ECCPoW 블록체인의 영지식 증명 검증 구현을 위한 사전 분석

윤민호, 권강륜, 김승민, 이흥노

광주과학기술원

minho.yoon@gm.gist.ac.kr, gryun0330@gm.gist.ac.kr, seungminkim@gm.gist.ac.kr, heungno@gist.ac.kr

Preliminary Analysis for Implementing Zero-Knowledge Proof Verification on ECCPoW Blockchain

Minho Yoon, Gangryun Kwon, Seungmin Kim, Heung-No Lee

Gwangju Institute of Science and Technology (GIST)

요 약

본 논문은 부호이론 기반 합의 알고리즘인 ECCPoW 기반 블록체인에 영지식 증명을 도입하기 위한 기초 연구로서, 블록 검증 과정의 로직을 분석하고 영지식 회로 구현 시의 연산 특성을 검토한다. 이를 위해 블록체인 데이터를 검증하는 Python 코드를 통해 4단계(기본 논리, 난이도, MixHash, LDPC) 검증의 정합성을 확인하고, 각 단계의 연산 특성을 영지식 친화도 관점에서 평가한다. 분석 결과, 산술 연산 위주의 기본 및 난이도 검증은 효율적인 반면, MixHash의 비트 단위 연산과 LDPC의 무작위 행렬 생성(PRPG) 과정이 주요 병목 구간임을 규명한다. 이는 실용적인 시스템 구현을 위해 Lookup Table 활용이나 프로토콜 레벨의 최적화가 필수적임을 시사한다.

Ⅰ. 서 론

2008년에 등장한 최초의 블록체인인 비트코인 [1]은 채굴자들이 무작위 Nonce 값을 소모적으로 대입하여 유효한 블록을 생성하는 합의 알고리즘을 기반으로 동작한다. 그러나 비트코인 채굴 연산에 최적화된 특정 용도용 집적 회로(ASIC)의 등장으로, CPU 및 GPU를 사용하는 일반 노드의 채굴 참여가 확률적으로 불가능에 가까워지는 중앙화 문제가 발생하였다. 이후 등장한 이더리움 [2]은 기존 채굴 문제에 방향성 비순환 그래프(DAG)를 도입하여 이를 해결하고자 했으나, 완전한 ASIC 저항성을 확보하지는 못하였다. 이에 대한 대안으로 제시된 Error Correction Code Proof of Work(ECCPoW) [3]는 저밀도 패리티 검사 부호(LDPC) 디코딩 문제를 활용하여, PoW 기반 블록체인의 주요 해결 과제인 ASIC 저항성을 실현한 합의 알고리즘이다.

ECCPoW 합의 알고리즘을 채택한 신규 블록체인 네트워크가 생태계를 확장하기 위해서는 외부 자본과 유저의 유입이 필수적이다. 이를 위해서는 기존 블록체인에 기록된 유저의 자산을 새로운 블록체인으로 옮겨올 수 있는 크로스-체인 브릿지가 요구된다. 브릿지는 특정 유저가 기존 체인에서 자산을 동결(Lock)시킨 것을 확인하고, 새로운 체인에서 동일한 가치의 토큰을 발행(Mint)하는 역할을 수행한다. 그러나 대부분의 브릿지는 중앙화된 검증 방식에 의존하고 있어, 개인 키 유출로 인한 대규모 해킹 사고가 지속적으로 발생하였다. 이러한 보안 문제를 해결하기 위해, 중앙화된 검증 주체 없이도 영지식 증명(Zero-Knowledge Proof)을 통해 누구나 증명을 생성함으로써, 블록체인의 블록 및 Lock 이벤트를 수학적으로 검증하는 방식이 주목받고 있다 [4].

본 논문에서는 ECCPoW의 블록 검증 과정을 분석하여, 영지식 회로 구현 시 요구되는 연산의 특성과 친화도를 단계별로 평가한다. 이를 통해 실제 영지식 증명 검증 시스템 설계 시 발생하는 비친화적 요소를 구체적으로 식별하여, 실용적인 구현을 위한 토대를 마련한다.

Ⅱ. 본 론

i) ECCPoW 블록체인 분석

ECCPoW 블록체인은 Merge 이전의 PoW 이더리움의 합의 알고리즘인 Ethash를 LDPC 디코딩 문제로 교체하여 구현된다 [3]. ECCPoW 블록체인에서 채굴자는 결정론적으로 생성된 무작위 Parity Check Matrix H와 nonce로 구해진 해시 벡터 v을 바탕으로 LDPC 디코더에 (v, H)를 입력해 출력 c를 얻는다. c가 코드워드 조건을 만족하면 채굴 성공이고, 그렇지 않다면 새로운 nonce 값으로 채굴 과정을 반복하게 된다. 이러한 ECCPoW는 블록체인 네트워크의 난이도가 증가할수록, LDPC 코드워드의 길이가 증가하기 때문에, 고정된 연산 구조를 반복할수록 유리한 ASIC의 설계를 방지한다. 이러한 ECCPoW의 블록체인 헤더는 기존 이더리움의 헤더 내용에 추가로 코드워드 및 그 길이를 포함하여 구성된다.

Algorithm 1 ECCPoW Mining Process

Initialization:
$H \leftarrow$ GenerateParityCheckMatrix()
$N \leftarrow$ GenerateRandomNonce() ▷ Initialize nonce
loop
  $S \leftarrow$ Keccak(N) ▷ Generate seed from current nonce
  $V \leftarrow$ GenerateHashVector(S)
  $W \leftarrow$ Decoding(V, H) ▷ Decode vector V using H
  if W is a valid codeword **then**
    $N_{valid} \leftarrow N$
    return (N_{valid}, W) ▷ Mining Success
  else
    $N \leftarrow$ GenerateRandomNonce() ▷ Mining Failed: Retry
  end if
end loop

그림 1. ECCPoW 채굴 과정 의사코드

ECCPoW 블록체인에서 헤더 내용을 바탕으로 블록을 검증하는 과정은 (1) 기본 논리 검증 (2) 난이도 조절 검증 (3) MixHash 검증 (4) LDPC 검증의 4단계로 구성된다. 우선, 기본 논리 검증은 블록체인의 연결성과 메타데이터의 논리적 정합성을 확인하는 과정이다. 현재 블록 헤더의 parentHash 값이 이전 블록 헤더의 값과 일치하는지, 블록의 타임스탬프가 유효한지, 트랜잭션 처리에 사용된 가스의 총량이 제한을 넘지 않는지 등의 기초적인 검증을 수행한다. 이 이후에는 난이도 조절 검증이 수행된다. ECCPoW 네트워크는 블록 생성 간격을 일정하게 유지하기 위해 이전 블록의 난이도와 생

성 소요 시간을 기반으로 목표 난이도를 동적으로 조절한다. 검증 노드는 정의된 난이도 공식에 따라 현재 시점의 난이도를 계산하고, 이를 블록 헤더에 기록된 실제 난이도 값과 일치하는지 확인한다. 다음으로, MixHash 검증 단계에서는 Nonce와 MixHash를 제외한 블록 헤더의 모든 필드를 Recursive Length Prefix (RLP) 방식으로 인코딩하여 SealHash를 생성한다. 이를 Nonce와 결합하여 Keccak-512 해시를 하여 생성된 Digest가 블록 헤더의 MixHash 값과 일치하는지 확인한다. 마지막으로, 합의 알고리즘의 핵심인 LDPC 검증 과정이 이루어진다. 블록 헤더의 parentHash 값을 시드로 하여 Parity Check Matrix H를 생성한다. 그 후 블록 헤더에 포함된 코드워드를 비트 벡터 v로 변환하고, 행렬 H와의 곱셈 연산($H \cdot v$)을 통해 이 결과가 0이 되는지 검증한다.

본 연구에서는 ECCPoW의 영지식 증명 시스템 설계에 앞서, ECCPoW 프로토콜의 정확한 검증 로직을 재확인하고 연산 특성을 분석하기 위해 Python 기반의 검증기를 구현하였다. 해당 구현체는 실제 ECCPoW 합의 알고리즘을 사용하는 Worldland 메인넷의 블록 데이터(Block #5,832,693)를 대상으로 4단계의 검증을 수행하여, 모든 과정이 정상적으로 통과됨을 확인하였다.

ii) ECCPoW 영지식 구현 분석

구현된 ECCPoW 블록체인의 검증 과정을 영지식 증명 시스템으로 변환하기 위해서는 기존의 검증 로직을 산술 회로 형태로 설계하는 과정이 선행된다. 영지식 증명 시스템에서 주로 사용되는 회로 작성 언어인 Circom은 코드를 Rank-1 Constraint System (R1CS)으로 변환하는데, 이 시스템은 모든 논리적 검증 절차를 유한체 상의 덧셈과 곱셈 연산만으로 구성해야 한다는 제약이 존재한다. 그러므로 기존의 조건문이나 비트 연산 등은 회로 내에서 오버헤드로 이어질 수 있으며, 효율적인 시스템 설계를 위해서 각 검증 단계의 연산 특성을 영지식 친화도 관점에서 분석을 진행하였다.

우선, 처음과 두 번째에 해당하는 과정인 기본 논리 검증과 난이도 조절 검증 단계는 영지식 증명 회로 구성에 있어 친화적이다. 타임스탬프 유효성 확인, 가스 한도 검사, 난이도 목표값 계산 등은 정수의 사칙연산과 크기 비교만으로 이루어져 있다. 비교 연산을 산술 연산으로 변환하기 위해서는 추가적인 영지식 비교기 회로가 요구되지만, 제약 조건의 수가 크게 증가하지 않는 영지식 친화적인 연산이다. 따라서 이 단계들은 회로의 크기를 과도하게 늘리지 않아, 안정적으로 영지식 구현이 가능하다.

Operation	Constraints
Is Equal	'A = B ?' -> '(A-B) = 0 ?' $x \times out = 0$ $x \times inv = 1 - out$
Bitwise	• Constraint 1: Reconstruction $x = \sum_{i=0}^{k-1} 2^i \cdot b_i$ • Constraint 2: Boolean Check $\forall i \in [0, k-1]: \quad b_i \cdot (b_i - 1) = 0$

그림 2. 비산술 연산의 R1CS 제약 조건 변환 예시.

다음으로, MixHash 검증 단계는 영지식 비친화적인 단계이다. 이 과정의 핵심인 Keccak-512 해시 함수는 입력값을 비트 단위로 쪼개어 XOR, AND, NOT 및 비트 회전과 같은 논리 연산을 수십 라운드 반복 수행한다. 유한체 기반의 산술 회로에서 이러한 비트 연산을 처리하기 위해서는 하나의 필드 요소를 비트 단위로 분해하고 이를 다시 검증하는 복잡한 과정이 필요하다. 이러한 이유로, 해시 함수가 사용되는 검증 과정은 수많은 제약 조건을 생성하게 하여 증명 생성 시간을 크게 증가시킨다. 따라서 효율적인 구현을 위해서는 비트 연산의 결과를 미리 계산해 둔 Lookup Table을 활용하여 회로 내 연산량을 줄이거나, 장기적으로는 Keccak-512를 Poseidon과 같은 영지식 친화적인 해시 함수로 대체하는 등의 최적화 전략 도입이 요구된다.

마지막으로, LDPC 검증 단계도 영지식 비친화적인 단계이다. Parity Check Matrix H와 코드워드 벡터 v의 곱셈 연산을 영지식 회로의 덧셈과 곱셈 게이트로 변환하는 과정 자체는 용이하다. 그러나 모듈러 2 연산을 기반으로 하는 LDPC 코드워드 검증과 달리, 영지식 회로는 큰 소수 유한체에서 동작하기 때문에, 변수가 0 또는 1의 값만 가지도록 강제하는 이진 제약 조건을 모든 행렬 요소에 추가해야 하는 비용이 존재한다. 또한, 행렬 H가 검증 과정에서 영지식 비친화적인 Pseudo-Random Pattern Generator (PRPG) 과정을 통해 생성된다. PRPG 로직을 회로 내부에서 그대로 구현할 경우 막대한 오버헤드가 발생한다. 이러한 비효율성을 개선하기 위해서는 회로 내부에서 H 행렬을 매번 생성하는 대신, 블록 헤더에 H 행렬에 대한 정보를 미리 포함하도록 프로토콜을 수정하여 회로 구성을 간소화하는 구체적인 최적화 설계가 필요하다.

Ⅲ. 결 론

본 논문은 ECCPoW 블록체인에 영지식 증명을 도입하기 위한 선행 연구로서, 블록 검증 과정의 로직을 분석하고 각 단계별 연산 특성과 영지식 회로 구현 시의 친화도를 평가하였다. 연구를 위해 구현된 Python 검증기를 통해 메인넷 데이터에 대한 4단계(기본 논리, 난이도 조절, MixHash, LDPC) 검증의 정확성을 확인하였으며, 이를 바탕으로 회로 변환 시의 효율성을 검토하였다.

분석 결과, 단순 산술 연산 위주의 기본 논리 검증과 난이도 조절 검증은 영지식 증명 회로 구성에 효율적인 것으로 나타났다. 반면, MixHash와 LDPC 검증은 높은 연산 비용을 유발하는 비친화적 요소로 식별되었다. 따라서 실용적인 시스템 구현을 위해서는 Lookup Table 활용, 영지식 친화적 해시 함수 도입, 그리고 행렬 생성 과정의 개선과 같은 최적화 전략을 통해 연산 오버헤드를 절감해야 한다.

본 연구에서 도출된 분석 결과와 제안된 최적화 방안은 향후 중앙화된 검증 주체 없이도 안전하게 자산을 이동할 수 있는 ECCPoW 기반의 탈중앙화 크로스-체인 브릿지 설계를 위한 토대가 될 것으로 기대된다.

ACKNOWLEDGMENT

본 연구는 과학기술정보통신부 및 정보통신기획평가원의 대학 ICT 연구센터지원사업의 연구결과로 수행되었음 (IITP-2026-RS-2021-II211835) 그리고 이 성과는 정부(과학기술정보통신부)의 재원으로 한국연구재단의 지원을 받아 수행된 연구임. (RS-2025-22932973)

참 고 문 헌

[1] S. Nakamoto. (2008). Bitcoin Whitepaper. [Online]. Available: https://bitcoin.org/bitcoin.pdf

[2] V. Buterin. (2014). Ethereum Whitepaper. [Online]. Available: https://ethereum.org/content/whitepaper/whitepaper-pdf/Ethereum_Whitepaper_-_Buterin_2014.pdf

[3] H. Kim, J. Jang, S. Park and H. -N. Lee, "Error-Correction Code Proof-of-Work on Ethereum," in IEEE Access, vol. 9, pp. 135942-135952, 2021.

[4] T. Xie et al., “ZkBridge: Trustless cross-chain bridges made practical,” in Proc. ACM SIGSAC Conf. Comput. Commun. Secur., New York, NY, USA, 2022, pp. 3003-3017.