

스케이트 AMM: 단일 상태 머신 내 유동성 통합

Siddharth Lalwani

Skate

March 5, 2025

Abstract

기존의 자동 시장 조성자들 (Automated Market Makers, AMMs) 은 각 블록체인별로 작동하는 모놀리식 (monolithic) 구조로 인한 근본적인 한계를 가지고 있으며 유동성의 파편화, 가격 곡선의 분리 및 비효율적인 자본 활용 등을 발생시킵니다. 본 문서에서 우리는 서로 상이한 블록체인 생태계에서 하나의 표준 가격 곡선을 유지하며, 복수의 가상머신 환경 (예를 들어, EVM, 솔라나, MoveVM, TonVM) 에서 단일 AMM 이 끊임 없이 운영될 수 있도록 지원하는 스테이트리스 (stateless) 애플리케이션 디자인 패턴을 구현하는 이전에 없던 프로토콜, “Skate AMM(스케이트 AMM)”을 소개합니다. 어느 한 지원 체인에 유동성이 입금되면 해당 유동성은 하나의 통합 유동성 상태 (canonical liquidity state) 로 편입되며, 오직 전용 허브 체인에서만 유지됩니다. 토큰 커스터디는 여전히 개별 체인 (스포크) 에서 탈중앙화 방식으로 유지되나 핵심 가격 로직, 유동성 상태 및 틱데이터는 글로벌로 공유됩니다. 이러한 통합된 상태 (unified state) 는 정족수의 AVS 오퍼레이터 그룹 (리스테이킹된 노드 운영자들로, 슬래싱의 영향을 받을 수 있음) 이 허브-스포크 (hub-and-spoke) AVS 모델을 통해 허브에서 주변 컴포넌트 (periphery) 에 대한 콜백을 트리거하는 태스크들을 생성함으로써 상태 업데이트를 집합적으로 증명하는 EigenLayer 의 경제적 신뢰에 의해 보호됩니다. 또한 우리는 통합된 가격 곡선을 설명하는 새로운 수학 공식들을 소개하며, 리밸런싱 대비 손실 (loss-versus-rebalancing, LVR) 에 대한 분석을 제공하여 스케이트의 설계가 어떻게 유해한 흐름 (toxic flow) 및 차익 거래 손실을 최소화하는지 선보입니다.

Contents

1	배경	3
2	스테이트리스 애플리케이션 패턴: Kernel-Periphery 모델	3
2.1	Kernel (허브 체인)	3
2.2	Periphery (스포크 체인)	4
2.3	상태 동기화를 위한 EigenLayer AVS	4
2.4	Execution Network (실행 네트워크)	4
2.5	스케이트 토큰: 조정 및 보안	4
3	프로토콜 메커니즘 및 스왑 실행	4
3.1	스케이트 AMM 모델	4
3.2	스왑 실행 순서	5
3.3	유동성 공급 순서	6
3.4	견고한 보안 및 완화 전략	7
4	유동성 공급자	7
4.1	통합 AMM 의 이점을 누리는 LP	7
4.2	LP 수수료 최적화 및 글로벌 수수료 공유	7
4.3	FLAIR 지표에 대한 영향	9
5	스왑 규모 제한 및 로컬 유동성 제약 사항	9
5.1	체인별 수수료 컴포넌트	9
5.2	잔고 리밸런싱에 대한 인센티브	9
6	향후 비전: 스케이트 AMM v2 와 그 이후	10
6.1	스케이트 AMM v2: 크로스체인 확장	10
6.2	왕복 스왑거래를 통한 리밸런싱 향상	10
6.3	체인 내 유동성 제공	11
7	결론	11
A	형식적 도출 및 비교 분석	12
A.1	통합된 가격 및 유동성 인베리언트	12
A.2	LVR 측정	12
A.3	동적 수수료 조정 모델	12
A.4	유동성 이동을 위한 균형 모델	13
A	참조	14

1 배경

레이어 1 블록체인들과 이들의 다양한 가상 머신 (virtual machine, VM) 환경 (예를 들어, 이더리움 가상머신 (EVM), 솔라나 VM, TON VM, Move VM 등) 의 급속한 확장은 탈중앙화 금융 (decentralized finance, DeFi) 환경의 단편화를 불러 일으켰습니다. 전통적인 AMM 들 (automated market maker, 자동 시장 조성자) 에서 유동성은 각 체인별로 독립되어 있으며, 이로 인해 특히 신규 또는 유동성이 낮은 네트워크에서 자산간 가격 불균형 및 비효율적인 사용자 경험이 발생합니다. 예를 들어, 서로 다른 체인에 배포된 유니스왑 v3 포크들은 각각 개별적으로 유동성을 부트스트래핑해야 하며, 이는 종종 거래자들에게 유동성 풀의 부족과 더 높은 슬리피지를 발생시킵니다. 이러한 파편화 (fragmentation) 는 사용자들이 복잡한 크로스체인 상호작용을 수행하거나 최선이 아닌 교환 비율을 수용할 수밖에 없게 만들며, 유동성 공급자들 (liquidity providers, LPs) 이 자신의 자본을 각각의 독립된 풀에 분산시키도록 강제하여 수수료 수익은 감소하고 무상한 손실 (impermanent loss) 에 대한 노출이 증가합니다.

스케이트 (Skate) 는 스테이트리스 (stateless) 애플리케이션들의 개발을 지원하는 인프라를 제공하여 각 블록체인마다 유동성 상태가 개별적으로 관리되는 모놀리식 (monolithic) AMM 구조의 한계를 직접적으로 해결합니다. 스케이트 AMM 은 스테이트리스 애플리케이션 모델의 구현을 통해 해당 인프라를 활용하며, 전용 허브 체인에 배포된 *Kernel*(커널) 스마트 컨트랙트 하나를 통해 관리되는 글로벌 스테이트에 유동성을 통합시킵니다. 각 블록체인에 배포된 경량 *Periphery*(페리퍼리) 컨트랙트들은 사용자들의 상호작용, *EigenLayer* 자율 검증 서비스 (AVS) 서명 검증, 자산 커스터디, 그리고 자산의 정산을 관리하지만 가격 정보를 저장하지는 않습니다. 체인들 간 상태의 일관성은 스케이트 AVS 를 통해 유지되며, 모든 상태 업데이트에서 탈중앙화 및 무신뢰 검증이 보장됩니다.

이 방식은 통합된 가격 곡선을 형성하며 유동성 수준을 크게 향상시키고, LP 에 대한 유해한 흐름 (toxic flow) 을 감소시키며 자본 효율성을 극대화합니다. 또한 자산 발행자들에게는 다음과 같은 이점을 제공합니다:

- **즉각적인 시장 깊이와 안정성:** 통합된 가격 (unified pricing) 은 자산에 풍부한 유동성이 즉각적으로 공급되도록 하여 슬리피지와 차익거래를 감소시킵니다.
- **자본요건의 감소:** 단 한 번의 유동성 예치로 옴니체인 전반에서 풍부한 유동성이 확보되며, 필요한 자본이 감소합니다.
- **운영 단순화:** 자산 발행자들은 다수의 체인에 분산된 유동성 풀을 관리하는 대신 전략적 이니셔티브에 집중할 수 있습니다.

2 스테이트리스 애플리케이션 패턴: Kernel-Periphery 모델

스케이트 AMM 이 활용하는 스테이트리스 애플리케이션 디자인 패턴은 복수의 블록체인에서 독립적인 유동성 상태를 유지하는 것에 따른 비효율성을 해결합니다. 해당 아키텍처는 경량 클라이언트 컴포넌트가 계산 집약적 작업 및 통합 상태 관리를 처리하는 중앙 집중형 서버와 상호작용하는 전통적인 분산 컴퓨팅 시스템의 클라이언트-서버 패러다임을 참고하여 구현되었습니다. 전통적인 웹 애플리케이션이 경량 클라이언트를 활용해 계산 및 데이터 일관성 유지의 책임을 중앙 백엔드 서버에 위임하는 것처럼, 스케이트 모델은 블록체인 인프라에 유사한 원리를 적용합니다. 이 모델은 로컬 자산 관리 및 사용자 상호작용을 글로벌 유동성 관리, 가격 메커니즘, 그리고 전용 허브 체인에서 구현되는 인베리언트 강제 (invariant enforcement) 와 같은 공유 연산 로직과 명확히 구분합니다. 크로스체인 상호작용과 상태 업데이트는 스케이트의 AVS 가 제공하는 탈중앙화되고 경제적으로 보호된 검증을 통해 보장됩니다.

2.1 Kernel (허브 체인)

Kernel(커널) 스마트 컨트랙트는 지정된 허브 체인에 배포되며 글로벌 유동성 풀과 가격, 그리고 틱데이터를 관리합니다. 해당 스마트 컨트랙트는 집중 유동성 인베리언트 (예를 들어 Uniswap v3 에서 사용됨) 를 활용하되 이를 통합된 멀티체인 컨텍스트에 적용하는 핵심 AMM 로직을 구현합니다. 모든 거래 및 유동성 변화는 *Kernel* 에서 연산되며, *Kernel* 은 상수곱 인베리언트 (constant-product invariant) 를 강제하여 글로벌 상태를 업데이트합니다. 단, *Kernel* 은 토큰 커스터디를 직접 관리하지 않고 모든 체인에서 사용자들이 제공한 유동성 포지션을 집계합니다.

2.2 Periphery (스포크 체인)

각 지원 체인에는 Periphery(페리퍼리) 컨트랙트가 배포되어 사용자 인터페이스 역할을 수행합니다. Periphery 는 로컬 토큰 커스터디를 관리하며 사용자의 스왑이나 유동성 공급 등의 의도를 기록하고, 이를 크로스체인 메시지를 통해 Kernel 에 전달합니다. Kernel 의 응답은 허브-스포크 모델에서 스케이트의 AVS 를 통해 콜백 형태로 해당 Periphery 컨트랙트에 전달됩니다. Periphery 컨트랙트는 어떠한 가격 계산도 수행하지 않으며, 오직 Kernel 의 글로벌 상태에 전적으로 의존합니다. 이러한 분리를 통해 온체인 복잡성이 최소화되고 모든 체인의 사용자들은 통합된 단일 시장을 경험할 수 있습니다.

2.3 상태 동기화를 위한 EigenLayer AVS

스케이트는 EigenLayer AVS 를 활용하여 Kernel 과 Periphery 컨트랙트 간의 안전한 조율을 수행합니다. AVS 는 분산된 운영자 네트워크로 구성되며, 이들 운영자들은 자산 리스테이킹을 통해 안전하고 신뢰를 최소화한 크로스체인 메시지 증명 방법을 제공합니다. 해당 프로토콜은 리스테이킹된 ETH 와 스테이킹된 \$SKATE 토큰을 모두 활용하는 이중 스테이킹 모델을 지원하여 경제적 보안을 강화합니다. 사용자가 Periphery 컨트랙트에 트랜잭션을 제출하면 해당 이벤트는 AVS 노드에 의해 Kernel 로 릴레이되며, Kernel 이 트랜잭션을 처리한 후 결과값은 AVS 에 의해 증명되어 적절한 Periphery 컨트랙트에 콜백으로 전달됩니다. 이 메커니즘은 상태 업데이트와 거래 실행이 경제적 인센티브를 받은 정족수의 운영자 그룹에 의해 검증되도록 보장합니다.

2.4 Execution Network (실행 네트워크)

스케이트 아키텍처 내 실행 네트워크는 사용자들이 Periphery(스포크 체인) 컨트랙트에서 시작한 액션을 수신하는 실행자 (Executors) 들로 구성됩니다. 실행자들은 사용자의 의도를 파악하여 이를 허브 체인의 Kernel 로 안전하게 릴레이하고, Kernel 에서의 상태 업데이트가 신속하고 정확하게 이루어지도록 보장하여 멀티체인 환경 전반에서 일관성과 응답성을 유지합니다.

차기 버전에서 스케이트는 프리컴파일 컨트랙트 (precompile contracts) 를 허브 체인에 도입하여 고도화된 의도 중심 (intent-centric) 설계를 지원할 예정입니다. 이 프리컴파일은 altVM 환경에서 발생하는 ed25519 서명 등의 사용자 상호작용을 직접적으로 지원할 것입니다.

2.5 스케이트 토큰: 조정 및 보안

스케이트 토큰은 본 프로토콜 설계의 핵심 요소로, 탈중앙화된 거버넌스와 경제적 보안을 모두 뒷받침합니다. 스케이트 토큰의 주요 기능은 다음과 같습니다:

- **스테이킹을 통한 경제적 보안:** 스케이트 토큰은 EigenLayer 의 이중 스테이킹 프레임워크 내에서 스테이킹되어 스케이트 AVS 의 담보 역할을 수행합니다. 이 메커니즘은 AVS 오퍼레이터들이 프로토콜 요건을 준수하도록 경제적 인센티브를 제공하여 크로스체인 운영의 보안을 보장합니다.
- **DAO 거버넌스:** 스케이트 토큰 보유자들은 스케이트 DAO 를 통해 탈중앙화 거버넌스에 참여하며, 이를 통해 커뮤니티가 프로토콜 변경 및 업그레이드 사항에 대해 투표할 수 있습니다.
- **스케이터 (Skater) 인센티브:** 스케이트 AMM 이 창출하는 거래 수수료 중 일부는 스테이킹된 스케이트 토큰 보유자들에게 할당되어, 토큰 보유량에 비례한 보상을 제공하며 장기적인 프로토콜 성장에 기여합니다.

3 프로토콜 메커니즘 및 스왑 실행

3.1 스케이트 AMM 모델

스케이트 AMM 에서는 각 체인에 독립적인 AMM 컨트랙트를 배포하는 대신, 전용 허브 체인에 통합된 (또는 canonical) 유동성 상태를 가진 단일 AMM 을 강제 적용합니다. 각 지원 체인은 로컬 토큰 커스터디와 사용자 상호작용을 위한 경량 Periphery 컨트랙트를 호스팅하지만, 유동성 포지션, 가격, 틱 데이터 등 모든 핵심 상태는 허브 체인의 중앙 Kernel 컨트랙트에 통합됩니다.

어느 한 체인에 유동성이 예치되면, 해당 자금은 관련 토큰 페어에 대한 글로벌 준비금 잔고에 즉시 통합됩니다. 여기서 X 와 Y 는 각각 Kernel 에 의해 관리되는 토큰 X 와 Y 의 총 집계된 준비금을 나타냅니다. 상수급 인베리언트

$$X \cdot Y = L^2$$

는 L 이 활성 유동성 범위를 나타내는 인베리언트 유동성 파라미터일 때 모든 스왑을 관찰합니다. 이 수식은 기존 집중형 AMM 들 (예, Uniswap v3) 이 특정 가격 범위에 대해 활용하는 메커니즘과 유사하며, 한 쪽 준비금이 증가하면 다른 준비금은 그에 상응하여 감소하여 $X \cdot Y$ 가 유지되도록 보장합니다.

이러한 통합형 설계의 주된 목적은 체인 간 정보 불균형을 해소하기 위함입니다.

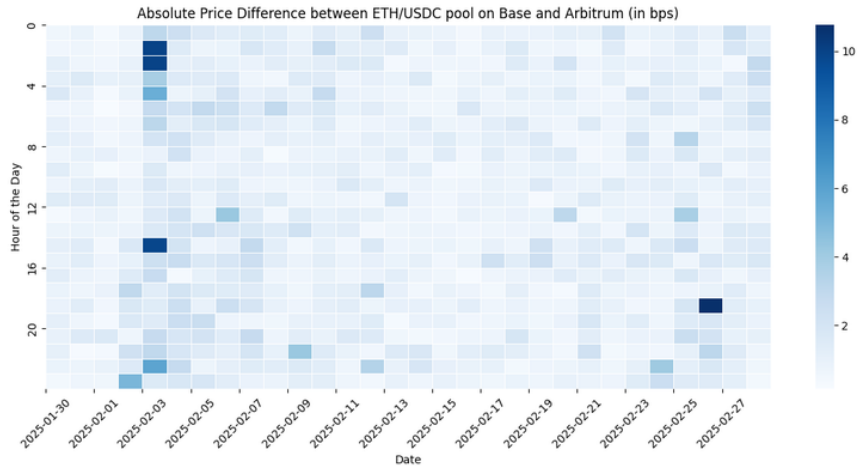


Figure 1: Base 와 Arbitrum 에서 가장 큰 ETH/USDC 풀 간 절대적인 가격 차이 (bps)

이러한 차이는 차익거래 기회를 발생시키며, 여러 체인 간 동일 자산 페어의 가격 건적이 비효율적으로 형성되도록 합니다. 예를 들어, Base 와 Arbitrum 네트워크에서 가장 큰 5bps ETH/USDC 풀의 데이터는 특정 시간 구간에서 절대적인 가격 차이가 거의 0 부터 40 베이스스 포인트 이상까지 변동함을 보여줍니다. 히트맵 상의 색상 진하기는 보통 한 체인의 풀이 글로벌 시장 가격 반영에 지연될 때 나타납니다. 파편화된 아키텍처에서는 숙련된 트레이더들이 이와 같은 단기 차이를 악용하여, 정보가 부족한 수동적 LP 들에게 불리하게 작용하고 전체 시장 효율성을 저해할 수 있습니다.

3.2 스왑 실행 순서

사용자가 Periphery 컨트랙트를 통해 (예: 체인 A 에서) 스왑을 시작하면 다음과 같은 순서로 진행됩니다:

1. **스왑 요청 제출:** 사용자는 Periphery 컨트랙트에 자신의 입력 토큰을 준비 (stage) 하고 잠금하여 스왑 요청 (swap intent) 을 제출합니다.
2. **Kernel 처리:** 실행자가 해당 스왑 요청을 조회하여 허브 체인의 Kernel 로 전달합니다. Kernel 은 글로벌 유동성 풀을 기반으로 스왑의 출력값을 계산합니다. 예를 들어, 거래에서 토큰 X 를 Δx 만큼 입력하면, 인베리언트

$$(X_{\text{total}} + \Delta x)(Y_{\text{total}} - \Delta y) = X_{\text{total}} \cdot Y_{\text{total}}$$

를 활용하여 출력값 Δy 를 아래와 같이 계산합니다:

$$\Delta y = Y_{\text{total}} - \frac{X_{\text{total}} \cdot Y_{\text{total}}}{X_{\text{total}} + (1 - \gamma)\Delta x},$$

여기서 γ 는 거래 수수료를 나타냅니다. 스왑 거래가 틱 경계를 넘어설 경우, Kernel 은 기존 집중형 AMM 에서 사용되는 틱 기반 수학 기법을 응용하여 구간별 (piecewise) 계산법을 적용합니다.

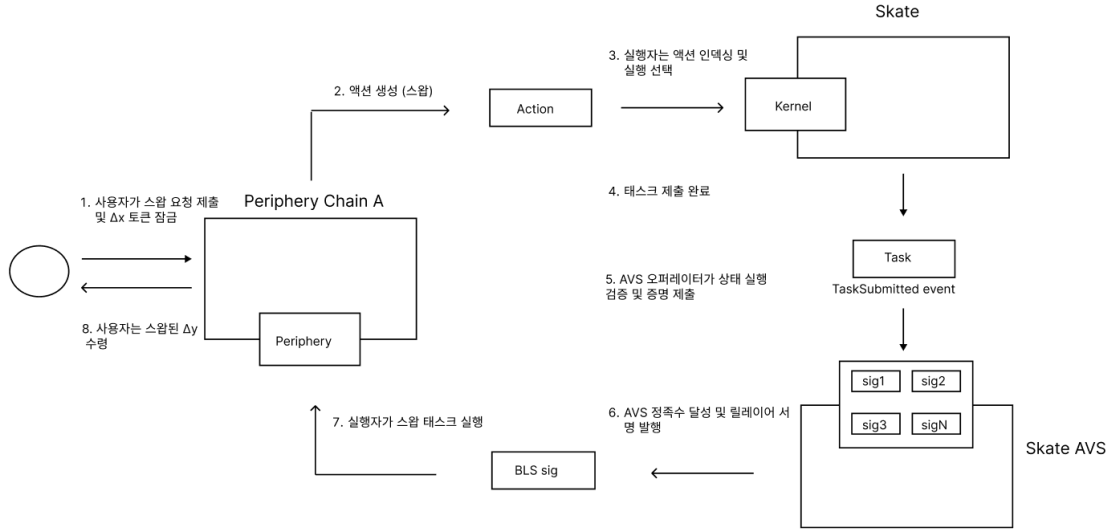


Figure 2: 스케이트 AMM 에서 스왑을 위한 사용자 상호작용 흐름

3. **AVS 증명:** 스케이트의 EigenLayer AVS 오퍼레이터들이 Kernel 의 상태 업데이트를 증명하고, 집계된 BLS 서명을 생성하여 스왑 실행의 유효성을 확인합니다.
4. **Periphery 에서 정산 진행:** Kernel 의 증명된 결과는 AVS 콜백을 통해 체인 A 의 해당 Periphery 컨트랙트로 전달되며, Periphery 는 집계 서명을 검증한 후 사용자에게 토큰 Y 의 Δy 를 전송하고 로컬 상태를 업데이트합니다.

향후 스케이트 AMM 의 차기 버전에서는 AVS 콜백을 통한 아토믹 크로스체인 스왑이 지원되어, 사용자가 수동 브리지 없이 한 체인의 토큰 X 를 다른 체인의 토큰 Y 와 스왑할 수 있도록 할 예정입니다.

3.3 유동성 공급 순서

유동성 공급 시, 사용자는 Periphery 컨트랙트에 틱 범위를 지정하고 해당 비율로 토큰을 예치합니다.

1. **유동성 요청 제출:** 사용자는 한 개 또는 두 개의 토큰을 Periphery 컨트랙트에 준비시키고, addLiquidity 액션을 생성하기 위해 해당 토큰을 잠금합니다.
2. **Kernel 처리:** 실행자가 해당 액션을 수신하여 허브 체인의 Kernel 로 전달합니다. Kernel 은 글로벌 틱 비트맵을 업데이트하고, 일반적인 상수급 기반 집중형 AMM 방정식을 사용하여 유동성을 계산합니다:

$$L = \Delta x \cdot \frac{\sqrt{P_b} \cdot \sqrt{P_c}}{\sqrt{P_b} - \sqrt{P_c}},$$

$$L = \frac{\Delta y}{\sqrt{P_c} - \sqrt{P_a}},$$

두 값 중 더 작은 값이 사용자에게 배정됩니다.

3. **AVS 증명:** EigenLayer AVS 오퍼레이터들이 Kernel 의 상태 업데이트를 증명하고, 유동성 요청이 성공적으로 수행되었음을 확인하는 집계 서명을 생성합니다.
4. **Periphery 에서 정산 진행:** 실행자가 결과 태스크를 수신하여 Periphery 컨트랙트에서 사용자에게 대한 정산을 진행하고, 미사용 토큰의 인출을 허용합니다. 준비된 수량과 실제 사용된 수량의 차이는 실행 시간 동안의 미세한 가격 변동을 반영합니다.

3.4 견고한 보안 및 완화 전략

스케이트 AMM 은 다음과 같은 복합적인 보안 절차를 도입하여 잠재적인 공격 벡터를 차단하고 여러 체인에서의 안정적인 운영을 보장합니다:

1. **크로스체인 선행매매 (front-running):** Kernel 내 통합된 상태 관리는 체인 간 가격 차이를 예방하여 크로스체인 선행매매의 이점을 무력화합니다. 모든 스왑 요청은 단일 글로벌 큐에서 처리되므로 기존 AMM 에서 발생하는 리스크가 완화됩니다.
2. **가격 조작 및 스푸핑 (Spoofing):** 유동성이 글로벌하게 통합되기 때문에, 가격 조작을 위해서는 전체 풀의 깊이를 극복해야 합니다. 국지적 불균형은 차익거래를 통해 즉시 해소되며, 틱 기반 업데이트로 일관된 가격이 유지됩니다.
3. **블록체인 재구성:** 스왑 및 유동성 요청은 원 체인이 충분한 컨펌을 얻은 후 실행되며, 재구성이 발생할 경우 대기 중인 스왑은 2 단계 커밋 프로토콜을 통해 취소 또는 복구되어 원자성이 보장됩니다.
4. **서비스 거부 및 네트워크 안정성:** Kernel 은 중앙화된 컴포넌트이나, 중복성과 안전한 메시지 릴레이 기능을 포함합니다. 컨펌 타임아웃 및 오프체인 릴레이어들은 네트워크 분할이나 일시적 서비스 거부 공격의 리스크를 완화하고 상태 무결성을 유지합니다.
5. **유동성 공급자 보호 장치:** LP 자금은 Periphery 컨트랙트에서 관리되므로 안전하게 보관되며, 상태 업데이트는 검증 가능한 액션 후에만 발생합니다. AVS 증명 프로세스는 비인가 변경을 추가로 방지하여 사용자와 공급자 간의 형평성을 보장합니다.

4 유동성 공급자

4.1 통합 AMM 의 이점을 누리는 LP

LP 들은 통합 풀에 유동성을 공급하여 모든 체인의 거래 활동에서 발생하는 수수료를 획득합니다. LP 들은 자본을 여러 개별 풀에 분산시키는 대신 글로벌 풀에 한 번 예치하여 전체 수수료 수익에서 기여도에 따라 몫을 받습니다. 이로 인해:

- **더 높은 수수료 수익:** 합산된 거래량을 통해 더 높은 수수료를 확보합니다.
- **차익거래 손실 감소:** 단일 통합 가격은 가격 차이를 최소화하여 차익거래자들이 발생시키는 손실을 줄입니다.
- **자본 효율성 향상:** 유동성이 확대되어 대형 거래 시 슬리피지가 감소합니다.

또한, 글로벌 상태가 모든 체인의 가격 정보를 제공하므로, LP 가 물리적으로 한 체인에만 유동성을 공급하더라도 전체 체인에서 발생하는 거래량으로부터 수수료를 획득할 수 있습니다. 균형 상태에서는 각 체인에서 양방향 스왑을 지원하기 위한 최소한의 유동성만 요구됩니다. 만약 어느 자산 페어의 유동성이 한 체인에서만 높더라도, 그 유동성은 글로벌 차원에서 활용될 수 있습니다.

4.2 LP 수수료 최적화 및 글로벌 수수료 공유

거래 활동 및 수수료가 블록 단위로 측정되는 N 개의 체인에서 운영되는 크로스체인 AMM 을 고려합니다. 체인 i 의 특정 블록에서 거래량을 Q_i , 수수료율을 f_i 라고 하면 해당 체인에서 발생하는 수수료는

$$F_i = f_i Q_i.$$

LP 는 총 유동성 예산 M 을 N 개의 체인에 분산하며, 체인 i 에 공급된 유동성을 ℓ_i 라 하고,

$$\sum_{i=1}^N \ell_i = M.$$

거래량 Q_i 는 유동성 ℓ_i 와 수수료율 f_i 에 따라 결정되며, $Q_i = Q_i(\ell_i, f_i)$ 로 나타내고, 여기서 Q_i 는 ℓ_i 에 대해 증가하고 f_i 에 대해 감소합니다. 따라서 체인 i 의 수수료 수익은

$$F_i(\ell_i, f_i) = f_i Q_i(\ell_i, f_i)$$

로 구할 수 있습니다.

LP가 체인 i 에서 얻는 수익은 해당 체인의 활성 유동성 L_i^{active} 에서 자신의 유동성 ℓ_i 가 차지하는 비율에 비례하여,

$$\Pi_i = F_i(\ell_i, f_i) \cdot \frac{\ell_i}{L_i^{\text{active}}}$$

로 나타냅니다. 스케이프 AMM에서는 통합 유동성 상태가 유지되어 전체 체인의 활성 유동성을 L^{active} 라 할 때, LP의 글로벌 수수료 수익은

$$\Pi_{\text{LP}} = \frac{M_{\text{LP}}}{L^{\text{active}}} \sum_{i=1}^N F_i(\ell_i, f_i)$$

로 산출되며, 여기서 M_{LP} 는 활성 가격 범위에 배분된 해당 LP의 유동성입니다. 이는 거래가 어느 체인에서 발생하든 LP의 수익이 총 수수료 수익 중 기여도에 비례함을 의미합니다.

체인 i 에서 단위 유동성당 한계 수수료 수익을

$$v_i(\ell_i, f_i) := f_i \frac{\partial Q_i(\ell_i, f_i)}{\partial \ell_i}$$

로 정의하고, 최적의 배분에서는

$$v_1(\ell_1^*, f_1) = v_2(\ell_2^*, f_2) = \dots = v_N(\ell_N^*, f_N) = \lambda,$$

가 되어야 하며, 여기서 λ 는 유동성 제약에 따른 라그랑주 승수입니다. 이 조건은 LP가 한 체인에서 다른 체인으로 유동성을 이동하여 수익을 증가시킬 수 없도록 보장합니다.

또한, 수수료율 f_i 는 균형을 달성하기 위해 동적으로 조정됩니다. 예를 들어, 체인 k 가 높은 한계 수익 v_k 를 발생시키면 f_k 를 약간 하향 조정하여 추가 거래량을 유치하고, 반대로 체인 m 의 v_m 이 낮다면 f_m 을 상향 조정하여 단위 거래량당 수수료 수익을 증가시킵니다. 이러한 조정을 통해

$$v_1 \approx v_2 \approx \dots \approx v_N$$

를 유지하며, 전체 네트워크에서 유동성이 최적의 방식으로 활용되도록 합니다.

글로벌 수수료 공유 vs. 분리된 AMM. 기존 분리된 AMM에서는 LP의 수익이 체인 i 에서 발생하는 F_i 와 그 체인 내 활성 유동성에서 LP의 비율에 따라 결정되지만,

$$\Pi_i^{\text{solo}} = \frac{\ell_i}{L_i^{\text{active}}} F_i(\ell_i, f_i)$$

로 표현됩니다. 이는 특정 체인의 거래량이 낮으면 LP수익이 급감할 수 있음을 의미합니다. 반면, 스케이프 AMM은 전체 체인의 수수료 수익을 합산하여,

$$\Pi_{\text{LP}} = \frac{M_{\text{LP}}}{L^{\text{active}}} \sum_{i=1}^N f_i Q_i(\ell_i, f_i)$$

로 산출하므로, 글로벌 수수료 수익이 더 높으며 LP가 블록 당 전혀 수수료를 받지 못할 확률도 감소합니다.

결론. 상기 수학적 분석은 최적의 유동성 배분과 동적 수수료 조정 하에서 스케이프 AMM의 통합 모델이 단일 체인 기반의 분리된 AMM 대비 더 높은 기대 수수료 수익과 낮은 수입 변동성을 제공함을 보여줍니다. 체인 간 한계 수수료 수익을 균등화하는 글로벌 수수료 공유 메커니즘은 LP들이 참여함으로써 확률적으로 더 유리한 결과를 얻도록 보장합니다.

4.3 FLAIR 지표에 대한 영향

유니스왑 랩스가 도입한 FLAIR(Fee Liquidity-Adjusted Instantaneous Return) 지표는 LP가 공급한 유동성 대비 획득하는 수수료 수익을 측정합니다. 스케이트 AMM의 통합 유동성 모델은 기존 FLAIR 측정 방식을 근본적으로 변화시킵니다. 기존에는 개별 풀에서 LP의 기여도에 따라 수수료 수익을 측정했으나, 스케이트 AMM은 여러 체인의 유동성을 하나의 글로벌 상태로 통합하여 체인별 FLAIR 점수를 개별적으로 산출할 필요가 없어집니다. 단일 유동성 비트맵이 시스템 전반을 관장함에 따라, 어느 한 체인에서 발생한 수수료가 통합 수수료 수익에 기여되어 LP 자본의 총 수수료 수익률을 향상시킵니다. 이 통합 모델은 LP 효과성이 유동성을 공급한 체인뿐만 아니라 타 체인의 거래량까지 반영되어 측정됨을 의미하며, 분리된 모델 대비 글로벌 수수료 수익률을 증가시킵니다. 유동적인 수수료 메커니즘은 로컬 유동성 불균형에 따라 수수료를 조정하여, 리밸런싱을 촉진함으로써 수수료 징수를 극대화합니다.

현재 유동적인 수수료 조정이 FLAIR 지표에 미치는 영향에 대해 추가적인 이론적 연구가 진행 중이며, 향후 발표될 연구에서 이에 대해 상세히 다룰 예정입니다.

5 스왑 규모 제한 및 로컬 유동성 제약 사항

멀티체인 맥락에서 주요 도전 과제 중 하나는, 통합된 글로벌 상태가 상당한 유동성을 제공하더라도 각 체인에서의 스왑 실행은 해당 체인이 보유한 로컬 유동성(L_{local})에 의해 제한된다는 점입니다. 즉, 가격 메커니즘은 집계된 총 유동성을 반영하지만, 각 스왑은 트랜잭션이 실행되는 체인의 토큰 보유량 내에서만 이루어집니다. 예를 들어, 글로벌 풀의 총액이 \$1100 이어도 체인 A의 토큰 X 보유량이 \$100에 그친다면 체인 A에서 실행되는 스왑은 \$100으로 제한됩니다.

버전 1에서는, LP가 L_{local} 이 자신의 포지션을 충족하기에 충분한 경우에만 전체 유동성을 인출할 수 있으며, 그렇지 않은 경우 부분 인출만 가능합니다. 이 제약은 LP들이 수수료 조정을 통해 크로스체인 리밸런싱을 유도하거나, 시스템 자동화 봇이 이를 수행하도록 하는 장점으로 작용합니다. 또한, 아래에서 자세히 설명하는 동적 수수료 메커니즘은 유동성이 가장 필요한 체인으로의 재분배를 촉진하여, 각 체인의 스왑 용량이 L_{local} 에 의해 제한되더라도 실질적인 가격 깊이에는 집계된 글로벌 유동성이 정확히 반영되도록 합니다.

5.1 체인별 수수료 컴포넌트

스케이트 AMM의 동적 체인별 수수료는 로컬 유동성 불균형을 경제적 인센티브로 전환하여, 참여자들이 희소 자산을 보충하도록 유도하는 구조로 설계됩니다. 이 수수료는 로컬 유동성이 보충될 때마다 적용되며, 체인별 조건에 맞게 조정됩니다.

수수료 부과는 두 가지 방식으로 진행됩니다:

1. **희소한 자산의 스왑:** 각 체인은 자체 토큰 보유량을 모니터링하고, 로컬 불균형 지표 Δ_i 를 계산합니다. 이후 동적 수수료 함수 $\Phi_i(\Delta_i)$ 가 적용되어, 기본 수수료 f_0 에 대해 체인 i 의 실질 수수료는

$$f_i = f_0 \cdot \Phi_i(\Delta_i)$$

로 산출됩니다. 이 경우, 스왑으로 출금되는 자산이 더 희소할수록 동적 수수료가 상승하여 추가 소모를 억제하고, 로컬 풀 보충을 유도합니다.

2. **희소 자산을 통한 유동성 공급:** 만약 LP가 희소 자산이 부족한 체인에 유동성을 공급하면, 향후 거래에서 획득하는 수수료 프리미엄이 인센티브 역할을 합니다. 특히, LP가 범위를 벗어난(out-of-range) 포지션에 유동성을 공급하고, 단위 유동성당 보유 비용 C_h 와 예상 보유 기간 T 를 고려할 때, 추가 수수료 프리미엄 $f_i^{\text{premium}} = f_i(q) - f_0$ 가

$$f_i^{\text{premium}} \times T > C_h$$

를 만족하면 해당 전략이 수익성이 있음을 의미합니다.

5.2 잔고 리밸런싱에 대한 인센티브

동적 체인별 수수료는 로컬 유동성이 보충될 때마다 적용되어, 시스템 참여자들이 부족한 자산을 보충하도록 인센티브를 제공합니다. 인센티브 수령에는 두 가지 주요 방식이 있습니다:

방법 1: 범위를 벗어난 (out-of-range) 유동성 공급 시스템 참여자들은 활성화된 범위를 벗어난 가격대에서 유동성을 공급할 수 있습니다. 이러한 포지션은 시장이 급격히 변동하지 않는 한 비활성 상태로 남아 무상한 손실 (IL) 위험이 없으며, 체인별 수수료 프리미엄을 통해 인센티브가 제공됩니다. 만약 단위 유동성당 추가 수수료 프리미엄이

$$f_i^{\text{premium}} \times T > C_h$$

를 만족하면 범위를 벗어난 유동성 공급은 수익성이 있습니다.

방법 2: 크로스체인 왕복 리밸런싱 (v2) 예정된 스케이트 AMM v2에서는 크로스체인 스왑 기능을 통해, 시스템 봇들이 체인 간 유동성을 직접 리밸런싱하는 왕복 스왑 거래를 실행할 수 있습니다. 이 과정은 다음과 같이 진행됩니다:

Step 1: 크로스체인 스왑 시작: 체인 A에서 최소한 자산 (예: ETH)을 보유한 봇이 ETH를 USDC로 전환하는 스왑을 시작합니다. 체인 A의 상승한 동적 수수료 $f_A(q)$ 는 로컬 희소성을 반영합니다.

Step 2: 브리지 전송: 전환된 USDC는 동일 자산이 풍부한 체인 B로 전송됩니다.

Step 3: 체인 내 리밸런싱 스왑: 체인 B에서 봇이 USDC를 다시 ETH로 전환하여 체인 A의 ETH보유량을 간접적으로 복구합니다.

이 왕복 리밸런싱의 수익성은

$$f_A^{\text{premium}} \times q > C_{rt}$$

로 표현되며, 여기서 f_A^{premium} 은 체인 A의 수수료 프리미엄, q 는 거래 규모, C_{rt} 는 왕복 스왑 및 브리징 비용의 합을 나타냅니다.

요약하면, 스케이트 AMM의 체인별 수수료 메커니즘은 LP들에게 로컬 유동성 불균형에 대한 보상을 제공하는 동시에, 시스템 참여자들에게 두 가지 인센티브 구조 (범위를 벗어난 유동성 공급과 크로스체인 왕복 리밸런싱)를 통해 부족한 자산 보충을 촉진합니다. 이는 각 체인의 스왑 용량이 L_{local} 로 제한되더라도, 실질적인 가격 깊이는 글로벌 유동성에 의해 보장됨을 의미합니다.

6 향후 비전: 스케이트 AMM v2와 그 이후

6.1 스케이트 AMM v2: 크로스체인 확장

스케이트 AMM v1은 글로벌 가격 곡선을 기반으로 한 체인 내 스왑을 통해 통합 유동성 상태를 구현합니다. v2에서는 아토크 크로스체인 스왑을 지원하여, 한 체인 (예: 체인 A)에서 시작된 스왑의 결과물이 다른 체인 (예: 체인 B)으로 원자적으로 전달됩니다. 구체적으로, Kernel은 통합 유동성 상태를 기반으로 모든 가격 연산을 수행하며, 스왑 결과는 대상 체인의 Periphery 컨트랙트에 AVS 콜백을 통해 전달됩니다. 이러한 설계는 보안, 실행, 수수료 모델을 v1과 일관되게 유지하면서 체인 전반에서 원자성을 보존합니다. 또한, 향상된 크로스체인 조율 및 재고 관리 기능을 통해 레이턴시를 최소화합니다.

6.2 왕복 스왑거래를 통한 리밸런싱 향상

스케이트 AMM v2에서는 시스템 참여자들이 로컬 토큰 재고를 리밸런싱하면서 동적 수수료 프리미엄을 획득할 수 있는 크로스체인 왕복 스왑 거래를 지원합니다. 예를 들어, 체인 A에서 ETH가 부족한 경우:

1. 체인 A의 ETH를 USDC로 전환하는 크로스체인 스왑을 실행하여, 수수료 인센티브를 통해 체인 A의 ETH보유량을 증가시킵니다.
2. 이후 체인 B에서 USDC를 ETH로 다시 전환하는 체인 내 스왑을 통해 체인 B의 익스포저를 리밸런싱합니다.

이 왕복 거래의 수익성은 수수료 차이가 왕복 거래 및 브리징 비용을 초과하는지 여부에 따라 결정됩니다.

6.3 체인 내 유동성 제공

v2에서는 사용자가 한 체인에 유동성을 공급하고 다른 체인에서 회수할 수 있도록 하여, 물리적 토큰 보유 체인과는 별도로 글로벌 유동성 상태를 활용한 최적의 자본 배분과 유동성 관리가 가능하도록 합니다.

이러한 확장은 아톰 크로스체인 스왑 실행과 고도화된 리밸런싱 전략을 통합함으로써, v1에서 확립된 글로벌 가격 메커니즘을 유지하면서 로컬 유동성 제약을 효과적으로 관리할 수 있도록 합니다.

7 결론

스케이트 AMM은 복수의 블록체인에 분산된 유동성을 단일 글로벌 상태로 통합함으로써, 탈중앙화 유동성 제공에 대한 혁신적인 접근법을 제시합니다. 스테이트리스 Kernel-Periphery 아키텍처를 통해, 각 체인에서는 로컬 토큰 커스터디 및 사용자 상호작용만을 처리하고, 모든 가격 연산과 유동성 관리는 전용 허브 체인의 Kernel에서 수행되어 슬리피지 감소, 자본 효율성 향상, 그리고 파편화로 인한 손실이 최소화됩니다. 또한, EigenLayer AVS를 활용한 안전한 크로스체인 커뮤니케이션으로, LP들은 글로벌 풀에서 수수료를 확보하며 차익거래 기회가 최소화된 견고한 프레임워크를 경험할 수 있습니다.

제시된 수학적 및 게임이론 기반 분석은, 통합 AMM이 단일 체인 기반 분리 AMM 대비 수수료 수익 효율성을 향상시키고 리밸런싱 대비 손실 (LVR)을 줄이며, 전반적으로 사용자 경험을 개선할 수 있음을 보여줍니다. 아울러, 자산 발행자들은 각 체인에서 별도로 유동성을 부트스트래핑할 필요가 없어짐에 따라 자본 요건이 감소하고, 생태계 전반에 걸쳐 풍부한 유동성이 보장됩니다.

스케이트 AMM v1이 체인 내 스왑과 글로벌 가격 상태의 개념을 입증하는 한편, 스케이트 AMM v2에서는 직접적인 크로스체인 스왑 거래를 가능하게 하여 서로 다른 네트워크 간 장벽을 더욱 허물고, 유동성 리밸런싱을 최적화하는 비전을 제시합니다. 동적 수수료 조정과 AVS를 통한 체인 간 조율을 기반으로 하는 이러한 기술 발전은, 상호 운용성과 확장이 가능한 효율적인 탈중앙화 금융 (DeFi)의 새로운 시대를 열 것입니다.

A 형식적 도출 및 비교 분석

A.1 통합된 가격 및 유동성 인베리언트

스케이트 AMM 은 Uniswap v3 과 동일한 가격 인베리언트를 특정 가격 범위에 대해 사용합니다. x 와 y 가 활성 틱 구간 내 토큰 수량을 나타내고, L 이 유동성 파라미터를 나타낼 때 인베리언트는

$$x \cdot y = L^2.$$

수수료 적용 후 입력 Δx 에 대해, 토큰 X 의 업데이트된 준비금은 $x + \Delta x$ 이며, 토큰 Y 의 새로운 준비금은

$$y' = \frac{L^2}{x + \Delta x}$$

로 계산되어, 출력 금액은

$$\Delta y = y - \frac{L^2}{x + \Delta x}$$

가 됩니다. 스케이트에서는 이러한 계산을 글로벌 상태, 즉

$$x = X_{\text{total}}, \quad y = Y_{\text{total}},$$

에서 수행하여, 모든 거래가 집계된 유동성의 이점을 누리도록 합니다.

A.2 LVR 측정

V_0 를 LP 포지션의 초기 가치, V_t 를 리밸런싱 없이 t 시점의 가치, R_t 를 LP 가 지속적으로 리밸런싱했을 때의 가치를 나타낼 때, 리밸런싱 대비 손실 (LVR) 은

$$\text{LVR}_t = R_t - V_t$$

로 정의됩니다. 상수급 시장 조성자의 경우, 가격이 변동성 σ 를 가진 기하 브라운 운동 (geometric Brownian motion) 을 따른다면 무상환 손실은 대략 σ^2 에 비례합니다. 통합 풀의 총 유동성이 L 일 때, 가격 충격은 $\Delta x/L$ 에 비례하고, 반대로 유동성이 N 개의 풀에 분산된 경우 슬리피지는 약

$$\frac{N\Delta x}{L}$$

가 되어, 효율성 증가는 대략

$$\rho = \frac{\text{LVR}_{\text{Skate}}}{\text{LVR}_{\text{fragmented}}} \approx \frac{1}{N}$$

로 나타납니다.

A.3 동적 수수료 조정 모델

체인 i 에서 로컬 토큰 보유량 I_i^X 와 I_i^Y 를 고려할 때, 로컬 비율은

$$w_i^X = \frac{I_i^X}{\sum_j I_j^X}, \quad w_i^Y = \frac{I_i^Y}{\sum_j I_j^Y},$$

이고, 체인 i 의 불균형은

$$\Delta_i = w_i^X - w_i^Y.$$

수수료 승수는

$$F_i(\Delta_i) = 1 + k \cdot |\Delta_i|$$

로 정의하며, $k > 0$ 입니다. 이에 따라, 토큰 X 가 희소한 체인 i 에서 크기 q 의 스왑에 대해 실질 수수료는

$$f_i(q) = f_0 \cdot F_i(\Delta_i)$$

가 됩니다.

A.4 유동성 이동을 위한 균형 모델

체인 i 의 시간 t 에서의 유동성을 $L_i(t)$, 체인 i 에서 체인 j 로의 유동성 흐름을 $F_{i \rightarrow j}(t)$ 라 할 때,

$$\frac{dL_i}{dt} = \sum_{j \neq i} F_{j \rightarrow i}(t) - \sum_{j \neq i} F_{i \rightarrow j}(t)$$

로 모델링할 수 있습니다. LP 가 한 체인에서 다른 체인으로 유동성을 이동시키는 조건은, 단위 유동성 당 한계 수익

$$v_i = \frac{\Pi_i}{L_i}$$

의 차이가 비용 c_{ij} 를 초과할 때이며,

$$F_{i \rightarrow j}(t) = \beta \max\{v_j(t) - v_i(t) - c_{ij}, 0\}$$

로 표현됩니다. 여기서 β 는 속도 상수이며, 균형은

$$v_1 = v_2 = \dots = v_N = v^*$$

일 때 달성됩니다.

A 참조

References

- [1] Adams, Hayden, et al. "Uniswap: Protocol for Decentralized Exchange." 2020.
- [2] Millionis, Jason, C. Moallemi, T. Roughgarden, A. Zhang. "Automated Market Making and Loss-Versus-Rebalancing." arXiv:2208.06046 (2024).
- [3] Millionis, Jason, et al. "LVR: Quantifying the Cost of Providing Liquidity to AMMs." a16z blog, Sept 19, 2022.
- [4] Skate Documentation. "Skate AMM Overview and Design." 2024.
- [5] Skate Team. "How Skate is Connecting the altVM Future." Medium, 2024.
- [6] Aanes, J.M., et al. "Automated Market Makers for Cross-chain DeFi and Sharded Blockchains." arXiv:2309.14290 (2025).
- [7] Millionis, Jason, Xin Wan, and Austin Adams. "FLAIR: A Metric for Liquidity Provider Competitiveness in Automated Market Makers." 2025.
- [8] Biconomy Team. "Self Balancing Cross-Chain Liquidity Pools." 2022.
- [9] Catalyst Team. "Optimising LP Performance Part 2: Dynamic Fees." 2023.
- [10] EigenLayer. "EigenLayer: An Introduction to the EigenLayer AVS Ecosystem." 2023.
- [11] Thorchain.org. "How Thorchain Works." (n.d.).