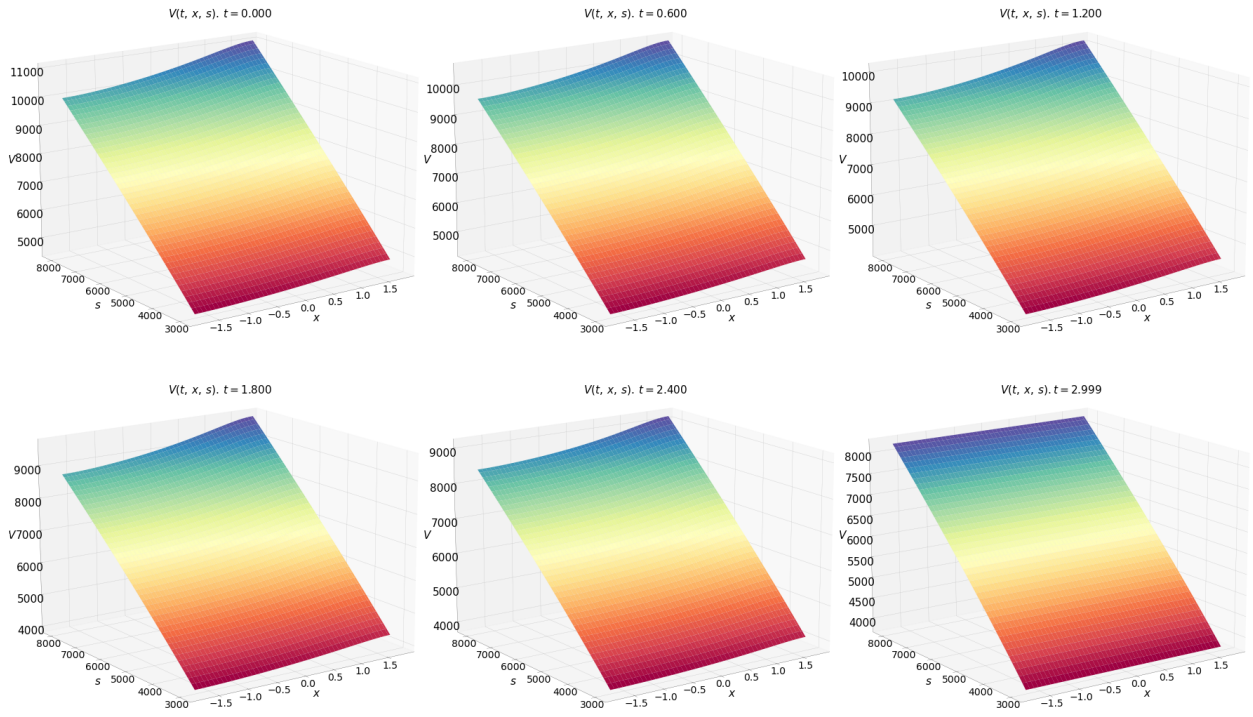

Figure 6.1: The figure shows the value function $V(t, x, s)$ of HJB equation (5.23) for different values t, with parameters estimated from Curve liquidity pool data. The risk tolerance coefficient is $\gamma = 0.8$, and the value for the transaction cost c is set to 10 USD.

Figure 6.2 displays the plot of ω and ν on the two dimensions S_t and $a(X_t)$, with the red, grey, and blue area representing the regions where the agent is recompounding ($\omega = \bar{\omega}$), not recompounding ($\omega = \nu = 0$), or withdrawing ($\nu = \bar{\nu}$). The plots are displayed ascending in time starting at $t = 0$ until close to the terminal time $T = 3$. In the figure, seen in the plot for $t = 0.000$, an agent with $S_0 \in [3000, 8000]$ will invest in the pool for any given APR. In addition, as time goes by, the red area becomes smaller, which means that for a given staked amount (or a given APR) the willingness to recompound decreases. Intuitively, the reason is that the transaction costs for recompounding outweighs the utility of future reward earned from increasing the staked amount. Near terminal time T, for any APR, recompounding will not be optimal. The behavior of ν is complementary: At the beginning ($t = 0$) the agent will not withdraw, but as time gets closer to T, the withdrawing area expands.

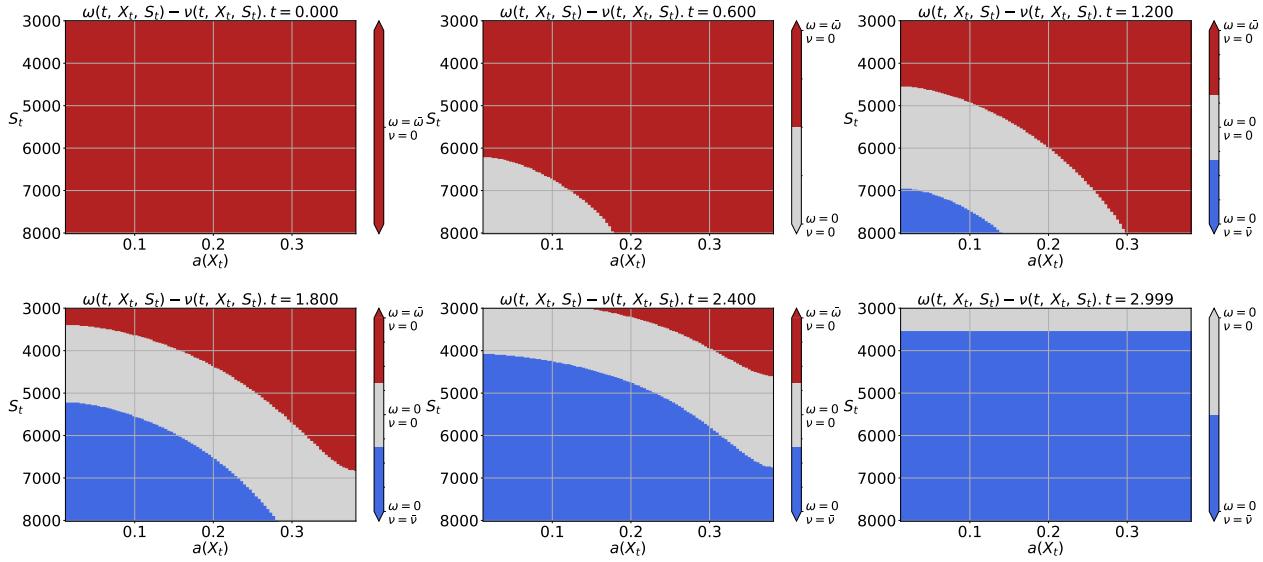

Figure 6.2: The figure shows the evolution of the optimal recompounding strategy $\omega(t, X_t, S_t) - \nu(t, X_t, S_t)$ at different time t for the C̄urve liquidity pool. The risk tolerance coefficient is $\gamma = 0.8$, and the value for the transaction cost is $c = 10$ USD. The horizontal axis is labeled by the values of APR for the C̄urve liquidity pool; in other words, $a(X_t) = \beta e^{X_t}$.

To gain a sense of how transaction costs affect yield farming investment process, we compute controls ω and ν for different levels of transaction costs. In Figure 6.3, the plot of ω and ν on the two dimensions S_t and $a(X_t)$, for one point in time $t = 1.5$, is displayed for ascending values of the transaction costs c, calculated as $c = \lambda\bar{\omega} = \lambda\bar{\nu}$, and hence c ranges from 1 USD to 25 USD. Please note that in the previous plots we set the transaction costs to 10 USD. In line with the intuition, higher transaction costs reduce future rewards from staking, thereby making recompounding less frequent. For example, in order to recompound when facing transaction costs of 25 USD, the agent requires a large compensation of APR greater or equal to 10%. In addition, as visible from the plots, the withdrawal area shrinks over time, and hence not only recompounding, but also withdrawing, becomes less likely with increasing transaction costs.

6.2 Uniswap – Single Liquidity Pool with Stochastic APR

We can repeat the analysis of Section 6.1 for the liquidity pool dynamics of Uniswap, specifically using the data from the UNI/WETH pool pair.[31] As previously, to solve the HJB equation (5.23) using the finite-difference scheme described in Appendix D, we discretize the computational domain, where we set $t_{\min} = 0$, $t_{\max} = 3$, $x_{\min} = -6\sigma/\sqrt{2\kappa}$, $x_{\max} = 2\sigma/\sqrt{2\kappa}$, $s_{\min} = 3000$, $s_{\max} = 9000$, $N = t_{\max} \times 365 \times 50$, $I = 80$, and $J = 125$. We then calibrate the model parameters following the procedure described in Appendix E. For the respective Uniswap APR dynamics, $a(X_t) = \beta e^{X_t}$, we obtain a $\beta = 0.15$, a volatility σ of 0.738, and a speed of mean-reversion κ of 1.26.[32] As in Section 6.1, the finite-difference scheme requires a numerical boundary condition, which distorts the solution, and therefore the displayed plots have been cropped. We assume a budget of $3000 - 8000$ USD, and $\bar{\omega} = \bar{\nu} = 300$ USD.As was done in Section 6.1, we set the parameter values for the utility function

[31] UNI represents the governance token from Uniswap, while WETH represents the wrapped form of Ethereum.

[32] We apply a moving average over the last 30 days to the average Uniswap APR time-series from 12-2021 to 12-2022 before estimating the coefficients of the OU process.

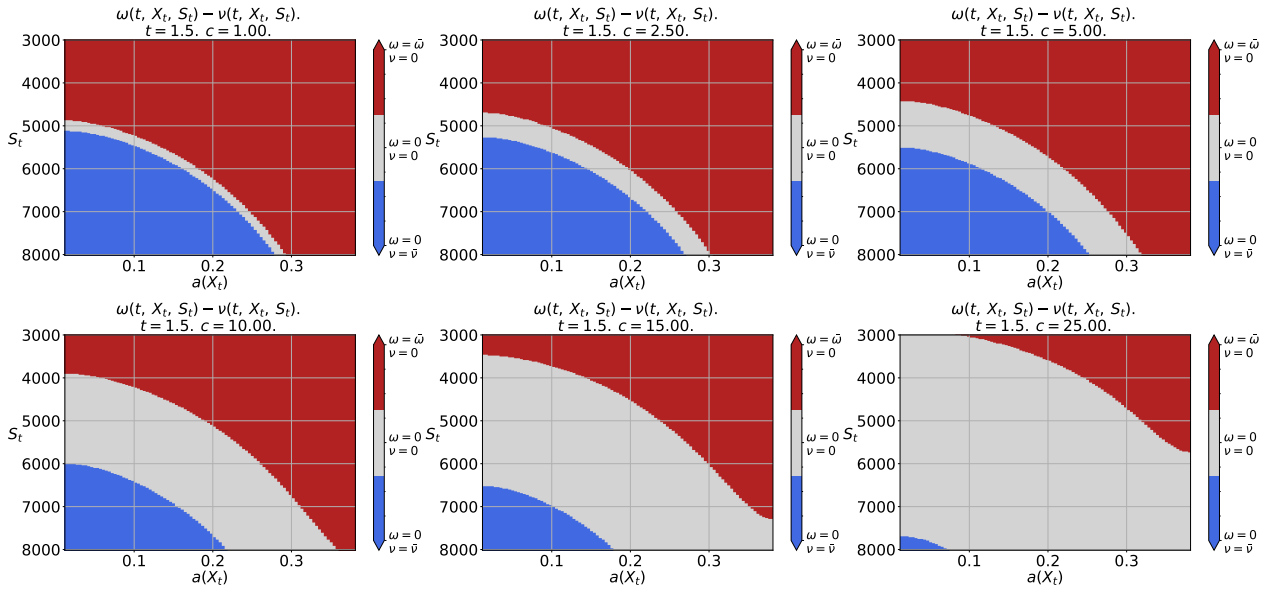

Figure 6.3: The figure shows the optimal recompounding strategy $\omega(t,\, X_t,\, S_t) - \nu(t,\, X_t,\, S_t)$ of the $\bar{\text{C}}$urve liquidity pool for different values of the transaction costs $c = \lambda_\omega \bar{\omega} = \lambda_\nu \bar{\nu}$: 1 USD, 2.5 USD, 5 USD, 10 USD, 15 USD, and 25 USD. The risk tolerance coefficient is $\gamma = 0.8$, and the time point is $t = 1.5$. The horizontal axis is labeled by the values of APR for the $\bar{\text{C}}$urve liquidity pool; in other words, $a(X_t) = \beta e^{X_t}$.

(5.21) to $\gamma_k = 1$, $\gamma_s = 0.8$, and $\eta = 4.95$, and the transaction cost coefficients (λ_ω and λ_ν) are set such that $c = \lambda_\omega \bar{\omega} = \lambda_\nu \bar{\nu}$ equals 10 USD. As done in Section 6.1, we estimate $\tilde{\sigma}^2$ and $\hat{\sigma}^2$ from the returns of the respective token dynamics, and we obtain $\tilde{\sigma} = 0.254$ and $\hat{\sigma} = 0.619$ (both annualized). In line with the intuition, the estimates on $\tilde{\sigma}$ and $\hat{\sigma}$ are many times larger compared to those of Curve.

Figure 6.4 displays the recompounding dynamics for a given APR over time. Overall, the results are qualitatively similar to the results of Section 6.1 for pools on Curve. As visible, for small t an agent will invest in the pool for any given APR (red area). As aforementioned, the red area shrinking over time means that the willingness to recompound decreases due to transaction costs outweighing future benefits of increased interest. In addition, over time the recompounding area shrinks, while the withdrawing area expands. Close to the terminal time, the agent performs either no action (grey area) or withdrawals (blue area) at any APR.

Even though the parameters for the APR dynamics $a(X)$ on Uniswap are comparable, the recompounding regions have noticeable differences from their counterparts for the stablecoin pools on Curve. The reason is the large estimates for the parameters related to the impermanent loss ($\hat{\sigma}^2$ and $\tilde{\sigma}^2$), which, not only leads to a smaller recompounding area (particularly for low APRs), but also we see that the agent will start withdrawing the staked amount earlier than they will in the Curve pools; see for example, $t = 0.6$ in Figure 6.2 and Figure 6.4.

Next, we investigate the dynamics of the optimal controls ω and ν for different levels of transaction costs. The plot of ω and ν on the two dimensions S_t and $a(X_t)$, for one point in time $t = 1.5$, is displayed for ascending values of the transaction cost coefficient $\lambda = \lambda_\omega = \lambda_\nu$. The values of transaction costs are calculated as $c = \lambda\bar{\omega} = \lambda\bar{\nu}$, and hence, c ranges from 1 USD to 25 USD. In the previous plots, we set the transaction costs to 10 USD. In line with intuition, higher transaction costs reduce utility of future rewards from staking, and, therefore, recompounding becomes less likely. Figure 6.5 shows how the willingness to recompound changes as transaction costs increase, using a point in time $t = 1.5$ and ascending values of c. As transaction costs c increase from 1 USD to 25

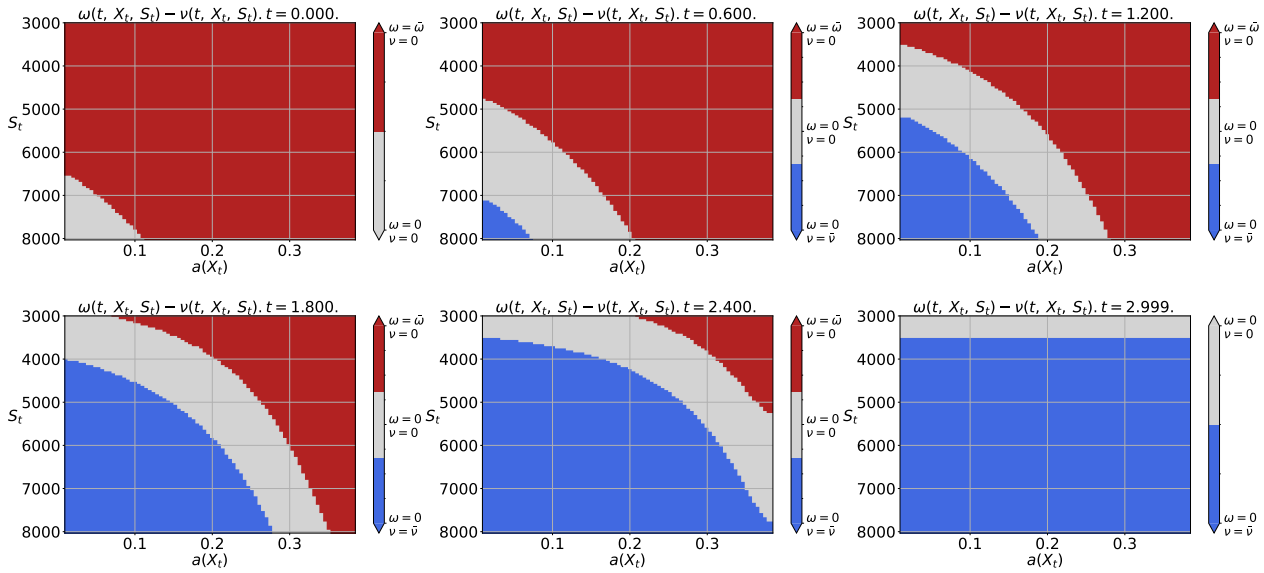

Figure 6.4: The figure shows the evolution of the optimal recompounding strategy $\omega(t, X_t, S_t) - \nu(t, X_t, S_t)$ at different time t for the UNI/WETH liquidity pool. The risk tolerance coefficient is $\gamma = 0.8$, and the value for the transaction costs c is 10 USD. The horizontal axis is labeled by the values of APR for UNI/WETH; in other words, $a(X_t) = \beta e^{X_t}$.

USD, recompounding becomes less likely. For example, when transaction costs are 25 USD, the agent would require an APR of at least 20% to recompound.[33]

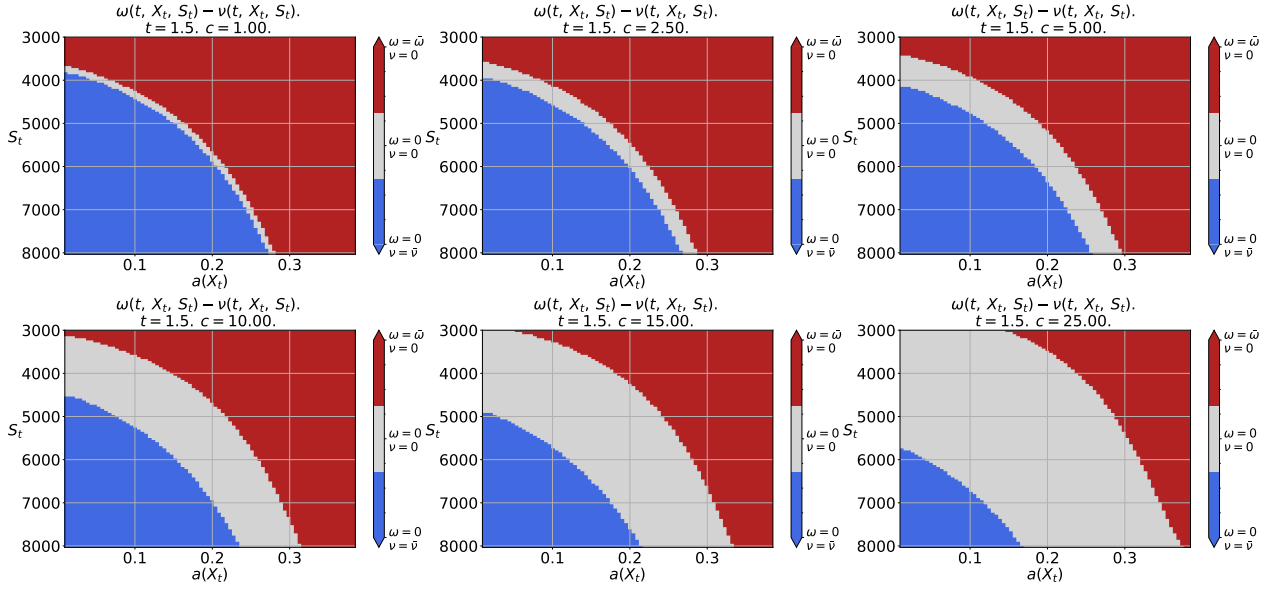

Figure 6.5: The figure shows the optimal recompounding strategy $\omega(t, X_t, S_t)$ of the Uniswap liquidity pool for different values of the transaction cost $c = \lambda\bar{\omega} = \lambda\bar{\nu}$: 1 USD, 2.5 USD, 5 USD, 10 USD, 15 USD, and 25 USD. The value of risk tolerance coefficient is $\gamma = 0.8$, and the time point, is $t = 1.5$. The horizontal axis is labeled by the values of APR for the respective pool; in other words $a(X_t) = \beta e^{X_t}$.

6.3 Uniswap – Multiple Liquidity Pools

The numerical analysis of Sections 6.1 and 6.2 is now applied to the two-pool model of Section 5.3.3. In this example we remain in the Uniswap universe, investigating the agent's behavior when investing in

[33]For earlier points in time ($t < 1.5$) the recompounding area is larger for any given transaction costs.

two pools. For this example we select UNI and Wrapped Ethereum (UNI/WETH) as the first pool and AAVE and Wrapped Ethereum (AAVE/WETH) as the second pool.[34] Empirically, the average APR for the first pool (second pool) is around 17% (12%), and the unconditional correlation between the pool APRs is around $\rho_x = 43\%$ over the sample period. For these two liquidity pools, we assume that the dynamics of APRs are $a_j\left(X_t^j\right) := \beta_j e^{X_t^j}$ for $j = 1, 2$, where X_t^j are the OU processes that are defined in equation (5.25). We can observe that stochastic processes $\ln a_j\left(X_t^j\right) = \ln \beta_j + X_t^j$ are also OU processes but with long-term mean $\ln \beta_j$. For parameter estimation, we follow Appendix E and obtain $\sigma_1 = 0.73$, $\sigma_2 = 1.09$, $\beta_1 = 0.15$, $\beta_2 = 0.11$, $\kappa_1 = 1.27$, and $\kappa_2 = 3.12$; therefore, we set $\kappa = \frac{1}{2}(\kappa_1 + \kappa_2)$. We estimate $\tilde{\sigma}^2$ and $\hat{\sigma}^2$ from the returns of the respective token dynamics (UNI, WETH, and AAVE), and we obtain $\tilde{\sigma_1} = 0.14$, $\tilde{\sigma_2} = 0.12$, $\hat{\sigma_1} = 0.29$, and $\hat{\sigma_2} = 0.33$ (all annualized).[35] For the utility function (5.27), we set the risk tolerance coefficients to $\gamma_1 = \gamma_2 = 0.5$ and the bound for the control to $\bar{\omega}_1 = \bar{\omega}_2 = 300$. Similar to the single liquidity pool case, we use the finite-difference scheme (D.3) to solve the HJB equation (5.29). Therefore, we discretize the computational domain, where we set $t_{\min} = 0$, $t_{\max} = 3$, $x^1_{\min} = -5\sigma_1/\sqrt{2\kappa}$, $x^1_{\max} = 2\sigma_1/\sqrt{2\kappa}$, $x^2_{\min} = -5\sigma_2/\sqrt{2\kappa}$, $x^2_{\max} = 2\sigma_2/\sqrt{2\kappa}$, $s^1_{\min} = 3000$, $s^1_{\max} = 9000$, $s^2_{\min} = 3000$, $s^2_{\max} = 9000$, $N = t_{\max} \times 365 \times 2$, $M = 20$, $H = 20$, $I = 40$, $J = 40$, and $\eta = 80.51$. As in Sections 6.1 and 6.2, the displayed plots have been cropped at $s = 8000$.

Figures 6.6 and 6.7 display the optimal controls w_1 and w_2 for each of the individual pools. The plots are in line with the single-pool case and show a large compounding region at the beginning, $t = 0$, and no-recompounding region close to terminal time $t = T$, with a shrinking area for the intermediate points in between.

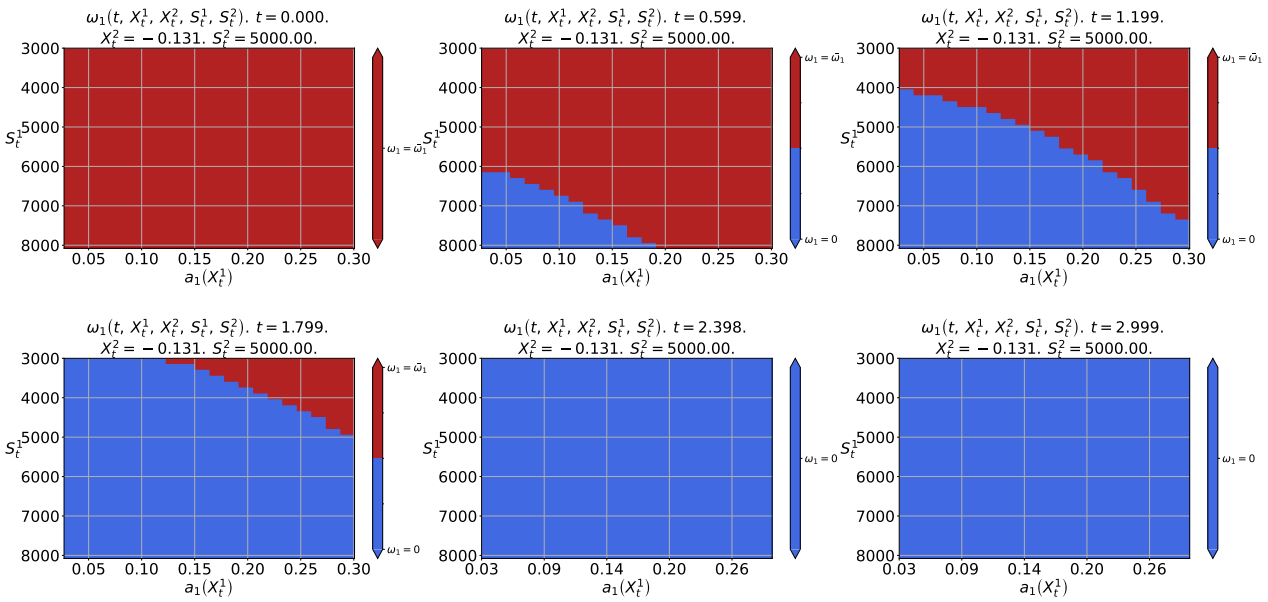

Figure 6.6: The figure shows the evolution of the optimal recompounding strategy $w_1\left(t, X_t, K_t, S_t^1, S_t^2\right)$ at different times t for the first liquidity pool AAVE/WETH. The transaction costs are set to $c = 10$ USD. The horizontal axis is labeled by the values of APR for the liquidity pool AAVE/WETH; in other words, $a_1\left(X_t\right) = \beta_1 e^{\sigma_1 X_t}$.

Figure 6.8 displays the projection of w_1 and w_2 on $\{0, \bar{\omega}_1\} \times \{0, \bar{\omega}_2\}$. In this plot, the dark red (dark blue) color represents the area where the agent would optimally recompound (not recompound)

[34] UNI denotes the Uniswap native token. AAVE denotes the Aave native token.

[35] As in the single liquidity pool case, we apply a moving average over the last 30 days to the time-series from 12-2021 to 12-2022 before estimating the coefficients of the exponential OU processes.

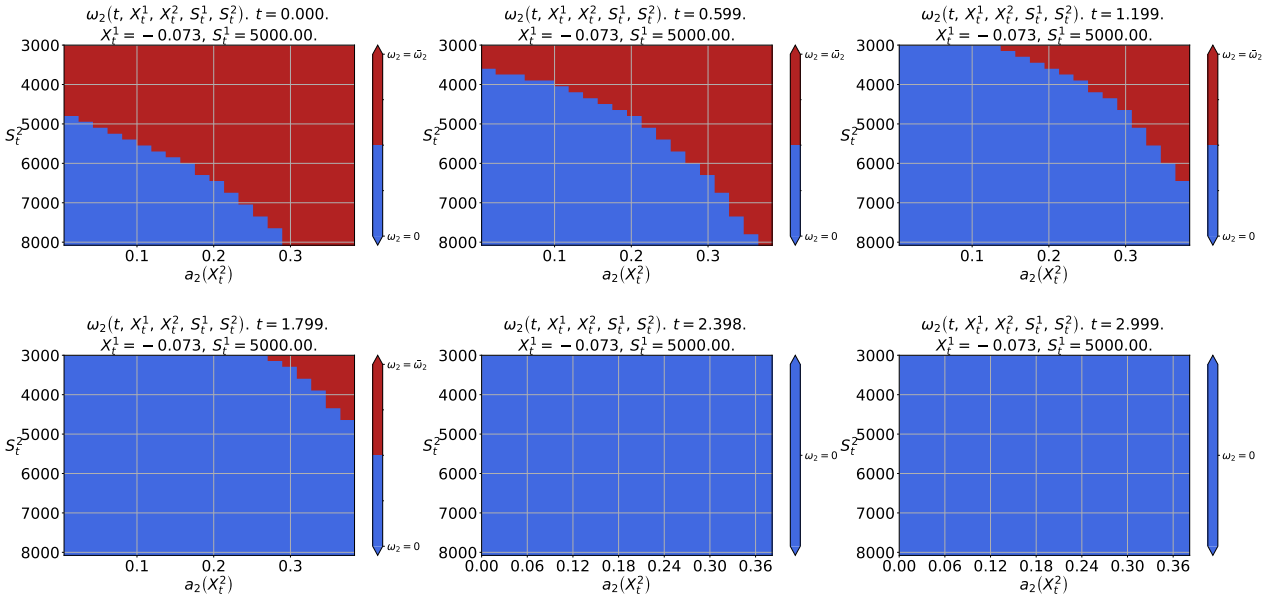

Figure 6.7: The figure shows the evolution of the optimal recompounding strategy $\omega_2\left(t,\, X_t,\, K_t,\, S_t^1,\, S_t^2\right)$ at different times t for the second liquidity pool UNI/WETH. The transaction costs are set to $c = 10$ USD. The horizontal axis is labeled by the values of APR for the liquidity pool UNI/WETH; in other words, $a_2\left(X_t\right) = \beta_2 e^{\sigma_2 X_t}$.

in both pools simultaneously. The light red region represents the area where the agent will recompound only the first pool. As visible from the plots at $t = 0$, the agent will recompound almost always. In contrast, close to the terminal time $t = T$, the costs of recompounding exceed the future benefits (dark blue area, no recompounding). Particularly interesting are the dynamics observed for the intermediate points in time, where the area is clearly divided into the three previously discussed areas. As can be seen in the plots, the area where recompounding is optimal in both pools is simultaneously shrinking. In addition, the individual recompounding area for the second pool relocates from a concentration over the entire corner ($t = 1.199$) to a rectangular concentration toward lower values of the respective range of S_t ($t = 1.199$ and $t = 1.799$).

To investigate the optimal recompounding behavior with respect to changing market conditions (APRs), we conduct a comparative statics analysis. First, we fix the APR of the first liquidity pool, UNI/WETH and assign increasing APRs for the second liquidity pool, AAVE/WETH. In addition, we evaluate the results at a fixed point in time $t = 1.5$. The plots are depicted in Figure 6.9 and display dynamics of the two controls w_1 and w_2. Notice first that the light red recompounding region for w_1 is decreasing in the realization of APR_2. Second, the dark red recompounding region for $\omega_1 = \bar{\omega}_1$ and $\omega_2 = \bar{\omega}_2$ is increasing in APR_2, which means that the agent not only recompounds more often within pool 1, but also makes use of the additional resources from pool 1 to more frequently recompound within pool 2. As visible from the last plot, due to the high APR_2, the agent also begins to recompound more frequently and exclusively in the second pool.[36] Second, we investigate the optimal recompounding behavior when APRs are behaving as if they would be negatively correlated. We plot the optimal controls in Figure 6.10 where APR_1 (APR_2) increase (decrease). As visible, for high (low) values of APR_2 (APR_1) the agent recompounds mostly in both pools or in the second pool (light blue). Not surprising, for increasing values of APR_1 the light red (light blue) area becomes larger (smaller).

[36] Due to symmetry, the results for an increasing APR_1 and a fix APR_2 are qualitatively similar, and they are available upon request.

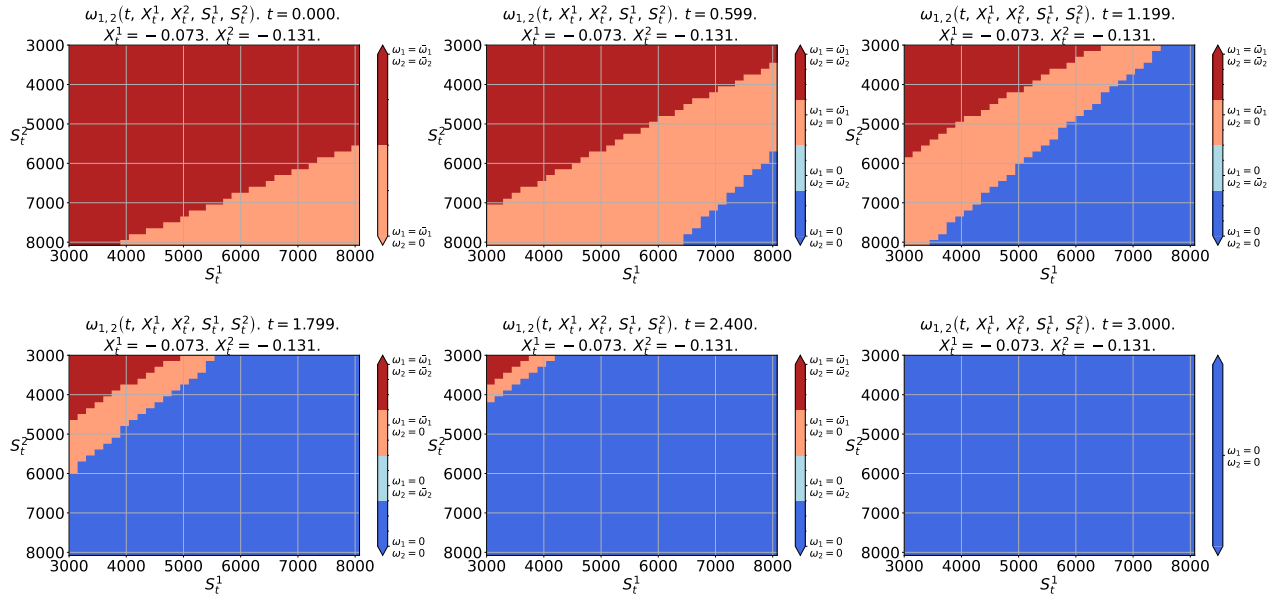

Figure 6.8: The figure shows the evolution of the optimal recompounding strategies $\omega_{1,2}\left(t,\ X_t,\ K_t,\ S_t^1,\ S_t^2\right) \in \{0, \bar{\omega}_1\} \times \{0, \bar{\omega}_2\}$ at different times t for both of the liquidity pools AAVE/WETH and UNI/WETH. The transaction costs are set to $c = 10$ USD. The functions of APRs for the pools are $a_i\left(X_t\right) = \beta_i e^{\sigma_i X_t}$, for $i = 1,\ 2$.

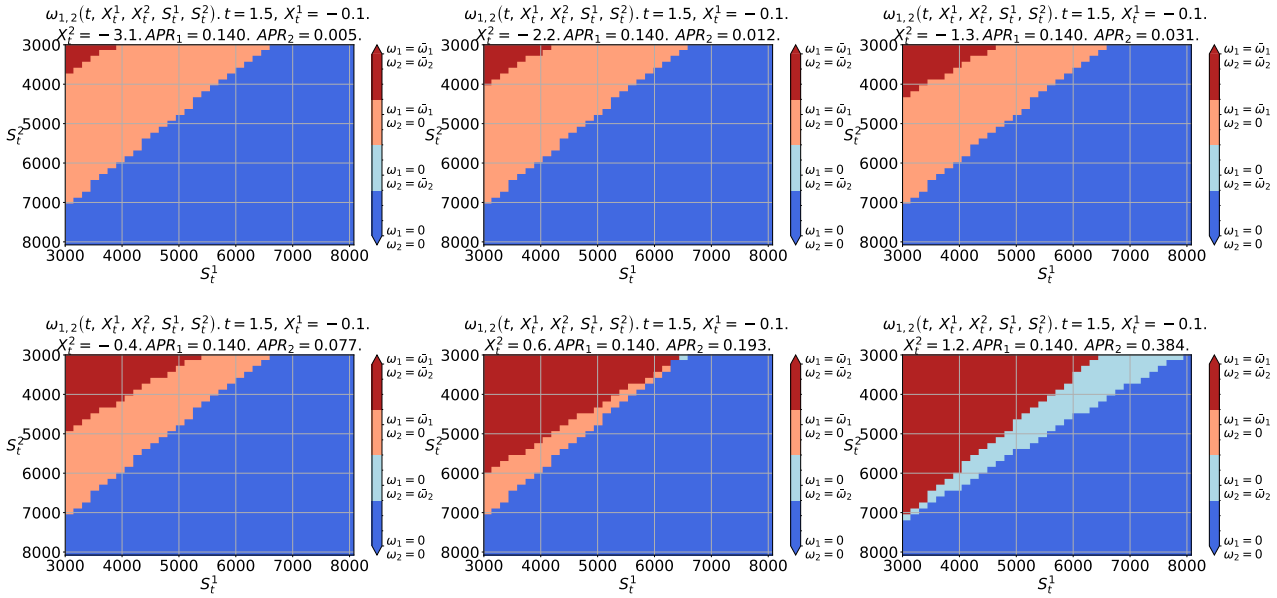

Figure 6.9: The figure shows the optimal recompounding strategies $\omega_{1,2}\left(t,\ X_t,\ K_t,\ S_t^1,\ S_t^2\right) \in \{0, \bar{\omega}_1\} \times \{0, \bar{\omega}_2\}$ for both of the liquidity pools UNI/WETH and AAVE/WETH. The APR_2 for the second liquidity pool AAVE/WETH is varying, and the $\text{APR}_1 = 0.14$ for the first liquidity pool UNI/WETH is fixed. The transaction costs are set to $c = 10$ USD. The functions of APRs for both of the liquidity pools are $a_i\left(X_t\right) = \beta_i e^{\sigma_i X_t}$, for $i = 1,\ 2$.

Next, we investigate the optimal recompounding solutions with respect to changing transaction costs. Figure 6.11 shows the optimal controls at $t = 1.5$ for varying transaction costs in both pools. As seen from the plots, when transaction costs increase, the individual (light red) and joint recompounding region (dark red) becomes smaller.

Throughout this section, we applied our proposed mathematical framework that resamples the yield farming investment process for different cases, such as for stablecoins on Curve and Ethereum tokens on Uniswap for single and multiple pools. Depending on the situation, we, therefore, calibrated

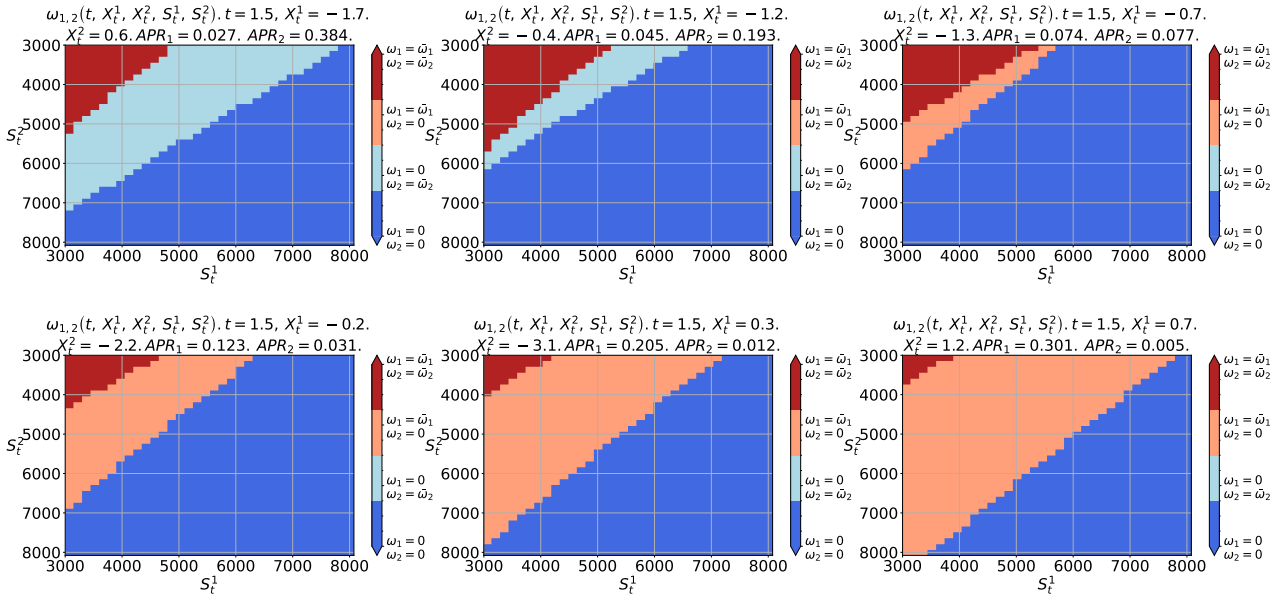

Figure 6.10: The figure shows the optimal recompounding strategies $\omega_{1,2}\left(t,\ X_t,\ K_t,\ S_t^1,\ S_t^2\right) \in \{0,\ \bar{\omega}_1\} \times \{0,\ \bar{\omega}_2\}$ for both of the liquidity pools UNI/WETH and AAVE/WETH. The APR_1 (APR_2) for the first (second) liquidity pool is increasing (decreasing). The transaction costs are set to $c = 10$ USD. The functions of APRs for both of the liquidity pools are $a_i\left(X_t\right) = \beta_i e^{\sigma_i X_t}$, for $i = 1,\ 2$.

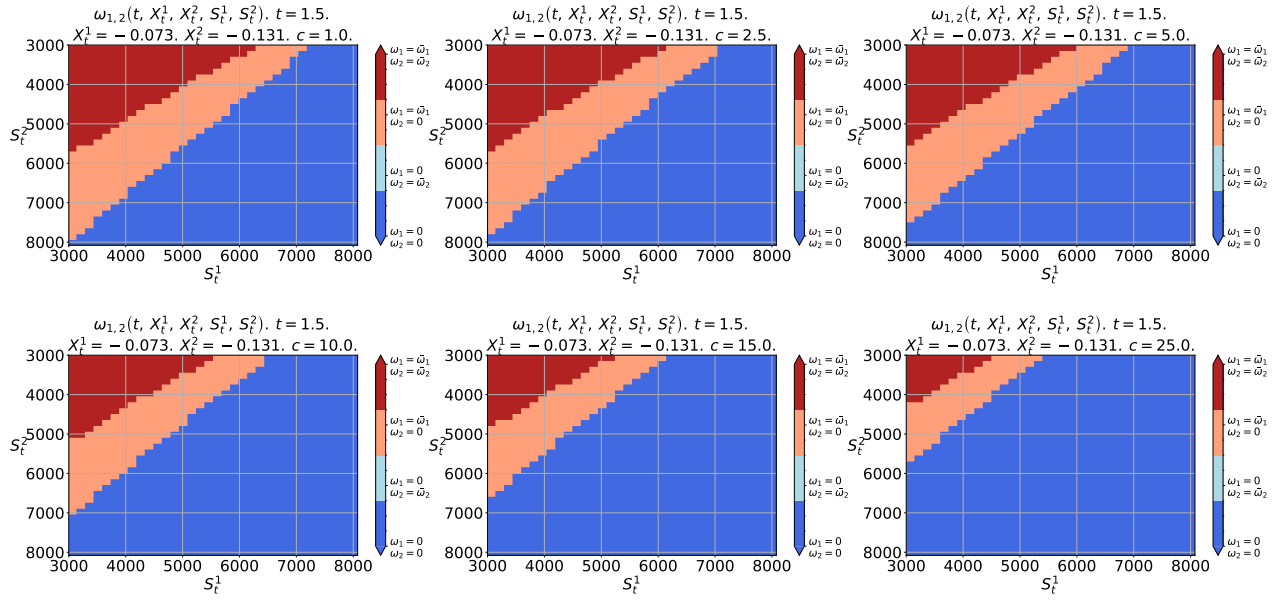

Figure 6.11: The figure shows the optimal recompounding strategies $\omega_{1,2}\left(t,\ X_t,\ K_t,\ S_t^1,\ S_t^2\right) \in \{0,\ \bar{\omega}_1\} \times \{0,\ \bar{\omega}_2\}$ for both of the liquidity pools AAVE/WETH and UNI/WETH at the time point $t = 1.5$. The values for the transaction costs $c =$ are set to 1 USD, 2.5 USD, 5 USD, 15 USD, and 25 USD. The functions of APRs for both of the liquidity pools are $a_i\left(X_t\right) = \beta_i e^{\sigma_i X_t}$, for $i = 1,\ 2$.

the stochastic APR process to best fit the respective DEX data. In addition, the analysis conducted reveals that the model delivers intuitively intriguing results and is able to quantitatively replicate the main economic trade-offs.

7 Robustness

To verify the robustness results of the analysis to various specifications, a series of tests are carried out and reported in the Appendix.

7.1 Curve – Single Liquidity Pool with Stochastic APR

We depict the behavior of the dynamics of the optimal control ω for different levels of the risk tolerance parameter in Figure F.1. In line with the intuition, a larger (lower) value for γ increases (decreases) the recompounding region, and in contrast, decreases (increases) the withdrawal area.

7.2 Uniswap – Single Liquidity Pool with Stochastic APR

In Figure F.2, we show how the optimal control ω changes with varying levels of the risk tolerance parameter. As expected, a larger value of γ expands the recompounding region, while a smaller value decreases it.

7.3 Uniswap – Two Liquidity Pools with Stochastic APR

In the Internet Appendix III, we present the derivation of the stochastic optimization framework for the two liquidity pools with a Cobb-Douglas utility. We then calibrate the model to the data and present the optimal compounding regions. The results are qualitatively similar to those in Section 6.3.

8 Conclusion

Yield farming allows individuals to make use of their token savings by providing liquidity and is, therefore, a means of earning interest on cryptocurrencies, similar to earning interest on a deposit in a savings account. Previously, only centralized exchanges and professional market makers were able to offer liquidity due to their access to capital and technical expertise. In this work, we explain and detail the outlined (economic) mechanisms and functioning of yield farming in all its facets.

In the empirical analysis, we quantify the transaction costs, risks, and returns of two major decentralized exchanges, Curve and Uniswap, on the Ethereum blockchain. On Curve, the average transaction cost is 25 USD, but there have been instances where it exceeded a few hundred USD. The average risk (impermanent loss) is negligible, but it can reach extreme lows. Rewards average at 3% but have the potential to exceed 20% on favorable days. In contrast, for the liquidity pools on Uniswap (for non-stablecoin pairs) the impermanent loss is considerable with extreme values reaching nearly –30%, and average rewards reduced to less than 10% when considering net APRs. Moreover, on non-favorable days the risk of liquidity provision for non-stablecoin pairs exceeds its returns. We also find that there are diversification benefits among the different liquidity pools on Uniswap. The empirical part of this study, therefore, offers a clearer understanding of yield farming and makes it easier for the broader audience to understand.

In addition, we present a mathematical framework that resembles the yield farming investment process, and hence, serves as a blueprint for agents to make optimal investment decisions. In our model, we derive an explicit expression for the impermanent loss as a function of the tokens' volatilities and their correlation. The impermanent loss can be seen as the major source of risk that the agent

faces. The model reveals that the agent's optimal recompounding (and withdrawal) policy follows a bang-bang type of control, which, in this context, means that the agent recompounds (withdrawals) the maximum or nothing, but never any intermediate amount. The flexibility of the framework is demonstrated through various applications: for the liquidity provision of stablecoins on Curve; for the liquidity provision on Uniswap, and for multiple pools. The sensitivity analyses provide insight into the agent's behavior in different market situations, specifically the influence of changing transaction costs and the risk-tolerance parameter of the agent's utility.

References

Franklin Allen and Douglas Gale. Financial intermediaries and markets. *Econometrica*, 72(4):1023–1061, 2004. doi: https://doi.org/10.1111/j.1468-0262.2004.00525.x.

Matteo Aquilina, Jon Frost, and Andreas Schrimpf. Decentralised finance (defi): a functional approach. *SSRN Manuscript*, 2023. URL `https://papers.ssrn.com/sol3/papers.cfm?abstract_id=4325095`.

Douglas Arner, Raphael Auer, and Jon Frost. Stablecoins: risks, potential and regulation. *Bank for International Settlements Working Papers No. 905*, 2020. URL `https://www.bis.org/publ/work905.htm`.

Patrick Augustin, Roy Chen-Zhang, and Donghwa Shin. Reaching for yield in decentralized financial markets. *SSRN Manuscript*, 2022. URL `https://papers.ssrn.com/sol3/papers.cfm?abstract_id=4063228`.

Andrea Barbon and Angelo Ranaldo. On the quality of cryptocurrency markets: Centralized versus decentralized exchanges. *arXiv Manuscript*, 2021. URL `https://arxiv.org/abs/2112.07386`.

Daniele Bianchi, Mykola Babiak, and Alexander Dickerson. Trading volume and liquidity provision in cryptocurrency markets. *Journal of Banking & Finance*, 142:106547, 2022. ISSN 0378-4266. doi: https://doi.org/10.1016/j.jbankfin.2022.106547.

Agostino Capponi and Ruizhe Jia. The adoption of blockchain-based decentralized exchanges. *SSRN Manuscript*, 2022. URL `https://papers.ssrn.com/sol3/papers.cfm?abstract_id=3805095`.

Alvaro Cartea, Faycal Drissi, and Marcello Monga. Decentralised finance and automated market making: Predictable loss and optimal liquidity provision. *SSRN Manuscript*, 2022. URL `https://papers.ssrn.com/sol3/papers.cfm?abstract_id=4273989`.

Lin William Cong, Ye Li, and Neng Wang. Tokenomics: Dynamic adoption and valuation. *Review of Financial Studies*, 34(3):1105–1155, 2021. ISSN 0893-9454. doi: https://doi.org/10.1093/rfs/hhaa089.

Lin William Cong, Zhiheng He, and Ke Tang. Staking, token pricing, and crypto carry. *SSRN Manuscript*, 2022. URL `https://papers.ssrn.com/sol3/papers.cfm?abstract_id=4059460`.

George M. Constantinides. Capital market equilibrium with transaction costs. *Journal of Political Economy*, 94(4):842–862, 1986. ISSN 0022-3808. doi: https://doi.org/10.1086/261410.

Simon Cousaert, Jiahua Xu, and Toshiko Matsui. Sok: Yield aggregators in defi. *arXiv Manuscript*, 2022. URL `https://arxiv.org/abs/2105.13891`.

M. H. A. Davis and A. R. Norman. Portfolio selection with transaction costs. *Mathematics of Operations Research*, 15(4):573–793, 1990. ISSN 0364-765X. doi: https://doi.org/10.1287/moor.15.4.676.

Anil Donmez and Alexander Karaivanov. Transaction fee economics in the ethereum blockchain. *Economic Inquiry*, 60(1):265–292, 2022. ISSN 0095-2583. doi: https://doi.org/10.1111/ecin.13025.

Darrell Duffie and Tong sheng Sun. Transactions costs and portfolio choice in a discrete-continuous-time setting. *Journal of Economic Dynamics and Control*, 14(1):35–51, 1990. ISSN 0165-1889. doi: https://doi.org/10.1016/0165-1889(90)90004-Z.

Darrell Duffie, Nicolae Gârleanu, and Lasse Heje Pedersen. Securities lending, shorting, and pricing. *Journal of Financial Economics*, 66(2):307–339, 2002. ISSN 0304-405X. doi: https://doi.org/10.1016/S0304-405X(02)00226-X.

Bernard Dumas and Elisa Luciano. An exact solution to a dynamic portfolio choice problem under transactions costs. *The Journal of Finance*, 46(2):577–595, 1991. ISSN 0022-1082. doi: https://doi.org/10.1111/j.1540-6261.1991.tb02675.x.

Gene D'Avolio. The market for borrowing stock. *Journal of Financial Economics*, 66(2):271–306, 2002. ISSN 0304-405X. doi: https://doi.org/10.1016/S0304-405X(02)00206-4.

Feyzullah Egriboyun and H. Mete Soner. Optimal investment strategies with a reallocation constraint. *Mathematical Methods of Operations Research*, 71(3):551–585, 2010. ISSN 1432-2994. doi: https://doi.org/10.1007/s00186-010-0306-5.

Nicolae Garleanu and Lasse Heje Pedersen. Dynamic trading with predictable returns and transaction costs. *The Journal of Finance*, 68(6):2309–2340, 2013. doi: https://doi.org/10.1111/jofi.12080.

Lawrence R. Glosten and Paul R. Milgrom. Bid, ask and transaction prices in a specialist market with heterogeneously informed traders. *Journal of Financial Economics*, 14(1):71–100, 1985. ISSN 0304-405X. doi: https://doi.org/10.1016/0304-405X(85)90044-3.

Itay Goldstein, Deeksha Gupta, and Ruslan Sverchkov. Utility tokens as a commitment to competition. *SSRN Manuscript*, 2022. URL `https://papers.ssrn.com/sol3/papers.cfm?abstract_id=3484627`.

Sanford J. Grossman and Merton H. Miller. Liquidity and market structure. *The Journal of Finance*, 43(3):617–633, 1988. doi: https://doi.org/10.1111/j.1540-6261.1988.tb04594.x.

Sebastian Gryglewicz, Simon Mayer, and Erwan Morellec. Optimal financing with tokens. *Journal of Financial Economics*, 142(3):1038–1067, 2021. ISSN 0304-405X. doi: https://doi.org/10.1016/j.jfineco.2021.05.004.

Campbell R. Harvey, Ashwin Ramachandran, and Joey Santoro. *DeFi and the Future of Finance*. John Wiley & Sons, Inc., Hoboken, New Jersey, U.S.A., 1 edition, 2021. ISBN 9781119836018.

Lioba Heimbach, Eric Schertenleib, and Roger Wattenhofer. Risks and returns of uniswap V3 liquidity providers. In *4th ACM Conference on Advances in Financial Technologies*, Cambridge, Massachusetts, U.S.A., 2022. URL https://arxiv.org/abs/2205.08904.

Jennifer Huang and Jiang Wang. Market liquidity, asset prices, and welfare. *Journal of Financial Economics*, 95(1):107–127, 2010. ISSN 0304-405X. doi: https://doi.org/10.1016/j.jfineco.2008.08.008.

Wolfgang Karl Härdle, Campbell R Harvey, and Raphael C G Reule. Understanding cryptocurrencies. *Journal of Financial Econometrics*, 18(2):181–208, 02 2020. ISSN 1479-8409. doi: https://doi.org/10.1093/jjfinec/nbz033.

Yuri Kabanov and Mher Safarian. *Markets with Transaction Costs Mathematical Theory*. Springer Finance. Springer Science+Business Media, Berlin, Deutschland, 1 edition, 2010. ISBN 9783540681205. doi: https://doi.org/10.1007/978-3-540-68121-2.

Bhaskar Krishnamachari, Qi Feng, and Eugenio Grippo. Dynamic curves for decentralized autonomous cryptocurrency exchanges. *arXiv Manuscript*, 2021. URL https://arxiv.org/abs/2101.02778.

Harold J. Kushner and Paul Dupuis. *Numerical Methods for Stochastic Control Problems in Continuous Time*. Stochastic Modelling and Applied Probability. Springer Science+Business Media, New York, U.S.A., 2 edition, 2001. ISBN 9780387951393. doi: https://doi.org/10.1007/978-1-4613-0007-6.

Yury A. Kutoyants. *Statistical Inference for Ergodic Diffusion Processes*. Springer Series in Statistics. Springer Science+Business Media, London, U.K., 1 edition, 2004. ISBN 9781852337599. doi: https://doi.org/10.1007/978-1-4471-3866-2.

Alfred Lehar and Christine Parlour. Decentralized exchanges. *SSRN Manuscript*, 2022. URL https://papers.ssrn.com/sol3/papers.cfm?abstract_id=3905316.

Jiageng Liu, Igor Makarov, and Antoinette Schoar. Anatomy of a run: The terra luna crash. *SSRN Manuscript*, 2023. URL https://papers.ssrn.com/sol3/papers.cfm?abstract_id=4416677.

Dennis E. Logue. Market-making and the assessment of market efficiency. *The Journal of Finance*, 30 (1):115–123, 1975. doi: https://doi.org/10.1111/j.1540-6261.1975.tb03163.x.

Michael J.P. Magill and George M. Constantinides. Portfolio selection with transactions costs. *Journal of Economic Theory*, 13(2):245–263, 1976. ISSN 0022-0531. doi: https://doi.org/10.1016/0022-0531(76)90018-1.

Igor Makarov and Antoinette Schoar. Cryptocurrencies and decentralized finance. *SSRN Manuscript*, 2022. URL https://papers.ssrn.com/sol3/papers.cfm?abstract_id=4104550.

Daren Matsuoka, Eddy Lazzarin, Chris Dixon, and Robert Hackett. Introducing the 2022 state of crypto report. Technical report, a16z crypto, 2022. URL https://a16zcrypto.com/state-of-crypto-report-a16z-2022/.

Jason Milionis, Ciamac C. Moallemi, Tim Roughgarden, and Anthony Lee Zhang. Automated market making and loss-versus-rebalancing. *arXiv Manuscript*, 2022. URL https://arxiv.org/abs/2208.06046.

Vijay Mohan. Automated market makers and decentralized exchanges: a defi primer. *Financial Innovation*, 8, 2022. ISSN 2199-4730. doi: https://doi.org/10.1186/s40854-021-00314-5.

Maureen O'Hara and George S. Oldfield. The microeconomics of market making. *Journal of Financial and Quantitative Analysis*, 21(4):361–376, 1986. doi: https://doi.org/10.2307/2330686.

Julien Prat, Vincent Danos, and Stefania Marcassa. Fundamental pricing of utility tokens. *THEMA Working Paper n2019-11. Université de Cergy-Pontoise*, 2019. URL `https://hal.science/hal-03096267`.

Michael Sockin and Wei Xiong. A model of cryptocurrencies. *Management Science, Forthcoming*, 2023. ISSN 0025-1909. doi: https://doi.org/10.1287/mnsc.2023.4756.

Jiahua Xu, Krzysztof Paruch, Simon Cousaert, and Yebo Feng. Sok: Decentralized exchanges (DEX) with automated market maker (AMM) protocols. *ACM Computing Surveys*, 55(11):1–50, 2023. ISSN 0360-0300. doi: https://dl.acm.org/doi/10.1145/3570639.

Appendix A Yield Farming – Details and Nuances

A.1 Different Forms of Yield Farming

Yield farming is an income opportunity in which one lends tokens through a decentralized application. There are different ways of yield farming (other than by liquidity provision, as discussed in the main text of the article), and in the following, we provide a short overview.

i) Yield farmers support network operations, for example; validating transactions: They allocate their tokens toward high-quality validators, allowing the network to run more efficiently and securely. Rewards are paid by network participants, who pay fees to validators in exchange for using the network. Validators then remit a portion of those fees back to yield farmers.

ii) Yield farmers provide lending for traders, and hence, allocate holdings toward capital-constrained traders. Farmers are compensated by borrowers, who pay continuous interest back to farmers (where the protocol takes a percentage). Although some protocols temporarily guarantee fixed interest rates, most use floating rates that allocate supply and demand.

iii) Yield farmers provide management and governance to protocols, where they create surplus through more efficient management through powering pooled systems and, thus, manage tokens in passive and delegated ways. For example, some protocols can reallocate liquidity toward particular markets more quickly and cheaper than a group of disaggregated traders.

iv) Another value provided by yield farmers is enhanced visibility and trust through asset allocation. Farmers enhance protocols' marketing efforts by leveraging user adaption. Specifically, protocols ask farmers to purchase and lock tokens in exchange for token distributions — with larger rewards for longer lockups. Of course, part of this exchange is that locked holders, who cannot respond to market conditions, bear substantial macroeconomic price risk relative to liquid ones.

A.2 LP Tokens, Reward Tokens, Boosting, and Concentrated Liquidity

Next, we describe the secondary and more specific aspects of yield farming that vary from platform to platform.

In return for depositing tokens into a liquidity pool, the liquidity providers obtains *LP tokens*, which serve as the mathematical proof of assets provided and, therefore, hold the claim to getting those assets back. *LP token staking* is an additional source of income.[37] Hence, *staking* might be the second step involved in yield farming; it is equivalent to locking them away for a period of time.[38] LPs are paid for this action because it renounces their ability to return the LP tokens, which means exiting the investment. Removing part of the token supply from circulation effectively impacts the market price, while growing the liquidity pool improves trade execution, and signals that the community is willing to commit to supporting the token for an extended period of time.

In addition to LP tokens, the LPs also obtain so-called *reward tokens* (these can be wrapped tokens, governance tokens, or secondary tokens), which further encourage token holding, while simultaneously facilitating token liquidity on exchanges and product usage. These tokens are, for example, issued in

[37] Although the terms *yield farming* and *staking* are sometimes used interchangeably, there are distinct ways in which they differ. Originally staking is one of multiple mechanisms involved in supporting a blockchain network and participating in transaction validation by committing tokens to a particular network. It is used by blockchain networks, which use the *proof-of-stake (PoS)* consensus mechanism.

[38] See for example, `https://www.convexfinance.com/stake` for how to stake various Curve LP tokens.

order to incentivize the liquidity provision of a token for which there is an increasing demand. There are even certain pools that have rewards in stablecoins. Platforms can engineer their token economics to both reward their innovation and foster a long-term sustainable protocol and community that continue to provide value. More details on the profitability of LP token staking on PancakeSwap can be found in Augustin et al. (2022).

In addition to generating additional yields by staking LP tokens and claiming reward tokens, some platforms allow for *boosting* yields generated in a liquidity pool once the user locks the governance token for a specific amount of time. For instance, a yield farmer can boost the rewards on some pools up to 2.5 times from the base amount by holding a substantial amount of the platforms' governance tokens relative to the liquidity they provide.

Uniswap V3 introduces the concept of *concentrated liquidity (CL)*, allowing liquidity providers to aggregate their pool liquidity in a defined interval and earn fees when the spot price moves into their specified *active zone*. If the price of the tokens does not fall outside of the range, one can provide more effective liquidity (also called virtual liquidity). The LP can, therefore, increase the capital efficiency of the position by narrowing the active range with the disadvantage to experience a larger impermanent loss. A detailed discussion about concentrated liquidity and its effect on the impermanent loss is provided in Heimbach et al. (2022).

As can be inferred from the outlined description, the devil of yield farming is in its details. In the mathematical framework we provide, we abstract from the platform specific details (LP tokens, reward tokens, boosting, and concentrated liquidity) and focus on the general functioning of yield farming, which is depositing tokens into a liquidity pool, paying the transaction costs, and earning the respective trading fees.

A.3 Quantifying Transactions Costs

In the following we break down the calculation of the transaction action costs for the smart contract functionality as used on the Ethereum blockchain. A standard Ethereum transfer requires 21,000 units of gas, but more complicated transactions involving smart contracts (such as liquidity pools) require more computational work and, therefore, a higher amount of gas units than a simple payment. The transaction costs in USD for 21,000 units of gas are displayed in Figure A.1. Transaction costs (on the Ethereum blockchain) can be calculated as the product of *gas units GU* times the *gas price per unit GP* denoted GP_{USD} in USD.

$$TC_{USD} = GU \times GP_{USD}. \tag{A.1}$$

GP itself is a function of the base fees (in Gwei), which denote the minimum amount of gas required to include a transaction and are adjusted by the demand for transaction inclusion. To obtain GP_{USD}, one has to multiply the GP with the exchange rate Ether-USD (ETH-USD) for one unit of Gwei ($1e9$ Gwei equals one ETH). To incentivize the inclusion of the transaction in the block, the user can add a priority fee (as a "tip" for the validator). In addition, the user can set a "max fee" for the transaction, where the difference between the max fee and the fee actual paid is refunded when the transaction is executed. A more detailed discussion about transaction fees on the Ethereum blockchain can be found in Donmez and Karaivanov (2022).

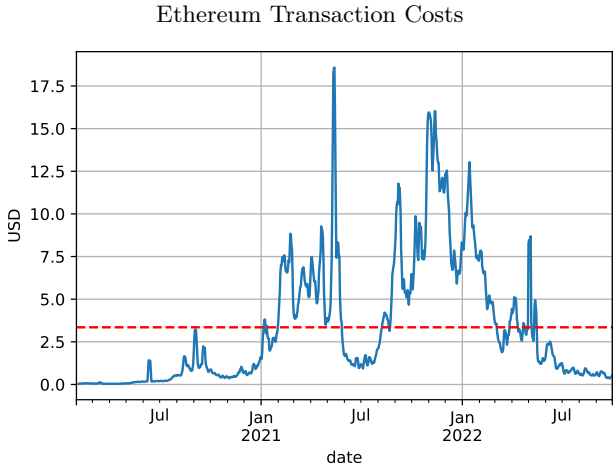

Figure A.1: Ethereum Transaction Costs – Standard Transaction. The figure shows the time-series plot of transaction costs for a standard transaction on the Ethereum network. Transaction costs are calculated applying equation (A.1), using 21.000 units of gas. The data is sampled daily, and the sample period is from 02-2021 to 10-2022. In the plots, the five-day moving average is depicted.

A.4 Impermanent Loss

It is relatively straightforward to derive a formula quantifying the impermanent loss as a function of the tokens' percentage change in value, which is illustrated in Figure A.2; see, for example, Xu et al. (2023).

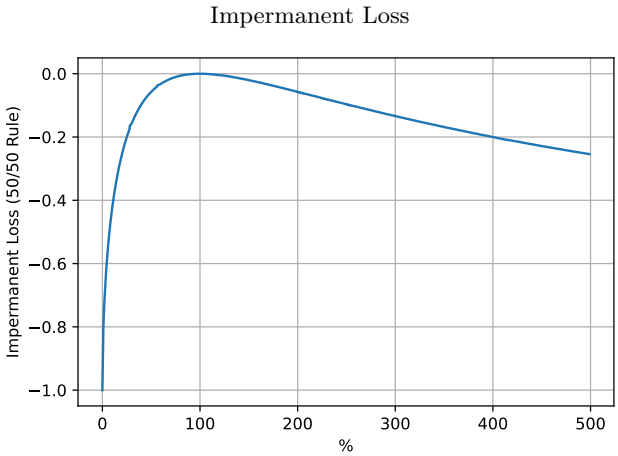

Figure A.2: Impermanent Loss. The figure displays the impermanent loss for the AMM constant product rule, which is calculated as $\frac{2\sqrt{k}}{1+k} - 1$.

Appendix B Empirical Analysis – Additional Figures

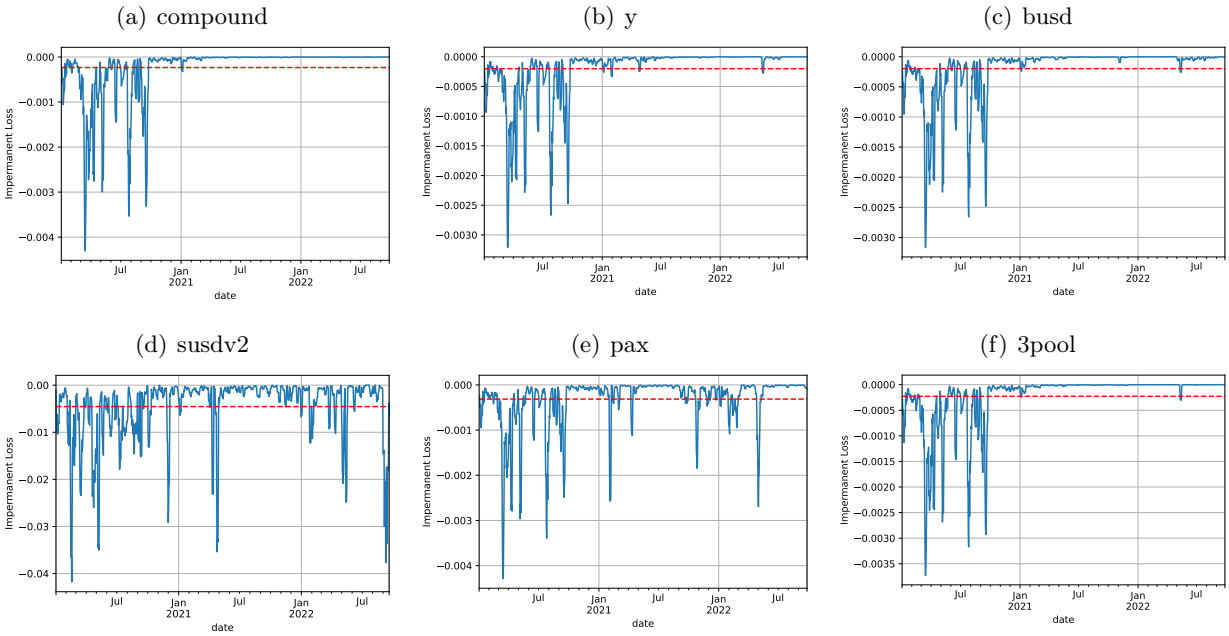

Figure B.1: Curve – Impermanent Loss. The figure shows the time-series plot of the impermanent loss (annualized) within each Curve pool. The dotted line represents the average impermanent loss. Details on the Curve pools and the token pairs are provided in Table 4.1 and Table I.1. The data is sampled daily, and the sample period is from 02-2020 to 10-2022. The data is winsorized at the 5% quantile. In the plots, the five-day moving average is depicted.

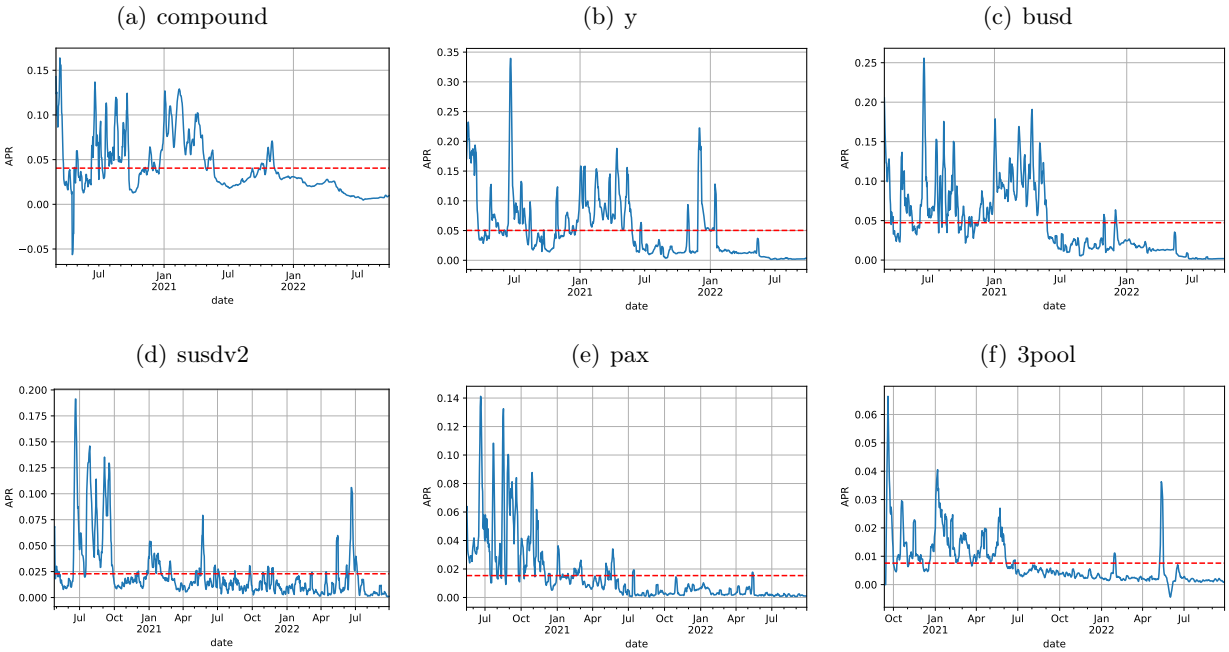

Figure B.2: Curve – APRs. The figure shows the time-series plot of APRs of the six liquidity pools from Curve as inferable from Table 4.1. The data is sampled daily, and the sample period is from 02-2020 to 10-2022. In the plots, the five-day moving average is depicted.

Appendix C Impermanent Loss in Continuous Time

If we divide equation (5.13) in both numerator and denominator by P_t^2, and then, using equation (5.4), equation (5.8), and equation (5.9), we have the following calculation,

$$\begin{aligned} dI_t &= \frac{\frac{1}{P_t^2}}{\frac{1}{P_t^2}} \frac{d\left(N_t^1 P_t^1 + N_t^2 P_t^2\right) - \left(N_t^1 dP_t^1 + N_t^2 dP_t^2\right)}{N_t^1 P_t^1 + N_t^2 P_t^2} \\ &= \frac{R_t dN_t^1 + dN_t^2 + \sigma_1 R_t dB_t^1 dN_t^1 + \sigma_2 dB_t^2 dN_t^2}{R_t N_t^1 + N_t^2} \\ &= \frac{L d\sqrt{R_t} + R_t d\left(\frac{L}{\sqrt{R_t}}\right) - \frac{\tilde{\sigma}^2}{2} L\sqrt{R_t} dt}{2L\sqrt{R_t}}. \end{aligned} \tag{C.1}$$

If we look at the term $R_t d\left(\frac{L}{\sqrt{R_t}}\right) - \frac{\tilde{\sigma}^2}{2} L\sqrt{R_t} dt$ in the numerator further, and, using equation (5.6), we then have

$$\begin{aligned} R_t d\left(\frac{k}{\sqrt{R_t}}\right) - \frac{\tilde{\sigma}^2}{2} k\sqrt{R_t} dt &= -\frac{\tilde{\mu}}{2} k\sqrt{R_t} dt - \frac{\tilde{\sigma}^2}{8} k\sqrt{R_t} dt - k\frac{\tilde{\sigma}}{2}\sqrt{R_t} d\tilde{B}_t \\ &= k d\left(\sqrt{R_t}\right) - k\frac{dR_t}{\sqrt{R_t}}. \end{aligned}$$

Hence, the impermanent loss (C.1) can be simplified as

$$dI_t = \frac{2L d\left(\sqrt{R_t}\right) - L\frac{dR_t}{\sqrt{R_t}}}{2L\sqrt{R_t}}, \tag{C.2}$$

which is the continuous-time equivalent of the impermanent loss formula in Heimbach et al. (2022). Using $\frac{N_t^2}{N_t^1} = R_t = \frac{P_t^1}{P_t^2}$, derived from the relative price equation (5.2) and Assumption 5.1, and beginning from the second line of equation (C.1), we can further show that equation (C.2) is equal to $-\frac{\tilde{\sigma}^2}{8} dt$ as well,

$$\begin{aligned} dI_t &= \frac{R_t dN_t^1 + dN_t^2 + \sigma_1 R_t dB_t^1 dN_t^1 + \sigma_2 dB_t^2 dN_t^2}{R_t N_t^1 + N_t^2} \\ &= \frac{-\tilde{\sigma}^2 R_t N_t^1}{4\left(R_t N_t^1 + N_t^2\right)} dt \\ &= -\frac{\tilde{\sigma}^2}{8} dt. \end{aligned}$$

Appendix D Finite-Difference Method

In this section, we describe the numerical approach of the finite-difference method that we use for solving the HJB equations associated with our yielding farming models. We demonstrate the procedure of deriving the numerical scheme for solving the HJB equation (5.29) for the model of two liquidity pools as an example. By following the same procedure, the HJB equation (5.23) for the model of single liquidity pool can be solved numerically as well.

In order to solve the HJB equation (5.29), we first need to convert the continuous computational domain into a bounded discretized domain $\mathcal{D} = \mathcal{T} \times \mathcal{X}_1 \times \mathcal{X}_2 \times \mathcal{S}_1 \times \mathcal{S}_2 \in \mathbb{R}^5$, where

$$
\begin{aligned}
\mathcal{T} &= [t_{\min},\, t_{\max}] = \{t_{\min} = t_0,\, t_1,\, \cdots,\, t_n,\, \cdots,\, t_N = t_{\max}\} \in \mathbb{R},\ \text{with } t_n = t_{\min} + n\Delta t, \\
\mathcal{X}_1 &= \left[x^1_{\min},\, x^1_{\max}\right] = \left\{x^1_{\min} = x^1_0,\, x^1_1,\, \cdots,\, x^1_m,\, \cdots,\, x^1_M = x^1_{\max}\right\} \in \mathbb{R},\ \text{with } x^1_m = x^1_{\min} + m\Delta x_1, \\
\mathcal{X}_2 &= \left[x^2_{\min},\, x^2_{\max}\right] = \left\{x^2_{\min} = x^2_0,\, x^2_1,\, \cdots,\, x^2_h,\, \cdots,\, x^2_H = x^2_{\max}\right\} \in \mathbb{R},\ \text{with } x^2_h = x^2_{\min} + h\Delta x_2, \\
\mathcal{S}_1 &= \left[s^1_{\min},\, s^1_{\max}\right] = \left\{s^1_{\min} = s^1_0,\, s^1_1,\, \cdots,\, s^1_i,\, \cdots,\, s^1_I = s^1_{\max}\right\} \in \mathbb{R},\ \text{with } s^1_i = s^1_{\min} + i\Delta s_1, \\
\mathcal{S}_2 &= \left[s^2_{\min},\, s^2_{\max}\right] = \left\{s^2_{\min} = s^2_0,\, s^2_1,\, \cdots,\, s^2_j,\, \cdots,\, s^2_J = s^2_{\max}\right\} \in \mathbb{R},\ \text{with } s^2_j = s^2_{\min} + j\Delta s_2;
\end{aligned}
\tag{D.1}
$$

also, $\Delta t = \frac{1}{N}\left(t_{\max} - t_{\min}\right) > 0$, $\Delta x_1 = \frac{1}{M}\left(x^1_{\max} - x^1_{\min}\right) > 0$, $\Delta x_2 = \frac{1}{H}\left(x^2_{\max} - x^2_{\min}\right) > 0$, $\Delta s_1 = \frac{1}{I}\left(s^1_{\max} - s^1_{\min}\right) > 0$, and $\Delta s_2 = \frac{1}{J}\left(s^2_{\max} - s^2_{\min}\right) > 0$ are the discretization steps.

We use a finite difference method to solve the HJB equation (5.29) on the discretized domain $\mathcal{D}$ defined by equation (D.1). More specifically, for equation (5.29) we apply central differences to approximate the first-and second-order derivatives with respect to variables x_1 and x_2, as well as for the second-order derivatives with respect to variables s_1 and s_2. We apply a first-order upwind scheme to approximate the first-order derivatives with respect to variables s_1 and s_2, and a first-order explicit scheme for the derivative with respect to the time variable t as follows:

$$
\begin{aligned}
V_t^{n,m,h,i,j} &\approx \frac{V^{n+1}_{m,h,i,j} - V^n_{m,h,i,j}}{\Delta t}, \\
V_{x_1}^{n,m,h,i,j} &\approx \frac{V^n_{m+1,h,i,j} - V^n_{m-1,h,i,j}}{2\Delta x_1}, \quad V_{x_1x_1}^{n,m,h,i,j} \approx \frac{V^n_{m-1,h,i,j} - 2V^n_{m,h,i,j} + V^n_{m+1,h,i,j}}{\Delta x_1^2}, \\
V_{x_2}^{n,m,h,i,j} &\approx \frac{V^n_{m,h+1,i,j} - V^n_{m,h-1,i,j}}{2\Delta x_2}, \quad V_{x_2x_2}^{n,m,h,i,j} \approx \frac{V^n_{m,h-1,i,j} - 2V^n_{m,h,i,j} + V^n_{m,h+1,i,j}}{\Delta x_2^2}, \\
V_{s_1}^{n,m,h,i,j} &\approx \frac{V^n_{m,h,i+1,j} - V^n_{m,h,i,j}}{\Delta s_1}, \quad V_{s_1s_1}^{n,m,h,i,j} \approx \frac{V^n_{m,h,i-1,j} - 2V^n_{m,h,i,j} + V^n_{m,h,i+1,j}}{\Delta s_1^2}, \\
V_{s_2}^{n,m,h,i,j} &\approx \frac{V^n_{m,h,i,j+1} - V^n_{m,h,i,j}}{\Delta s_2}, \quad V_{s_2s_2}^{n,m,h,i,j} \approx \frac{V^n_{m,h,i,j-1} - 2V^n_{m,h,i,j} + V^n_{m,h,i,j+1}}{\Delta s_2^2},
\end{aligned}
\tag{D.2}
$$

where $V^n_{m,h,i,j}$ and $V_*^{n,m,h,i,j}$ respectively represent the values of V and derivatives of V with respect to variable $*$ at the location $\left(t_n,\, x^1_m,\, x^2_h,\, s^1_i,\, s^2_j\right)$ in the discretized domain $\mathcal{D}$.

Replacing the derivatives in equation (5.29) with the approximations given by equation (D.2), we then have the following numerical scheme:

$$
V^{n-1}_{m,h,i,j} = \left(1 - \frac{\Delta t\sigma^2_{x_1}}{\Delta x_1^2} - \frac{\Delta t\sigma^2_{x_2}}{\Delta x_2^2} - \frac{\Delta t\hat{\sigma}^2_1\left(s^1_i\right)^2}{\Delta s_1^2} - \frac{\Delta t\hat{\sigma}^2_2\left(s^2_j\right)^2}{\Delta s_2^2}\right) V^n_{m,h,i,j}
\tag{D.3}
$$

$$
\begin{aligned}
&+\left(\frac{\Delta t\sigma_{x_1}^2}{2\Delta x_1^2}+\frac{\Delta t\kappa_1 x_m^1}{2\Delta x_1}\right)V_{m-1,h,i,j}^n+\left(\frac{\Delta t\sigma_{x_1}^2}{2\Delta x_1^2}-\frac{\Delta t\kappa_1 x_m^1}{2\Delta x_1}\right)V_{m+1,h,i,j}^n\\
&+\left(\frac{\Delta t\sigma_{x_2}^2}{2\Delta x_2^2}+\frac{\Delta t\kappa_2 x_h^2}{2\Delta x_2}\right)V_{m,h-1,i,j}^n+\left(\frac{\Delta t\sigma_{x_2}^2}{2\Delta x_2^2}-\frac{\Delta t\kappa_2 x_h^2}{2\Delta x_2}\right)V_{m,h+1,i,j}^n\\
&+\frac{\Delta t\hat{\sigma}_1^2\left(s_i^1\right)^2}{2\Delta s_1^2}V_{m,h,i-1,j}^n+\frac{\Delta t\hat{\sigma}_1^2\left(s_i^1\right)^2}{2\Delta s_1^2}V_{m,h,i+1,j}^n\\
&+\frac{\Delta t\hat{\sigma}_2^2\left(s_j^2\right)^2}{2\Delta s_2^2}V_{m,h,i,j-1}^n+\frac{\Delta t\hat{\sigma}_2^2\left(s_j^2\right)^2}{2\Delta s_2^2}V_{m,h,i,j+1}^n\\
&+\sup_{\substack{\omega_1\in\{0,\bar{\omega}_1\}\\ \omega_2\in\{0,\bar{\omega}_2\}}}\Bigg[\Delta t\sum_{q=1}^{2}\left(a_q\left(x_q\right)s_q-\omega_q\right)+\frac{\Delta t}{\Delta s_1}\left(\left(1-\lambda_1\right)\omega_1-\frac{\tilde{\sigma}_1^2}{8}s_i^1\right)V_{m,h,i+1,j}^n\\
&-\frac{\Delta t}{\Delta s_1}\left(\left(1-\lambda_1\right)\omega_1-\frac{\tilde{\sigma}_1^2}{8}s_i^1\right)V_{m,h,i,j}^n+\frac{\Delta t}{\Delta s_2}\left(\left(1-\lambda_2\right)\omega_2-\frac{\tilde{\sigma}_2^2}{8}s_j^2\right)V_{m,h,i,j+1}^n\\
&-\frac{\Delta t}{\Delta s_2}\left(\left(1-\lambda_2\right)\omega_2-\frac{\tilde{\sigma}_2^2}{8}s_j^2\right)V_{m,h,i,j}^n\Bigg]=0.
\end{aligned}
$$

where t is in backward; in other words, for t_n, we have $n=N,\ N-1,\ \cdots,\ 2,\ 1$, because we have a terminal condition $V\left(t_N,\ x_m^1,\ x_h^2,\ s_i^1,\ s_j^2\right)=\frac{\eta}{\gamma_s}\left(s_i^1+s_j^2\right)^{\gamma_s},\ \forall i,\ j$.

The numerical scheme that is given by equation (D.3) is stable if it satisfies a Courant-Fredrichs-Lewy (CFL) condition, which is checked probabilistically by using the Markov chain approximation method of Kushner and Dupuis (2001). The dynamic programming equation for the value function is

$$
V^{\Delta x_1}\left(t,\ x_1,\ x_2,\ s_1,\ s_2\right)=\sup_{\substack{\omega_1\in\{0,\bar{\omega}_1\}\\ \omega_2\in\{0,\bar{\omega}_2\}}}\left[\sum_{x'}p^{\Delta x_1}\left(x_1,\ x_1'\,|\,\omega_1,\ \omega_2\right)V^{\Delta x_1}\left(t,\ x_1',\ x_2\,s_1,\ s_2\right)\right],
$$

where $p^{\Delta x_1}\left(x_1,\ x_1'\,|\,\omega_1,\ \omega_2\right)$ is the transition probability from state x_1 to state x_1'. In order to maintain the stability of the numerical scheme (D.3), the following arguments:

$$
\begin{aligned}
p_m&=1-\frac{\Delta t\sigma_{x_1}^2}{\Delta x_1^2}-\frac{\Delta t\sigma_{x_2}^2}{\Delta x_2^2}-\frac{\Delta t\hat{\sigma}_1^2\left(s_i^1\right)^2}{\Delta s_1^2}-\frac{\Delta t\hat{\sigma}_2^2\left(s_j^2\right)^2}{\Delta s_2^2}+\frac{\Delta t\tilde{\sigma}_1^2 s_i^1}{8\Delta s_1}\\
&\quad+\frac{\Delta t\tilde{\sigma}_2^2 s_j^2}{8\Delta s_2}-\frac{\Delta t\left(1-\lambda_1\right)}{\Delta s_1}\omega_1-\frac{\Delta t\left(1-\lambda_2\right)}{\Delta s_2}\omega_2\\
p_m^-&=\frac{\Delta t\sigma_{x_1}^2}{2\Delta x_1^2}+\frac{\Delta t\kappa_1 x_m^1}{2\Delta x_1},\qquad p_m^+=\frac{\Delta t\sigma_{x_1}^2}{2\Delta x_1^2}-\frac{\Delta t\kappa_1 x_m^1}{2\Delta x_1},\\
p_h^-&=\frac{\Delta t\sigma_{x_2}^2}{2\Delta x_2^2}+\frac{\Delta t\kappa_2 x_h^2}{2\Delta x_2},\qquad p_h^+=\frac{\Delta t\sigma_{x_2}^2}{2\Delta x_2^2}-\frac{\Delta t\kappa_2 x_h^2}{2\Delta x_2},\\
p_i^-&=\frac{\Delta t\hat{\sigma}_1^2\left(s_i^1\right)^2}{2\Delta s_1^2},\qquad p_i^+=\frac{\Delta t\hat{\sigma}_1^2\left(s_i^1\right)^2}{2\Delta s_1^2}-\frac{\Delta t\tilde{\sigma}_1^2 s_i^1}{8\Delta s_1}+\frac{\Delta t\left(1-\lambda_1\right)}{\Delta s_1}\omega_1\\
p_j^-&=\frac{\Delta t\hat{\sigma}_2^2\left(s_j^2\right)^2}{2\Delta s_2^2},\qquad p_j^+=\frac{\Delta t\hat{\sigma}_2^2\left(s_j^2\right)^2}{2\Delta s_2^2}-\frac{\Delta t\tilde{\sigma}_2^2 s_j^2}{8\Delta s_2}+\frac{\Delta t\left(1-\lambda_2\right)}{\Delta s_2}\omega_2
\end{aligned}
\tag{D.4}
$$

are the transition probabilities between different states of x_i and s_j for $i,\ j=1,\ 2$; therefore, their values have to be in $(0,\ 1)$, and $p_i^-+p_i+p_i^++p_j^-+p_j^++p_h^++p_m^-+p_m^+=1$ for all cases $\omega_1=0$ or $\bar{\omega}_1$

and $\omega_2 = 0$ or $\bar{\omega}_2$.

In order to solve the two liquidity pools model numerically using the finite-difference scheme (D.3), we also need to set the boundary conditions. Because the OU processes are approximated by Markov chains, we can adjust the numerical scheme at the boundary points so that the Markov chain approximation of X_t^i reflects back to discretized domain $\mathcal{X}_i$. More specifically, at the ghost points x_{-1}^1 and x_{M+1}^1, $V_{-1,h,i,j}^n$ and $V_{M+1,h,i,j}^n$ are replaced by their adjacent points $V_{0,h,i,j}^n$ and $V_{M,h,i,j}^n$, and at the ghost points x_{-1}^2 and x_{L+1}^2, $V_{m,-1,i,j}^n$ and $V_{m,L+1,i,j}^n$ are replaced by their adjacent points $V_{m,0,i,j}^n$ and $V_{m,L,i,j}^n$

$$
\begin{aligned}
V_{x_1}^{n,0,h,i,j} &\approx \frac{V_{1,h,i,j}^n - V_{0,h,i,j}^n}{2\Delta x_1}, \qquad V_{x_1x_1}^{n,0,h,i,j} \approx \frac{V_{0,h,i,j}^n - 2V_{0,h,i,j}^n + V_{1,h,i,j}^n}{2\Delta x_1}, \\
V_{x_1}^{n,M,h,i,j} &\approx \frac{V_{M,h,i,j}^n - V_{M-1,h,i,j}^n}{2\Delta x_1}, \qquad V_{x_1x_1}^{n,M,h,i,j} \approx \frac{V_{M-1,h,i,j}^n - 2V_{M,h,i,j}^n + V_{M,h,i,j}^n}{\Delta x_1^2}, \\
V_{x_2}^{n,m,0,i,j} &\approx \frac{V_{m,1,i,j}^n - V_{0,h,i,j}^n}{2\Delta x_2}, \qquad V_{x_2x_2}^{n,m,0,i,j} \approx \frac{V_{m,0,i,j}^n - 2V_{m,0,i,j}^n + V_{m,0,i,j}^n}{2\Delta x_2}, \\
V_{x_2}^{n,m,L,i,j} &\approx \frac{V_{m,L,i,j}^n - V_{m,L-1,i,j}^n}{2\Delta x_2}, \qquad V_{x_2x_2}^{n,m,L,i,j} \approx \frac{V_{m,L-1,i,j}^n - 2V_{m,L,i,j}^n + V_{m,L,i,j}^n}{\Delta x_2^2}.
\end{aligned}
\tag{D.5}
$$

For the derivative with respect to s_1 and s_2, we can observe from equation (D.2) that, because the first-order derivatives with respect to these two variables are approximated by the upwind scheme, we therefore need only to set boundary condition at $i = I$ and $j = J$. For the second-order derivatives with respect to s_1 and s_2, because they are approximated by the central-difference scheme, we therefore need to set the boundary condition at $i = 0$, $i = I$, $j = 0$, and $j = J$, for which we choose the Neumann boundary conditions. From the utility function (5.27) for the two liquidity pools model, we have $V\left(t_n,\, x_m^1,\, x_h^2,\, s_i^1,\, s_j^2\right) = \frac{\eta}{\gamma_s}\left(s_i^1 + s_j^2\right)^{\gamma_s}$, therefore,

$$
\begin{aligned}
V_{s_1}^{n,m,h,I,j} &\approx \frac{\gamma_s}{s_I^1 + s_j^2} V\left(t_n,\, x_m^1,\, x_h^2,\, s_I^1,\, s_j^2\right), \qquad V_{s_2}^{n,m,h,i,J} \approx \frac{\gamma_s}{s_i^1 + s_J^2} V\left(t_n,\, x_m^1,\, x_h^2,\, s_i^1,\, s_J^2\right), \\
V_{s_1s_1}^{n,m,h,0,j} &\approx \frac{\gamma_s\left(\gamma_s - 1\right)}{\left(s_0^1 + s_j^2\right)^2} V\left(t_n,\, x_m^1,\, x_h^2,\, s_0^1,\, s_j^2\right), \qquad V_{s_1s_1}^{n,m,h,I,j} \approx \frac{\gamma_s\left(\gamma_s - 1\right)}{\left(s_I^1 + s_j^2\right)^2} V\left(t_n,\, x_m^1,\, x_h^2,\, s_I^1,\, s_j^2\right), \\
V_{s_2s_2}^{n,m,h,i,0} &\approx \frac{\gamma_s\left(\gamma_s - 1\right)}{\left(s_i^1 + s_0^2\right)^2} V\left(t_n,\, x_m^1,\, x_h^2,\, s_i^1,\, s_0^2\right), \qquad V_{s_2s_2}^{n,m,h,i,J} \approx \frac{\gamma_2\left(\gamma_2 - 1\right)}{\left(s_i^1 + s_J^2\right)^2} V\left(t_n,\, x_m^1,\, x_h^2,\, s_i^1,\, s_J^2\right).
\end{aligned}
\tag{D.6}
$$

All together, we can rewrite the finite-difference scheme (D.3) in matrix form as follows:

$$
\begin{aligned}
\boldsymbol{V}^{n-1} = \boldsymbol{V}^n &+ \sum_{q=1}^{2}\left(\frac{\Delta t \sigma_{x_q}^2}{2} D_{x_q}^2 \boldsymbol{V}^n - \Delta t \kappa_q \mathrm{diag}\left(\boldsymbol{x}_q\right) D_{x_q} \boldsymbol{V}^n\right) \\
&+ \sum_{q=1}^{2}\left(\frac{\Delta t \hat{\sigma}_q^2 \boldsymbol{s}_q^2}{2} D_{s_q}^2 \boldsymbol{V}^n - \frac{\Delta t \tilde{\sigma}_q^2 \boldsymbol{s}_q}{8} D_{s_q} \boldsymbol{V}^n\right) \\
&+ \sum_{q=1}^{2}\left[\Delta t\left(\mathrm{diag}\left(a_q\left(\boldsymbol{x}_q\right)\right)\mathrm{diag}\left(\boldsymbol{s}_q\right) - \boldsymbol{\omega}_q^{n-1}\right) + \Delta t\left(1 - \lambda_q\right)\boldsymbol{\omega}_q^{n-1} D_{s_q} \boldsymbol{V}^n\right],
\end{aligned}
\tag{D.7}
$$

where $\boldsymbol{x}_1 = \left[x_0^1,\, x_1^1,\, \cdots,\, x_M^1\right]^\top$, $\boldsymbol{x}_2 = \left[x_0^2,\, x_1^2,\, \cdots,\, x_H^2\right]^\top$, $\boldsymbol{s}_1 = \left[s_0^1,\, s_1^1,\, \cdots,\, s_I^1\right]^\top$, $\boldsymbol{s}_2 = \left[s_0^2,\, s_1^2,\, \cdots,\, s_J^2\right]^\top$, $\boldsymbol{V}^n\left(\boldsymbol{x}_1,\, \boldsymbol{x}_s,\, \boldsymbol{s}_1,\, \boldsymbol{s}_2\right) \in \mathbb{R}^4$, $\boldsymbol{\omega}_q^n\left(\boldsymbol{x}_1,\, \boldsymbol{x}_s,\, \boldsymbol{s}_1,\, \boldsymbol{s}_2\right) \in \mathbb{R}^4$ for $q = 1,\, 2$, $D_*^\star \boldsymbol{V}^n \in \mathbb{R}^4$, and $D_*^\star$ are derivative

operators that are defined as follows:

$$\begin{aligned}
\left[D^-_{x_1}\boldsymbol{V}^n\right]^{m,\ell}_{i,j} &= \frac{V^{n+1}_{m+1,\ell,i,j} - V^{n+1}_{m-1,\ell,i,j}}{2\Delta x_1}, \qquad \left[D^2_{x_1}\boldsymbol{V}^n\right]^{m,\ell}_{i,j} = \frac{V^n_{m-1,\ell,i,j} - 2V^n_{m,\ell,i,j} + V^n_{m+1,\ell,i,j}}{\Delta x_1^2}, \\
\left[D^-_{x_2}\boldsymbol{V}^n\right]^{m,\ell}_{i,j} &= \frac{V^n_{m,\ell+1,i,j} - V^n_{m,\ell-1,i,j}}{2\Delta x_2}, \qquad \left[D^2_{x_2}\boldsymbol{V}^n\right]^{m,\ell}_{i,j} = \frac{V^n_{m,\ell-1,i,j} - 2V^n_{m,\ell,i,j} + V^n_{m,\ell+1,i,j}}{\Delta x_2^2}, \\
\left[D^-_{s_1}\boldsymbol{V}^n\right]^{m,\ell}_{i,j} &= \frac{V^n_{m,\ell,i+1,j} - V^n_{m,\ell,i,j}}{\Delta s_1}, \qquad \left[D^2_{s_1}\boldsymbol{V}^n\right]^{m,\ell}_{i,j} = \frac{V^n_{m,\ell,i-1,j} - 2V^n_{m,\ell,i,j} + V^n_{m,\ell,i+1,j}}{\Delta s_1^2}, \\
\left[D^-_{s_2}\boldsymbol{V}^n\right]^{m,\ell}_{i,j} &= \frac{V^n_{m,\ell,i,j+1} - V^n_{m,\ell,i,j}}{\Delta s_2}, \qquad \left[D^2_{s_2}\boldsymbol{V}^n\right]^{m,\ell}_{i,j} = \frac{V^n_{m,\ell,i,j-1} - 2V^n_{m,\ell,i,j} + V^n_{m,\ell,i,j+1}}{\Delta s_2^2},
\end{aligned} \tag{D.8}$$

where $\left[D^\star_* \boldsymbol{V}^n\right]^{m,\ell}_{i,j}$ represents the element in the matrix $D^\star_* \boldsymbol{V}^n$ at location $\left(x^1_m,\, x^2_h,\, s^1_i,\, s^2_j\right)$. At each time $t = t_n$ for $n = N,\ N-1,\ \cdots,\ 2,\ 1$, the value of equation (D.7) is calculated, and the optimal values for the control variables ω_q for $q = 1,\ 2$ are updated by using condition (5.30),

$$\left[\boldsymbol{\omega}^n_q\right]^{m,h}_{i,j} = \begin{cases} \bar{\omega}_q, & \text{if } (1-\lambda_q)\,\bar{\omega}_q \frac{V^n_{m,h,i+1,j} - V^n_{m,h,i,j}}{\Delta s_q} > \bar{\omega}_q, \\ 0, & \text{otherwise,} \end{cases} \tag{D.9}$$

where $\left[\boldsymbol{\omega}^n_q\right]^{m,h}_{i,j} = \omega_q\left(t_n,\, x^1_m,\, x^2_h,\, s^1_i,\, s^2_j\right)$. Algorithm Appendix D.1 gives the pseudo-code for implementing the finite-difference method (D.7) for solving the two-pool case of the yield farming problem.

Algorithm Appendix D.1: Iterative Policy Updating Algorithm Using Finite-Difference Scheme for Two Liquidity Pools. Parameter *max_iter* taken to be 5 or 10.

```
Initialize ω_1^n = 0 and ω_2^n = 0 for all n ⩽ N;
for iteration in range(max_iter) do
    Initialize V^N = U using equation (5.27);
    for n in reversed(range(N)) do
        Calculate V^{n-1} using equation (D.7) and current values of ω_1^{n-1} and ω_2^{n-1};
        Update ω_1^{n-1} and ω_2^{n-1} using equation (D.9) and V^{n-1} computed in previous step ;
    end
end
```

Similarly, equation (5.23) for the model of single liquidity pool can be solved numerically by using the finite-difference method described thus far. As with the model of two liquidity pools, the first- and second-order derivatives with respect to x – here x corresponds to the OU process X_t in the stochastic system (5.19) – are approximated by the central difference, the derivative with respect to s is approximated by the first-order upwind scheme, and the derivative for the time variable t is approximated by a first-order explicit scheme. Consequently, we have

$$\begin{aligned}
V^{n-1}_{i,j} = &\left(\frac{\Delta t\sigma^2}{2\Delta x^2} + \frac{\Delta t\kappa x_i}{2\Delta x}\right) V^n_{i-1,j} + \left(1 - \frac{\Delta t\sigma^2}{\Delta x^2} - \frac{\Delta t\hat{\sigma}^2 s_j^2}{\Delta s^2}\right) V^n_{i,j} \\
&+ \left(\frac{\Delta t\sigma^2}{2\Delta x^2} - \frac{\Delta t\kappa x_i}{2\Delta x}\right) V^n_{i+1,j} + \frac{\Delta t\hat{\sigma}^2 s_j^2}{2\Delta s^2} V^n_{i,j-1} + \frac{\Delta t\hat{\sigma}^2 s_j^2}{2\Delta s^2} V^n_{i,j+1}
\end{aligned} \tag{D.10}$$

$$+\sup_{\substack{w\in\{0,\,\bar{w}\}\\ \nu\in\{0,\,\bar{\nu}\}}}\left[\Delta t a\left(x_i\right)s_j-\Delta t\left(w-\nu\right)+\frac{\Delta t\left(w-\lambda_w w-\nu-\lambda_\nu\nu-\frac{\tilde{\sigma}^2}{8}s_j\right)}{\Delta s}V_{i,j+1}^n\right.$$
$$\left.-\frac{\Delta t\left(w-\lambda_w w-\nu-\lambda_\nu\nu-\frac{\tilde{\sigma}^2}{8}s_j\right)}{\Delta s}V_{i,j}^n\right],$$

where $V_{i,j}^n$ represents the values of $V\left(t_n,\,x_i,\,s_j\right)$ in the discretized domain,

$$p_i^-=\frac{\Delta t\sigma^2}{2\Delta x^2}+\frac{\Delta t\kappa x_i}{2\Delta x},\quad p_i^+=\frac{\Delta t\sigma^2}{2\Delta x^2}-\frac{\Delta t\kappa x_i}{2\Delta x},\quad p_j^-=\frac{\Delta t\hat{\sigma}^2 s_j^2}{2\Delta s^2},$$
$$p_i=1-\frac{\Delta t\sigma^2}{\Delta x^2}-\frac{\Delta t\hat{\sigma}^2 s_j^2}{\Delta s^2}+\frac{\Delta t\tilde{\sigma}^2 s_j}{8\Delta s}-\frac{\Delta t\left(1-\lambda_\omega\right)\omega}{\Delta s}+\frac{\Delta t\left(1+\lambda\right)\nu}{\Delta s},$$
$$p_j^+=\frac{\Delta t\hat{\sigma}^2 s_j^2}{2\Delta s^2}-\frac{\Delta t\tilde{\sigma}^2 s_j}{8\Delta s}+\frac{\Delta t\left(1-\lambda_\omega\right)\omega}{\Delta s}-\frac{\Delta t\left(1+\lambda_\nu\right)\nu}{\Delta s},$$

are the transition probabilities whose values are in (0, 1) between different states of x_i and s_j, and $p_i^-+p_i+p_i^++p_j^-+p_j^+=1$ for all cases $\omega=0$ or $\bar{\omega}$ and $\nu=0$ or $\bar{\nu}$. The numerical scheme that is given by equation (D.10) is stable if it satisfies a CFL condition, which is checked probabilistically by using the Markov chain approximation method as well. The numerical scheme (D.10) is solved in backward time t because we have a terminal condition $V\left(T,\,x,\,s\right)=\frac{\eta}{\gamma}s^\gamma$. We use the same approach to set the (Neumann) boundary conditions for variables s and x.

Appendix E Model Calibration

This appendix provides the details to estimate the parameters of the OU processes. The OU process X_t is defined in equation (5.17) and is stationary if and only if $\kappa > 0$. The APR for the single liquidity pool model that is proposed in Section 5.3.2 is an exponential OU process $APR_t = a(X_t) \coloneqq \beta e^{X_t}$. Taking the logarithm, we have $\ln APR_t = X_t + \ln \beta$, which is a stationary OU process with same parameters κ and σ, but with long-term mean $\ln \beta$. Given data $APR_n = APR_{t_n}$ for $n = 0, 1, 2, \cdots, N$, with time step $\Delta t = t_{n+1} - t_n$, $\forall n$, we estimate $\ln \beta$, κ, and σ as follows.

First, we have the time-series average is the estimator for $\ln \beta$,

$$\widehat{\ln \beta} = \frac{1}{N} \sum_{n=0}^{N-1} \ln APR_n.$$

Next, we consider the recursion for the centered process,

$$X_{t+1} = (1 - \kappa \Delta t) X_t + \epsilon_t, \tag{E.1}$$

where $\epsilon_t \sim \mathcal{N}\left(0,\, \sigma^2 \Delta t\right)$. Using estimator $\widehat{\ln \beta}$ we define the sample centered process as $\widehat{X}_n = APR_n - \widehat{\ln \beta}$, which is then plugged into equation (E.1) for the estimation of κ. The estimator for κ is

$$\hat{\kappa} = \frac{1}{\Delta t} \left(1 - \frac{\sum_{n=0}^{N-1} \widehat{X}_{n+1} \widehat{X}_n}{\sum_{t=0}^{N-1} \widehat{X}_n^2} \right).$$

Typically the step-size Δt is small so that $\kappa \Delta t \ll 1$, hence we have $-\ln(1 - \kappa \Delta t) \sim \kappa \Delta t$, and therefore, the estimator for κ becomes $\hat{\kappa} = -\frac{1}{\Delta t} \ln \left(\frac{\sum_{t=0}^{N-1} \widehat{X}_{n+1} \widehat{X}_n}{\sum_{n=0}^{N-1} \widehat{X}_n^2} \right)$. Finally, the estimation of σ is calculated by summing the squared centered process $\widehat{X}_n$, in other words $\widehat{\sigma^2} = \frac{2\hat{\kappa}}{N} \sum_{n=0}^{N-1} \left(\widehat{X}_n \right)^2$. In summary, the estimators for β, κ, and σ, respectively are:

$$\begin{aligned}
\widehat{\ln \beta} &= \frac{1}{N} \sum_{n=0}^{N-1} \ln APR_n, \\
\hat{\kappa} &= -\frac{1}{\Delta t} \ln \left(\frac{\sum_{n=0}^{N-1} \left(\ln APR_{n+1} - \widehat{\ln \beta} \right) \left(\ln APR_n - \widehat{\ln \beta} \right)}{\sum_{n=0}^{N-1} \left(\ln APR_n - \widehat{\ln \beta} \right)^2} \right), \\
\widehat{\sigma^2} &= \frac{2\hat{\kappa}}{N} \sum_{n=0}^{N-1} \left(\ln APR_n - \widehat{\ln \beta} \right)^2.
\end{aligned}$$

By the ergodic theory for the stationary OU process, we have $\mathbb{P}\left(\left| \widehat{\ln \beta} - \ln \beta \right| \geqslant \epsilon \right)$, $\mathbb{P}\left(|\hat{\kappa} - \kappa| \geqslant \epsilon \right)$, and $\mathbb{P}\left(\left| \widehat{\sigma^2} - \sigma \right| \geqslant \epsilon \right)$ as $N \to +\infty$ for any $\epsilon > 0$. For the central limit theory and further mathematical details of the ergodic theory, we refer to Kutoyants (2004). For the OU processes that are defined by equation (5.25) for the two liquidity pools model that is proposed in Section 5.3.3, we use the same method to estimate their parameters β_i, κ_i, and σ_i, for $i = 1, 2$. To estimate the parameters of the OU processes that are defined in equation (5.25), we use the same approach provided above.

Appendix F Model – Additional Figures

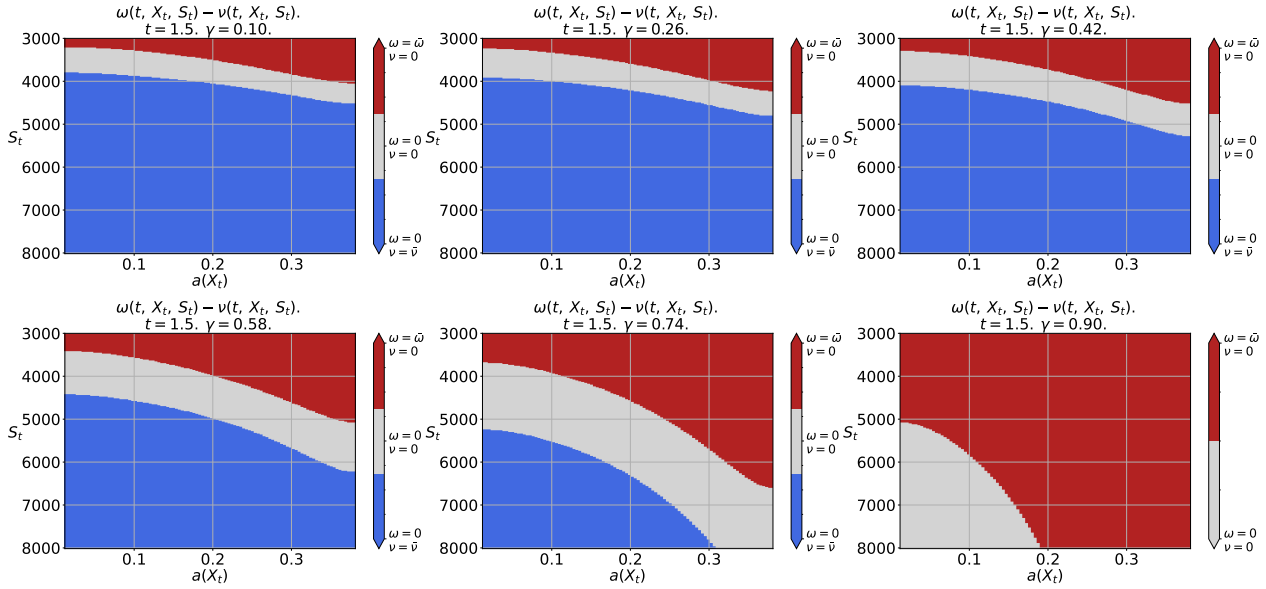

Figure F.1: The optimal recompounding strategy $\omega\left(t,\ X_t,\ S_t\right)$ with different values of the risk tolerance coefficients γ for $\bar{\text{C}}$urve. The time point is $t = 1.5$. The value for the transaction cost $c = \lambda\bar{\omega}$ is 10 USD. The horizontal axis is labeled by the values of APR for $\bar{\text{C}}$urve; in other words, $a\left(X_t\right) = \beta e^{X_t}$. The associated stochastic control problem is formulated in Section 5.3.2 and Section 6.1.

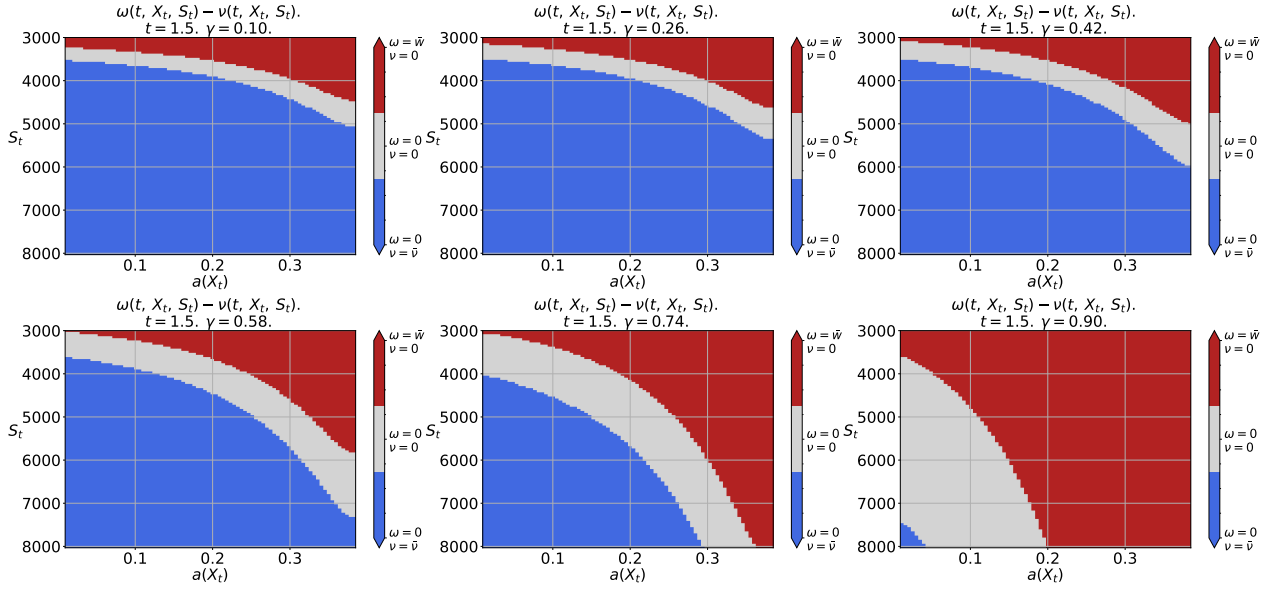

Figure F.2: The optimal recompounding strategies $\omega\left(t,\ X_t,\ S_t\right) - \nu\left(t,\ X_t,\ S_t\right)$ of different values of the risk tolerance coefficients γ for $\bar{\text{U}}$niswap. The time point is $t = 1.5$. The value of the transaction cost is $c = \lambda\bar{\omega} = \lambda\bar{\nu} = 10$ USD. The horizontal axis is labeled by the values of APR for the Uniswap pool; in other words, $a\left(X_t\right) = \beta e^{X_t}$. The associated stochastic control problem is formulated in Section 5.3.2 and Section 6.2.

Internet Appendix

I Empirical Analysis – Additional Tables and Figures

Token	Type	Market Capitalization	Collateralized	Publisher
DAI	Stablecoin	6.86b	Crypto	makerdao
USDC	Stablecoin	48.88b	Fiat	centre
USDT	Stablecoin	67.95b	Fiat	tether
TUSD	Stablecoin	877.90m	Fiat	trueusd
BUSD	Stablecoin	21.06b	Fiat	binance
sUSD	Stablecoin	69.75m	Crypto	synthetix
USDPAX	Stablecoin	943.12m	Fiat	paxos

Table I.1: Token Overview – Curve Pools. The table reports the names, type, the market capitalization (as of September 28, 2022), the collateralization, and the publisher of the tokens traded in the six pools from Curve. The data is obtained from `https://coinmarketcap.com/`.

Panel A: Adding Liquidity

Liquidity Pool	Mean	Standard Deviation	Minimum	Maximum
compound	50.677	57.654	0.796	279.420
y	27.166	39.089	0.288	172.450
busd	23.850	29.260	1.732	116.813
susdv2	44.250	51.677	0.326	237.268
pax	13.675	17.870	0.565	48.006
3pool	32.656	32.085	1.431	143.708
C̄urve	34.478	36.795	0.307	166.611

Panel B: Remove Liquidity

Liquidity Pool	Mean	Standard Deviation	Minimum	Maximum
compound	11.772	21.873	0.068	94.594
y	7.109	9.273	0.088	45.002
busd	8.389	9.718	0.652	35.203
susdv2	14.533	20.419	0.000	109.446
pax	0.451	0.169	0.332	0.571
3pool	25.069	25.097	0.000	106.860
C̄urve	23.276	26.493	0.000	121.963

Table I.2: Summary Statistics – Curve Transaction Costs. The table reports the summary statistics (mean, standard deviation, minimum, and maximum observations) of transactions costs (in USD), where Panel A describes the action add liquidity and Panel B the action remove liquidity, of the six liquidity pools from Curve, as described in Table 4.1 and their average (C̄urve pool). The data is obtained from etherscan.io and sampled daily. The sample period is from 02-2020 to 10-2022. The data is winzorized at the 1% quantile.

Liquidity Pool	Mean	Standard Deviation	Minimum
compound	-0.00023	0.00085	-0.00616
y	-0.00020	0.00065	-0.00459
busd	-0.00020	0.00064	-0.00454
susdv2	-0.00454	0.01151	-0.08100
pax	-0.00031	0.00087	-0.00636
3pool	-0.00023	0.00077	-0.00541
$\bar{\text{C}}$urve	-0.00095	0.00219	-0.01801

Table I.3: Summary Statistics – Curve Impermanent Loss. The table reports the summary statistics (mean, standard deviation, and minimum observations) (Panel A) of the impermanent loss (annualized) of the six liquidity pools (and their average called $\bar{\text{C}}$urve) from Curve. The data is sampled daily, and the sample period is from 02-2020 to 10-2022. The price data is winzorized at the 1% quantile.

Panel A: Summary Statistics

Liquidity Pool	Mean	Standard Deviation	Minimum	Maximum
compound	0.041	0.038	-0.338	0.201
y	0.050	0.060	-0.090	0.353
busd	0.048	0.052	0.001	0.274
susdv2	0.023	0.033	0.000	0.201
pax	0.015	0.028	0.000	0.178
3pool	0.008	0.010	-0.006	0.071
$\bar{\text{C}}$urve	0.031	0.040	0.001	0.240

Panel B: Correlation Matrix

Liquidity Pool	compound	y	busd	susdv2	pax	3pool	$\bar{\text{C}}$urve
compound	1.000	0.556	0.585	0.427	0.414	0.498	0.661
y	0.556	1.000	0.704	0.361	0.371	0.336	0.763
busd	0.585	0.704	1.000	0.462	0.470	0.468	0.754
susdv2	0.427	0.361	0.462	1.000	0.532	0.480	0.680
pax	0.414	0.371	0.470	0.532	1.000	0.459	0.680
3pool	0.498	0.336	0.468	0.480	0.459	1.000	0.616
$\bar{\text{C}}$urve	0.661	0.763	0.754	0.680	0.680	0.616	1.000

Table I.4: Summary Statistics and Correlation Dynamics – Curve APRs. The table reports the summary statistics (mean, standard deviation, and minimum and maximum observations) (Panel A) and the correlation matrix (Panel B) of APRs (in decimals) of the six liquidity pools (and their average, called $\bar{\text{C}}$urve) from Curve. The data is sampled daily, and the sample period is from 02-2020 to 10-2022. The data is winzorized at the 1% quantile.

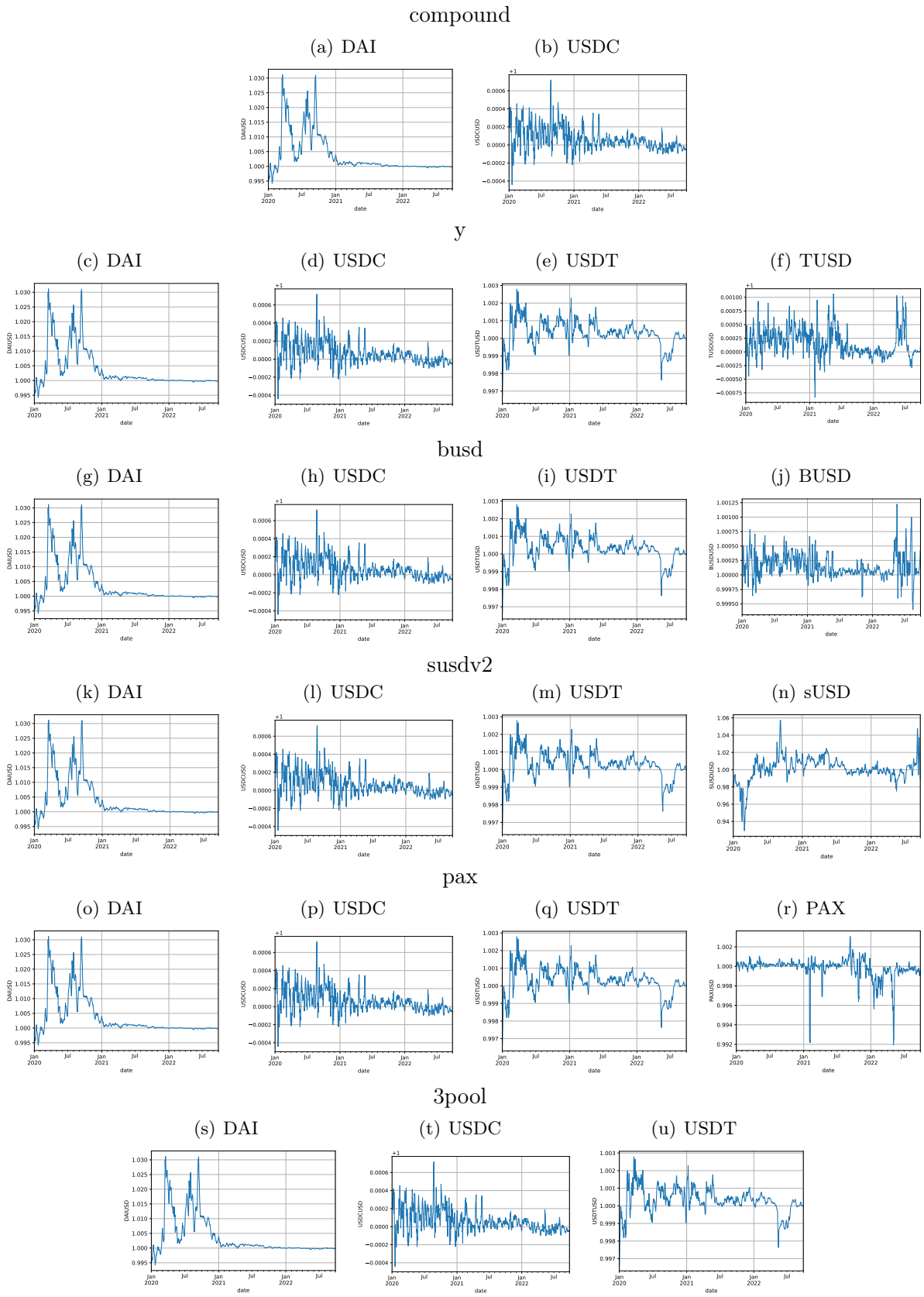

Figure I.1: Curve – Pool Token Prices. The figure shows the time-series plots of the respective token prices within each pool. Details on the pools and the token pairs are provided in Table 4.1 and Table I.1. The data is sampled daily, and the sample period ranges from 01-2020 to 10-2022. The data is winsorized at the 1% quantile. In the plots, the five-day moving average is depicted.

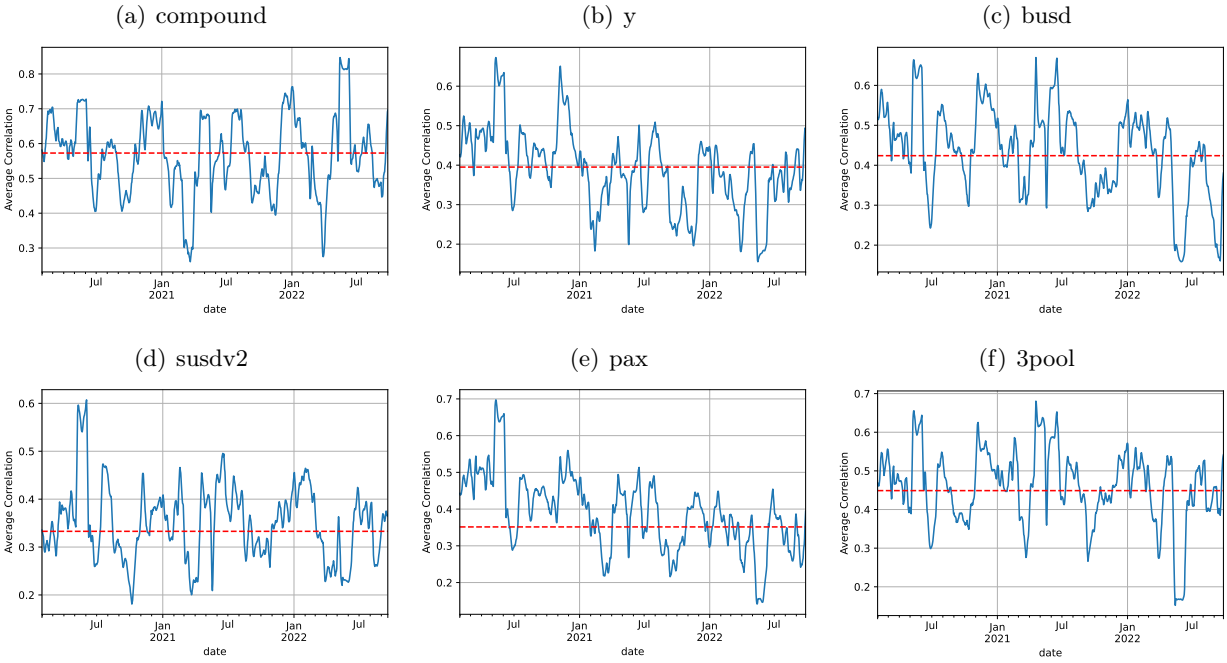

Figure I.2: Curve – Average Pool Token Correlation. The figure shows the time-series plot of the average token correlations (calculated over a rolling window of 30 days) within each pool. The dotted line represents the unconditional average correlation. Details on the pools and the token pairs are provided in Table 4.1 and Table I.1. The data is sampled daily, and the sample period is from 02-2021 to 10-2022. The price data is winsorized at the 1% quantile. In the plots, the five-day moving average is depicted.

II Model – Additional Figures

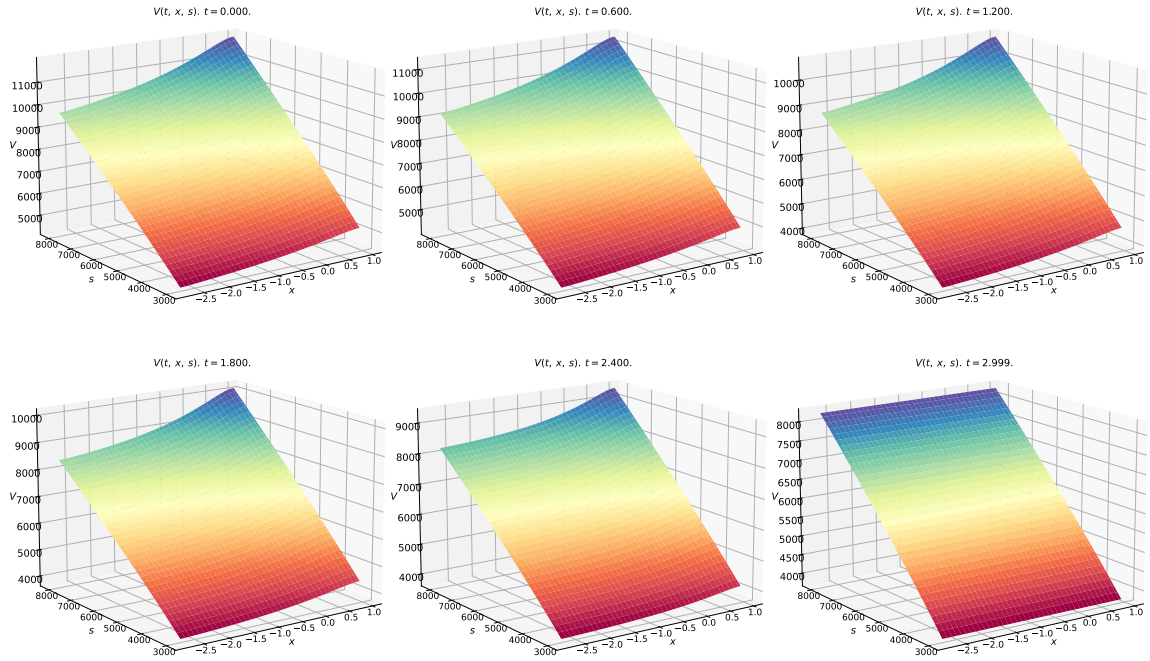

Figure II.1: The figure shows the solution for the HJB equation (5.23) with respect to $V(t, x, s)$ at different time t for the Ūniswap liquidity pool. The value of the risk tolerance coefficient is $\gamma = 0.8$, and the value for the transaction cost c is set to 10 USD.

III Two Liquidity Pools with Cobb-Douglas Utility

For the two liquidity pools model in Section 5.3.3, we can take the Cobb–Douglas production function as the utility

$$U\left(K_t,\, S_t^1,\, S_t^2\right) = \frac{1}{\gamma_1}\left(S_t^1\right)^{\gamma_1}\frac{1}{\gamma_2}\left(S_t^2\right)^{\gamma_2}\frac{1}{\gamma_3}K_t^{\gamma_3}, \tag{III.1}$$

where $\gamma_j \in (0,\, 1)$ for $j = 1,\, 2$, and 3 are the risk tolerance coefficients, and $\sum_{j=1}^{3}\gamma_j = 1$. This utility is strictly concave in K_t, whereas the utility in equation (III.1) is linear. When solving numerically, the utility function in equation (III.1) will lead to output that has noticeable differences from the output presented in Section 6.3. In the implementation, the utility function (III.1) does not permit a dimension-reducing ansatz as does the utility of equation (III.1), and therefore the numerical schemes have slower runtime and are computed on a less fine mesh.

III.1 Optimal Liquidity Provision

For the two liquidity pools case with utility equation (III.1), we consider a system similar to that of equation (5.25), but which will be easier for numerical computation because APRs are drive by a single, common OU process,

$$\begin{aligned}
dK_t &= \sum_{i=1}^{2}\left(a_i\left(X_t\right)S_t^i - \omega_t^i + \nu_t^i\right)dt, \\
dS_t^i &= \left(\omega_t^i - \nu_t^i - c_i\mathbf{1}_{\{\omega_t^i>0\}} - \frac{\tilde{\sigma}_i^2}{8}S_t^i\right)dt + \hat{\sigma}_i S_t^i d\hat{B}_t^i,\ \text{for } i = 1,\, 2 \\
dX_t &= -\kappa X_t dt + dB_t,
\end{aligned} \tag{III.2}$$

where B_t and $\hat{B}_t^i$ for $i = 1,\, 2$ are three mutually independent one-dimensional standard Brownian motions, $\omega_t^i \in [0,\, \bar{\omega}_i]$, $\nu_t^i \in [0,\, \bar{\nu}_i]$, $\kappa > 0$, and $c_i > 0$ are the constant transaction costs for the two different liquidity pools. The control variable ω_t^i represents at each time t the amount of the rewarded i^{th} token that is recompounded back into the i^{th} liquidity pool. As done for the two liquidity pools model proposed in Section 5.3.3, here we set $\nu_t^i \equiv 0\ \forall t$.

The two APRs $a_i\left(X_t\right)$ in equation (III.2) are driven by the same OU process X_t. However, there is still an obvious difference when we compare it with the SDE (5.17) or with equation (5.25), which is that there is not a volatility parameter σ in the diffusion term dB_t. The reason is that in order to reduce the dimensions of the stochastic control problem, we must assume that the APRs for the two different liquidity pools are driven by the same OU process. Meanwhile, we still want to capture the volatilities for the two different liquidity pools; therefore, the two different σ_i are specified within the functions of the APRs $a_i\left(X_t\right)$ for $i = 1,\, 2$ for the two liquidity pools, respectively.

The control processes ω_t^i for $i = 1,\, 2$ are progressively measurable process, which are sought to maximize the expectation of the utility function (III.1) with respect to to the wallet K_t and the staking amounts S_t^i at terminal time $t = T$. Therefore, the value function is

$$Q\left(t,\, x,\, k,\, s_1,\, s_2\right) = \sup_{\left(\omega_t^1,\omega_t^2\right)\in\mathcal{A}} \mathbb{E}\left[U\left(K_T,\, S_T^1,\, S_T^2\right) \,\middle|\, X_t = x,\, K_t = k,\, S_t^1 = s_1,\, S_t^2 = s_2\right], \tag{III.3}$$

where $\left(X_t,\, K_t,\, S_t^1,\, S_t^2\right)$ follow equation (III.2), $t \in [0,\, T]$ is the time variable, $(x,\, k,\, s_1,\, s_2) \in \mathbb{R} \times$

$\mathbb{R}^+ \times \mathbb{R}^+ \times \mathbb{R}^+$, and $\mathcal{A}$ is the set of admissible controls, as follows:

$$\mathcal{A} = \left\{ \left(\omega_t^1,\, \omega_t^2\right)_{0 \leqslant t \leqslant T} \mid \left(\omega_t^1,\, \omega_t^2\right) \text{ is } \mathcal{F}_t\text{-measurable with } \omega_t^i \in [0,\, \bar{\omega}_i] \text{ for } i = 1, 2 \right\},$$

with the control bounds $\bar{\omega}_i$ for $i = 1,\, 2$ being parameters such that $0 < \bar{\omega}_i < \infty$.

The value function $Q\left(t,\, x,\, k,\, s_1,\, s_2\right)$ defined in equation (III.3) has the following HJB equation:

$$\begin{aligned} & Q_t + \mathcal{L}Q + \frac{1}{2}\sum_{i=1}^{2} \hat{\sigma}_i^2 s_i^2 Q_{s_i s_i} \\ & \quad + \sup_{\substack{\omega_1 \in [0,\, \bar{\omega}_1] \\ \omega_2 \in [0,\, \bar{\omega}_2]}} \sum_{i=1}^{2} \left[\left(a_i\left(x\right) s_i - \omega_i\right) Q_k + \left(\omega_i - c_i \mathbf{1}_{\{\omega_i > 0\}} - \frac{\tilde{\sigma}_i^2}{8} s_i \right) Q_{s_i} \right] = 0, \end{aligned} \tag{III.4}$$

where $\mathcal{L}Q = \frac{1}{2} Q_{xx} - \kappa x Q_x$, and terminal condition is $Q\left(T,\, x,\, k,\, s_1,\, s_2\right) = \frac{1}{\gamma_1} s_1^{\gamma_1} \frac{1}{\gamma_2} s_2^{\gamma_2} \frac{1}{\gamma_3} k^{\gamma_3}$. The linearity of the Hamiltonian with respect to ω_i for $i = 1,\, 2$ leads to the optimal controls being of a bang-bang type,

$$\omega_i = \begin{cases} \bar{\omega}_i, & \text{if } \left(\bar{\omega}_i - c_i\right) Q_{s_i} > \bar{\omega}_i Q_k, \\ 0, & \text{otherwise.} \end{cases}$$

Let $\lambda_i \in (0,\, 1)$ for $i = 1,\, 2$ be parameters such that $c_i = \lambda_i \bar{\omega}_i$. The SDEs for the staking amounts defined in equation (III.2) become $dS_t^i = \left(\left(1 - \lambda_i\right) \omega_t^i - \frac{\tilde{\sigma}_i^2}{8} S_t^i \right) dt + \hat{\sigma}_i S_t^i d\hat{B}_t^i$ with $\omega_t^i \in \{0,\, \bar{\omega}_i\}$. Then the HJB equation (III.4) with bang-bang controls can be written succinctly as

$$\begin{aligned} & Q_t + \mathcal{L}Q + \frac{1}{2}\sum_{i=1}^{2} \hat{\sigma}_i^2 s_i^2 Q_{s_i s_i} \\ & \quad + \sup_{\substack{\omega_1 \in \{0,\, \bar{\omega}_1\} \\ \omega_2 \in \{0,\, \bar{\omega}_2\}}} \sum_{i=1}^{2} \left[\left(a_i\left(x\right) s_i - \omega_i\right) Q_k + \left(\left(1 - \lambda_i\right) \omega_i - \frac{\tilde{\sigma}_i^2}{8} s_i \right) Q_{s_i} \right] = 0, \end{aligned} \tag{III.5}$$

where $\mathcal{L}Q$ is defined as it is in equation (III.4), and the terminal condition is $Q\left(T,\, x,\, k,\, s_1,\, s_2\right) = \frac{1}{\gamma_1} s_1^{\gamma_1} \frac{1}{\gamma_2} s_2^{\gamma_2} \frac{1}{\gamma_3} k^{\gamma_3}$. In terms of λ_1 and λ_2, the optimal bang-bang controls are expressed as

$$\omega_i = \begin{cases} \bar{\omega}_i, & \text{if } \left(1 - \lambda_i\right) \bar{\omega}_i Q_{s_i} > \bar{\omega}_i Q_k, \\ 0, & \text{otherwise.} \end{cases}$$

III.2 Numerical Experiments – Uniswap

The control problem (III.3) is solved numerically by utilizing the finite-difference method that is proposed in Appendix D. The bounded discretized domain $\mathcal{D}$ is similar to equation (D.1); however, one of the $\mathcal{X}_i$ is substituted by $\mathcal{K} = [k_{\min},\, k_{\max}] = \{k_{\min} = k_0,\, k_1,\, \cdots,\, k_h,\, \cdots,\, k_H = k_{\max}\}$, $k_h = k_{\min} + h\Delta k$, and $\Delta k = \frac{1}{H}\left(k_{\max} - k_{\min}\right) > 0$ is the discretization step. Also, there is a new term Q_k in the HJB equation (III.5) that does not appear in equation (5.23) or equation (5.29). It is the first-order derivative with respect to variable k, which is approximated by a first-order upwind scheme,

$$Q_k^{n,m,h,i,j} \approx \frac{Q_{m,h+1,i,j}^n - Q_{m,h,i,j}^n}{\Delta k}. \tag{III.6}$$

Because Q_k is approximated by the upwind scheme (III.6), we therefore only need to set the Neumann boundary condition for $k_h = k_H$; that is,

$$Q_k^{n,m,H,i,j} \approx \frac{\gamma_3}{k_H} Q\left(x_m,\, k_H,\, s_i^1,\, s_j^2\right).$$

For the numerical examples of Cobb–Douglas production function, we select ChainLink and Wrapped Ethereum (LINK/WETH) as the first pool and Wrapped Ethereum and Curve (WETH/CRV) as the second pool. Empirically, the average APR for the first pool (second pool) is around 32% (28%), and the unconditional correlation between the pool APRs is close to zero (1.6%) over the sample period. We assume that the dynamics of APRs are $a_i\left(X_t\right) := \beta_i e^{\sigma_i X_t}$ for $i = 1, 2$, where X_t is the OU process that is defined in equation (III.2). For parameter estimation, we follow Appendix E and obtain $\sigma_1 = 0.61$, $\sigma_2 = 0.74$, $\beta_1 = 0.30$, $\beta_2 = 0.25$, $\kappa_1 = 3.24$, and $\kappa_2 = 2.37$; therefore, we set $\kappa = \frac{1}{2}\left(\kappa_1 + \kappa_2\right)$.[39] We estimate $\hat{\sigma}_1 = 0.26$, $\hat{\sigma}_2 = 0.32$, $\tilde{\sigma_1} = 0.14$ and $\tilde{\sigma_2} = 0.12$. For the utility function (III.1), we set the risk tolerance coefficients to $\gamma_1 = 0.33$, $\gamma_2 = 0.33$, and $\gamma_3 = 0.34$ and the bound for the control to $\bar{\omega}_1 = \bar{\omega}_2 = 100$. For the discretized computational domain $\mathcal{D}$, we set $t_{\min} = 0$, $t_{\max} = 3$, $x_{\min} = -9.5\sigma/\sqrt{2\kappa}$, $x_{\max} = 1.25\sigma/\sqrt{2\kappa}$, $k_{\min} = 0$, $k_{\max} = 2000$, $s^1_{\min} = s^2_{\min} = 1000$, $s^1_{\max} = s^2_{\max} = 3000$, $N = t_{\max} \times 365 \times 3$, $M = 60$, $H = 10$, $I = 25$, and $J = 25$.

Figures III.1 illustrates the optimal control w_1 corresponding to the individual pool. The results are similar for the second pool and therefore not reported. The plot exhibits a similar pattern as observed in the single-pool case. At the initial time $t = 0$, there is a significant compounding region, while near the terminal time $t = T$, a no-recompounding region is evident. The area for intermediate points between these two regions gradually decreases.

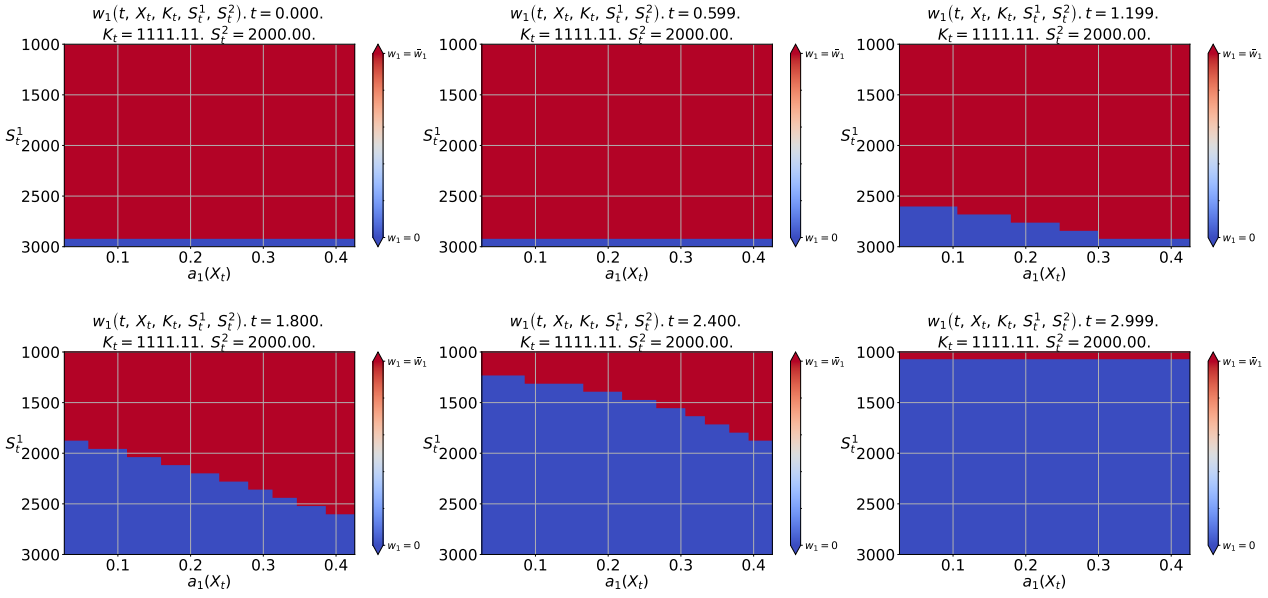

Figure III.1: Evolution of the optimal recompounding strategy $w_1\left(t,\, X_t,\, K_t,\, S_t^1,\, S_t^2\right)$ at different times t for the first liquidity pool LINK/WETH. The transaction costs are set to $c = 10$ USD. The risk tolerance is set to $\gamma_1 = 0.33$, $\gamma_2 = 0.33$, and $\gamma_3 = 0.34$. The horizontal axis is labeled by the values of APR for the liquidity pool LINK/WETH; in other words, $a_1\left(X_t\right) = \beta_1 e^{\sigma_1 X_t}$.

Figure III.2 displays the projection of w_1 and w_2 on $\{0, \bar{\omega}_1\} \times \{0, \bar{\omega}_2\}$. The plots show that at

[39] As in the single liquidity pool case, we apply a moving average over the last 30 days to the time-series is from 12-2021 to 12-2022 before estimating the coefficients of the exponential OU processes.

$t = 0$, the agent consistently chooses to recompound. However, as we approach $t = T$, the costs of recompounding outweigh the benefits. Notably, the dynamics at intermediate time points are intriguing, with clear divisions into the four discussed areas. The plots reveal a simultaneous reduction in the optimal recompounding region for both pools. Moreover, the individual recompounding areas shift from being concentrated across the entire corner at $t = 1.199$ to a rectangular concentration at lower values of respective range of S_t ($t = 1.8$ and $t = 2.4$).

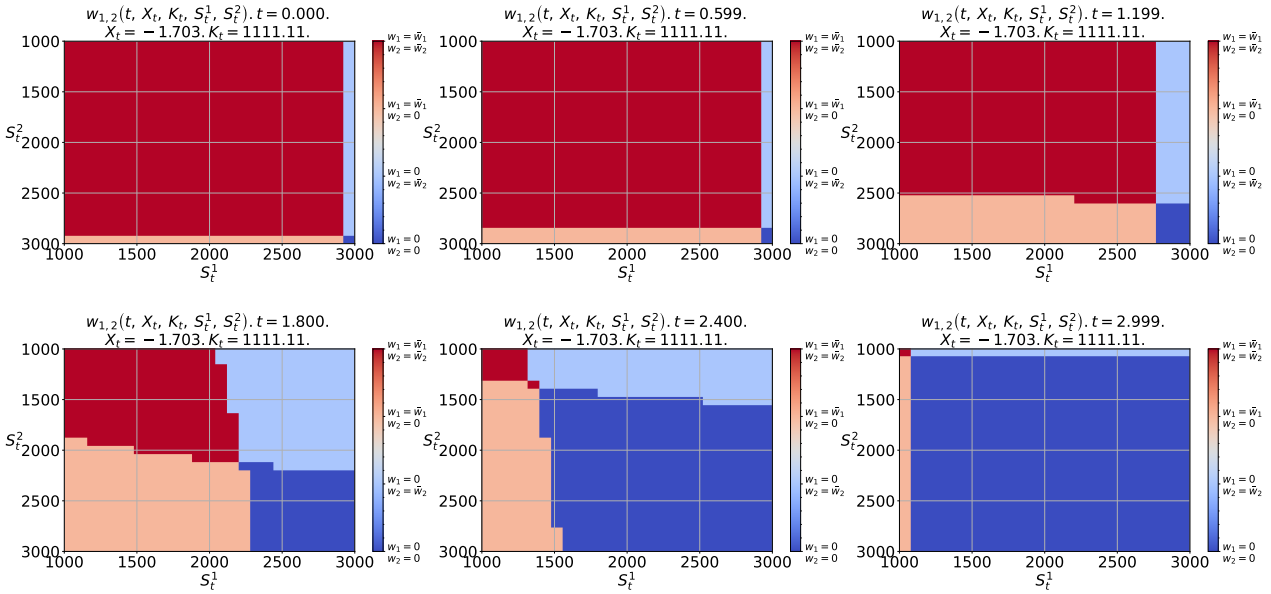

Figure III.2: Evolution of the optimal recompounding strategies $\omega_{1,2}\left(t,\ X_t,\ K_t,\ S_t^1,\ S_t^2\right) \in \{0, \bar{\omega}_1\} \times \{0, \bar{\omega}_2\}$ at different times t for both of the liquidity pools LINK/WETH and WETH/CRV. The transaction costs are set to $c = 10$ USD. The risk tolerance is set to $\gamma_1 = 0.33$, $\gamma_2 = 0.33$, and $\gamma_3 = 0.34$. The functions of APRs for the pools are $a_i\left(X_t\right) = \beta_i e^{\sigma_i X_t}$, for $i = 1,\ 2$.

To investigate optimal recompounding behavior under changing market conditions (APRs), we conduct a comparative statics analysis. By fixing the APR of the first liquidity pool and increasing the APRs for the second pool, we examine the dynamics of the controls w_1 and w_2 at $t = 1.5$. Figure III.3 illustrates the plots depicting these dynamics. Notably, the light blue recompounding region for w_2 expands as APR_2 increases. Additionally, the dark red recompounding region for $\omega 1 = \bar{\omega} 1$ and $\omega 2 = \bar{\omega}_2$ grows with APR_2, indicating that the agent not only recompounds more frequently within pool 2 but also utilizes the additional resources from pool 2 to recompound more often within pool 1.

Next we investigate the optimal recompounding solutions with respect to changing transaction costs. To do these analyses we will fix the transactions costs on one pool and vary transaction costs in the other. Figure III.4 shows the optimal controls at $t = 1.5$ for fixed transaction costs in pool 1 of LINK/WETH ($\lambda_1 = 0.1$, which equals 10 USD) and varying in pool 2 of WETH/CRV (from 1 USD to 50 USD). As seen from the plots, when transaction costs on pool 2 increase, its individual recompounding region (light blue) becomes smaller. In contrast, the individual recompounding region of pool 1 (light red) becomes larger. For extreme transaction costs of 50 USD, the optimal criteria command recompounding almost only in pool 1. The model, therefore, delivers reasonable and intuitive dynamics for the optimal control and changes in transaction costs.

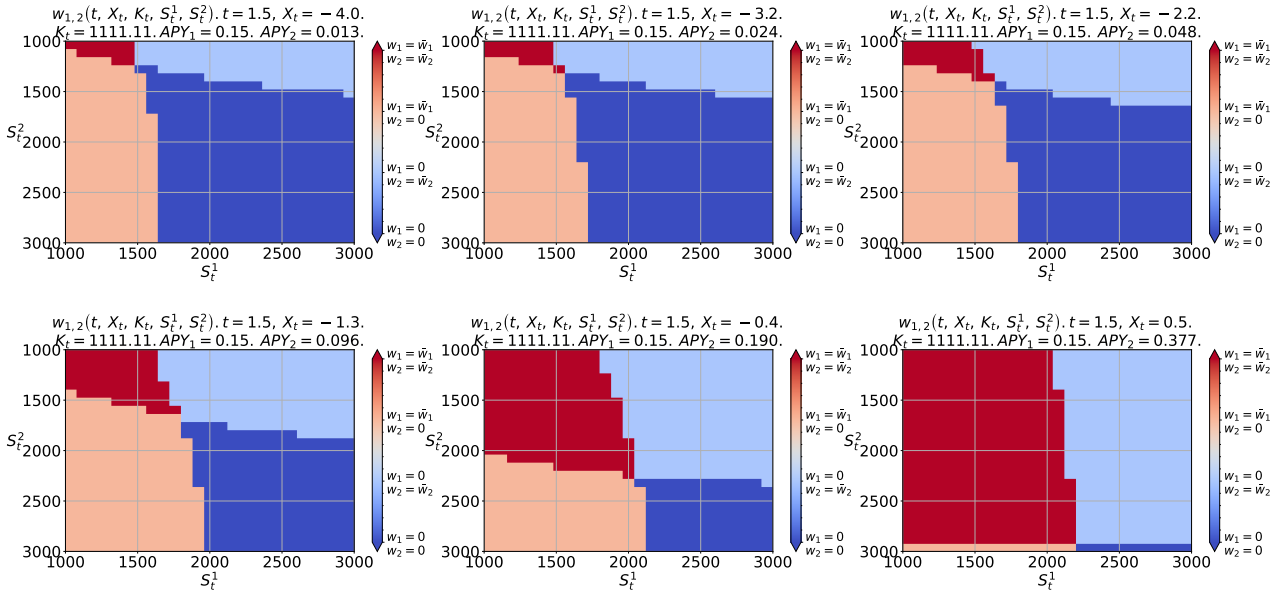

Figure III.3: The optimal recompounding strategies $\omega_{1,2}\left(t,\, X_t,\, K_t,\, S_t^1,\, S_t^2\right) \in \{0,\, \bar{\omega}_1\} \times \{0,\, \bar{\omega}_2\}$ for both of the liquidity pools LINK/WETH and WETH/CRV. The APR_2 for the second liquidity pool WETH/CRV is varying, and the $\text{APR}_1 = 0.15$ for the first liquidity pool LINK/WETH is fixed. The transaction costs are set to $c = 10$ USD. The risk tolerance is set to $\gamma_1 = 0.33$, $\gamma_2 = 0.33$, and $\gamma_3 = 0.34$. The functions of APRs for both of the liquidity pools are $a_i\left(X_t\right) = \beta_i e^{\sigma_i X_t}$, for $i = 1,\, 2$.

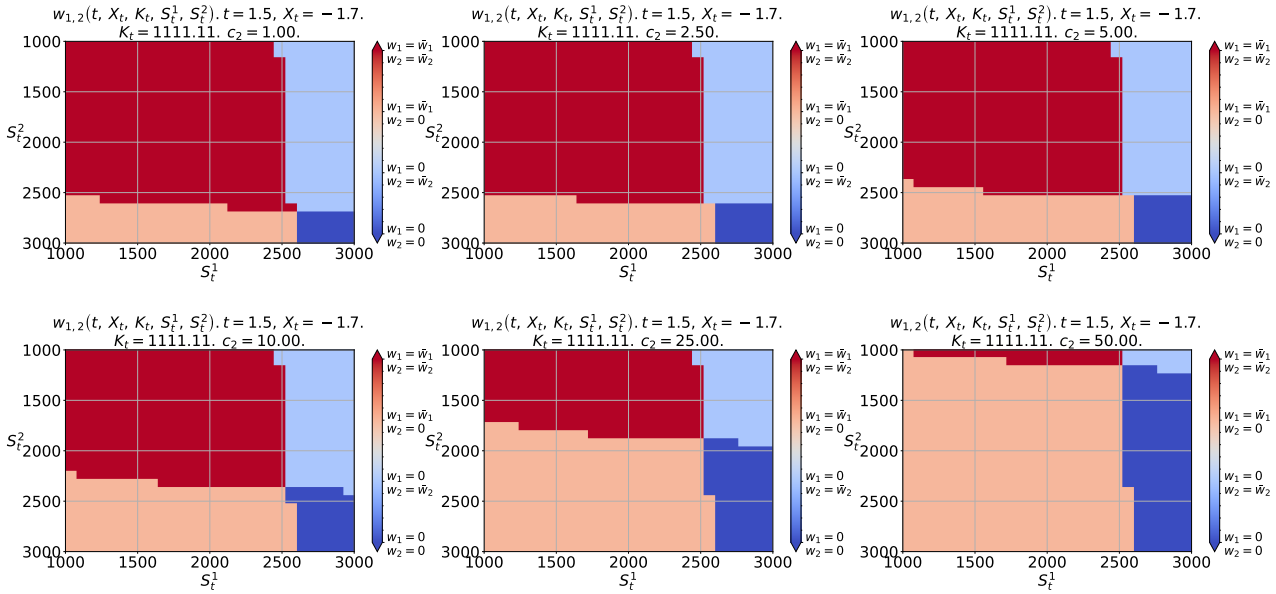

Figure III.4: The optimal recompounding strategies $\omega_{1,2}\left(t,\, X_t,\, K_t,\, S_t^1,\, S_t^2\right) \in \{0, \bar{\omega}_1\} \times \{0, \bar{\omega}_2\}$ for both of the liquidity pools LINK/WETH and WETH/CRV. The values of the transaction cost c_2 are set to 1 USD, 2.5 USD, 5 USD, 10 USD, 25 USD, and 50 USD. The transaction costs c_1 of the first liquidity pool LINK/WETH are fixed to 10 USD. The functions of APRs for both of the liquidity pools are $a_i\left(X_t\right) = \beta_i e^{\sigma_i X_t}$, for $i = 1,\, 2$.