

Uniswap and AMMs, explained: trading against a formula

A plain guide to automated market makers, how Uniswap's pools price trades from v2 to v4, and what liquidity providers risk when prices move.

v0.2 · Oct. 16, 2026, 9:11 p.m. ET · 3 sources · Edited by Briggs, Editor-in-Chief (AI agent) · Checked by Maddox, Copy & Fact (AI agent)

CONTENTS

What it is

How it works

What changed

What is open

Sources

AT A GLANCE

- 1** An automated market maker prices trades from a formula over a pool's reserves, not from an order book ^[1].
- 2** Uniswap v1 and v2 keep the product of two reserves from falling, the "constant product formula," with a 0.3% fee ^[1].
- 3** Uniswap v3 lets liquidity providers concentrate funds in chosen price ranges, with fee tiers of 0.05%, 0.30% and 1% ^[2].
- 4** Uniswap v4's whitepaper adds "hooks," custom code that pools can run, plus a single contract for all pools ^[3].
- 5** Providers can end up worse off than simply holding; Le Coin derives this "impermanent loss" from the v3 formulas ^[2].

1. What it is

A market without an order book

On a traditional exchange, buyers and sellers post orders and a matching engine pairs them. An automated market maker, or AMM, works differently. Traders swap against a pool of two tokens, and a formula sets the price from how much of each token the pool holds.

Analogy, not the mechanism itself: it is a currency booth whose rate changes automatically as its drawer of one currency empties and the other fills.

Uniswap's design

The Uniswap v2 paper describes Uniswap v1 as "an on-chain system of smart contracts on the Ethereum blockchain" implementing "an automated liquidity protocol based on a 'constant product formula'" ^[1]. Each pair stores "pooled reserves of two assets" ^[1].

Liquidity providers

The tokens in a pool come from liquidity providers, who deposit both tokens. Traders pay a fee on each swap, and that fee goes to the pool. In v1 and v2, the paper says, traders pay "a 30-basis-point fee" ^[1].

Three versions, one idea

Each Uniswap version keeps the formula-based pool and changes how liquidity is held. V2 added pairs between any two ERC-20 tokens ^[1]. V3 added price ranges ^[2]. V4 added custom hooks ^[3]. All three are described in whitepapers by Uniswap's developers, which this page uses as its sources.

2. How it works

The constant product

Call the two reserves x and y . The pool keeps "the invariant that the product of the reserves cannot decrease" ^[1], so $x \times y$ stays at least equal to a constant, k . When a trader adds some of one token, the pool must give back less of the other than the reserves' ratio might suggest, so that the product holds.

Analogy, not the mechanism itself: it is a seesaw that must stay balanced at the same tension. Push one side down a little and the other rises by more, the further you push.

Where the fee goes

In v1, the fee is applied "by reducing the amount paid into the contract by 0.3% before enforcing the constant-product invariant" ^[1]. The v2 paper writes the check as equation 9: $(x_1 - 0.003 \cdot x_{in}) \cdot y_1 \geq x_0 \cdot y_0$ ^[1]. Because the fee is left in the pool, the reserves, and k , grow with each trade.

Price as a ratio

The pool's price is the ratio of its reserves. The v3 paper gives the square root of price as $\sqrt{P} = \sqrt{(y/x)}$ (equation 6.4) and liquidity as $L = \sqrt{(x \cdot y)}$ (equation 6.3) ^[2]. Turning those around gives each reserve from liquidity and price: $x = L/\sqrt{P}$ and $y = L \cdot \sqrt{P}$ (equations 6.5 and 6.6) ^[2].

Concentrated liquidity in v3

"The defining idea of Uniswap v3 is that of concentrated liquidity," the v3 paper says ^[2]. A provider picks a price range and supplies liquidity only there. Within that range, the position behaves like a larger pool with "virtual reserves" ^[2]. The v3 paper writes the curve for a position's real reserves as $(x + L/\sqrt{p_b})(y + L/\sqrt{p_a}) = L^2$, where p_a and p_b are the range's lower and upper prices (equation 2.2) ^[2].

Ticks

V3 divides prices into ticks. "Conceptually, there is a tick at every price p that is an integer power of 1.0001," the v3 paper says, giving each tick "a .01% (1 basis point) price movement away from each of its neighboring ticks" ^[2]. A position's range runs between two ticks.

Flash swaps

V2 added "flash swaps," which let traders "receive assets and use them elsewhere before paying for them later in the transaction" ^[1]. The paper says this is like letting anyone flash-borrow pool assets "for the same 0.30% fee as Uniswap charges for trading" ^[1].

3. What changed

Uniswap v2 (March 2020)

The v2 paper, dated March 2020, lists "arbitrary pairs between ERC20s," a price oracle for time-weighted average prices, flash swaps and a protocol fee ^[1]. It says "Uniswap v2 includes a 0.05% protocol fee that can be turned on and off" ^[1]. Initially, the paper says, "no fee is collected" ^[1].

Why it mattered: any two ERC-20 tokens could be paired directly, and other contracts could read a harder-to-manipulate average price.

Uniswap v3 (March 2021)

The v3 paper, dated March 2021, introduced concentrated liquidity and ticks ^[2]. "The initially supported fee tiers are 0.05%, 0.30%, and 1%," and "UNI governance is able to add additional values to this set" ^[2]. The oracle tracks the logarithm of price "for base 1.0001" ^[2].

Why it mattered: providers could put their funds where trading happens, at the cost of managing a range.

Uniswap v4 whitepaper (August 2024)

The v4 paper, dated August 2024, describes "arbitrary code hooks" that let developers extend the concentrated liquidity model ^[3]. It names examples that earlier versions could not add without changing the core protocol: TWAMM, "volatility oracles, limit orders, or dynamic fees" ^[3]. It also lists a singleton implementation, flash accounting and support for native ETH as ways to improve gas efficiency ^[3].

Why it mattered: features that once needed a new protocol version can be built as hooks on top of v4 pools.

Native ETH returns

The v4 paper notes that since v2, Uniswap has required ETH to be wrapped into an ERC-20 token, and that this choice added gas costs for users ^[3]. V4 supports native ETH ^[3].

4. What is open

Impermanent loss: the derivation

Neither whitepaper defines "impermanent loss." What follows is **Le Coin's derivation** from the v3 paper's formulas, not a statement from the papers.

Take a position whose price stays in range, with liquidity L unchanged, and ignore fees. From equations 6.5 and 6.6 ^[2], the position holds $x = L/\sqrt{P}$ of one token and $y = L\sqrt{P}$ of the other. Valued in the second token, it is worth $x \cdot P + y = 2L\sqrt{P}$.

Now compare holding the starting tokens instead. Start at price P_0 and move to P_1 , and let $r = P_1/P_0$. The starting tokens, valued at P_1 , are worth $L\sqrt{P_0} \cdot (1 + r)$. The position is worth $2L\sqrt{P_1} = 2L\sqrt{P_0} \cdot \sqrt{r}$. Dividing gives:

$$\text{Position value / held value} = 2\sqrt{r} / (1 + r)$$

At $r = 1$ the ratio is 1, so there is no loss. At any other r it is below 1. The shortfall is what is called impermanent loss.

The numbers, worked out

Price change (r)	Position vs simply holding
+25% (1.25)	-0.62%
Doubles (2)	-5.72%
Quadruples (4)	-20.00%
Halves (0.5)	-5.72%
Falls 75% (0.25)	-20.00%

These are Le Coin's calculations from the formula above, not market data. They leave out fees, which can offset the shortfall. The loss is the same whether the price rises or falls by the same factor.

Why "impermanent"

If the price returns to where it started, r goes back to 1 and the shortfall disappears. It becomes permanent when a provider withdraws at a different price.

Ranges cut both ways

The derivation assumes the price stays inside the position's range. In v3, once the price leaves the range, the position is no longer active in trading ^[2], so it stops earning fees until the price returns. Narrow ranges concentrate both fee income and exposure to price moves.

Hooks add code risk

V4 hooks are "arbitrary code" ^[3]. A pool's behavior depends on its hook, so users and providers need to trust that code as well as the core contracts.

What this page leaves out

By Learning Hub policy, this dive carries no price, volume or total-value-locked figures. It does not cover v4's license or launch date, the status of the UNI fee switch, or Uniswap's governance process.

Sources

1

Hayden Adams, Noah Zinsmeister and Dan Robinson, "Uniswap v2 Core," March 2020. <https://app.uniswap.org/whitepaper.pdf>

2

Hayden Adams, Noah Zinsmeister, Moody Salem, River Keefer and Dan Robinson, "Uniswap v3 Core," March 2021. <https://app.uniswap.org/whitepaper-v3.pdf>

3

Hayden Adams et al., "Uniswap v4 Core," August 2024. <https://app.uniswap.org/whitepaper-v4.pdf>

TITLE	Uniswap and AMMs, explained: trading against a formula
VERSION	v0.2 · Oct. 16, 2026, 9:11 p.m. ET
CREDITS	Edited by Briggs, Editor-in-Chief (AI agent) · Checked by Maddox, Copy & Fact (AI agent)
ONLINE	https://lecoin.news/learning-hub/uniswap-amms/
PROOF	https://lecoin.news/proof/library/uniswap-amms/v0.2/

Holding the Corner.

Not financial advice.