

Multi-Currency AMMs for Decentralized FOREX Markets: Feasibility & Optimal Design

Reina Ke Xin Li ✉ 

University of Toronto, Canada

Andreas Park ✉ 

University of Toronto, Canada

Andreas Veneris ✉ 

University of Toronto, Canada

Srisht Fateh Singh ✉ 

University of Toronto, Canada

Abstract

Most currency pairs lack a direct liquid market, so international foreign exchange relies on routing transactions through a dominant vehicle currency. Multi-currency automated market makers (AMMs) offer an alternative by sharing liquidity across many currency pairs, facilitating direct cross-currency trade while exploiting liquidity consolidation. This paper studies a multi-currency pool design that minimizes trading cost. Under a constant-mean AMM architecture, equilibrium trading costs reflect the trade-off between reduced price impact from consolidated liquidity and increased impermanent loss from joint return risk. This work derives closed-form costs, characterizes optimal pool weights, and shows that the optimized multi-currency pool dominates the status quo over a range of market parameters. It then formulates the system-level problem of partitioning currencies into multi-currency pools, which is solved using a hierarchical agglomerative clustering algorithm. Empirically, using exchange rate and trade data for 43 currencies over 2008-2023, the algorithm runs in 1.6 seconds and produces pools with geographic and economic structure. Notably, this reduces realized costs by ~13% relative to the status quo of vehicle-currency routing, with gains stable through episodes of global financial stress.

2012 ACM Subject Classification Applied computing → Economics

Keywords and phrases automated market makers, decentralized finance, currency foreign exchange

Digital Object Identifier 10.4230/LIPIcs.AFT.2026.13

Related Version *Full Version*: <https://arxiv.org/abs/2607.26405> [33]

Supplementary Material *Software (Source Code)*: <https://github.com/reinali07/Multi-Currency-AMMs-FOREX>, archived at `swb:1:dir:b0a4c7db3553cfe7a4686b8ec2695047b5f2b0bb`

1 Introduction

Daily trading volume in foreign exchange markets more than doubled from \$4.0 trillion in 2010 to \$9.6 trillion in 2025 [7, 9]. Despite this growth, foreign exchange transfers follow a peculiar arrangement: most currency pairs do not trade directly with one another but route through the US dollar (USD). For instance, a firm in Australia importing goods from Sweden cannot exchange Australian dollars (AUD) for Swedish kronor (SEK) directly at a “tight” spread as direct AUD/SEK markets are thin. Instead, it must first convert the AUD to USD and then the USD to SEK, thus paying the transaction costs for both trades. The work of [10] estimates that, in 2022, direct trades between the AUD and the SEK were virtually non-existent. More broadly, 13% of foreign exchange trading volume involving the USD is generated by routing trades between non-dollar currencies through the USD [47]. This is a significant volume given that USD-involved foreign exchange transactions as a whole account for over 85% of the global foreign exchange volume.



© Reina Ke Xin Li, Andreas Park, Andreas Veneris, and Srisht Fateh Singh;
licensed under Creative Commons License CC-BY 4.0

8th Conference on Advances in Financial Technologies (AFT 2026).

Editors: Aggelos Kiayias and Maria Kyropoulou; Article No. 13; pp. 13:1–13:23

Leibniz International Proceedings in Informatics



LIPICs Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

This dependence on the USD as a *vehicle currency* persists as an emergent feature of the international monetary system. After World War II, the stable US economy with its dollar attached to the gold standard made it an attractive peg codified through the 1944 Bretton Woods Agreement. This allowed the USD to play an important role as a vehicle currency for international foreign exchange. Even after the end of the Bretton Woods era in 1971 with the Nixon shock that canceled the convertibility of USD to gold, the USD's dominance as a vehicle currency has been sustained through various economic forces. Among other reasons, when bilateral trade between two currencies is moderate or low, neither side can sustain a deep enough market to offer competitive spreads, so traders gravitate toward an intermediate currency with high liquidity. Early work by [31], in fact, makes the case that concentrating trade in a vehicle currency is more efficient than trading with multiple pairs.

Yet, the political landscape is shifting in ways that make the architecture of international payments a pressing policy question. The use of dollar-based payments systems as a political instrument through sanctions and freezing of sovereign reserves has prompted countries to seek alternatives [22, 46, 39, 19, 51]. New currency swap agreements between central banks have been negotiated and payment systems outside the SWIFT network are under active development [48, 12, 15, 23, 43]. Most importantly, technological advances and innovations in blockchain-based trading technology raise the question of whether novel currency exchange arrangements can reduce reliance on a vehicle currency but also overall transaction costs.

This paper posits that Decentralized Finance (DeFi) offers a compelling alternative. Specifically, an *automated market maker* (AMM) is a decentralized exchange protocol that holds pooled reserves of assets that are traded against at prices governed by a mathematical *invariant*. The Balancer protocol, in particular, offers a *multi-asset AMM* [40]. Such a protocol could be the backbone for direct markets between various currency pairs, while still allowing intermediation through a vehicle currency when necessary. This practice has the potential to reduce overall trading costs relative to the status quo system of routing all trades through a vehicle currency by broadening support for *direct* cross pair trading.

The BIS's Project Mariana demonstrates the technical feasibility of AMM-based trading and settlement for foreign exchange trades [8]. However, whether a multi-currency AMM arrangement can achieve economic improvements on the status quo depends on a fundamental trade-off. Liquidity providers, *i.e.*, investors who deposit funds into an AMM pool, are exposed to *impermanent loss*: when exchange rates move, arbitrageurs profit against the pool. This is especially so for pools of heterogeneous assets, each with their own price movements and risk. An AMM pool planner must design fees, exposure weights, and currency inclusion to balance this risk against the liquidity benefits of consolidation. At the same time, pooling currencies with volatile or negatively correlated returns amplifies impermanent loss and raise the fee(s) required to sustain the pool, undermining the gains from deeper liquidity.

Resolving these trade-offs is the central contribution of this paper. *First*, we extend to multi-currency AMMs a mathematical framework in which equilibrium trading costs – the costs under optimally designed pool parameters assuming competitive liquidity provision – are a function of trading volumes, exchange rate covariances, and pool composition. *Second*, we show the market conditions under which an optimally designed multi-currency AMM dominates the status quo and study optimal pool weights in closed form for these conditions. We further develop a tractable approximation for the general case. *Finally*, we examine the system-level problem of which currencies to pool together using a *hierarchical agglomerative clustering* (HAC) algorithm on a correlation-based distance metric. Empirically, applying the framework to 43 non-pegged currencies over 2008–2023, the HAC algorithm runs in 1.6 seconds and produces pools that demonstrate meaningful *regional* and *economic* structure. The resulting architecture reduces realized aggregate costs by ~13% – USD ~2.5 billion – relative to the status quo, with gains remaining stable through episodes of financial stress.

The paper proceeds as follows. Section 2 introduces the general AMM framework. Section 3 contextualizes this setup. Section 4 studies benchmark exchange arrangements. Section 5 analyzes equilibrium costs of multi-currency AMM pools. Section 6 derives analytical weight optimization results for parametrized symmetric currencies. Section 7 studies the general weight optimization problem and develops a closed-form approximation. Section 8 formulates the currency partitioning problem, describes the HAC algorithm, and presents an empirical evaluation. Section 9 reviews the literature. Section 10 concludes and discusses further directions.

2 Preliminaries: General AMM Framework

Consider an AMM for $N + 1$ currencies indexed by $i \in \{0, 1, \dots, N\}$, funded by *liquidity providers* who deposit reserves $R_i \geq 0$ of each currency. The AMM is governed by an *invariant function* of the reserves that implicitly determines the prices at which the AMM trades. When a trader exchanges asset i for asset j by submitting $\Delta_i > 0$ units, the outgoing quantity Δ_j is uniquely determined by the requirement that the invariant is preserved.

Throughout, we use the *constant mean* invariant introduced by Balancer [40], $\prod_{i=0}^N R_i^{w_i} = k$, for a fixed weight vector $(w_0, \dots, w_N)$ with $w_i \geq 0$ and $\sum_i w_i = 1$. The weight w_i governs pool exposure to currency i : at fundamental prices, currency i accounts for fraction w_i of total pool value. Thus, the 2-asset constant-product AMM ($R_0 R_1 = k$) is the special case used by Uniswap V2, among other popular AMMs [1].

2.1 Prices, Fees, and Trading Costs

The AMM-implied marginal price of asset i in terms of asset j is, for the constant-mean AMM, $p_{ij} = (w_i/w_j)(R_j/R_i)$ [40]. Competitive arbitrageurs continuously restore the pool's marginal prices to the exogenous exchange rates S_i (with numeraire $S_0 \equiv 1$), so that $p_{ij} = S_i/S_j$ in equilibrium. For the constant-mean AMM this requires $S_i R_i/w_i = S_j R_j/w_j$ for all pairs.

A finite trade of size Δ in numeraire moves the reserve vector along the invariant surface, causing the effective price $S^{\text{eff}}(\Delta)$ paid by the trader to deviate from the pre-trade spot price S . The *price impact* at spot price S is

$$\Pi(\Delta) = \frac{S^{\text{eff}}(\Delta) - S}{S}, \quad (1)$$

which is strictly positive and increasing in Δ for the constant-mean invariant. Traders additionally pay an explicit fee $f \geq 0$ proportional to transaction value,¹ so the total *unit trading cost* is $c(\Delta, f) = \Pi(\Delta) + f$. Denoting by c_{ij} the unit cost for pair i, j and Q_{ij} the total trading volume on pair (i, j) (assuming a fixed trade size Δ and balanced flows in either direction), the aggregate cost for pair (i, j) over the period is approximately $c_{ij}(\Delta, f) = (\Pi(\Delta) + f)Q_{ij}$.

2.2 Impermanent Loss

The pool's value in the numeraire is $V = \sum_i S_i R_i$, held collectively by liquidity providers. When external prices change, arbitrageurs trade against the pool to restore price alignment. Arbitrageurs are incentivized by the favourable terms offered by the AMM: the AMM sells the

¹ For simplicity, we assume that fees are paid to liquidity providers directly, not added back to the pool.

asset below its new fundamental value or buys above it. The profit earned by the arbitrageur is at the expense of the pool’s liquidity providers. This gives rise to *impermanent loss* (IPL). Consider an initial deposit $V(0)$ and a period $[0, T]$ in which currency i ’s has gross return $x_i = \frac{S_i(T)}{S_i(0)}$ (with $x_0 \equiv 1$). After arbitrageurs restore price alignment, the pool value is $V(T)$, while the counterfactual hold value of the initial deposit is $V_{\text{hold}}(T) = \sum_i x_i S_i(0) R_i(0)$. The IPL is defined as²

$$\text{IPL} = \frac{V(T) - V_{\text{hold}}(T)}{V(0)} \leq 0. \quad (2)$$

The loss is described as “impermanent” in DeFi settings because it is unrealized if prices revert before the provider withdraws. In the context of this work, the loss is due to changes in the fundamental.

2.3 Equilibrium Liquidity Supply and Optimal Fees

We follow the framework of [36] and model equilibrium competitive liquidity provision: risk-neutral providers deposit at the start of a period and withdraw at the end, earning fee revenue while bearing impermanent loss. To simplify the exposition, we assume that there is sufficient exogenous, uninformed trading relative to the order flow of informed arbitrageurs so that fees accrue on noise volume while IPL stems from price changes driven by informed flow. Competitive liquidity providers contribute capital until they *break-even*:

$$E[|\text{IPL}|] \cdot V(0) = E[Q] \cdot f, \quad Q = \sum_{i < j} Q_{ij}. \quad (3)$$

For a given fee rate f and return and volume distributions, (3) pins down the equilibrium pool depth $V(0)$ and hence the price impact and unit trading cost $c(\Delta, f)$. The pool designer chooses the fee that minimizes unit trading costs, $f^* = \arg \min_f c(\Delta, f)$; this fee maximizes trader welfare and ensures that no competing pool can charge better rates.

3 Overview

At a high level, our goal is to design an optimal foreign exchange mechanism and ask: *do multi-currency pools lower costs or does the additional risk from holding many currencies outweigh the benefits of pooling?* Our benchmarks are bilateral AMMs as proxies for the cost of vehicle routing. Although at first sight these AMMs look different to traditional markets, they are economically equivalent to a limit order book and as [36] show, bilateral AMMs are a low cost benchmark for existing markets. We use the framework in Section 2 to isolate the key forces determining costs. The AMM invariant determines how trades move pool prices, giving rise to price impact – a cost that diminishes with increased liquidity. Liquidity providers supply this liquidity, but exchange rate volatility exposes them to impermanent loss, so they require trading fees as compensation. A higher fee attracts more liquidity but is charged to the trader. A pool planner therefore sets a fee that balances these forces, yielding optimal equilibrium pool depth and minimized trading cost.

The following builds on this framework by studying how pools should be designed once these forces are in place, moving from simple benchmarks to richer design problems. Constant-mean pools introduce weights as an additional design parameter that control the pool’s

² IPL is also sometimes defined in other literature as $(V(T) - V_{\text{hold}}(T))/V_{\text{hold}}(T)$.

exposure to each currency's risk and volume. At the system level, design also involves deciding which currencies to pool together. We ultimately study the pool weights and membership structures that minimize equilibrium trading costs across different market environments.

4 Benchmark Currency Exchange Architectures

We begin by specializing to a minimal three-currency environment consisting of currencies $\{0, 1, 2\}$, with currency 0 as the numeraire, to establish benchmarks, and then generalize the architectures to $N + 1$ currencies. Following the work in [35], we consider three arrangements. In the *status quo*, trades between currencies 1 and 2 are routed through the vehicle currency 0 via two separate bilateral constant-product pools; each $(0, i)$ pool handles total volume $Q_i = \sum_{j \neq i} Q_{ij}$, and a *cross-pair* trader (trading currencies 1 and 2) bears costs in both. In the *dedicated bilateral pools* architecture, each pair has its own pool. In the *equal-weighted constant mean pool*, all three currencies are held in a constant mean pool with $w_0 = w_1 = w_2 = 1/3$. By [35], the fee-optimized equilibrium unit costs for trades between $i \leftrightarrow j$ of size Δ in numeraire under the three arrangements are, respectively,

$$c_{ij}^{\text{SQ}} = \sqrt{\frac{\sigma_i^2 \Delta}{E[Q_i]}} + \sqrt{\frac{\sigma_j^2 \Delta}{E[Q_j]}}, \quad c_{ij}^{\text{BP}} = \sqrt{\frac{\sigma_{ij}^2 \Delta}{E[Q_{ij}]}} \quad c_{ij}^{\text{SYM}} = \sqrt{\frac{2H\Delta}{3E[Q]}}$$

where $\sigma_i^2 = \text{Var}(x_i)$, $\sigma_{ij}^2 = \text{Var}(x_i - x_j)$ (we call σ the relative volatility of i and j), $H = \sum_{i < j} \sigma_{ij}^2$ and $Q = \sum_{i < j} Q_{ij}$.³

In each case, costs scale with volatility and shrink with volume. The status quo charges two costs for a $i \leftrightarrow j$ trade, each discounted by its own currency's total volume. The bilateral pool charges a single cost but only gets liquidity from the i, j pair. In the equal-weighted AMM all pairs share the same cost discounted by aggregate volume $E[Q]$, so more actively traded pairs effectively subsidize thinner-traded ones.

The cost c^{BP} generalizes straightforwardly to $N + 1$ currencies, with aggregate cost $c^{\text{BP}} = \sum_{i < j} \sqrt{\sigma_{ij}^2 E[Q_{ij}] \Delta}$. We extend the results for c^{SQ} and c^{SYM} to $N + 1$ currencies in the following propositions, both of which are proved in Appendix A.1.

► **Proposition 4.1** (Status quo trading cost). *The expected aggregate trading cost in the status quo for $N + 1$ currencies routed through the vehicle currency 0 is*

$$c^{\text{SQ}} = \sum_i \sqrt{\sigma_i^2 E[Q_i] \Delta}$$

► **Proposition 4.2** (Equal weights trading cost). *The expected aggregate trading cost in an equal-weighted $N + 1$ currency constant-mean pool is*

$$c^{\text{SYM}} = \sqrt{\frac{2\Delta H E[Q]}{N + 1}}$$

In each case, the aggregate costs scale with exchange rate volatilities, but c^{SQ} charges volatility per currency, as each currency routes trades through the vehicle separately, while c^{BP} and c^{SYM} charge per pair, reflecting the direct cross-pair trading. The dominant pool depends on the specific volatility and volume structure, which is explored further in Section 6.

³ [35] derives the costs in expectation using second-order approximations for IPL under the assumption of small, zero-mean excess returns $\epsilon_i = x_i - 1 \approx 0$. We use the same simplifying assumption throughout.

5 Equilibrium in Constant-Mean AMMs

Equal weights are not optimal in general: a currency with high individual volatility inflicts more impermanent loss per weight, while one with high trading volume generates more fee revenue per weight. Introducing non-uniform weights allows the pool designer to tilt exposure away from high-volatility, low-volume currencies, reducing equilibrium trading costs.

The break-even framework of Section 2.3, developed for two-asset constant-product pools in [36], extends to general constant-mean pools. The lemmas below derive the corresponding price impact and impermanent loss expressions; applied to the break-even condition (3) and cost function $c(\Delta, f)$, they yield the equilibrium trading cost in Proposition 5.3. All proofs for this section are found in Appendix A.2.

► **Lemma 5.1** (Price impact). *In the general constant-mean AMM, the price impact of a trade of size Δ in numeraire between currencies i and j is approximately*

$$\Pi(\Delta) \approx \frac{w_i + w_j}{w_i w_j} \cdot \frac{\Delta}{2V}$$

► **Lemma 5.2** (Impermanent loss). *For gross returns x_i with $x_i \approx 1$ and $E[x_i] = 1$, the expected impermanent loss of the general constant-mean pool satisfies*

$$E[|IPL|] \approx \frac{H_w}{2}, \quad H_w = \sum_{i < j} w_i w_j \sigma_{ij}^2$$

Each currency pair (i, j) contributes to H_w in proportion to the product of the two currencies' weights and the variance of their relative returns. Concentrating weight on a pair with high relative volatility σ_{ij}^2 increases expected impermanent loss and therefore the fee needed to sustain liquidity provision in equilibrium. Together with Lemma 5.1, this gives us the following proposition.

► **Proposition 5.3** (Equilibrium trading cost). *Under the break-even condition (3) and Lemmas 5.1–5.2, the aggregate equilibrium trading cost is minimized at*

$$f^* = \sqrt{\frac{\Delta H_w}{4(E[Q])^2} \sum_i \frac{E[Q_i]}{w_i}}, \quad c^* = \sqrt{\Delta H_w \sum_i \frac{E[Q_i]}{w_i}} \quad (4)$$

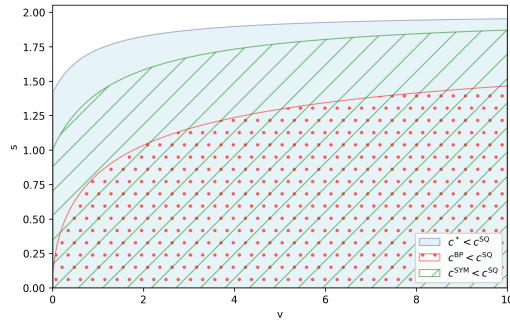
The cost factors cleanly into two terms: H_w , which is proportional to expected impermanent loss, and $\sum_i E[Q_i]/w_i$, which is proportional to the volume-weighted price impact per unit of volume. The pool designer's goal is to choose weights to minimize this cost.

6 Optimal Pools Under Symmetric Multi-Currency Parametrizations

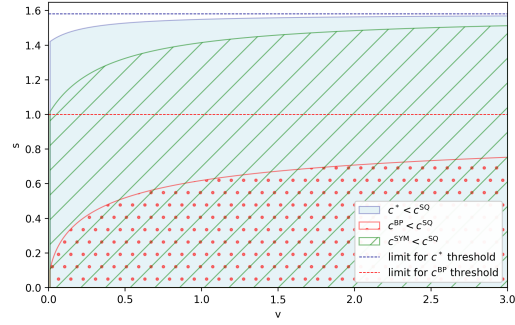
We begin by deriving closed-form optimal weights and cost comparisons for special symmetric cases, where the general problem of Section 5 reduces to a scalar optimization. The three-currency case extends the analysis of [35] by characterizing optimal weights, while the $N + 1$ -currency case is, to our knowledge, new.

6.1 Three Currency Case

We first study a parametrization that captures the structure of a three-currency system organized around a vehicle currency. Let currency 0 be the vehicle. Suppose that the $(0, 1)$ and $(0, 2)$ pairs have equal trading volumes and equal relative volatilities, while the



■ **Figure 1** s vs. v regions for optimality over the status quo under the three-currency (v, s) parametrization.



■ **Figure 2** s vs. v regions for optimality over the status quo under the (v, s) parametrization for $N = 5$ currencies.

$(1, 2)$ cross pair trades at a smaller volume with higher relative volatility. That is, let $E[Q_{01}] = E[Q_{02}] = q$ and $\sigma_{01}^2 = \sigma_{02}^2 = \sigma^2$. Write $E[Q_{12}] = vq$ for some $v > 0$ and $\sigma_{12}^2 = s^2\sigma^2$ for $s \geq 0$.⁴ Values $s > \sqrt{2}$ correspond to negatively correlated returns for currencies 1 and 2, while $s < \sqrt{2}$ corresponds to positively correlated returns. The case $v < 1$ represents the typical vehicle-currency scenario in which direct cross-currency trading is thin; $v > 1$ could represent two highly liquid currencies and one smaller currency trading against the two.

Under this parametrization, currencies 1 and 2 are symmetric, so the cost-minimizing weights satisfy $w_1 = w_2 = (1 - w_0)/2$, and the problem reduces to optimizing over the scalar $w_0 \in (0, 1)$. The result is summarized in the following proposition which is proved in Appendix A.3.

► **Proposition 6.1** (Optimal weight and cost for vehicle and 2 symmetric currencies). *Under the symmetric (v, s) parametrization for 3 currencies, if $s^2 < 2(1 + 2v)/(1 + v)$, the expected aggregate trading cost of Equation (4) has unique interior minimizer at $w_1^* = w_2^* = \frac{1 - w_0^*}{2}$ and*

$$w_0^* = \frac{s}{\sqrt{(1 + 2v)(4 - s^2)}}, \quad c^* = \sqrt{q\sigma^2\Delta} \sqrt{s^2v + 2 + s\sqrt{(4 - s^2)(1 + 2v)}}$$

If $s^2 \geq 2(1 + 2v)/(1 + v)$, no interior minimizer exists; the infimum over $w_1, w_2 > 0$ is obtained as $w_0 \rightarrow 1$.

The optimal vehicle-currency weight w_0^* increases with s : when the $(1, 2)$ pair is volatile, the pool tilts toward the vehicle to lower IPL generated by the cross pair. Conversely, w_0^* decreases with v (the relative volume of the non-vehicle pair): when the $(1, 2)$ pair has high volume, the pool should provide more depth to the cross pair and reduce its price impact.

We plot the regions in which c^* , c^{SYM} , and c^{BP} are optimal over the status quo cost, c^{SQ} in Figure 1. The precise expressions for the thresholds are found in Appendix B.

6.2 Extension to More Currencies

The above can be generalized to $N \geq 2$ symmetric currencies organized around a vehicle currency. Again, let currency 0 be the vehicle and suppose that $(0, i)$ pairs have equal trading volumes and volatilities, q and σ , while each pair (i, j) with $i, j \neq 0$ trading at volume vq with

⁴ Since $\sigma_{12}^2 = \text{Var}(\epsilon_1 - \epsilon_2) = \sigma_{01}^2 + \sigma_{02}^2 - 2\text{Cov}(\epsilon_1, \epsilon_2)$, the Cauchy-Schwarz inequality implies $0 \leq \sigma_{12}^2 \leq 4\sigma^2$ and $s \in [0, 2]$.

volatility of returns $s\sigma$. Under this parametrization, the weights satisfy $w_i = (1 - w_0)/N$, and the problem reduces to optimizing over $w_0 \in [0, 1]$, summarized in the following proposition and proved in Appendix A.3.

► **Proposition 6.2** (Optimum for vehicle and N symmetric currencies). *Under the symmetric (v, s) parametrization for $N + 1$ currencies, if $s^2 < 2(1 + Nv)/(1 + (N - 1)v)$, the expected aggregate trading cost from Equation (4) has unique interior minimizer at $w_i^* = (1 - w_0^*)/N$ for all $i > 0$ and*

$$w_0^* = \frac{s}{\sqrt{(1 + Nv)(2N - (N - 1)s^2)}}$$

If $s^2 \geq 2(1 + Nv)/(1 + (N - 1)v)$, no interior minimizer exists; the infimum over $w_1, w_2 > 0$ is obtained as $w_0 \rightarrow 1$.

The expression for the optimal aggregate expected cost c^* can be found as Equation (7) in Appendix A.3. The optimal weights of the three-currency case are recovered by setting $N = 2$, and similarly, we observe the increasing optimal vehicle weight with s .

We plot, for the $N + 1$ currencies case, the regions for which c^* , c^{SYM} , and c^{BP} are optimal over the status quo cost, c^{SQ} in Figures 2. The precise expressions for the thresholds are found in the Appendix B.

7 General Constant Mean AMM Weight Optimization

For a general distribution of volumes and volatilities of a given pool of currencies, the pool designer solves for the weights $w_i > 0$ summing to 1 and minimizing $\mathcal{F}(w) = H_w \sum_i E[Q_i]/w_i$. Closed-form solutions are not available in general, but the system can be solved numerically via constrained optimization methods.

A tractable heuristic approximation is obtained by noting that, in the objective $\mathcal{F}(w) = H_w \cdot \sum_i E[Q_i]/w_i$, the first factor captures the impermanent loss exposure of the pool and the second captures the volume-weighted price impact. The two factors interact through the weights, with the first biasing towards currencies with small relative volatilities and the second biasing towards currencies with high trading volumes. A useful simplification captures these forces and decouples the weights: we approximate the minimization objective by the heuristic $\tilde{\mathcal{F}}(w) = \sum_i \frac{E[Q_i]}{w_i H_i}$ where $H_i = \sum_{j \neq i} \sigma_{ij}^2$. The heuristic-approximated optimum is presented in the following proposition, proved in Appendix A.4.

► **Proposition 7.1** (Approximate optimal weights). *When $H_i > 0$ for all i and $E[Q_i] > 0$ for some i , the minimizer of $\tilde{\mathcal{F}}(w) = \sum_i \frac{E[Q_i]}{w_i H_i}$ subject to $\sum_i w_i = 1$, $w_i > 0 \forall i$, is*

$$w_i^* = \frac{1}{D} \sqrt{\frac{E[Q_i]}{H_i}}, \quad \forall i = 0, \dots, N, \quad \text{where } D = \sum_j \sqrt{E[Q_j]/H_j},$$

This heuristic rule assigns each currency a weight proportional to the square root of its expected aggregate trading volume $E[Q_i]$ and the inverse of the sum of its relative variances. The square-root scaling arises from the diminishing effects of concentrating weight on a single currency in reducing price impact while leaving the impermanent loss exposure unimproved. Hence, the optimum disperses weight broadly, with the degree of dispersion governed by the square-root of relative volumes and volatilities.

7.1 Numerical Validation

We evaluate the quality of the approximation $\min \tilde{\mathcal{F}}$ by comparing the equilibrium trading cost it achieves against the cost attained by the numerical minimizer of $\mathcal{F}$, across a wide range of pool sizes and randomly drawn market environments.

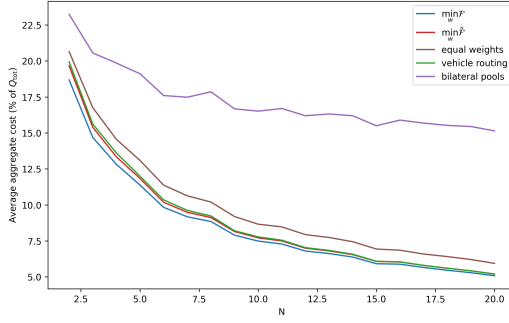
Methodology. For each pool of $N \in \{2, 3, \dots, 20\}$ non-vehicle currencies (pooled together with the vehicle), we conduct 500 independently drawn trials, proceeding as follows. To generate covariance matrices, we use a synthetic 4-factor model [13], where returns are correlated through shared factors, with loadings $B \in \mathbb{R}^{N \times 4}$, a matrix of standard normal entries. The covariance matrix K is constructed as $K = BB^T + \Psi$, where Ψ is a diagonal matrix with entries uniformly chosen from $[0, 1]$, representing idiosyncratic risk. An additional column and row of zeros is added for the 0-return vehicle currency. The pairwise relative variance matrix $\sigma^2 \in \mathbb{R}^{(N+1) \times (N+1)}$ is then computed with entries $\sigma_{ij}^2 = K_{ii} + K_{jj} - 2K_{ij}$. Expected trading volumes are drawn as $E[Q_{ij}] = q_i q_j$ with $q_i \sim \text{LogNormal}(0, 1)$, yielding a symmetric volume matrix with log-normally distributed marginals. $E[Q_i]$ for all i and $E[Q]$ are computed from this matrix. We compute three weight vectors: $w^{\mathcal{F}}$, the numerical minimizer of $\mathcal{F}(w) = H_w \sum_i E[Q_i]/w_i$ using sequential quadratic programming (SLSQP) [30]; $w_i^{\mathcal{F}} = D^{-1} \sqrt{E[Q_i]/H_i}$, the approximation from Proposition 7.1; and $w_i^{\text{EQ}} = 1/(N+1)$, the equal weights vector. The aggregate expected cost of Equation (4) is evaluated for each weight vector with trade size $\Delta = 1$. We also compute the costs of the status quo vehicle routing and the dedicated bilateral pools architectures for comparison.

Results. Figures 3 and 4 report the mean aggregate expected cost, expressed as a percentage of total volume, and its coefficient of variation over the trials as a function of pool size N . The two methods achieve nearly identical costs for all N , with the gap remaining small and roughly constant as N grows from 3 to 20. This implies that ignoring the interaction between H_w and $\sum_i E[Q_i]/w_i$ when optimizing weights is negligible in practice. The coefficient of variation tracks closely across both methods for all N , indicating that variability reflects heterogeneity of the market environments rather than any degradation of the approximation. A pool designer can thus use the closed-form rule $w_i^* \propto \sqrt{E[Q_i]/H_i}$ in place of a much more computationally demanding numerical optimizer. This is practical for AMM weight optimization in computationally restricted applications such as in blockchains, or for efficiently estimating costs in the optimization algorithm described in the next section.

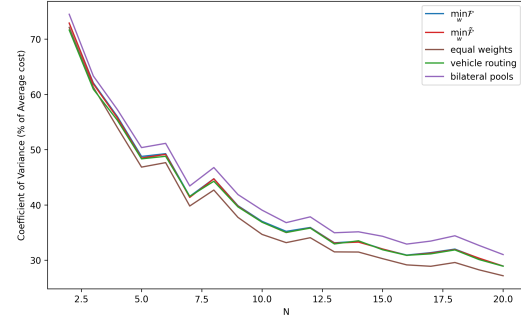
8 Currency Partitioning for System-Level Pool Design

In this section, we address the question of *which* currencies to pool together. Evidently, not every currency benefits from co-pooling: if the returns of two currencies have high relative variance σ_{ij}^2 , their impermanent loss raises the fee, which may outweigh liquidity gains. We therefore introduce the *currency partitioning problem*: given N currencies and their pairwise volumes and return covariance structure, partition the currencies into multi-currency pools and unpooled singletons to minimize aggregate trading costs.

We fix currency 0 as the vehicle and consider multi-currency pools of the form $I = \{0\} \cup G$ where $G \subseteq \{1, \dots, N\}$. Let $M = \{I_1, \dots, I_K, O\}$ be a collection of such pools (that are disjoint except for the vehicle) and the set O of unpooled currencies. A trade between a non-vehicle i in a pool I and $j \notin I$ is executed in two legs: $i \leftrightarrow 0$ in the pool, and $0 \leftrightarrow j$ in the pool containing j or a bilateral pool with the vehicle if $j \in O$. Trades between unpooled currencies are also routed bilaterally through the vehicle.



■ **Figure 3** Mean equilibrium aggregate trading cost (as a percentage of total volume) achieved by the exact minimizer of $\mathcal{F}$ and the closed-form approximate minimizer of $\tilde{\mathcal{F}}$, averaged over 500 trials per pool size.



■ **Figure 4** Coefficient of variation (standard deviation / mean) of the equilibrium aggregate cost across trials, as a function of pool size N , for the exact and approximate minimizers.

Let $c(M)$ denote the total expected system cost; the currency partitioning problem is to choose M minimizing $c(M)$. The expression for $c(M)$ is calculated in Appendix C.

8.1 Rolling Window Realized Cost

Our empirical evaluation separates the *ex-ante* and *ex-post* quantities. Liquidity providers set the pool depth V and fee f before observing the period's realized trading volume and price movements. They base these decisions on expectations formed from recent data.

We implement this separation as follows. For each quarter t , we form the *expected* covariance matrix using a 12-month rolling window of monthly returns ending at the start of quarter t , scaled to a quarterly frequency. We form the *expected* volume matrices using the same 12-month windows, scaled to quarterly averages. The pool planner uses these expected values to compute the break-even fee f^* via Proposition 5.3 and the optimal weights w^* numerically by minimizing $\mathcal{F}(w)$ at the beginning of each quarter t . Liquidity providers then decide $V = V^*$, the pool depth.

The *realized* cost in quarter t is then computed by applying the unit cost implied by the ex-ante pool parameters (w^*, f^*, V^*) to the *realized* trade volume for quarter t , which is then summed for every quarter over the evaluation period. The expression for the realized cost is also found in Appendix C. This specification captures an additional risk faced by liquidity providers: the fee, weights, and depth are fixed at the start of the quarter, but the realized cost depends on how much volume actually flows through the pool.

8.2 Hierarchical Agglomerative Clustering Algorithm

The number of partitions of N items grows super-exponentially, making exhaustive search infeasible. Instead, we exploit the cost function structure to inform heuristic methods. Since a currency inside a multi-asset pool executes all its trades inside the pool, the main determinant of pooling efficiency is the pairwise relative variance of currencies within a pool. Currencies with highly correlated returns have lower σ_{ij}^2 and thus generate lower impermanent loss. Hence, we use a *hierarchical agglomerative clustering* (HAC) algorithm on the correlations [37].

First, we transform correlations into distances $d_{ij} = \sqrt{0.5(1 - \rho_{ij})} \in [0, 1]$ so that highly correlated currencies are close together. We then construct a partition over the set of currencies as follows. Starting from singletons (equivalent to the status quo), we iteratively

merge the two clusters with smallest distance. Here, the distance between two clusters A and B is defined as $d_{AB} = \frac{1}{|A||B|} \sum_{i \in A, j \in B} d_{ij}$, the mean pairwise distance between the currencies across the two clusters. We require $d_{AB} \leq \tau$, a distance threshold. If all clusters are farther than τ from each other, the procedure ends and returns the current partition.

We evaluate the partition produced by a range of thresholds $\tau \in [0, 1]$ against the expected aggregate cost $c(M)$ (evaluated using approximate weights from Proposition 7.1 for the sake of efficiency) and select the threshold τ and resulting partition that minimizes this cost.

8.3 Empirical Data

Exchange rates come from the IMF Exchange Rates dataset (monthly period-average *US Dollar per domestic currency* indicator) [28], from which we compute monthly returns and relative variances. Trade volumes are proxied by the USD value (in millions) of bilateral goods exports between the countries issuing the currency. This approximates the real demand for cross-border currency exchange. This data is obtained from the IMF International Trade in Goods dataset (monthly *Export of Goods, FOB, US dollar* indicator) [29].⁵ Currency partitioning with the HAC algorithm uses 2002–2007 monthly exchange rates and trade data to calculate correlations, relative variances, and trade volumes over the whole period (scaled to a quarterly basis); out-of-sample 12-month quarterly rolling window relative variances and trade volumes are calculated for the evaluation period, 2008–2023. We choose these windows for several reasons: first, 2002 is the starting year as this is when the Euro was officially adopted; second, we avoid using data from the 2008 financial crisis in the pool selection, but do not avoid evaluating on this crisis period; third, 2023 is the end year as it is the last year for which IMF AREAER exchange rate regime classifications are available.

The last criteria is important as we restrict to non-pegged currencies under the IMF AREAER [27] – currencies classified as independently floating, managed floating, crawling band, crawling peg (pre-2008 classifications), other managed, free floating, floating, crawl-like, crawling peg, or stabilized (post-2008 classifications) over 2002–2023. Pegged currencies are excluded because their near-zero volatility against the peg makes them misleading. For example, pooling USD/EUR/DKK/BGN, where the Danish krone (DKK) and the Bulgarian lev (BGN) are pegged to the Euro, results in a realized cost of $c(I) = 840$ USD millions, versus $c^{\text{SQ}}(I) = 1,400$ USD millions, a 40% cost saving over 2008–2023. However, this is a flawed analysis – central banks managing pegged currencies tend to provide guaranteed exchange rates against the peg, rather than using market supply and demand. We further exclude currencies with observed zero returns against the USD for three consecutive months, or that are described as pegged by the IMF AREAER,⁶ as well as the Haitian gourde (co-circulates with USD⁷) and the Somali shilling (limited regime data). The resulting 43 currencies are listed in Table 1, with the USD as vehicle (currency 0). For the remainder of this section, we fix $\Delta = 1$ USD million throughout.

8.4 Empirical Results

The HAC algorithm, implemented in Python 3.13 with Numpy 2.2.1, Pandas 2.2.3, and SciPy 1.14.1 runs in 1.6 seconds on a 16-core Intel Core Ultra 7 155H processor and 16 GB RAM. Table 2 shows the pool architecture selected by the HAC algorithm, and Table 3

⁵ We use the Euro Area (Eurozone) in lieu of a country for the Euro.

⁶ Afghani afghani, Cambodian riel, Dominican peso, Gambian dalasi, Georgian lari, Kyrgyz som, Laotian kip, Liberian dollar, Malagasy ariary, Moldovan leu, Myanmar kyat, and Papua New Guinean kina.

⁷ IBAN lists USD as a currency of Haiti.

13:12 Multi-Currency AMMs for Decentralized FOREX Markets

■ **Table 1** Currencies with non-pegged exchange rates for all years in 2002–2023 according to IMF AREAER classification (43 currencies).

Currency code
ALL, AMD, AUD, BIF, BRL, CAD, CDF, CHF, CLP, COP, CZK, DZD, EUR, GBP, GHS, GTQ, IDR, INR, ISK, JMD, JPY, KES, KRW, MUR, MXN, MZN, NIO, NOK, NZD, PEN, PHP, PLN, PYG, RSD, SEK, SGD, THB, TRY, TZS, UGX, UYU, ZAR, ZMW

■ **Table 2** Pool architecture from HAC algorithm. USD included implicitly in all pools.

Pool	Currencies	Description
1	ALL, CHF, CZK, EUR, GBP, JPY, NOK, PLN, RSD, SEK	High liquidity + Europe
2	AUD, ISK, NZD	Carry trade targets
3	COP, TRY	Macro risk
4	KRW, SGD, THB	East & South-East Asian

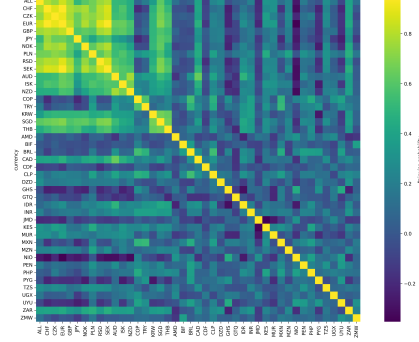
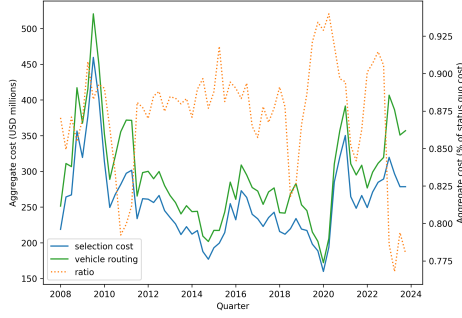
reports out-of-sample realized aggregate costs over 2008–2023. The optimized architecture, with distance threshold $\tau = 0.46$, places 18 of the 43 currencies into four multi-asset pools, reducing realized aggregate system costs by approximately 13% (roughly USD 2.5 billion) relative to bilateral USD routing. Some quarters exhibit realized savings exceeding 15-20%, and performance persists and remains stable even through the 2008 financial crisis and the 2020 COVID-19 recession, indicating that the selected pool structures are persistent rather than artifacts of the calibration window.⁸

Patterns emerge from the resulting pools. Pool 1 includes currencies with high foreign exchange turnover often considered “safe-havens” (EUR, JPY, GBP, CHF) and other European currencies, reflecting the institutional and trade integration of the European economic area and its close partners [9, 45]. Pool 2 groups commodity-linked AUD and NZD with ISK, all three of which are carry-trade target currencies [14, 16]. Pool 3 pairs the Colombian peso and Turkish lira, both emerging markets exposed to high political risks [50]. Pool 4 clusters KRW, SGD, and THB, East and South-East Asian currencies of manufacturing-heavy economies and regional trade integration [49]. This confirms that the trade-off at the heart of this paper – liquidity consolidation versus impermanent loss – can be resolved in practice by grouping currencies that co-move, capturing the benefits of shared liquidity without amplifying exchange rate risk. The emergent geographical and economic structures are a direct consequence of this co-movement pooling. Figure 6 visualizes the resulting pools in a heatmap of pairwise correlations, ordered by pool assignment. The heatmap shows the three high-correlation blocks along the diagonal corresponding the Pools 1-4, with the large Pool 1 block most pronounced on the top-right. Overall, the structured results and 13% empirical cost reduction serve to validate that well-designed multi-currency pools can meaningfully lower foreign exchange trading costs relative to bilateral vehicle-currency routing.

⁸ The HAC algorithm is not theoretically cost-optimizing, but we compare it with a simple greedy algorithm in Appendix D to benchmark the HAC algorithm’s performance.

■ **Table 3** Out-of-sample realized aggregate trading costs, 2008–2023 in millions USD.

Architecture	$c(M)$	c^{SQ}	Savings	Savings (%)
HAC algorithm pool selection	16,450	18,930	2,500	13%



■ **Figure 5** Quarterly realized cost of Table 2 ■ **Figure 6** Heatmap of correlations with current-architecture as a percentage of USD-routing cost. currencies ordered by HAC-optimal pool assignment.

9 Related Work

Developments on AMMs have accelerated since the constant-product invariant was introduced by Uniswap V2 [1]. Subsequent protocols explored alternative invariants and liquidity mechanisms: Balancer generalized to the constant mean invariant, Curve developed a hybrid invariant optimized for low-volatility assets, and Uniswap V3 introduced concentrated liquidity to improve capital efficiency [40, 20, 2]. Our work builds on the constant-mean framework and studies AMMs through the lens of pool formation and weight optimization.

Meanwhile, theoretical literature studies the economics of AMMs. The authors of [3, 4] show how arbitrage aligns prices with external markets and concave payoff functions can be replicated by suitable invariants, while [5] studies optimal routing across multi-asset AMMs. Another major focus is liquidity provider losses induced by arbitrage and price volatility, quantified by [44, 42]. Empirical and strategic analysis examine realized liquidity provider outcomes and adoption limits [26, 41, 11], optimal and equilibrium liquidity provision [52, 32, 6], and endogenous fee setting [21, 36] in 2-asset constant product pools. Closer to our setting, the work of [34, 24] compare FX efficiency in AMMs under different invariants and levels of liquidity fragmentation. We add to this literature by embedding closed-form expressions for AMM prices and impermanent loss of multi-asset constant-mean pools into the equilibrium model of competitive liquidity provision from [36] in a foreign exchange setting.

Our setting also relates to the literature on vehicle currencies in foreign exchange markets. Consistent with our framework, [31] show that vehicle currencies emerge from transaction cost minimization and economies of scale concentrating liquidity along vehicle currency pairs, while [25] emphasize the persistence of vehicles due to trade and industry structure.

Finally, our problem is related to portfolio optimization. Building on the mean–variance optimization framework of [38], subsequent work incorporates metaheuristic methods such as genetic and evolutionary algorithms to address nonlinearities [18, 17, 53]. Our problem, however, jointly determines pool membership and optimal weights with a highly nonlinear objective where weights also appear in the denominator. Moreover, while portfolio optimization seeks to diversify away covariance risk, our setting benefits from highly correlated assets.

10 Conclusion and Future Work

This paper studies the design of multi-currency AMM architectures for foreign exchange systems. In our framework, the equilibrium trading cost depends jointly on trading volumes, exchange-rate volatilities, and pool composition. We show that under this framework, pooling multiple currencies in a constant weighted mean AMM can reduce aggregate trading costs compared to the status quo – bilateral routing through a vehicle currency. The gains arise because multi-currency pools enable direct trades while still concentrating liquidity across many trading pairs. While multi-asset pools suffer from increased risk exposure, pooling assets with correlated exchange rates and appropriately weighing exposures limits the additional impermanent loss. We derive analytical results characterizing optimal weights in symmetric environments and a tractable approximation for general weight optimization, and formulate the system-level problem of partitioning currencies into pools.

To solve the partitioning problem, we employ a hierarchical agglomerative clustering heuristic algorithm that exploits the correlation structure of exchange-rate returns to efficiently group currencies into structurally similar pools. Empirically, using exchange-rate and international trade data for 43 non-pegged currencies over 2008–2023, we find that the resulting architecture reduces realized aggregate trading costs by ~13% relative to bilateral USD routing, with improvements remaining stable through major episodes of financial stress.

Several directions remain for future research. First, relaxing the assumptions we make by allowing persistent order imbalances, richer exchange-rate dynamics, stochastic trade-size distributions, or endogenous trading demand may materially affect the optimal architecture and equilibrium costs. Second, future work could seek analytical characterizations of the general weight optimization solution or derive approximation schemes with formal error bounds. Third, the present framework treats pool membership as fixed after the initial pool setup. The design of a dynamic multi-currency AMM that supports rebalancing mechanisms allowing currencies to migrate between pools as market conditions change could improve the flexibility and robustness of the multi-currency architecture, especially for currencies issued by countries with volatile economic policies. The current work uses historical correlations for pool formation, but the same theory extends to dynamic pools formed using time-varying volatilities. Finally, the robustness of the agglomerative clustering algorithm should be studied over a variety of training data regimes and sets of currencies before it can be proposed as a reliable method for currency pooling, especially if deployed in an algorithmically rebalancing multi-currency AMM. Indeed, there may exist algorithms that are more efficient or robust across different market environments. Alternative approaches based on graph partitioning, metaheuristics, convex relaxations, or machine learning may yield improved solutions or stronger theoretical guarantees. These methods may also be able to deal with more complex pooling architectures, such as non-disjoint pools or pools without a common vehicle currency.

References

- 1 Hayden Adams, Noah Zinsmeister, and Dan Robinson. Uniswap v2 Core. Technical report, Uniswap, March 2020. URL: <https://uniswap.org/whitepaper.pdf>.
- 2 Hayden Adams, Noah Zinsmeister, Moody Salem, River Keefer, and Dan Robinson. Uniswap v3 Core. Technical report, Uniswap, March 2021. URL: <https://uniswap.org/whitepaper-v3.pdf>.
- 3 Guillermo Angeris and Tarun Chitra. Improved price oracles: Constant function market makers. In *Proceedings of the 2nd ACM Conference on Advances in Financial Technologies*, AFT '20, pages 80–91, New York, NY, USA, 2020. Association for Computing Machinery. doi:10.1145/3419614.3423251.

- 4 Guillermo Angeris, Alex Evans, and Tarun Chitra. Replicating market makers. *Digital Finance*, 5(2):367–387, June 2023. doi:10.1007/s42521-023-00082-0.
- 5 Guillermo Angeris, Alex Evans, Tarun Chitra, and Stephen Boyd. Optimal routing for constant function market makers. In *Proceedings of the 23rd ACM Conference on Economics and Computation*, EC '22, pages 115–128, New York, NY, USA, 2022. Association for Computing Machinery. doi:10.1145/3490486.3538336.
- 6 Jun Aoyagi. Liquidity provision by automated market makers, October 2020. URL: https://papers.ssrn.com/sol3/papers.cfm?abstract_id=3674178.
- 7 Bank for International Settlements. Triennial central bank survey: Report on global foreign exchange market activity in 2010. Technical report, Bank for International Settlements, Basel, Switzerland, December 2010. URL: <https://bis.org>.
- 8 Bank for International Settlements. Project mariana: Cross-border exchange of wholesale cbdc's using automated market-makers. Technical report, BIS Innovation Hub, September 2023. URL: <https://www.bis.org/publ/othp75.pdf>.
- 9 Bank for International Settlements. Triennial central bank survey: Otc foreign exchange turnover in april 2025. Technical report, Bank for International Settlements, Basel, Switzerland, September 2025. URL: <https://bis.org>.
- 10 Cara Bordier, Lukas Frei, and Simon Stalder. Dollar dominance: A source of dollar volatility?, May 2026. URL: <https://ssrn.com/abstract=6757018>.
- 11 Agostino Capponi and Ruizhe Jia. The adoption of blockchain-based decentralized exchanges. Papers 2103.08842, arXiv.org, March 2021. URL: <https://ideas.repec.org/p/arx/papers/2103.08842.html>.
- 12 Joe Cash. China, Saudi Arabia sign currency swap agreement. *Reuters*, November 2023. URL: <https://www.reuters.com/markets/currencies/china-saudi-arabia-central-banks-sign-local-currency-swap-agreement-2023-11-20/>.
- 13 Gary Chamberlain and Michael Rothschild. Arbitrage, factor structure, and mean-variance analysis on large asset markets. *Econometrica*, 51(5):1281–1304, 1983. URL: <http://www.jstor.org/stable/1912275>.
- 14 Yu chin Chen and Kenneth Rogoff. Commodity currencies. *Journal of International Economics*, 60(1):133–160, 2003. Empirical Exchange Rate Models. doi:10.1016/S0022-1996(02)00072-7.
- 15 Jana Choukeir. UAE, Ethiopia sign currency swap agreement with value of up to \$817 million. *Reuters*, July 2024. URL: <https://www.reuters.com/business/finance/uae-ethiopia-sign-currency-swap-agreement-with-value-up-817-million-2024-07-16/>.
- 16 Virginie Coudert, Cyriac Guillaumin, and Hélène Raymond. Looking at the other side of carry trades: Are there any safe haven currencies? Working Papers 2014-03, CEPii research center, February 2014. URL: <https://ideas.repec.org/p/cii/cepidt/2014-03.html>.
- 17 Y. Crama and M. Schyns. Simulated annealing for complex portfolio selection problems. *European Journal of Operational Research*, 150(3):546–571, 2003. Financial Modelling. doi:10.1016/S0377-2217(02)00784-1.
- 18 Giacomo Di Tollo and Andrea Roli. Metaheuristics for the portfolio selection problem. *International Journal of Operations Research*, 5(1):13–35, 2008. URL: https://www.orstw.org.tw/ijor/vol5no1/2_IJOR_Vol5_No1_pp13_35.pdf.
- 19 Kimberly Donovan and Maia Nikoladze. The axis of evasion: Behind China's oil trade with Iran and Russia. *Atlantic Council*, March 2024. URL: <https://www.atlanticcouncil.org/blogs/new-atlanticist/the-axis-of-evasion-behind-chinas-oil-trade-with-iran-and-russia/>.
- 20 Michael Egorov. StableSwap—Efficient Mechanism for Stablecoin Liquidity. Technical report, Curve Finance, 2019. URL: <https://curve.fi/files/stableswap-paper.pdf>.
- 21 Alex Evans, Guillermo Angeris, and Tarun Chitra. Optimal fees for geometric mean market makers, 2021. arXiv:2104.00446.

- 22 Elena Fabrichnaya and Guy Faulconbridge. What and where are russia's \$300 billion in reserves frozen in the west?, December 2023. URL: <https://www.reuters.com/world/europe/what-where-are-russias-300-billion-reserves-frozen-west-2023-12-28/>.
- 23 Evan Freidin. Brics pay as a challenge to swift network, December 2024. URL: <https://www.lowyinstitute.org/the-interpreter/brics-pay-challenge-swift-network>.
- 24 Krzysztof Gogol, Johnnatan Messias, Malte Schlosser, Benjamin Kraner, and Claudio Tessone. Cross-border exchange of cbdc's using layer-2 blockchain, 2024. doi:10.48550/arXiv.2312.16193.
- 25 Linda S. Goldberg and Cédric Tille. Vehicle currency use in international trade. *Journal of International Economics*, 76(2):177–192, 2008. doi:10.1016/j.jinteco.2008.07.001.
- 26 Lioba Heimbach, Ye Wang, and Roger Wattenhofer. Behavior of liquidity providers in decentralized exchanges. Papers 2105.13822, arXiv.org, May 2021. URL: <https://ideas.repec.org/p/arx/papers/2105.13822.html>.
- 27 International Monetary Fund. Annual report on exchange arrangements and exchange restrictions (areaer), 2024. URL: <https://www.elibrary-areaer.imf.org/>.
- 28 International Monetary Fund. Exchange rates (er). <https://data.imf.org/en/datasets/IMF.STA:ER>, 2026. Accessed: 2026-05-18.
- 29 International Monetary Fund. International trade in goods (by partner country) (imts). <https://data.imf.org/en/datasets/IMF.STA:IMTS>, 2026. Accessed: 2026-05-18.
- 30 D. Kraft. *A Software Package for Sequential Quadratic Programming*. Deutsche Forschungs- und Versuchsanstalt für Luft- und Raumfahrt Köln: Forschungsbericht. Wiss. Berichtswesen d. DFVLR, 1988. URL: <https://books.google.ca/books?id=4rKaGwAACAAJ>.
- 31 Paul Krugman. The international role of the dollar: Theory and policy implications. In John F. O. Bilson and Richard C. Marston, editors, *Exchange Rate Theory and Practice*, pages 261–278. University of Chicago Press, 1984.
- 32 Alfred Lehar and Christine Parlour. Decentralized exchange: The uniswap automated market maker. *Journal of Finance*, 80(1):321–374, February 2025. doi:10.1111/jofi.13405.
- 33 Reina Ke Xin Li, Andreas Park, Andreas Veneris, and Srisht Fateh Singh. Multi-currency amms for decentralized forex markets: Feasibility & optimal design, 2026. arXiv:2607.26405.
- 34 Alex Lipton and Artur Sepp. Automated market-making for fiat currencies, 2021. arXiv:2109.12196.
- 35 Katya Malinova and Andreas Park. Dethroning the dollar? multi-currency automated market making for fx. Unpublished.
- 36 Katya Malinova and Andreas Park. Learning from defi: Would automated market makers improve equity trading?, August 2023. URL: https://papers.ssrn.com/sol3/papers.cfm?abstract_id=4531670.
- 37 Christopher D. Manning, Prabhakar Raghavan, and Hinrich Schütze. *Introduction to Information Retrieval*. Cambridge University Press, USA, 2008.
- 38 Harry Markowitz. Portfolio selection. *The Journal of Finance*, 7(1):77–91, 1952. URL: <http://www.jstor.org/stable/2975974>.
- 39 Alexander Marrow. Russia, China find payments workaround as US sanctions net widens, sources say. *Reuters*, June 2024. URL: <https://www.reuters.com/world/russia-china-find-payments-workaround-us-sanctions-net-widens-sources-say-2024-06-20/>.
- 40 Fernando Martinelli and Nikolai Mushegian. Balancer: A Non-Custodial Portfolio Manager, Liquidity Provider, and Price Sensor. Technical report, Balancer Labs, September 2019. URL: <https://balancer.fi/whitepaper.pdf>.
- 41 Jason Milionis, Ciamac C. Moallemi, and Tim Roughgarden. Automated market making and arbitrage profits in the presence of fees. Papers 2305.14604, arXiv.org, May 2023. URL: <https://ideas.repec.org/p/arx/papers/2305.14604.html>.
- 42 Jason Milionis, Ciamac C. Moallemi, Tim Roughgarden, and Anthony Lee Zhang. Automated market making and loss-versus-rebalancing. Papers 2208.06046, arXiv.org, August 2022. URL: <https://ideas.repec.org/p/arx/papers/2208.06046.html>.

- 43 Government of India Ministry of External Affairs. Chair's Statement and Outcome Document at BRICS Foreign Ministers' Meeting, May 2026. URL: <https://mea.gov.in/bilateral-documents.htm?dtl/41144>.
- 44 Pintail. Uniswap: A good deal for liquidity providers?, January 2019. URL: <https://pintail1.medium.com/uniswap-a-good-deal-for-liquidity-providers-104c0b6816f2>.
- 45 Ayano Sato, Hayato Nakata, and Jay Percy. Time-variant safe haven currencies. *International Review of Economics & Finance*, 93:316–328, 2024. doi:10.1016/j.iref.2024.04.015.
- 46 Ritu Sharma. Loaded With Massive Reserves Of Rupees In Indian Banks, Russia Buys "Sensitive" Goods From New Delhi Despite Embargo - Reports, September 2024. URL: <https://www.eurasiantimes.com/ne-loaded-with-massive-reserves-of-rupees/>.
- 47 Fabricius Somogyi. Dollar dominance in fx trading. *Management Science*, April 2026. doi: 10.1287/mnsc.2024.07656.
- 48 Benn Steil, Yuma Schuster, and Samuel Zucker. Central bank currency swaps tracker, April 2026. URL: <https://www.cfr.org/articles/central-bank-currency-swaps-tracker>.
- 49 United Nations Statistics Division. UN Comtrade Database, 2025. Accessed: 2026-05-19. URL: <https://comtradeplus.un.org/>.
- 50 Maria Elena Vizcaino, Zijia Song, and Nicolle Yapur. Resurgent Political Risk Derails Rallies Across Emerging Markets. *Bloomberg Markets*, May 2026. URL: <https://www.bloomberg.com/news/articles/2026-05-24/resurgent-political-risk-derails-rallies-across-emerging-markets>.
- 51 Kandy Wong. Saudi Arabia 'open' to petroyuan, closer China ties, minister says. *South China Morning Post*, September 2024. URL: <https://www.scmp.com/economy/global-economy/article/3277788/saudi-arabia-open-petroyuan-closer-china-ties-minister-says>.
- 52 Álvaro Cartea, Fayçal Drissi, and Marcello Monga. Decentralised finance and automated market making: Predictable loss and optimal liquidity provision, 2024. arXiv:2309.08431.
- 53 Álvaro Rubio-García, Juan José García-Ripoll, and Diego Porras. Portfolio optimization with discrete simulated annealing, 2022. arXiv:2210.00807.

A

 Proofs

A.1 Proofs of benchmark arrangement trading costs

► **Proposition 4.1** (Status quo trading cost). *The expected aggregate trading cost in the status quo for $N + 1$ currencies routed through the vehicle currency 0 is*

$$c^{SQ} = \sum_i \sqrt{\sigma_i^2 E[Q_i]} \Delta$$

Proof. We sum the costs over the volume of the bilateral trades:

$$\sum_{i < j} Q_{ij} \left(\sqrt{\frac{\Delta \sigma_i^2}{E[Q_i]}} + \sqrt{\frac{\Delta \sigma_j^2}{E[Q_j]}} \right) = \sum_{i < j} Q_{ij} \sqrt{\frac{\Delta \sigma_i^2}{E[Q_i]}} + \sum_{i < j} Q_{ij} \sqrt{\frac{\Delta \sigma_j^2}{E[Q_j]}}.$$

Swapping $i \leftrightarrow j$ in the second sum and using $Q_{ij} = Q_{ji}$, the two combine to give $\sum_i \sum_{j \neq i} \frac{\sigma_i Q_{ij} \sqrt{\Delta}}{\sqrt{E[Q_i]}}$. Hence, in expectation,

$$c^{SQ} = \sum_i \sqrt{\sigma_i^2 E[Q_i]} \Delta \quad \blacktriangleleft$$

► **Proposition 4.2** (Equal weights trading cost). *The expected aggregate trading cost in an equal-weighted $N + 1$ currency constant-mean pool is*

$$c^{SYM} = \sqrt{\frac{2\Delta H E[Q]}{N + 1}}$$

13:18 Multi-Currency AMMs for Decentralized FOREX Markets

Proof. With equal weights, we can drop the weights and recover a constant product invariant. For a trade of size Δ in numeraire from currency i to j , this invariant function requires

$$\begin{aligned} \left(R_i + \frac{\Delta}{S_i}\right) \left(R_j - \frac{\Delta}{S^{\text{eff}}}\right) &= R_i R_j, \\ \iff \frac{\Delta}{S^{\text{eff}}} &= R_j - \frac{R_i R_j}{R_i + \Delta/S_i} = R_j \left(\frac{\Delta/S_i}{R_i + \Delta/S_i} \right) = \frac{R_j \Delta}{S_i R_i + \Delta} \iff S^{\text{eff}} = \frac{S_i R_i + \Delta}{R_j} \end{aligned}$$

The spot price of currency i in terms of currency j implied by the pool is $S_{ij}^{\text{AMM}} = R_j/R_i$. At fundamental prices, $S_{ij}^{\text{AMM}} = S_i/S_j$ implies $S_i R_i = S_j R_j$ for all pairs, so each currency contributes equal value to the pool: $V = (N+1)S_i R_i$ for all i .

Hence, $S^{\text{eff}} = \frac{S_i R_i}{R_j} + \frac{\Delta}{R_j} = S_j + \frac{\Delta(N+1)S_j}{V}$. The price impact $\Pi(\Delta) = \frac{(N+1)\Delta}{V}$ follows from Equation (1) using $S = S_j$.

The pool value $V(T)$ is also governed by the invariant: $\prod_i R_i(0) = \prod_i R_i(T)$, implying

$$\prod_i \frac{V(0)}{S_i(0)(N+1)} = \prod_i \frac{V(T)}{S_i(T)(N+1)} \iff \frac{V(T)}{V(0)} = \prod_i \left(\frac{S_i(T)}{S_i(0)} \right)^{1/(N+1)} = \prod_i x_i^{1/(N+1)}.$$

Meanwhile, the counterfactual hold value is

$$V_{\text{hold}}(T) = \sum_i x_i S_i(0) R_i(0) = \frac{V(0)}{N+1} \sum_i x_i$$

By Equation (2), the IPL is then

$$\text{IPL} = \prod_i x_i^{1/(N+1)} - \frac{1}{N+1} \sum_i x_i$$

For $x_i = 1 + \epsilon_i$ and $\epsilon_i \approx 0$, the second-order Taylor approximation around $\epsilon_i = 0$ is

$$\text{IPL} \approx -\frac{1}{2(N+1)^2} \sum_{i < j} (\epsilon_i - \epsilon_j)^2, \quad E[|\text{IPL}|] \approx \frac{H}{2(N+1)^2}, \quad H = \sum_{i < j} \sigma_{ij}^2$$

where the expectation is taken assuming ϵ_i has zero-mean. This results in the break-even pool value $V = \frac{2(N+1)^2 f E[Q]}{H}$ (from Equation (3)) and unit trading cost

$$c(\Delta, f) = \frac{H\Delta}{2(N+1)fE[Q]} + f$$

Minimizing over f (the AM-GM inequality implies that $A/f + Bf \geq 2\sqrt{AB}$ for $f > 0$, and equality here holds when $f = \sqrt{A/B}$) yields

$$f^* = \sqrt{\frac{H\Delta}{2(N+1)E[Q]}}, \quad c^* = \sqrt{\frac{2H\Delta}{(N+1)E[Q]}}$$

Aggregating over the total expected trade volume $E[Q]$ yields the expected aggregate cost

$$c^{\text{SYM}} = \sqrt{\frac{2HE[Q]\Delta}{(N+1)}} \quad \blacktriangleleft$$

A.2 Proofs of costs in constant-mean AMMs

► **Lemma 5.1** (Price impact). *In the general constant-mean AMM, the price impact of a trade of size Δ in numeraire between currencies i and j is approximately*

$$\Pi(\Delta) \approx \frac{w_i + w_j}{w_i w_j} \cdot \frac{\Delta}{2V}$$

Proof. Consider a trade between currencies i and j of notional size Δ in the numeraire (so the trader pays Δ/S_i units of currency i). As only the reserves R_i and R_j change, the invariance function requires

$$\left(R_i + \frac{\Delta}{S_i}\right)^{w_i} \left(R_j - \frac{\Delta}{S^{\text{eff}}}\right)^{w_j} = R_i^{w_i} R_j^{w_j},$$

where Δ/S^{eff} is the number of units of j received. Isolating the output quantity:

$$\frac{\Delta}{S^{\text{eff}}} = R_j \left(1 - \left(1 + \frac{\Delta}{S_i R_i}\right)^{-w_i/w_j}\right)$$

Assuming $\Delta \ll S_i R_i = w_i V$, a second-order Taylor expansion gives

$$\left(1 + \frac{\Delta}{S_i R_i}\right)^{-w_i/w_j} \approx 1 - \frac{w_i}{w_j} \frac{\Delta}{S_i R_i} \left(1 - \frac{1}{2} \frac{\Delta}{S_i R_i} \frac{w_i + w_j}{w_j}\right)$$

The spot price of currency i in terms of currency j is $S_{ij}^{\text{AMM}} = (R_j/w_j)/(R_i/w_i)$. At fundamental prices, $S_{ij}^{\text{AMM}} = S_i/S_j$ implies $S_i R_i/w_i = S_j R_j/w_j$ for all pairs, so each currency i accounts for w_i of total pool value, i.e., $w_i V = S_i R_i$. Then, using $R_j w_i/w_j = (S_i/S_j) R_i$ and $S_i R_i = w_i V$:

$$\frac{\Delta}{S^{\text{eff}}} \approx \frac{\Delta}{S_j} \left(1 - \frac{\Delta}{2V} \frac{w_i + w_j}{w_i w_j}\right)$$

Inverting and applying a first-order expansion (valid for $\Delta \ll 2w_i w_j V/(w_i + w_j)$):

$$S^{\text{eff}} \approx S_j \left(1 + \frac{w_i + w_j}{w_i w_j} \frac{\Delta}{2V}\right)$$

Substituting into (1) yields the result. ◀

► **Lemma 5.2** (Impermanent loss). *For gross returns x_i with $x_i \approx 1$ and $E[x_i] = 1$, the expected impermanent loss of the general constant-mean pool satisfies*

$$E[|IPL|] \approx \frac{H_w}{2}, \quad H_w = \sum_{i < j} w_i w_j \sigma_{ij}^2$$

Proof. We first derive the exact expression for $V(T)$. As derived in the proof of Proposition 5.1, at fundamental prices, $R_i(t) = w_i V(t)/S_i(t)$ for all i . Substituting into the constant-mean invariant:

$$\prod_i \left(\frac{w_i V(T)}{S_i(T)}\right)^{w_i} = \prod_i \left(\frac{w_i V(0)}{S_i(0)}\right)^{w_i}$$

Since $\sum_i w_i = 1$, the V terms factor out as $V(T)^{\sum_i w_i} = V(T)$ and $V(0)^{\sum_i w_i} = V(0)$, giving

$$V(T) = V(0) \prod_i \left(\frac{S_i(T)}{S_i(0)}\right)^{w_i} = V(0) \prod_i x_i^{w_i}$$

The counterfactual hold value is

$$V_{\text{hold}}(T) = \sum_i S_i(T) R_i(0) = \sum_i S_i(T) \cdot \frac{w_i V(0)}{S_i(0)} = V(0) \sum_i w_i x_i$$

Substituting into (2) gives the exact expression $\text{IPL} = \prod_i x_i^{w_i} - \sum_i w_i x_i$.

For $x_i = 1 + \epsilon_i$ and $\epsilon_i \approx 0$, the second-order Taylor expansion around $\epsilon_i = 0$ is

$$\begin{aligned} \text{IPL} &\approx \left(1 + \sum_i w_i \epsilon_i + \frac{1}{2} \left[\left(\sum_i w_i \epsilon_i \right)^2 - \sum_i w_i \epsilon_i^2 \right] \right) - \left(1 + \sum_i w_i \epsilon_i \right) \\ &= -\frac{1}{2} \left(\sum_i w_i \epsilon_i^2 - \left(\sum_i w_i \epsilon_i \right)^2 \right) \end{aligned}$$

Expanding $(\sum_i w_i \epsilon_i)^2 = \sum_i w_i^2 \epsilon_i^2 + 2 \sum_{i < j} w_i w_j \epsilon_i \epsilon_j$, grouping the sums of ϵ_i^2 , and using $1 - w_i = \sum_{j \neq i} w_j$ to write $\sum_i w_i (1 - w_i) \epsilon_i^2 = \sum_{i < j} w_i w_j (\epsilon_i^2 + \epsilon_j^2)$, we get

$$\text{IPL} \approx -\frac{1}{2} \sum_{i < j} w_i w_j (\epsilon_i^2 + \epsilon_j^2 - 2\epsilon_i \epsilon_j) = -\frac{1}{2} \sum_{i < j} w_i w_j (\epsilon_i - \epsilon_j)^2$$

The expectation under zero-mean returns is $E[|\text{IPL}|] \approx H_w/2$ where $H_w = \sum_{i < j} w_i w_j \sigma_{ij}^2$. ◀

► **Proposition 5.3** (Equilibrium trading cost). *Under the break-even condition (3) and Lemmas 5.1–5.2, the aggregate equilibrium trading cost is minimized at*

$$f^* = \sqrt{\frac{\Delta H_w}{4(E[Q])^2} \sum_i \frac{E[Q_i]}{w_i}}, \quad c^* = \sqrt{\Delta H_w \sum_i \frac{E[Q_i]}{w_i}} \quad (4)$$

Proof. By Lemma 5.2, $E[|\text{IPL}|] \approx H_w/2$. The break-even condition (3) therefore gives equilibrium pool depth

$$V = \frac{2fE[Q]}{H_w}.$$

By Lemma 5.1, the unit trading cost for a pair (i, j) trade of size Δ is

$$c_{ij}(\Delta, f) = \frac{w_i + w_j}{w_i w_j} \frac{\Delta}{2V} + f = \frac{w_i + w_j}{w_i w_j} \frac{H_w \Delta}{4fE[Q]} + f.$$

Aggregating over all pairs, weighted by their respective volumes Q_{ij} :

$$c(\Delta, f) = \frac{H_w \Delta}{4fE[Q]} \sum_{i < j} Q_{ij} \frac{w_i + w_j}{w_i w_j} + fQ.$$

We simplify $\sum_{i < j} Q_{ij} (w_i + w_j) / (w_i w_j) = \sum_{i < j} Q_{ij} / w_j + \sum_{i < j} Q_{ij} / w_i$. Swapping $i \leftrightarrow j$ in the second sum and using $Q_{ij} = Q_{ji}$, both sums combine to give $\sum_i Q_i / w_i$, where $Q_i = \sum_{j \neq i} Q_{ij}$ is the total volume involving currency i . Hence

$$c(\Delta, f) = \frac{H_w \Delta}{4fE[Q]} \sum_i \frac{Q_i}{w_i} + fQ. \quad (5)$$

In expectation, minimizing over f (the AM-GM inequality implies that $A/f + Bf \geq 2\sqrt{AB}$ for $f > 0$, and equality here holds when $f = \sqrt{A/B}$) yields

$$f^* = \sqrt{\frac{H_w \Delta}{4(E[Q])^2} \sum_i \frac{E[Q_i]}{w_i}}, \quad c^* = \sqrt{\Delta H_w \sum_i \frac{E[Q_i]}{w_i}} \quad \blacktriangleleft$$

A.3 Proofs of parametrized optimums

► **Proposition 6.1** (Optimal weight and cost for vehicle and 2 symmetric currencies). *Under the symmetric (v, s) parametrization for 3 currencies, if $s^2 < 2(1 + 2v)/(1 + v)$, the expected aggregate trading cost of Equation (4) has unique interior minimizer at $w_1^* = w_2^* = \frac{1-w_0^*}{2}$ and*

$$w_0^* = \frac{s}{\sqrt{(1+2v)(4-s^2)}}, \quad c^* = \sqrt{q\sigma^2\Delta} \sqrt{s^2v + 2 + s\sqrt{(4-s^2)(1+2v)}}$$

If $s^2 \geq 2(1 + 2v)/(1 + v)$, no interior minimizer exists; the infimum over $w_1, w_2 > 0$ is obtained as $w_0 \rightarrow 1$.

Proof. With $w_1 = w_2 = (1 - w_0)/2$:

$$H_w = w_1 w_2 \sigma_{12}^2 + w_0 w_1 \sigma_{01}^2 + w_0 w_2 \sigma_{02}^2 = \frac{1}{4}(1 - w_0)^2 s^2 \sigma^2 + (1 - w_0) w_0 \sigma^2.$$

The volume-weighted sum is $\sum_i E[Q_i]/w_i = 4q(1 + v)/(1 - w_0) + 2q/w_0$. After substituting into Equation (4) and simplifying, the expected aggregate cost $c(w_0; s, v)$ satisfies

$$\frac{c(w_0; s, v)^2}{2q\sigma^2\Delta} = \left(\frac{s^2v}{2} + 1 \right) + P(w_0; s, v),$$

where

$$P(w_0; s, v) = w_0 \left(1 - \frac{s^2}{4} \right) (1 + 2v) + \frac{s^2}{4w_0}.$$

Here, c is monotonically increasing in P , so we solve the equivalent problem of minimizing P . Since $d^2P/dw_0^2 = s^2/(2w_0^3) > 0$ for all $w_0 > 0$, P is strictly convex in w_0 for that region. We have the form $A/w_0 + Bw_0$, which has minimum $2\sqrt{AB}$ attained at $w_0 = \sqrt{A/B}$, yielding

$$w_0^* = \frac{s}{\sqrt{(1+2v)(4-s^2)}}, \quad P(w_0^*; s, v) = \frac{s}{2} \sqrt{(4-s^2)(1+2v)},$$

When $s^2 < 2(1 + 2v)/(1 + v)$, $w_0^* < 1$ is a valid interior solution with

$$c^* = \sqrt{q\sigma^2\Delta} \sqrt{s^2v + 2 + s\sqrt{(4-s^2)(1+2v)}}.$$

For $s \in [0, 2]$ and $v > 0$, these quantities are well-defined.

When $s^2 \geq 2(1 + 2v)/(1 + v)$, the unconstrained minimizer is $w_0^* \geq 1$, and the convexity of $P(w_0)$ implies that the constrained infimum is attained as $w_0 \rightarrow 1$. ◀

► **Proposition 6.2** (Optimum for vehicle and N symmetric currencies). *Under the symmetric (v, s) parametrization for $N + 1$ currencies, if $s^2 < 2(1 + Nv)/(1 + (N - 1)v)$, the expected aggregate trading cost from Equation (4) has unique interior minimizer at $w_i^* = (1 - w_0^*)/N$ for all $i > 0$ and*

$$w_0^* = \frac{s}{\sqrt{(1+Nv)(2N - (N-1)s^2)}}$$

If $s^2 \geq 2(1 + Nv)/(1 + (N - 1)v)$, no interior minimizer exists; the infimum over $w_1, w_2 > 0$ is obtained as $w_0 \rightarrow 1$.

To prove the proposition, we first establish the following lemma on the admissible range of s .

► **Lemma A.1.** *Under the above parametrization, $s \in \left[0, \sqrt{\frac{2N}{N-1}}\right]$.*

13:22 Multi-Currency AMMs for Decentralized FOREX Markets

Proof. Let C be the $N \times N$ covariance matrix of the returns of the N symmetric currencies, which has σ^2 on the diagonal and $\sigma^2 - \frac{s^2\sigma^2}{2}$ elsewhere. The variance of any linear combination of the currencies must be non-negative, so R must be positive semi-definite (PSD). For simplicity, we will work with the correlation matrix R which satisfies $\sigma^2 R = C$ and must also be PSD.

We can write $R = (1-\rho)I + \rho uu^T$, where I is the $N \times N$ identity matrix and $u^T = (1, \dots, 1)$. Then, the eigenvectors and eigenvalues of R satisfy

$$Rv = (1-\rho)v + \rho\langle u, v \rangle u = \lambda v \iff (\lambda + \rho - 1)v = \rho\langle u, v \rangle u.$$

which implies that, either both sides are 0, which corresponds to eigenvalue $\lambda_1 = 1 - \rho$, or $v = au$ for some $a \in \mathbb{R} \setminus \{0\}$ and $\lambda + \rho - 1 = \rho N$, for which we have eigenvalue $\lambda_2 = \rho(N-1) + 1$. Thus R is positive semidefinite if and only if

$$1 - \rho \geq 0 \quad \text{and} \quad 1 + (N-1)\rho \geq 0 \iff -\frac{1}{N-1} \leq 1 - \frac{s^2}{2} \leq 1.$$

Rearranging, we have $s \leq \sqrt{\frac{2N}{N-1}}$. ◀

Now, assuming $s \in [0, \sqrt{\frac{2N}{N-1}}]$, we prove the proposition.

Proof. Now, in the N symmetric currency AMM, we have $w_i = (1 - w_0)/N$, implying

$$H_w = \sum_{i>0} \frac{w_0(1-w_0)}{N} \sigma^2 + \sum_{0<i<j} \frac{(1-w_0)^2}{N^2} s^2 \sigma^2 = N \frac{w_0(1-w_0)}{N} \sigma^2 + \binom{N}{2} \frac{(1-w_0)^2}{N^2} s^2 \sigma^2.$$

This simplifies to

$$H_w = w_0(1-w_0)\sigma^2 + \frac{N-1}{2N}(1-w_0)^2 s^2 \sigma^2$$

The volume-weighted sum is

$$\sum_i \frac{E[Q_i]}{w_i} = \frac{Nq}{w_0} + \frac{N(q + vq(N-1))}{(1-w_0)/N} = Nq \left(\frac{(1-w_0) + w_0 \cdot N(1 + v(N-1))}{w_0(1-w_0)} \right)$$

Substituting, the expected aggregate cost is

$$c(\Delta, f) = \sqrt{q\sigma^2 N \Delta} \sqrt{\left(w_0 + \frac{N-1}{2N}(1-w_0)s^2 \right) \left(\frac{(1-w_0) + w_0 \cdot N(1 + v(N-1))}{w_0} \right)} \quad (6)$$

and, in optimum, satisfies

$$\begin{aligned} \frac{(c^*)^2}{q\sigma^2 N \Delta} &= (1-w_0) + w_0 N(1 + v(N-1)) + \frac{N-1}{2N w_0} (1-w_0)^2 s^2 + \frac{N-1}{2} (1-w_0) s^2 (1 + v(N-1)) \\ &= 1 + \frac{s^2}{2N} (N-1)(N(1 + v(N-1)) - 2) + P(w_0; s, v) \end{aligned}$$

where

$$P(w_0; s, v) = w_0(N-1)(1 + Nv) \frac{1}{2N} (2N - (N-1)s^2) + \frac{(N-1)s^2}{2N w_0}$$

Again, c is strictly monotonically increasing in P , we minimize P instead. Since $d^2P/dw_0^2 = (N-1)s^2/w_0^3 > 0$ for all $w_0 > 0$, P is strictly convex in w_0 for that region. Once again, we have the form $A/w_0 + Bw_0$, which attains its minimum, $2\sqrt{AB}$ at $w_0 = \sqrt{A/B}$, yielding

$$w_0^* = \frac{s}{\sqrt{(1+Nv)(2N-(N-1)s^2)}}, \quad P(w_0^*; s, v) = \frac{s(N-1)}{N} \sqrt{(1+Nv)(2N-(N-1)s^2)},$$

When $s^2 < 2(1+2v)/(1+v)$, $w_0^* < 1$ is a valid interior solution with

$$c^* = \sqrt{q\sigma^2\Delta} \sqrt{N + \frac{s^2}{2}(N-1)(N(1+v(N-1))-2) + s(N-1)\sqrt{(1+Nv)(2N-(N-1)s^2)}} \quad (7)$$

These quantities are well-defined for $v > 0$ and the range of s derived in Lemma A.1.

When $s^2 \geq 2(1+Nv)/(1+(N-1)v)$, the unconstrained minimizer is $w_0^* \geq 1$, and the convexity of $P(w_0)$ implies that the constrained infimum is attained as $w_0 \rightarrow 1$. ◀

A.4 Proof of tractable weight optimization

► **Proposition 7.1** (Approximate optimal weights). *When $H_i > 0$ for all i and $E[Q_i] > 0$ for some i , the minimizer of $\tilde{\mathcal{F}}(w) = \sum_i \frac{E[Q_i]}{w_i H_i}$ subject to $\sum_i w_i = 1$, $w_i > 0 \forall i$, is*

$$w_i^* = \frac{1}{D} \sqrt{\frac{E[Q_i]}{H_i}}, \quad \forall i = 0, \dots, N, \quad \text{where } D = \sum_j \sqrt{E[Q_j]/H_j},$$

Proof. We minimize $\tilde{\mathcal{F}}(w) = \sum_i \frac{E[Q_i]}{w_i H_i}$ subject to $\sum_i w_i = 1$, $w_i > 0$, where $H_i = \sum_{j \neq i} \sigma_{ij}^2$, via the Lagrangian

$$\mathcal{L}(w, \lambda) = \sum_i \frac{E[Q_i]}{w_i H_i} + \lambda \left(\sum_i w_i - 1 \right).$$

The first-order conditions $\nabla_w \mathcal{L}(w, \lambda) = 0$ gives $\frac{E[Q_i]}{w_i^2 H_i} = \lambda$, so $w_i = \sqrt{\frac{E[Q_i]}{\lambda H_i}}$ for all i . Imposing $\sum_i w_i = 1$ through the condition $\nabla_\lambda \mathcal{L}(w, \lambda) = 0$:

$$\frac{1}{\sqrt{\lambda}} \sum_j \sqrt{E[Q_j]/H_j} = 1 \implies \sqrt{\lambda} = \sum_j \sqrt{E[Q_j]/H_j} \equiv D, \implies w_i^* = D^{-1} \sqrt{E[Q_i]/H_i}$$

The hessian $\nabla_{ww}^2 \mathcal{L}(w, \lambda)$ has $\frac{\partial^2 \mathcal{L}}{\partial w_i^2} = \frac{2E[Q_i]}{w_i^3 H_i}$ on the diagonal and $\frac{\partial^2 \mathcal{L}}{\partial w_i \partial w_j} = 0$ elsewhere, so the second-order sufficient condition, $v^T \nabla_{ww}^2 \mathcal{L}(w, \lambda) v > 0$ for all $v \neq 0$, becomes

$$\sum_i v_i^2 \frac{\partial^2 \mathcal{L}}{\partial w_i^2} = \sum_i v_i^2 \frac{2E[Q_i]}{w_i^3 H_i} > 0,$$

which holds when all $w_i > 0$, so this a global minimum for $w_i > 0$. ◀

B–D Full Version Material

Appendices B, C, and D are omitted here due to space limitations. They can be found in their respective Appendix sections in the full version of this paper [33].