Performance and Security Comparison of Flash Loan Protocols in DeFi

CSC2125 Blockchain Technology and Engineering - Course Project

Tsz Him Shek
Department of Computer Science
University of Toronto
Toronto, ON, Canada
shekanson@cs.toronto.edu

Tien Han
Department of Computer Science
University of Toronto
Toronto, ON, Canada
tien.han@cs.toronto.edu

***Abstract*—This project conducts a comparative analysis of flash loan protocols across four Ethereum-based decentralized finance (DeFi) platforms: Aave, Balancer, Uniswap V2, and Uniswap V3. Flash loans, which enable collateral-free borrowing within a single transaction block, are employed in various DeFi strategies, such as arbitrage and self-liquidation. The evaluation focuses on key performance metrics, including deployment costs, contract sizes, gas consumption, and fee structures. The study also reviews security enhancements implemented by each protocol to mitigate potential risks. The results provide a clearer understanding of the operational and security characteristics of these protocols within the DeFi ecosystem.**

***Index Terms*—Flash Loans, Decentralized Finance (DeFi), Aave, Balancer, Uniswap, Gas Costs, Security Risks.**

I. Introduction

Flash loans have emerged as a key innovation in decentralized finance (DeFi), enabling users to borrow assets without collateral, as long as the loan is repaid within the same transaction block. This unique characteristic leverages the atomic nature of blockchain transactions, where the entire transaction is reverted if the loan is not repaid, ensuring no risk to the lender. The no-risk feature has enabled a range of strategies in DeFi, such as arbitrage, liquidity management, and self-liquidation. However, while flash loans offer significant opportunities for capital efficiency, they also expose platforms to security vulnerabilities, including price manipulation and front-running attacks.

This study presents a comparative analysis of flash loan protocols across four Ethereum-based DeFi platforms: Aave, Balancer, Uniswap V2, and Uniswap V3. The evaluation focuses on performance metrics including deployment costs, contract sizes, gas consumption, and associated fee structures. Data was collected from the flash-loans-comparison repository, which provides standardized test scripts and Solidity contract implementations for each protocol. Tests were conducted using Foundry, simulating the protocols on a forked mainnet environment to gather gas cost data and assess operational efficiency.

In addition to performance evaluation, the study reviews security measures implemented by each protocol to mitigate risks associated with flash loan exploits. This includes examining post-attack security enhancements, such as Aave's risk parameters and Balancer's use of multiple oracles. The results offer insights into the operational and security characteristics of these protocols and provide a clearer understanding of their respective strengths and weaknesses in the DeFi ecosystem.

II. Related Work

Flash loans have become a fundamental feature of decentralized finance (DeFi), enabling uncollateralized borrowing within a single transaction block. Initially introduced by the Marble platform in 2018 [1], flash loans were later adopted and expanded by major DeFi protocols like Aave [2], dYdX, and Uniswap. These loans allow users to engage in various financial strategies such as arbitrage and liquidity management, significantly enhancing the efficiency of decentralized markets.

The rapid adoption of flash loans has led to a growing body of literature exploring their implications, including both their potential for profit and associated risks. Studies such as those by Mandin [3] document the widespread use of flash loans in DeFi ecosystems, revealing their role in expanding liquidity access while also highlighting vulnerabilities. Flash loans have been exploited for malicious purposes, such as price manipulation, front-running, and governance attacks. For instance, Lewis et al. [4] demonstrated how flash loans were used to manipulate governance in the MakerDAO protocol, while Kaihua et al. [5] explored price manipulation strategies in decentralized exchanges (DEXs) through flash loans.

Security concerns regarding flash loans have driven efforts to develop mitigation strategies. Aave's implementation of risk parameters and Balancer's use of decentralized oracles represent two notable approaches to securing flash loan transactions. Additionally, research on detecting flash loan-based attacks, such as that by Chen et al. [6], has advanced the ability to identify malicious price manipulation in real time, contributing to the development of more secure DeFi environments.

Comparative studies, such as those by Alhaidari et al. [7], have assessed the performance of different flash loan protocols in terms of gas consumption and transaction efficiency. These

studies provide valuable insights into the trade-offs between performance and security across various protocols, an essential consideration for users and developers when selecting the most suitable platform for flash loan operations.

III. Protocol Overview

In this section, we provide an overview of the three flash loan protocols that are the focus of this project: Aave, dYdX, and Uniswap. These protocols are integral to the decentralized finance (DeFi) ecosystem, each offering unique features that cater to different use cases within the DeFi space.

A. Aave

Aave is a prominent decentralized finance (DeFi) protocol offering lending and borrowing services, which has become one of the largest in the DeFi space. One of Aave's key innovations is its open-source, decentralized governance model, where AAVE token holders control key protocol decisions, such as asset listings and risk parameters.

Aave V3 [8] introduces several advancements, most notably High Efficiency Mode (E-Mode), which allows users to maximize borrowing power when using highly correlated assets like stablecoins or ETH derivatives. Isolation Mode offers enhanced risk management by enabling specific assets to have isolated debt ceilings, limiting exposure while still allowing borrowing.

Aave also stands out with its aTokens, which automatically accrue interest for users, providing a seamless way to earn yield. Additionally, Portals allow liquidity to seamlessly flow between different blockchain networks, enabling cross-chain borrowing and lending without complex interactions.

Aave's focus on capital efficiency and security is reflected in its dynamic risk parameters, which are adjusted through community governance. These features, coupled with Aave's robust liquidity pools, position it as a leading protocol in the DeFi ecosystem, providing both flexibility and safety to users.

B. Balancer

Balancer [9] is a decentralized automated market maker (AMM) that provides liquidity and trading services within the DeFi ecosystem. Unlike traditional AMMs that offer fixed ratios of assets in liquidity pools, Balancer supports flexible multi-asset pools, where liquidity providers can deposit up to eight different tokens in varying proportions. This flexibility allows for more efficient use of capital and reduced price slippage, which sets Balancer apart from other AMMs like Uniswap.

A notable feature of Balancer is its Flash Loans, a service that allows users to borrow assets without collateral, provided the loan is repaid within the same transaction. This feature leverages the consolidated liquidity within Balancer's Vault, which pools liquidity from various assets across different pools. Flash loans enable strategies like arbitrage and collateral swapping by allowing users to borrow funds, execute a strategy, and repay the loan within one atomic transaction. If the borrower fails to repay the loan, the entire transaction is reverted, ensuring that no risk is incurred by the protocol.

Additionally, Balancer introduces Flash Swaps, which allow traders to capitalize on price discrepancies between Balancer pools without needing to hold the input tokens. These swaps facilitate arbitrage and other trading strategies, with the trader being rewarded for identifying profitable opportunities, while the protocol ensures the transaction is completed with minimal slippage.

Balancer's unique combination of flexible pools, flash loans, and flash swaps provides a versatile platform for liquidity provision and trading, offering users significant capital efficiency and enabling a wide range of financial strategies within the DeFi ecosystem.

C. Uniswap

Uniswap [10] [11] is a decentralized exchange (DEX) platform built on Ethereum, utilizing an automated market maker (AMM) system to facilitate token swaps. Uniswap v2 [10], launched in 2020, introduced significant improvements over v1, including the ability to create arbitrary ERC-20 token pairs, making it more versatile. It also introduced flash swaps, allowing users to borrow tokens from liquidity pools and repay them in the same transaction, enabling strategies like arbitrage and collateral swaps. The v2 protocol includes a price oracle to track the time-weighted average price (TWAP), enhancing its use as a reliable price reference in other smart contracts.

Uniswap v3 [11], released in 2021, builds upon v2's foundation while adding major enhancements. A key feature of v3 is concentrated liquidity, which enables liquidity providers to allocate capital within specific price ranges, vastly improving capital efficiency. Additionally, v3 introduces multiple fee tiers (0.05%, 0.30%, and 1%), allowing liquidity providers to choose a fee structure that aligns with the volatility of the token pair. The protocol also features more robust price oracles, including a liquidity accumulator that tracks time-weighted liquidity, improving the accuracy and reliability of price data.

Both v2 and v3 rely on the same core constant product formula ($x \cdot y = k$) for price determination, but v3's flexibility and increased capital efficiency set it apart, making it the preferred platform for liquidity providers looking to optimize returns.

IV. Methodology

This study conducts a comparative analysis of flash loan implementations across four prominent Ethereum-based decentralized finance (DeFi) protocols: Aave, Balancer, Uniswap V2, and Uniswap V3. The evaluation focuses on key metrics such as deployment costs, contract sizes, gas consumption of flash loan functions, and associated fee structures. Moreover, this study evaluate the evolution of flash loan protocols in response to potential attack risks.

Data Collection

The analysis utilizes the flash-loans-comparison repository [12], which provides standardized Solidity contracts and test suites for each protocol. The repository's structure includes:

- `src/` Directory: Contains Solidity contract implementations for Aave (`AAVE.sol`), Balancer (`Balancer.sol`), Uniswap V2 (`UniswapV2.sol`), and Uniswap V3 (`UniswapV3.sol`).
- `test/` Directory: Houses test scripts (`FlashLoans.t.sol`) designed to interact with the respective flash loan functions and measure performance metrics.

For the security aspect, we gather information on security enhancements and mitigation strategies implemented by each protocol post-attack. This includes official documentation, security audit reports, and community discussions.

Procedure

1) Environment Setup: Install Foundry, a development toolkit for Ethereum. Clone the repository and initialize submodules to ensure all dependencies are available.
2) Configuration: Set the `ETH_RPC_URL` environment variable to point to an Ethereum mainnet node, facilitating accurate forking and testing.
3) Execution: Run the test suite using Foundry's `forge` tool with specific flags to enable gas reporting and optimization:

```
% forge test --fork-url $ETH_RPC_URL
   --gas-report --optimize --
   optimizer-runs 10
```

This command executes the tests against a forked mainnet environment, capturing data on deployment costs, contract sizes, gas usage, and function-specific metrics.

Data Analysis

Following the execution of the test suite, the resulting data undergoes a detailed analysis to assess and compare the performance of flash loan implementations across the selected protocols. This analysis focuses on several critical aspects:

- **Deployment Metrics:** Gas costs and bytecode sizes for deploying each contract. The gas expenditure provides insight into the initial setup costs associated with each protocol's flash loan functionality. The bytecode size reflects the complexity and potential impact on network resources.
- **Operational Costs:** Gas consumption statistics (minimum, average, maximum) for invoking the `flashLoan` functions. These metrics illuminate the efficiency and cost-effectiveness of executing flash loans within each protocol.
- **Fee Structures:** The specific fees imposed by each protocol for executing flash loans are identified and compared, highlighting the cost implications for users.

V. Results and Discussion

A. Deployment Metrics

TABLE I
Deployment Costs and Sizes for Flash Loan Protocols

Protocol	Deployment Cost (Gas)	Deployment Size (Bytes)
Aave	361,446	1,453
Balancer	316,017	1,243
Uniswap V2	231,692	853
Uniswap V3	226,722	830

Aave's higher deployment cost and size reflect its comprehensive feature set and security mechanisms. Balancer follows with moderate metrics, while Uniswap V2 and V3 exhibit lower deployment costs and sizes, indicative of their streamlined functionalities.

B. Operational Costs

TABLE II
Operational Gas Costs for Flash Loan Functions

Protocol	Function	Min Gas	Avg Gas	Max Gas	# Calls
Aave	flashLoan	174814	178256	209236	10
Balancer	flashLoan	59790	59790	59790	10
Uniswap V2	flashLoan	77823	78864	88239	10
Uniswap V3	flashLoan	78505	78505	78505	10

Aave has the highest and most variable gas usage, reflecting its complex internal logic. Balancer is the most efficient and consistent, with the lowest fixed gas cost. Uniswap V2 shows moderate cost with some variability, while Uniswap V3 offers stable execution with slightly higher gas than Balancer.

C. Fee Structures

TABLE III
Flash Loan Fee Structures of Protocols

Protocol	Flash Loan Fee (%)
Aave	0.05 [2]
Balancer	0.00 [13]
Uniswap V2	0.3009027 [14]
Uniswap V3	0.05 – 1.00 [15]

Balancer offers flash loans without fees, enhancing its appeal for cost-sensitive operations. Aave imposes a modest fee, while Uniswap's fees vary based on the specific pool, affecting the cost-effectiveness of flash loans.

D. Security Considerations

TABLE IV
Security Measures Implemented by Flash Loan Protocols

Protocol	Security Measures
Aave	Risk parameters, community governance [16]
Balancer	Multiple oracles, slippage controls [17]
Uniswap V2/V3	Time-weighted average price oracles, smart contract audits [18] [19]

Each protocol has implemented specific security measures to mitigate flash loan attack risks. Aave utilizes risk parameters and community governance; Balancer employs multiple

oracles and slippage controls; Uniswap incorporates time-weighted average price oracles and conducts smart contract audits.

The analysis reveals a trade-off between deployment complexity, operational efficiency, fee structures, and security measures among the protocols. Aave's robust features and security come with higher deployment costs, while Balancer offers fee-free flash loans but requires careful consideration of its security mechanisms. Uniswap's varying fees necessitate strategic selection based on specific use cases. Understanding these factors is crucial for users to select the most suitable protocol for their flash loan needs.

VI. Challenges and Limitations

While this study provides a comparative analysis of flash loan implementations across leading DeFi protocols, several challenges and limitations emerged during the evaluation process. One key limitation lies in the evolving nature of these protocols—features such as fee structures, contract logic, and oracle dependencies may change through governance or upgrades. This temporal volatility makes it difficult to capture a fully up-to-date snapshot, and the test results reflect the protocols at a specific point in time. Additionally, while test environments such as Foundry with mainnet forking offer realistic metrics (e.g., gas costs), they do not replicate real network conditions like mempool congestion, front-running risk, or MEV (Miner Extractable Value), which can impact flash loan viability in practice.

Another limitation involves the scope of testing. This analysis focused on flash loan functionality and associated metrics like gas usage and deployment cost, but it did not simulate more complex use cases such as multi-hop arbitrage or cross-protocol liquidations, which may exhibit different behavior. Furthermore, the Euler test case failed consistently, limiting the completeness of the cross-protocol comparison. Security assessments were primarily drawn from publicly available documentation and post-incident disclosures rather than from source-level formal verification or fuzzing, which constrains the depth of the vulnerability analysis.

Overall, while the analysis identifies clear trade-offs in efficiency, cost, and security measures among Aave, Balancer, and Uniswap, further research involving real-world use cases, security audits, and dynamic execution patterns would provide a more holistic understanding of flash loan behavior and risks.

VII. Conclusion

This study compared the performance and security of flash loan protocols across four DeFi platforms: Aave, Balancer, Uniswap V2, and Uniswap V3. We evaluated key metrics such as deployment costs, gas consumption, and fee structures, alongside an assessment of security measures implemented by each protocol.

Our analysis showed that Balancer is the most efficient in terms of gas usage and deployment costs, while Aave, with its robust security features, incurs higher operational costs. Uniswap V2 and V3 offered moderate gas efficiency, with V3 showing more consistent performance. The fee structures varied, with Balancer providing fee-free flash loans, unlike Aave and Uniswap, which have higher fees that impact cost-effectiveness.

In terms of security, each protocol employs different strategies to mitigate risks, including Aave's risk parameters, Balancer's multiple oracles, and Uniswap's time-weighted average price oracles. Despite these efforts, the analysis highlights ongoing challenges in securing flash loan transactions against attacks.

Overall, the findings underline the importance of balancing performance, cost, and security when choosing a flash loan protocol. Future research could expand on real-world use cases and incorporate more advanced security testing methods to better assess protocol vulnerabilities.

Acknowledgements

The authors would like to acknowledge the use of ChatGPT-4o, an AI language model developed by OpenAI, for assisting with rewriting and polishing the manuscript for fluency and readability. The original ideas, content, and research were solely authored by the authors, and the AI tool was used strictly for language enhancement and not for generating any academic or technical content.

References

[1] M. Wolff, "Introducing marble: A smart contract bank," https://medium.com/marbleorg/introducing-marble-a-smart-contract-bank-c9c438a12890, Jul. 2018, accessed: 2025-04-09.

[2] A. Protocol, "Flash loans," https://aave.com/docs/developers/flash-loans, 2024, accessed: 2025-04-09.

[3] J. Mandin, "Risk-free uncollateralized lending in decentralized markets: An introduction to flash loans," Bank of Canada, Staff Discussion Paper 2025-6, 2025, accessed: 2025-04-10. [Online]. Available: https://doi.org/10.34989/sdp-2025-6

[4] L. Gudgeon, D. Perez, D. Harz, B. Livshits, and A. Gervais, "The decentralized financial crisis," 2020. [Online]. Available: https://arxiv.org/abs/2002.08099

[5] K. Qin, L. Zhou, B. Livshits, and A. Gervais, "Attacking the defi ecosystem with flash loans for fun and profit," 2021. [Online]. Available: https://arxiv.org/abs/2003.03810

[6] Z. Chen, S. M. Beillahi, and F. Long, "Flashsyn: Flash loan attack synthesis via counter example driven approximation," in *Proceedings of the IEEE/ACM 46th International Conference on Software Engineering*, ser. ICSE '24. New York, NY, USA: Association for Computing Machinery, 2024. [Online]. Available: https://doi.org/10.1145/3597503.3639190

[7] A. Alhaidari, B. Palanisamy, and P. Krishnamurthy, "Protecting defi platforms against non-price flash loan attacks," 2025. [Online]. Available: https://arxiv.org/abs/2503.01944

[8] E. Frangella and L. Herskind, "Aave v3 technical paper," https://github.com/aave-dao/aave-v3-origin?tab=readme-ov-file, 2022, accessed: 2025-04-10.

[9] Balancer, "Balancer v2 - flash loans," https://balancer.gitbook.io/balancer-v2/products/the-vault/flash-loans, 2022, accessed: 2025-04-10.

[10] H. Adams, N. Zinsmeister, and D. Robinson, "Uniswap v2 core," https://app.uniswap.org/whitepaper.pdf, 2020, accessed: 2025-04-10.

[11] H. Adams, N. Zinsmeister, M. Salem, R. Keefer, and D. Robinson, "Uniswap v3 core," https://app.uniswap.org/whitepaper-v3.pdf, 2021, accessed: 2025-04-10.

[12] Jeiwan, "flash-loans-comparison," https://github.com/Jeiwan/flash-loans-comparison, 2021, accessed: 2025-04-09.

[13] B. Labs, "Protocol fees," https://balancer.gitbook.io/balancer-v2/concepts/fees, 2024, accessed: 2025-04-09.

[14] U. Labs, “Using flash swaps,” https://docs.uniswap.org/contracts/v2/guides/smart-contract-integration/using-flash-swaps, 2024, accessed: 2025-04-09.
[15] L. Xie, “Flash loan fees in uniswap v3,” https://uniswapv3book.com/milestone_5/flash-loan-fees.html, 2024, accessed: 2025-04-09.
[16] A. Protocol, “Risks in the aave protocol,” https://aave.com/docs/concepts/risks, 2024, accessed: 2025-04-09.
[17] B. Protocol, “Balancer integrates chainlink price feeds to help secure staked eth composable stable pools,” https://medium.com/balancer-protocol/balancer-integrates-chainlink-price-feeds-to-help-secure-staked-eth-composable-stable-pools-c649d8181510, Sep. 2023, accessed: 2025-04-09.
[18] RareSkills, “Understanding twap in uniswap v2,” https://www.rareskills.io/post/twap-uniswap-v2, 2023, accessed: 2025-04-09.
[19] U. Labs, “Uniswap protocol oracle,” https://docs.uniswap.org/concepts/protocol/oracle, 2024, accessed: 2025-04-09.