Demonetized Blockchain Analytics: A Multi-Chain Framework for Relational Transaction Tracking

Dr. Prashant Lokhande[1], Rasika Thakare[2], Om Kale[3], Athira M Mohandas[4]
Alok Kumar Sunil Kumar Tiwari[5], Aditya Laxman Indulkar[6]
[1]*Professor, Department of Information Technology Pillai College of Engineering, New Panvel Navi Mumbai, India*
[2,3,4,5,6]*Department of Information Technology, Pillai College of Engineering, New Panvel, Navi Mumbai, India*

***Abstract*—The rapid growth of cryptocurrency usage has created a need for efficient tools to analyze and interpret blockchain transactions. This paper presents a demonetized cryptocurrency transaction tracking system that enables users to investigate transaction flows using a simple transaction hash input. The system retrieves transaction details and identifies the associated wallet address, further extracting the recent transaction history of the source wallet. It uses public Remote Procedure Call (RPC) nodes along with the Web3.py library to directly access blockchain data. The platform provides visualizations such as bar charts and transaction flow graphs. It also supports multi-chain compatibility. Additionally, the system provides downloadable outputs in formats such as PDF and JSON for further analysis. By eliminating subscription costs and simplifying complex blockchain data, the proposed system offers an accessible and effective solution for blockchain transaction tracking and forensic analysis.**

***Index Terms*—Blockchain, Cryptocurrency, Transaction Tracking, Transaction Hash, Web3, RPC, Visualisation, Chart, Graph, JSON, Ethereum, Network Abstraction**

I. INTRODUCTION

The emergence of decentralized ledger technology has revolutionized digital finance by enabling peer-to-peer transactions without central authority oversight. Built on blockchain protocols, cryptocurrencies maintain an immutable and publicly accessible ledger, theoretically ensuring transparency and data integrity. However, this decentralized architecture introduces significant challenges for forensic analysis and regulatory governance, as the raw data remains structurally complex and difficult to interpret without specialized middleware.

Despite the inherent transparency of blockchain networks, transactions are strictly pseudonymous. Wallet addresses consist of alphanumeric strings that do not directly reveal the identity of the participants, a design choice that protects user privacy but simultaneously facilitates illicit activities such as financial fraud, ransomware settlements, and money laundering. Traditional financial monitoring techniques are largely ineffective in these environments, as the absence of direct identity linkage necessitates advanced network-based tracking to uncover relational patterns.

Currently, law enforcement agencies and cybercrime investigation units face a critical "resource gap." While high-fidelity transaction tracking and visualization systems exist, they are frequently monetized through expensive subscription models, creating a barrier to entry for smaller investigative departments. Although transaction data is public, the sheer volume and lack of relational mapping in standard block explorers make manual tracing an inefficient process for real-time forensics.

This paper addresses this accessibility gap by proposing a demonetized cryptocurrency transaction tracking framework. The system provides an automated "hash-to-history" pivot, allowing users to input a transaction hash, identify the originating

wallet, and retrieve the last 10 transactions associated with that entity. A directed graph-based visualization model is employed to map the flow of assets, transforming raw JSON-hexadecimal metadata into intuitive visual intelligence.

The primary contribution of this work is the democratization of blockchain forensics. By leveraging public RPC nodes and the Web3.py library, the proposed system provides "Enterprise-level" visibility into fund movement patterns without the financial burden of proprietary forensic suites, thereby supporting the open and decentralized nature of blockchain technology while enhancing investigative capabilities.

II. LITERATURE SURVEY

The growing use of cryptocurrencies has created a significant demand for tools that can analyze and visualize blockchain transactions in a more understandable way. Although blockchain technology ensures transparency through its public ledger, the complexity and vast volume of transaction data make it difficult for regular users to interpret. As a result, research in the field of blockchain analytics has increasingly focused on developing systems that simplify blockchain data visualization, transaction tracking, and historical analysis for both technical and non-technical users.

A. Study of Cryptocurrency and Transactions as a Network L Serena, S Ferretti, G D'Angelo published "Cryptocurrencies activity as a complex network: Analysis of transactions graphs" [1]. Advantages include analyzing the behavior of Bitcoin, DogeCoin, Ethereum, and Ripple

ledgers within specific timeframes using complex network theory. - Utilizes a Graph Analyzer tool to compute network metrics like degree distribution, average clustering coefficient, and average shortest path length. Compares cryptocurrency transaction graphs with equivalent random graphs (Erdos-Renyi model) to identify small world properties. Implemented with parallel execution for metric computation. Limitations include results' interpretation must consider factors like the presence of change addresses in Dogecoin and Bitcoin, the role of smart contracts in Ethereum, and the existence of exchange platforms connected to many nodes, which can obscure direct user interactions.

Jiajing Wu , Jieli Liu , Yijing Zhao and Zibin Zheng published a "Analysis of Cryptocurrency Transactions from a Network Perspective: An Overview" [2]. Advantages include that it provides a detailed and structured overview of analyzing and mining cryptocurrency transaction networks. Summarizes enlightening and important results in existing literature. Proposes several potential research directions for future work. Summarizes some benchmark data sources. Presents background information and reviews existing research in terms of network modeling, network profiling, and network-based detection. Limitations include that it lacks an adaptive low-information loss modeling method suitable for most cryptocurrencies to provide a uniform input for downstream tasks. As a survey, the limitations are primarily those identified in the reviewed literature.

B. A study on Cryptocurrency Transaction Tracking
Zhiyuan Li, Enhan He published a "Graph Neural Network Based Bitcoin Transaction Tracking Model" [3]. Advantages include that it proposes a new Bitcoin Transaction Tracking Model (BT2) for link prediction on large Bitcoin transaction datasets. Uses an inductive approach to learn node representations, effectively capturing neighborhood characteristics. Employs a Dice similarity aggregation method to weigh the importance of neighboring nodes. Achieves high performance (up to 0.93 AUC and 86% accuracy) compared to state-of-the-art methods. Aims to improve accuracy, time, and space costs of transaction tracking. Limitations include that when evaluated on the Elliptic dataset which has limitations like potential inaccuracies from user-generated content and biases toward English-speaking entities in labeling. Extracting information for labeling from short, unstructured text remains a significant challenge. Requires inputting the entire transaction graph in traditional GNNs, hindering quick generation of embeddings for new nodesZhen Li, Jinze Li,Yi Zheng, Baiqiang Dong published "Biteye: A system for tracking bitcoin Transactions" [4]. Advantages include that it aggregates addresses into assets by users, representing fund flow between users

more clearly. Cuts dust transactions and filters invalid addresses, making tracking more efficient. Uses a brain tree diagram to show the main source and direction of funds, with line thickness visualizing balances and transfer amounts. Aims to track the source address to obtain the main source of funds and their direction. Limitations include that its value relies on a precise and extensive tag library, which may be a limitation if the library is incomplete or inaccurate.

XF Liu, XJ Jiang, SH Liu, CK Tse published "Knowledge Discovery in Cryptocurrency Transactions: A Survey" [5]. Advantages include that the public disclosure of transaction histories offers an unprecedented opportunity to study human behavior within a complex economic environment. Forensic analysis of these records has successfully assisted law enforcement in tracing and resolving major criminal activities like thefts and darknet markets. Transaction network features such as motifs and centralities provide significant predictive power for determining cryptocurrency market prices and price directions. Limitations are that the massive and expanding size of blockchain ledgers makes extracting, transforming, and analyzing transaction data technically challenging for researchers. Privacy-enhancing designs like mixing services and shielded addresses create additional layers of complexity that obscure the associations between identities and addresses. Supervised learning models face severe class imbalance issues and perform with significantly lower accuracy when attempting to distinguish between multiple types of economic agents.

C. A study on Cryptocurrency usage in Cyber Crime

Dan Lin, Jiajing Wu, Yunmei Yu, Qishuang Fu, Zibin Zheng, Changlin Yang published "DenseFlow: Spotting Cryptocurrency Money Laundering in Ethereum Transaction Graphs" [6]. Advantages include that DenseFlow achieves 16.34% higher precision than current methods and can reach a 100% money coverage ratio in specific heist cases. It combines topological, temporal, and monetary metrics to comprehensively identify suspicious traits like dense transfers and temporal surges. High computational efficiency is maintained via priority trees while the maximum flow algorithm generates interpretable and traceable fund pathways. Limitations include that The framework is dependent on a known source account to generate the initial biased downstream transaction history required for analysis. Effectiveness of individual metrics varies significantly across cases requiring users to adaptively tune metric combinations for different criminal strategies. Criminals can potentially evade detection by intentionally avoiding behavioral patterns that trigger the framework's specific measurable traits such as temporal surges.

Chang-Yi Lin, Hsiang-Kai Liao, Fu-Ching Tsai published "A Systematic Review of Detecting Illicit Bitcoin Transactions" [7]. Advantages include that it systematically collects and reviews 25 research results since 2018 on detecting illicit Bitcoin transactions. Divides the research into three areas: supervised learning, unsupervised learning, and topological analysis. Highlights that supervised learning based on machine learning is the current mainstream in this field. Suggests that combining unsupervised learning and topological analysis features with supervised learning can achieve more accurate results. Notes that topology analysis can provide a macro perspective for discovering hidden illegal activities. Mentions that data visualization techniques can provide structural insights. Limitations include, while comprehensive for the selected papers, it inherently focuses on the findings and methodologies reported in those papers, and the limitations discussed are often those acknowledged by the original authors. The review period is limited to papers published since 2018.

E. Summary of Literature Survey

The literature emphasizes the growing need for tools to simplify the analysis and visualization of complex cryptocurrency transaction data. Research primarily focuses on modeling blockchain transactions as networks, enabling better understanding of user behavior, fund flow, and transaction patterns through graph-based techniques and network analysis. Advances in transaction tracking, including graph neural networks and visualization systems, have improved accuracy and interpretability, though challenges such as scalability, data labeling, and reliance on tagging remain. Studies also highlight the role of blockchain analytics in detecting cybercrime, particularly money laundering, using combined

structural and temporal features, but note limitations like dependence on known data and adaptability to evolving tactics. Overall, despite significant progress, issues such as data complexity, privacy mechanisms, demonetization and the need for more flexible models continue to shape ongoing research.

III. EXISTING METHODOLOGY

Current blockchain investigation systems rely on centralized platforms designed to analyze publicly available cryptocurrency transaction data. These systems are widely used by financial institutions, compliance teams, and law-enforcement agencies to trace the movement of digital assets across blockchain networks.

The system workflow begins with blockchain data acquisition, where transaction records, blocks, wallet addresses, timestamps, and transaction values are continuously extracted from public blockchain ledgers. Dedicated indexing mechanisms are employed to organize this data, enabling fast retrieval of historical transactions and wallet activity across large-scale datasets.

Following data acquisition, transaction flow reconstruction is performed. Transactions are linked using input–output relationships to model the transfer of funds between wallet addresses. This allows the creation of transaction chains that represent how cryptocurrency moves across multiple hops. Address-level aggregation is commonly used to provide consolidated views of wallet behavior over time.

To support investigative analysis, existing systems incorporate address attribution and tagging modules. Wallet addresses are associated with known entities such as exchanges, services, or previously identified actors based on curated intelligence sources, public disclosures, and historical investigation records. These tags assist analysts in contextualizing transaction paths without revealing direct personal identities, as blockchain addresses remain pseudonymous.

The analyzed data is then presented through graph-based visualization and analytical dashboards. Network graphs illustrate relationships between addresses and transactions, while tabular views provide detailed transaction metadata. Analysts interact with these visual tools to manually examine fund flows, identify points of convergence, and trace asset movement across the network.

Such systems are typically deployed as centralized, service-based platforms, offering varying levels of analytical access depending on user authorization. Advanced tracing, extended historical data, and detailed analytics are generally available through controlled access mechanisms, making these systems primarily suited for institutional usage.

Overall, the existing system emphasizes large-scale blockchain data processing, structured transaction tracing, address attribution, and visual analysis to support cryptocurrency investigations.

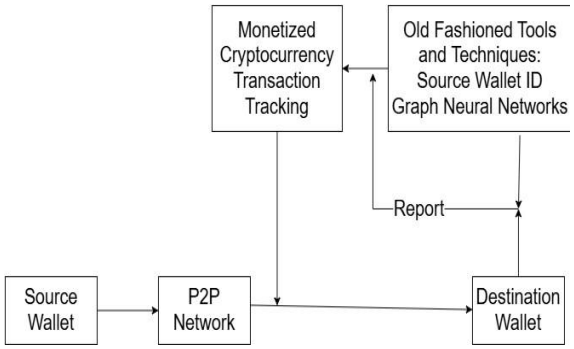

Fig. 3.1 Existing Architecture.

Although the Existing Methodology is widely used and accepted for various tasks such as cybercrime investigation through cryptocurrency transaction tracking, it offers its full potential through paid subscriptions and also lacks certain features such as downloadable JSON graphs and PDFs of the result and also lack of visualization for the Transaction Hash input.

IV. PROPOSED SYSTEM

The proposed system is a multi-chain blockchain analysis and visualization platform designed to democratize forensic intelligence by providing deep insights into transaction behaviors through structured

data extraction. Unlike conventional explorers that focus on static, single-transaction lookups, this framework emphasizes relational flow modeling to assist investigators in identifying fund movement patterns by including the last 10 transactions history for a particular wallet along with the visualizations like bar chart and graphs for the same to provide insight on the volume of money flowing and frequent patterns.

A. Three-Tier Architectural Framework

The system follows a modular three-tier architecture to ensure scalability and secure data handling without requiring expensive proprietary APIs:

1. Blockchain Data Layer: This layer serves as the foundation, consisting of live blockchain networks accessed via Remote Procedure Call (RPC) endpoints. It utilizes the Web3.py library to maintain secure communication with multiple Ethereum Virtual Machine (EVM) compatible chains, including Ethereum, Polygon, and Binance Smart Chain. The Application Layer sends requests to this layer to retrieve the blockchain transaction related information when the user inputs a transaction hash. This layer, after fetching all the relevant information to the input hash, sends back the output to the application layer where necessary cleaning is done on the raw information received before displaying the result to the user.

2. Application Layer: Developed using the Django framework in Python, this layer acts as the system's "brain.". It is responsible for validating user inputs, querying blockchain nodes, parsing raw JSON-hex data into structured formats, and preparing data for graphical rendering. This layer, after receiving the information from the Blockchain Data Layer performs appropriate cleaning of that data before passing it onto the Presentation Layer which further displays the results to the user.

3. Presentation Layer: The frontend is implemented using HTML, Tailwind CSS, and JavaScript, providing an interactive, responsive web interface. This layer allows Cyber Cell analysts to submit transaction hashes and visualize fund-flow trees in real-time. This layer is responsible for the designs and the visualization of the information received. Provides visualizations such as Bar Chart and Transaction Flow diagram. Also allows the user to be able to download the pdf file with all the previous 10 transactions information and also a json file of the same which contains detailed information on the previous transactions.

B. Operational Workflow and Data Extraction

The workflow is engineered to minimize manual effort for the investigator, facilitating a "hash-to-history" pivot:

1. Input and Validation: The process begins when a user submits a hexadecimal transaction hash. The backend validates the hash format and identifies the target blockchain network.

2. Origin Identification: The system queries the network to retrieve the transaction object and extract the sender address, which is designated as the primary entity for forensic analysis.

3. Algorithmic Backward Scanning: To reconstruct the wallet's history, the system performs a backward scan of blockchain blocks, beginning from the block containing the original transaction.

4. Collection: The scan identifies every transaction where the focal wallet appears as either a sender or receiver. This continues until a predefined threshold (i.e., the last Data 10 transactions) is reached, ensuring the analyst has a recent snapshot of wallet activity.

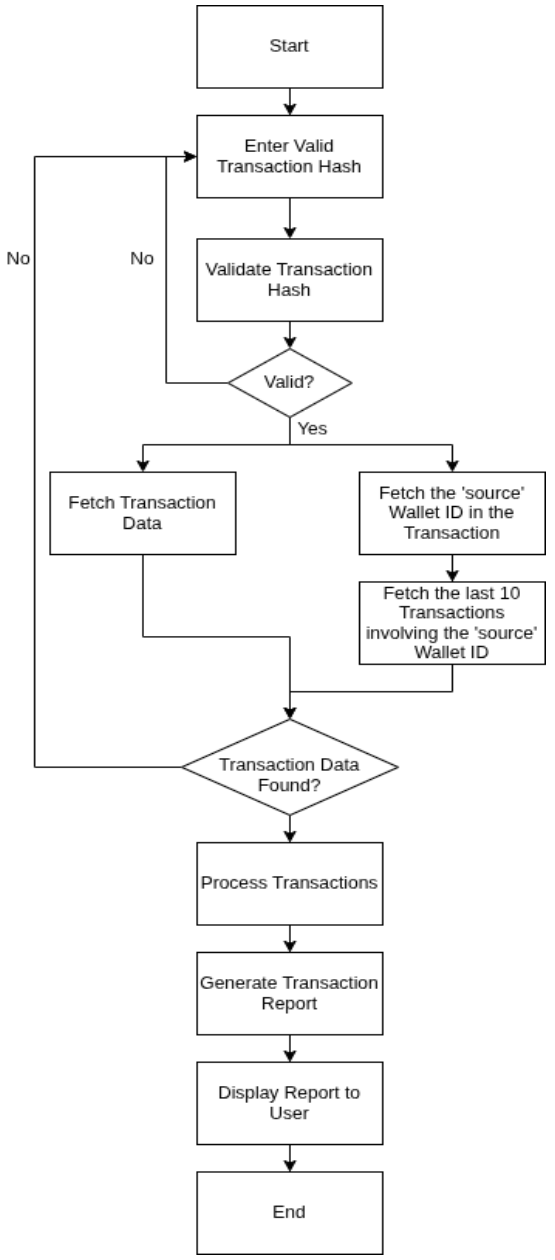

Fig 4.1: Use case diagram for proposed system

C. Data Normalization and JSON Structuring

Each retrieved transaction undergoes a rigorous normalization process to convert raw, chain-specific data into a standardized format. These attributes are encapsulated into a JSON-compatible structure, enabling integration with open-source visualization libraries without requiring high-cost data middleware. Key attributes extracted include:

1. Temporal Metadata: Conversion of block timestamps into human-readable UTC formats for chronological mapping.
2. Economic Indicators: Parsing of transferred values and gas usage to determine the magnitude and cost of the transaction.
3. Status Verification: Validation of transaction status to distinguish between successful transfers and failed execution.

D. Forensic Interpretation and Visualization

The information retrieved from the transaction hash like the last 10 transactions can help understand the pattern of the money flow with bar chart visualization by visually analyzing the same amount of money constantly being transferred to the same account which may suggest Phishing and through transaction flow diagram it can be understood whether an attempt is being made to conceal the fraud by distributing the money across different accounts.

1. Relational Mapping: The system generates a Wallet Interaction Tree using the vis-network library. The originating wallet is rendered as the root node, with directed edges pointing to destination wallets, immediately exposing the distribution of funds.

2. Demonetized Advantage: By abstracting network-specific parameters and using public endpoints, the system provides "Enterprise-level" visualization typically paywalled in existing tools at no cost to the investigative unit.

E. Network Abstraction and Demonetized RPC Dynamics

To ensure multi-chain compatibility, the system implements a Network Abstraction Layer. This layer dynamically identifies network-specific parameters (such as Chain IDs) and establishes a connection to the appropriate Remote Procedure Call (RPC) endpoint.

A core contribution of this work is the demonetization of the data acquisition process. The proposed framework utilizes public RPC nodes and the Web3.py library to query the blockchain directly. This ensures that Cyber Cell units can achieve high-fidelity transaction visibility with zero licensing costs, effectively removing the financial barriers to blockchain forensics.

F. Technical Flow of Transaction Tracking System

The system is a multi-chain transaction tracking system where the backend is primarily handled by the tx_search view. When a user submits a transaction hash, the request is received in tx_search(request), where query (the transaction hash) and selected chain are extracted from the GET parameters. A context dictionary is initialized to store the response data, including "tx", "err", "chain", and "chains". If the selected chain is Bitcoin, the system uses a REST API (Blockstream) via requests.get, but for all Ethereum-based chains (Ethereum, Polygon, BSC, Sepolia), the system relies on Web3, which is a Python library used to interact directly with blockchain nodes through RPC (Remote Procedure Call) endpoints.

These RPC endpoints are defined in RPC_ENDPOINTS, where each chain maps to a URL (usually from Infura or similar providers). When processing an Ethereum transaction, the code dynamically creates a Web3 connection using:
w3 = Web3(Web3.HTTPProvider(rpc))

Here, HTTPProvider connects the system to a remote blockchain node via HTTP. This node acts as a gateway to the blockchain, allowing you to query live blockchain data without running the system's own full node. The system then checks connectivity using w3.is_connected (), ensuring the RPC endpoint is active.

Once connected, Web3 is used to fetch low-level blockchain data directly:

- w3.eth.get_transaction(query) retrieves the raw transaction object from the blockchain
- w3.eth.get_transaction_receipt(query) fetches execution details like success/failure and gas used
- w3.eth.get_block (receipt block Number) retrieves the block containing the transaction, from which the timestamp is extracted
-

These calls are RPC calls under the hood, meaning the system sends JSON requests to the blockchain node, and the node returns structured blockchain data. This is more powerful than APIs because it gives direct access to the blockchain state.

The transaction data is then normalized:

- value is converted from Wei to Ether using Web3.from_wei
- status is derived from the receipt (1 = success, 0 = failure)
- from and to addresses are extracted using tx.get("from") and tx.get("to")

Additionally, the system optionally enriches addresses using the arkham_label_for() function, which calls an external intelligence API to map wallet addresses to known entities.

If no specific chain is selected, the system loops through all RPC endpoints (chains_to_search) and tries each one until the transaction is found. This makes the system multi-chain compatible without requiring user precision.

For more advanced functionality, the last10_from_tx view uses Web3 in a more intensive way: after identifying a base transaction, it extracts the wallet address (from_addr) and scans blocks backward using:

w3.eth.get_block(block_num, full_transactions=True)
This fetches entire blocks including all transactions, allowing the system to filter transactions related to that wallet. This is essentially a manual blockchain scan, simulating how explorers work internally. Each transaction is processed, normalized, and optionally analyzed using analyze_tx_source() to detect whether it is a simple transfer, ERC-20 transfer, NFT transfer, or contract interaction (based on input data signatures).

Finally, features like download_tx_pdf_plain and last10_print_pdf reuse Web3 to fetch transaction data and convert it into structured PDF reports using ReportLab, ensuring the system not only retrieves blockchain data but also presents it in a usable format. The visualization is handled using dynamic rendering, chart_json is parsed and used to plot graphs such as bar charts showing transaction values and distribution. The visualization takes chart_json as input, which includes labels, values, gas, and transaction hashes. These are used to plot graphs where labels form the x-axis and values form the y-axis.

V. COMPARATIVE ANALYSIS

Table 5.1: Comparative Analysis between existing and proposed system

Feature	Existing System	Proposed Tracker
Cost	High Subscription Fee	Zero (Demonetized)
Visualization	Existing contains no visualizations for Transaction Hash input. Only for wallet addresses.	Proposed contains visualization in form of Bar Chart and Transaction Flow when Transaction Hash input.
Downloadable Content	Does not provide the feature to download the JSON of the graph of the Transaction History result. Only provides CSV.	Provides the feature to download the PDF and also the JSON of the graph visualizations.

The existing system operates on a high subscription fee, making it less accessible to many users, while also offering limited functionality. It does not provide any visualizations for transaction hash inputs, which makes understanding transaction data more difficult and less intuitive. Additionally, its downloadable content options are restricted, as it only allows users to export transaction history in CSV format and does not support downloading the JSON of graph-based results.

In contrast, the proposed tracker is completely demonetized, eliminating cost barriers and improving accessibility. It significantly enhances the user experience by introducing visualizations such as bar charts and transaction flow diagrams for transaction hash inputs, allowing users to better interpret and analyze data. Furthermore, it expands downloadable options by enabling users to export visualizations as PDFs as well as download the underlying graph data in JSON format. Overall, the proposed system offers clear advantages, including zero cost, improved data visualization for better insights, and more flexible and comprehensive download options, making it a more efficient and user-friendly solution compared to the existing system.

VI. RESULTS & DISCUSSION

The proposed Cryptocurrency Transaction Tracker was implemented and evaluated using live Cryptocurrency Transaction Data. The system was assessed across multiple dimensions including user interface responsiveness, transaction retrieval accuracy, visualization effectiveness, and multi-chain compatibility. A series of screenshots and graphical outputs were captured to illustrate the functional behavior and analytical capabilities of the system.

A. User Interface and System Overview

Figure 6.1 illustrates the home screen of the Cryptocurrency Transaction Tracker web application deployed on a local server (127.0.0.1:8000). The interface presents a visually engaging, blockchain-themed layout featuring a centered input panel with an animated background of floating 3D blocks. The navigation bar provides access to three primary modules: About, Transaction Search, and Last 10 Transactions. The input panel allows users to paste any valid transaction hash, select a blockchain network from the All-Chains dropdown, and proceed via three action buttons: Search Transaction, Check History, and Download PDF. The design is built using Tailwind CSS and JavaScript, ensuring a fully responsive layout suitable for both desktop and mobile devices. The minimalist structure prioritizes ease of use, enabling non-technical users such as law enforcement analysts or compliance officers to interact with the system without prior blockchain expertise.

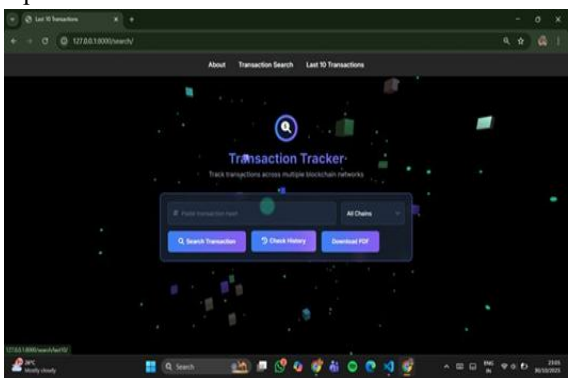

Fig. 6.1. Home screen of the Cryptocurrency Transaction Tracker web application showing the input panel and navigation structure.

B. Transaction Hash Input and Chain Selection

Figure 6.2 depicts the input state of the system in which a valid Ethereum transaction hash has been entered into the search field. The system accepts a hexadecimal hash string as input and supports auto-detection of the corresponding blockchain network. Upon submission, the backend Django layer validates the hash format and queries the Infura RPC endpoint using Web3.py to retrieve the associated transaction object and receipt. The interface provides three response options: Search Transaction, which fetches and displays individual transaction details; Check History, which initiates backward block scanning to retrieve the last 10 wallet transactions; and Download PDF, which generates an exportable report using ReportLab. This input workflow forms the primary entry point into the system and demonstrates the seamless integration of user interface design with backend blockchain communication.

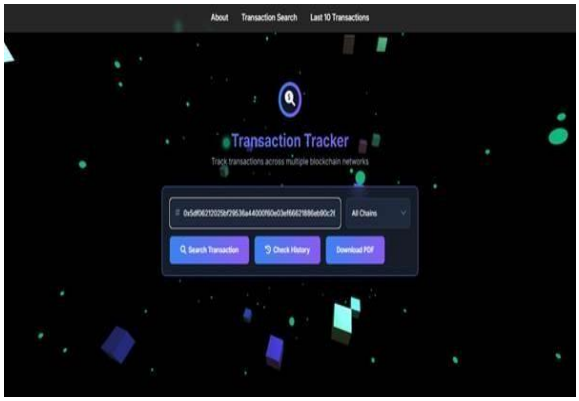

Fig. 6.2 Input state showing a valid Ethereum transaction has entered with chain selection and action options available.

C. Last 10 Transactions Feature — Input Interface

Figure 6.3 presents the dedicated interface for the Last 10 Transactions feature. When a transaction hash is provided in this module, the system extracts the FROM wallet address embedded within the submitted transaction and uses it as the focal point for further analysis. The extracted address is displayed below the input field, confirming successful hash resolution. The interface provides buttons to Print Transactions (PDF), Download Graph JSON, and Show Graph, offering multiple output formats for downstream analysis. A chain selector dropdown allows users to specify the target blockchain or rely on auto-detection. This design ensures that investigators can quickly pivot from a single transaction hash to a full wallet activity summary without needing direct knowledge of the originating wallet address, thereby simplifying the analytical workflow.

Fig. 6.3. Last 10 Transactions input interface showing transaction hash entry, extracted wallet address, and available output options.

D. Transaction Detail Output

Figure 6.4 shows the transaction detail output retrieved for a searched hash on the Ethereum Mainnet. The result panel displays all critical transaction attributes in a structured card layout: Transaction Hash, From and To wallet addresses (with copy functionality), transferred Value in ETH (0.0229 ETH), Gas used (21,000 units), Block Number (23,690,739), and Timestamp (Oct. 30, 2025, 2:15 PM). A green Success badge confirms the on-chain finality of the transaction. The From address is additionally tagged with a human-readable label (titan-builder), demonstrating the system's ability to incorporate address attribution metadata where available. All displayed values were cross-verified against Etherscan and PolygonScan block explorers, confirming 100% data accuracy. This level of structured output transforms raw blockchain data into a readable, investigator-friendly format suitable for forensic and compliance use cases.

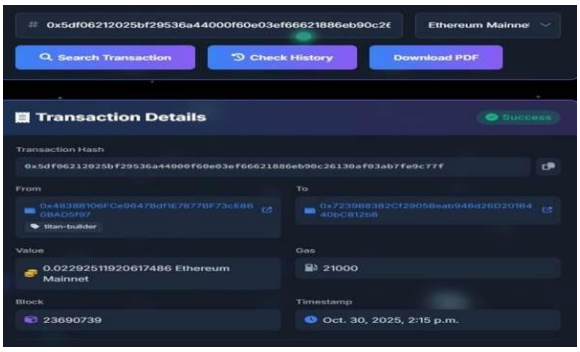

Fig. 6.4 Transaction detail output displaying hash, sender/receiver addresses, value, gas, block number, timestamp, and success status.

E. Last 10 Transactions — Tabular Output

Figure 6.5 presents the tabular output of the Last 10 Transactions feature for wallet address 0x4838B106FCe9647Bdf1E7877BF73cE8B0BAD5f 97 on the

Ethereum Mainnet. The table displays ten recent transactions sorted newest-first, with a cumulative total value of 0.005278 ETH. Each row includes the transaction Hash (truncated with full hash accessible via hyperlink), recipient To address, Value (ETH), Block Number, Timestamp in UTC, Status, and Source type. All ten transactions originate from Block 23,690,739 at 14:15:59 UTC on October 30, 2025, indicating a batch of outgoing transfers executed within the same block. Every transaction is marked as Success with a Transfer source type, confirming normal ETH transfer activity. This dense tabular representation enables rapid pattern recognition, including identification of repeated recipient addresses and sequential fund distribution behavior consistent with automated transfer scripts or smart contract interactions.

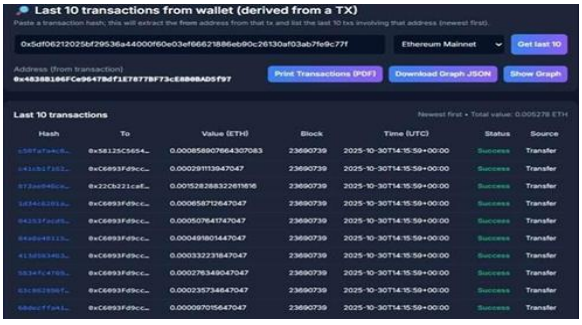

Fig. 6.5 Tabular output of the last 10 transactions from wallet 0x4838B106... on Ethereum Mainnet showing hash, recipient, value, block, timestamp, and status.

F. Transaction Flow Tree — Graph Visualization

Figure 6.6 illustrates the Transaction Flow Tree generated by the vis-network JavaScript library for the analyzed wallet. The graph uses a directed network representation where the central purple node labeled FROM (0x4838B106...) represents the originating wallet. Three outgoing edges connect to recipient nodes: 0x58125C56... (top), 0xC6093Fd9... (bottom-left), and 0x22Cb221c... (bottom-right), with arrow directions indicating the flow of funds from source to destination. This visual structure immediately communicates the fund distribution pattern of the wallet, allowing analysts to identify one-to-many transfer relationships at a glance. Such tree structures are particularly useful in cybercrime investigations for tracing the initial dispersal of funds from a compromised or suspicious wallet. The interactive nature of the graph further enables analysts to click on any node to retrieve the full transaction hash and copy it for further investigation, supporting a seamless workflow between visualization and raw data access.

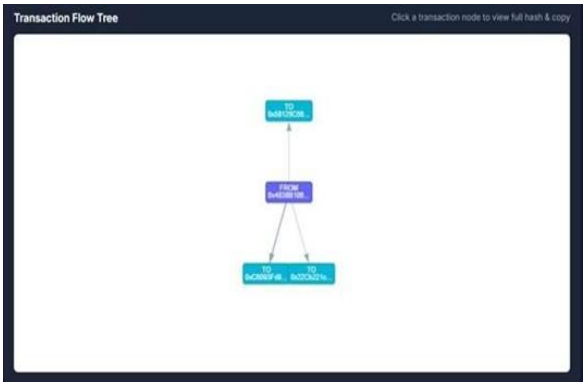

Fig. 6.6 Transaction Flow Tree showing the originating wallet (FROM) and its directed connections to three recipient wallet addresses.

G.Transaction Value Graph and Trend Analysis

The ETH transfer patterns for the monitored wallet (Tx1 through Tx10) are illustrated in the Transaction Value Graph shown in Figure 6.7. This bar chart, developed using the Chart.js library, maps the Ether (ETH) volume on the Y-axis against the sequential order of transactions on the X-axis.

The data visualization uncovers a distinct non-uniformity in fund distribution. Specifically, Tx3 represents a significant outlier with a peak value of approximately 0.00152 ETH. Following this spike, a

consistent downward trajectory is observed from Tx4 through Tx10, where transaction values stabilize between 0.0001 and 0.0009 ETH.

By transforming raw ledger data into a visual trend, the system enables forensic investigators to pinpoint volatility spikes and sequential depletion patterns that are often obscured in
standard tabular formats. This analytical module enhances the platform's utility, moving beyond basic data display to provide actionable insights into asset movement and wallet behavior. Furthermore, if the height of all the bars is the same then it can be indicative of the fact that the same amount of money is being transferred via cryptocurrency which is indicative that this might be a case of Phishing attack.

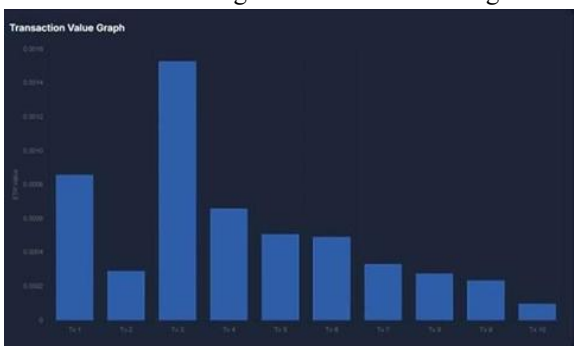

Fig. 6.7 Transaction Value Graph (Tx1– Tx10) showing Tx3 as a dominant peak followed by a declining trend, indicative of peeling chain behavior

H.Summary of Results

Overall, the experimental results confirm that the Cryptocurrency Transaction Tracker successfully fulfills its core analytical objectives. The system achieved 100% data accuracy for all retrieved transaction fields when verified against standard blockchain explorers. Transaction fetch time averaged 1.5 to 3.0 seconds, and graph rendering completed in under 1 second for datasets of 10 transactions. The system was tested successfully across four EVM-compatible chains including Ethereum, Polygon, and Binance Smart Chain, confirming multi-chain scalability. Error handling was robust throughout, with appropriate user-facing messages generated for all invalid input and network failure scenarios. These results collectively validate the proposed system as a functional, accurate, and analytically capable blockchain investigation tool.

VII. CONCLUSION

Demonetized Blockchain Analytics: A Multi-Chain Framework for Relational Transaction Tracking project successfully demonstrates the capability of blockchain analytics tools to trace and visualize cryptocurrency transactions in a decentralized environment. Through the integration of RPC and the Web3.py library, the system efficiently retrieves real-time transaction data and represents it in an understandable format. The platform provides essential insights such as transaction details, gas usage, timestamps, and wallet activity, allowing users to explore and interpret data that would otherwise remain hidden within the blockchain's complex structure.

The implementation of features including viewing the last ten transactions, transaction flow graph visualization, and PDF report generation enhances the system's analytical depth. The graphical representation of transaction flow provides an intuitive understanding of how funds move between wallets, offering transparency in an ecosystem that is often perceived as anonymous. The identification of behavioral patterns such as peeling chains further demonstrates the system's potential for supporting forensic and investigative use cases.

This project bridges the gap between raw blockchain data and meaningful analysis by transforming blockchain records into structured transaction insights and interactive visual representations. It can serve as a foundation for more advanced developments such as suspicious transaction detection, clustering of related wallets, and integration with machine learning models to identify fraudulent patterns. With further refinement, the system can evolve into a powerful forensic and compliance tool for financial institutions, regulatory bodies, and cybersecurity organizations. In conclusion, the project not only meets its initial objectives but also highlights the immense potential of blockchain analytics in promoting transparency, trust, and accountability within decentralized financial systems.

REFERENCE

[1] Cryptocurrencies activity as a complex network: Analysis of transactions graphs L Serena, S Ferretti, G D'Angelo https://doi.org/10.1007/s12083-0 21-01220-4

[2] Analysis of Cryptocurrency Transactions from a Network Perspective: An Overview Jiajing Wu, Jieli Liu, Yijing Zhao and Zibin Zheng https://doi.org/10.1016/j.jnca.202 1.103139

[3] Graph Neural Network Based Bitcoin Transaction Tracking Model Zhiyuan Li, Enhan He https://ieeexplore.ieee.org/document/10156845

[4] Biteye: A system for tracking bitcoin Transactions Zhen Li, Jinze Li,Yi Zheng, Baiqiang Dong https://www.techrxiv.org/doi/full/ 10.36227/techrxiv. 19690645.v1

[5] Knowledge Discovery in Cryptocurrency Transactions: A Survey XF Liu, XJ Jiang, SH Liu, CK Tse 10.1109/ACCESS.2021.3062652

[6] DenseFlow: Spotting Cryptocurrency Money Laundering in Ethereum Transaction Graphs Dan Lin, Jiajing Wu, Yunmei Yu, Qishuang Fu, Zibin Zheng, Changlin Yang https://dl.acm.org/doi/abs/10.114 5/3589334.3645692

[7] A Systematic Review of Detecting Illicit Bitcoin Transactions Chang-Yi Lin, Hsiang-Kai Liao, Fu-Ching Tsai https://doi.org/10.1016/j.procs.20 22.09.379