BlankChain: 스토리지 및 네트워크 비용 효율적인 초확장 블록체인

박상현 [O1] 박보성 [2] 문수묵 [1]

[1] 서울대학교 전기·정보공학부

[2](주)카카오

lukepark@snu.ac.kr, elin.97@kakaocorp.com, smoon@snu.ac.kr

BlankChain: Storage and Network Cost-efficient Ultra-scalable Blockchain

Sanghyeon Park[O1] BoSung Park[2] Soo-Mook Moon[1]

[1]Department of Electrical and Computer Engineering, Seoul National University

[2]Kakao Corp.

요 약

최근 등장하는 블록체인은 트랜잭션 처리 성능 향상을 위한 여러 장치가 포함되어 있으나, 초당 트랜잭션 처리량만을 중시해 저장용량과 네트워크 자원이 낭비되곤 한다. BlankChain은 과도한 트랜잭션 확장성을 달성하기 위해 빈 블록이 자주 형성되는 어느 블록체인에도 적용할 수 있는 방법론으로, 빈 블록을 정숫값으로 치환함으로써 스토리지 및 네트워크 비용 효율적이도록 개선한다.

1. 서 론

블록체인은 중앙화된 주체가 없는 분산 환경에서도 합의알고리즘을 통해 상태의 합의를 이룰 수 있게 하는 기술이다. 블록체인은 주로 비트코인(Bitcoin)과 같은 지불(payment) 시스템으로 활용되었는데[1], 분산 컴퓨팅 플랫폼을 표명한 이더리움(Ethereum)의 등장 이후부터는 탈중앙화 응용프로그램(Decentralized Application, DApp)의 기반 시스템으로 재조명받기 시작하였다[2].

블록체인이 지불 혹은 컴퓨팅 플랫폼으로써 유의미한 역할을 수행하기 위해서는 최소한 종래의 시스템과 유사한 성능을 제공해야 한다는 인식 하에, 많은 신규 블록체인들이 높은 TPS(Transaction Per Second, 초당 트랜잭션 처리량)를 확보하고자 제시되었다. 그러나 TPS 달성만을 목표해 다른 종류의 확장성인 스토리지 확장성, 네트워크 비용 확장성 측면을 고려하지 못한 경우가 많다.

특히 발생 트랜잭션의 총량이 작지만, 더 짧은 블록 생성 간극을 가진 블록체인에서는 자원 낭비의 문제가 극심하게 나타난다. 이러한 블록체인에서는 간헐적으로 혹은 자주 트랜잭션을 포함하지 않는 빈 블록(empty block)이 생성된다. 빈 블록은 어떠한 데이터도 포함하지 않기에 사실상 저장용량만을 차지하는 잉여 데이터에 해당한다. 그러나 빈 블록을 함부로 배제하는 것은 합의알고리즘의 무결성을 해치며, 결국 보안에 악영향을 끼치게 되므로 임의 삭제가 불가능하다.

본 연구에서는 빈 블록을 저장 및 전파하지 않으면서도 시스템의 보안성을 해치지 않는 방법을 제시한다. 이를 통해 저장용량의 절감 및 네트워크 자원 낭비를 방지했음을 보인다. 나아가 사용량에 따른 높은 TPS의 확보와 재조정, 축소가 용이함을 보인다. 또한, 이러한 방법론을 적용한 블록체인인 BlankChain을 제시한다. BlankChain은 특히 연산, 스토리지, 네트워크 자원이 한정된 사물인터넷(Internet of Things, IoTs) 기기에 널리 적용될 것으로 기대된다.

2. 온체인 분석 및 비교

본 장에서는 핵심 구조가 유사한 독립적인 두 블록체인인 이더리움과 클레이튼(Klaytn)의 온체인(on-chain) 데이터 분석을 통해, 확장성을 확보하기 위해 저장용량과 네트워크 자원의 낭비가 있음을 보인다.

* 이 논문은 2021년도 정부(과학기술정보통신부)의 재원으로 정보통신기획평가원의 지원을 받아 수행된 연구임(No.2021-0-00136, 다양한 산업 분야 활용성 증대를 위한 대규모/대용량 블록체인 데이터 고확장성 분산 저장 기술 개발)

블록 번호	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19
이더리움	218	194	322	195	148	253	340	352	262	304	205	261	221	230	160	286	271	307	311	384
조정	16	14	24	15	11	19	26	27	20	23	15	20	17	17	12	22	20	23	23	29
클레이튼	3	1	1	1	4	2	2	3	0	4	2	2	1	2	1	2	1	2	3	2

표 1 블록당 이더리움의 트랜잭션 수, 조정된 이더리움의 트랜잭션 수, 그리고 클레이튼의 트랜잭션 수
(이더리움 블록 번호 오프셋 12,282,023; 클레이튼 블록 번호 오프셋 57,419,120)

2.1 이더리움

이더리움은 오늘날 가장 사용량이 많은 블록체인 중 하나로, 재화의 거래뿐만 아니라 다양한 탈중앙화 응용프로그램의 구동을 위해 전세계적으로 널리 활용되고 있다. 그만큼 온체인 트랜잭션의 총 개수가 많으며, 특정 시간대의 구분 없이 전반적으로 고르게 트랜잭션 발생량이 분포되어 있다.

이더리움은 2021년 5월을 기준으로 평균 블록 생성 간극이 13초를 전후하고 있다. 전체 네트워크에서 블록 생성에 참여하는 노드의 숫자가 증가하거나 줄어들어도, 블록 생성 난이도를 조정하는 식으로 이 간극을 유지하고자 하는 자가제한 시스템(self-regulating system)이다. 이로부터 높은 수준의 보안성을 제공한다.

2.2 클레이튼

클레이튼은 이더리움의 소스코드를 복제(fork)해 만들어진 블록체인으로, 합의알고리즘 등의 일부 수정을 적용해 실산업에 적용할 수 있는 높은 수준의 확장성을 확보하고자 한 블록체인이다[3]. 클레이튼은 높은 TPS와 완결성을 제공하기 위해 평균 블록 생성 간극을 1초로 유지한다.

클레이튼은 한국 사용자들이 주로 사용하며, 이에 따라 사용자 수가 적어지는 새벽 시간대에는 트랜잭션 발생량이 희소해지는 경향이 있다. 짧은 블록 생성 간극과 적은 트랜잭션 발생은 곧 비어있거나 낮은 블록당 트랜잭션으로 이어진다.

2.3 온체인 데이터 분석

표 1은 이더리움과 클레이튼의 블록 당 트랜잭션 개수를 직접 비교한 것이다. 모니터링 도구로 이더리움은 Etherscan을[4], 클레이튼은 KlaytnScope를 이용했다[5]. 이더리움은 12,282,023번째 블록에서부터 12,282,042번째 블록까지를, 클레이튼은 57,419,120번째 블록에서부터 57,419,139번째 블록까지의 블록 당 트랜잭션 개수를 추적해 나타냈다. 조정된 트랜잭션 수는 이더리움의 블록 당 트랜잭션 수를 이더리움 평균 블록 생성 간극으로 나눈 후, 클레이튼의 평균 블록 생성 간극으로 곱한 수치이다. 이 조정된 트랜잭션 수를 클레이튼 트랜잭션 수와 비교해 블록체인의 활성화 정도를 비교할 수 있다.

	이더리움	클레이튼
전체 블록 수	12,336,652	57,994,460
전체 트랜잭션 수	1,092,255,756	179,820,269
평균 블록 간극	13.34s	1.01s
평균 TPS	16.39	5.93
평균 블록 크기	52,576 Bytes	4,855 Bytes
평균 블록당 트랜잭션 수	218.6	6.01

표 2 이더리움과 클레이튼의 온체인 데이터 비교
(평균 수치는 최근 7일간의 데이터로 산출)

관측이 진행된 4월 23일 새벽 5시경을 기준으로, 이더리움과는 대조적으로 클레이튼은 그 활성화 정도가 매우 낮으며, 대부분의 블록이 2개 내외의 트랜잭션만을 포함하고 있음을 알 수 있다. 심지어 57,419,128번째 블록의 경우 아예 트랜잭션을 포함하고 있지 않은 빈 블록임을 알 수 있다.

이러한 차이는 블록체인 전체 온체인 데이터를 가지고 비교했을 경우 더 극심하게 나타난다. 표 2는 이더리움과 클레이튼의 온체인 분석 결과의 요약이다.

클레이튼의 평균 블록 생성 간극이 이더리움 대비 약 13배가량 빠르므로, 늦게 등장한 블록체인이지만 전체 블록 수가 훨씬 많은 것을 알 수 있다. 이는 블록 헤더를 저장하기 위한 더 많은 저장용량과 블록을 전파하기 위한 더 많은 네트워크 자원을 요구한다는 의미이다.

또 하나 주목할만한 점은, 클레이튼 네트워크가 높은 TPS를 제공하기 위해 빠른 블록 생성 속도를 확보했음에도 불구하고 이더리움보다 낮은 TPS를 보인다는 점이다. 이는 사용자 수가 네트워크의 역량에 비해 충분히 확보되지 못한 상황임을 의미한다. 반대로 말하자면 현재 수요에 비해 컴퓨팅 자원이 과도하게 들어가고 있다는 의미이다.

3. BlankChain

본 연구에서 제시하는 BlankChain 구조는 저장용량과

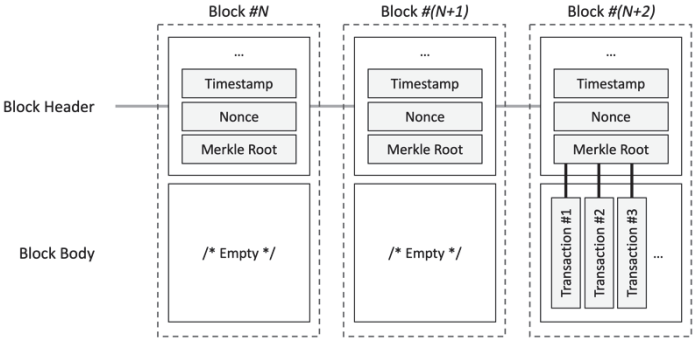

그림 1 일반적인 블록체인의 도식

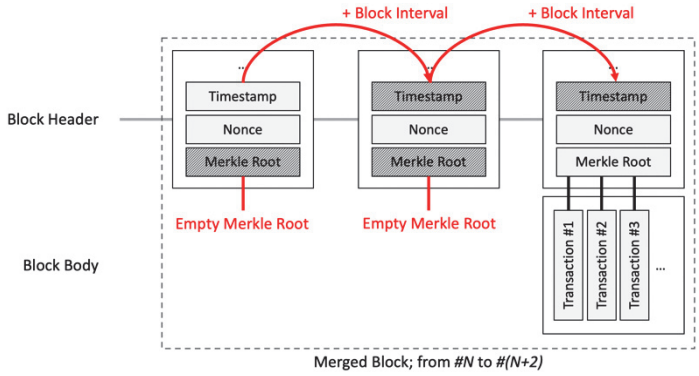

그림 2 BlankChain의 도식

네트워크 자원의 낭비를 줄이고, 필요에 따른 확장성 증감이 가능한 블록체인이다. 이러한 속성을 확보하기 위해 통합 블록(Merged Block) 개념을 도입한다.

3.1 블록 구조

그림 1에서처럼 블록은 블록 헤더(header)와 블록 바디(body)로 구성된다. 블록 헤더는 타임스탬프와 같이 블록체인의 메타데이터와 합의를 만족시키기 위한 값인 논스(nonce), 그리고 블록 바디의 요약 정보인 머클 루트(Merkle root) 등으로 구성된다. 한편 블록 바디는 트랜잭션 혹은 상태의 묶음으로 구성된다.

블록 해시(hash)는 블록 헤더를 입력으로 하는 암호화 해시함수의 출력이다. 이전 블록의 블록 해시를 본 헤더에 포함함으로써 체인 구조를 형성, 과거 블록의 임의 위조 및 변조를 막는 것이 보안의 핵심이다.

빈 블록의 경우 트랜잭션이 없으므로 블록 바디 부분이 비워져 있다. 이 경우 빈 블록의 머클 루트에 해당하는 값이 블록 헤더 필드에 채워진다. 빈 블록 역시 일반적인 블록처럼 채굴 과정을 거쳐 생성 및 전파 후 블록체인에 연결되며, 저장용량을 차지한다.

3.2 통합 블록 구조

통합 블록은 연속된 빈 블록들과, 뒤이어 존재하는 비어있지 않은 블록 하나로 구성된 묶음이다. BlankChain에서 채굴자는 빈 블록을 채굴할 경우 일반적인 블록 채굴과는 다른 행동을 취한다.

- 전파받은 트랜잭션이 존재하지 않는 경우: 다음 블록 역시 빈 블록으로 채굴을 시작하기 때문에, 다음 블록의 블록 헤더 값들을 논스를 제외하고는 결정론적으로 고정한다.
- 전파받은 트랜잭션이 존재하는 경우: 이어 채굴할 다음 블록은 트랜잭션이 포함된 블록이므로 일반적인 블록 채굴 방법을 따른다. 타임스탬프는 프로토콜에 의해 결정론적으로 정해진다.

BlankChain의 통합 블록 형성 과정에서 결정론적으로 구해지는 값들은 다음과 같다.

- 머클 루트: 빈 블록의 머클 루트는 항상 동일하다.
- 타임스탬프: 타임스탬프는 처음으로 채굴한 빈 블록에서부터, 프로토콜이 목표로하는 블록 간극을 더해가며 형성된다. 가령 블록 생성 간극을 1초에 맞추는 자가제한 시스템인 경우, 한 통합 블록 안에서 타임스탬프는 1씩 증가한다.

위와 같은 절차를 따르면, 통합 블록에서는 블록 생성에 요구된 논스 값을 제외하고는 불필요한 정보를 모두 삭제할 수 있다. 또한, 그림 2에서처럼 네트워크를 통해 전송하지 않아도 상대 노드가 로컬(local) 정보만으로 계산해 사용할 수 있다. 따라서 BlankChain에서는 오직 정수인 논스들로 구성된 리스트를 추가 전송하는 것으로 빈 블록 전송을 대체할 수 있으며, 또한 저장용량을 적게 소모할 수 있어 자원 효율적이다.

만일 확장성 증대가 요구될 경우 빈 블록의 개수가 줄어들어 통합 블록이 자연스럽게 와해될 것이며, 반대로 확장성 감소가 요구될 경우 통합 블록이 포함하는 빈 블록의 수가 많아지는 것으로 자연스럽게 네트워크 상황에 대응할 수 있다. 이를 통해 양방향 확장성에 대응하는 초확장 블록체인의 구현이 가능하다.

4. 결론 및 향후 연구

Blankchain은 저장용량과 통신 자원을 소모하는 빈 블록의 존재를 정수만으로 치환하는 방법을 제시한다. 이 방법을 적용하면 무의미한 정보인 빈 블록을 통합 블록의 형태로 압축해 저장 및 전송할 수 있다. 또한, 자연스럽게 확장성 증대 및 감소에 대응할 수 있다. 이를 통해 IoT 장비 등 하드웨어 자원이 부족한 환경에서도 블록체인의 원활한 구동이 가능하다.

참 고 문 헌

[1] Nakamoto Satoshi, “Bitcoin: A peer-to-peer electronic cash system,” 2008.
[2] Buterin Vitalik, “Ethereum Whitepaper,” 2013.
[3] Klaytn, “Position Paper v2.1,” 2021.
[4] Etherscan, https://etherscan.io
[5] Klaytnscope, https://scope.klaytn.com