고성능 분산 합의 알고리즘 동향 분석

Trend Analysis of High-Performance Distributed Consensus Algorithms

진희상 (H.-S. Jin, jinhs@etri.re.kr) 블록체인연구실 연구원
김동오 (D.-O. Kim, dokim@etri.re.kr) 블록체인연구실 책임연구원
김영창 (Y.-C. Kim, zerowin@etri.re.kr) 블록체인연구실 선임연구원
오진태 (J.-T. Oh, showme@etri.re.kr) 블록체인연구실 책임연구원/실장
김기영 (K.-Y. Kim, kykim@etri.re.kr) 블록체인·빅데이터연구단 책임연구원/단장

ABSTRACT

Recently, blockchain has been attracting attention as a high-reliability technology in various fields. However, the Proof-of-Work-based distributed consensus algorithm applied to representative blockchains, such as Bitcoin and Ethereum, has limitations in applications to various industries owing to its excessive resource consumption and performance limitations. To overcome these limitations, various distributed consensus algorithms have appeared, and recently, hybrid distributed consensus algorithms that use two or more consensus algorithms to achieve decentralization and scalability have emerged. This paper introduces the technological trends of the latest high-performance distributed consensus algorithms by analyzing representative hybrid distributed consensus algorithms.

KEYWORDS 분산 컴퓨팅, 분산 합의 알고리즘, 블록체인

Ⅰ. 서론

비트코인[1]의 등장으로 주목받기 시작한 블록체인 기술은 다양한 분야에서 고신뢰 기술로 주목받고 있다[2]. 특히, 2008년 리먼 브라더스 사태와 같은 제3의 신뢰 기관(TTP: Trusted Third Party)의 부정, 부패 등으로 인해 중앙화된 시스템에 대한 불신이 초래했으며, 중앙화된 시스템에서 벗어나 다수의 참여자가 직접 분산화된 원장을 운영하는 탈중앙화된 원장 기술 개발이 활발해졌다[3].

비트코인은 최초의 실용 가능한 수준의 탈중앙화 화폐 거래 기술로 주목받았고, 이더리움은 스마트 컨트랙트 개념을 도입하여 화폐뿐만 아니라 다양한 전자 계약이 가능한 탈중앙화 정보 거래 플랫폼으로 주목받았다[4].

하지만 비트코인과 이더리움의 분산 합의 알고

* DOI: https://doi.org/10.22648/ETRI.2022.J.370107
* 이 논문은 2021년도 정부(과학기술정보통신부)의 재원으로 정보통신기획평가원의 지원을 받아 수행된 연구임[No. 2021-0-00118, 대규모 노드를 위한 탈중앙화 합의체 구성 기술 개발].

리즘인 PoW(Proof of Work)는 컴퓨팅 연산 기반의 채굴 방식으로 인한 자원 낭비 및 성능 한계로 다양한 산업 분야 적용에 한계가 있었다[5]. 게다가 대형 마이닝풀(Mining Pool)의 등장으로 소수가 채굴을 독점함으로써 재중앙화 문제가 제기되었다[3].

PoW의 과도한 자원 소모 및 성능 한계의 해결을 위한 PoS(Proof of Stake), BFT(Byzantine Fault Tolerance) 등의 다양한 분산 합의 알고리즘이 등장했다[6]. 그러나 PoS 계열의 알고리즘은 과도한 자원 소모는 없지만 보유 자산으로 인한 재중앙화 문제가 발생했고, BFT 계열 알고리즘은 성능은 향상되었지만 합의 부하로 인한 확장성 문제가 발생했다[7].

이를 해결하기 위해, 최근에는 두 개 이상의 합의 알고리즘을 함께 사용하여 탈중앙성 및 확장성을 모두 달성하는 것을 목표로 하는 하이브리드 방식의 분산 합의 알고리즘들이 등장하였다[8-14].

하이브리드 방식의 분산 합의 알고리즘은 합의체를 선정하는 과정과 합의체가 블록을 합의하는 과정을 구분하고 각각 다른 합의 알고리즘을 사용하는 방식이다. 하이브리드 방식의 분산 합의 알고리즘은 확장성과 탈중앙성 중 확장성을 중요시하는 확장성 중심 합의 알고리즘과 탈중앙성을 중요시하는 탈중앙성 중심 합의 알고리즘으로 구분할 수 있다.

확장성 중심 합의 알고리즘은 PoS, DPoS(Delegated Proof of Stake) 등을 활용하여 고정된 소수 노드를 합의체로 선정한 후 BFT 계열의 합의를 수행하는 알고리즘이다. 확장성 중심 합의 알고리즘을 사용하는 대표적인 플랫폼으로는 Klaytn[8], Solana[9] 등이 있으며, 이는 소수 노드로 확장성이 높지만 고정된 합의체로 인해 탈중앙성이 낮으며 네트워크 공격에 취약하다.

탈중앙성 중심 합의 알고리즘은 VRF(Verifiable Random Function) 등을 활용하여 전체 참여 노드 중 일부를 임의로 뽑아 합의체로 선정한 후 BFT 계열의 합의를 수행하는 알고리즘이다. 탈중앙성 중심 합의 알고리즘을 사용하는 대표적인 플랫폼으로는 Algorand[10], PoN(Proof of Nonce)+BADA(Byzantine Agreement among Decentralized Agents)[11] 등이 있으며, 이는 합의체 수가 변동적이며 정해진 라운드마다 합의체가 임의로 변경되기 때문에 탈중앙성이 높고 네트워크 공격에 강하지만 합의체 선정 과정의 부하로 인한 성능 한계가 존재한다.

본고에서는 최근 사용되는 대표적인 확장성 중심 합의 알고리즘과 탈중앙성 중심 합의 알고리즘을 상세히 소개함으로써 하이브리드 방식의 분산 합의 알고리즘의 기술 동향을 분석한다.

본고의 구성은 다음과 같다. Ⅱ장에서는 확장성 중심 합의 알고리즘을 사용하는 Klaytn과 Solana를 소개하고, Ⅲ장에서는 탈중앙성 중심 합의 알고리즘을 사용하는 Algorand와 PoN+BADA를 소개한다. 마지막으로 Ⅳ장에서는 결론을 맺는다.

Ⅱ. 확장성 중심 합의 알고리즘

1. Klaytn

Klaytn[8]은 그라운드엑스가 2018년 공개한 하이브리드 블록체인 플랫폼이다. Klaytn은 허가된(Permissioned) Council 노드들이 블록 생성 및 합의를 담당하며, 공개된 네트워크를 통해서는 트랜잭션을 발생시키거나 블록 데이터를 읽을 수 있다.

Klaytn은 사전에 허가된 Council 노드만이 합의체가 될 수 있으며, 이중 임의로 선정된 합의체 노드들은 IBFT(Istanbul Byzantine Fault Tolerance)[15] 합의 알고리즘을 통해 블록을 합의한다. 이때 선정된 합의체 노드들을 Committee 노드라고 한다. Klaytn은 이러한 합의 방식을 통해 4,000TPS(Transaction per

Second)의 트랜잭션 처리 성능과 1초의 블록 생성 시간을 달성한다.

Committee 노드는 후보 블록을 생성하는 블록 제안 노드와 제안된 후보 블록을 검증하는 블록 검증 노드로 나뉜다. 블록 제안 노드는 Council 노드 중에서 임의의 순서로 정해지며, 토큰 예치 수량이 많을수록 더 많이 블록 제안 노드가 된다. 블록 제안 노드는 자신이 생성한 블록이 확정되면 보상을 받는다.

Klaytn은 많은 토큰을 가진 노드가 블록 생성을 독점하는 것을 막고 적은 토큰을 가진 노드에도 블록 생성 기회를 주기 위해 지니(Gini) 계수 G를 통해 선정 확률을 보정한다. 지니 계수 G가 포함된 블록 제안 노드의 선정 수식은 (1)[8]과 같다. 선정 수식에 따라 3,600블록(약 1시간) 동안의 블록 제안 노드 순서가 결정된다.

$$S_{new} = S^{\frac{1}{1+G}}$$

$$G = \frac{\sum_{i=0}^{n}\sum_{j=0}^{n} abs(S_i - S_j)}{2 \cdot n^2 \cdot mean(S)} \quad (1)$$

Klaytn에서 블록 검증 노드의 수는 22개(3f+1, f=7)로 고정되어 있으며[8], 현재 블록의 블록 제안 노드와 다음 블록의 블록 제안 노드는 항상 포함된다. 나머지 20개의 노드는 Council 노드 중에서 이미 포함된 2개의 노드를 제외하고 임의로 선정된다. 블록 검증 노드에 다음 블록 제안 노드를 포함하는 이유는 블록 전파 시간을 단축하기 위함이다.

선정된 블록 검증 노드들은 블록 제안 노드가 생성한 블록을 IBFT 합의 알고리즘을 통해 합의한다. IBFT는 그림 1과 같이 PBFT(Practical Byzantine Fault Tolerance)를 블록 단위로 바꾸고 Pre-prepare, Prepare, Commit의 3단계로 줄여 블록체인에 맞게 변형한 BFT 합의 알고리즘이다.

Pre-prepare 단계에서 블록 제안 노드(Proposer)는 블록을 생성하여 모든 블록 검증 노드(Validator)에

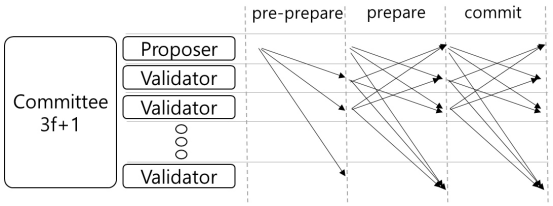

그림 1 IBFT 블록 합의 과정

전달한다. Prepare 단계에서 블록 검증 노드는 블록을 검증한 후 서명하여 모든 블록 검증 노드에 전달한다. Commit 단계에서 블록 검증 노드는 정족수에 해당하는 15개(2f+1, f=7)의 노드로부터 Prepare 메시지를 받은 후 서명하여 모든 검증 노드에 전달한다. 각 블록 검증 노드는 15개의 노드로부터 Commit 메시지를 받으면 블록을 확정 짓는다. 이때 블록 검증 노드 간 주고 받는 메시지의 복잡도는 $O(n^2)$이다.

Klaytn은 사전에 허가된 노드들만이 Council 및 Committee가 될 수 있고 합의 노드 수가 고정되어 있기 때문에 완전히 탈중앙화된 블록체인 네트워크는 아니다[8]. 또한 1시간 동안의 블록 제안 노드 순서가 미리 결정되기 때문에 블록 제안 노드를 예측하여 집중적으로 공격하는 방식에 취약하다고 할 수 있다.

2. Solana

Solana[9]는 2017년부터 개발된 퍼블릭 블록체인 플랫폼이다. Solana는 예치 방식의 PoS를 기반으로 리더 노드 및 검증 노드를 선정하며, 이후 TowerBFT를 사용하여 블록을 합의한다.

또한 트랜잭션의 발생 순서를 검증 가능하도록 하기 위해 이벤트에 대한 검증값을 생성하는 역사 증명(PoH: Proof of History) 알고리즘을 사용한다. Solana는 테스트넷에서 50,000TPS의 트랜잭션 처

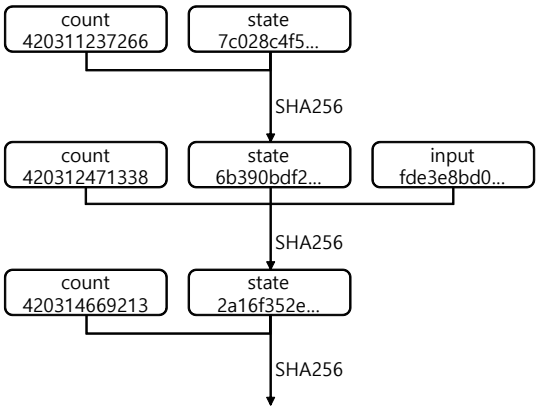

그림 2 PoH 생성 과정

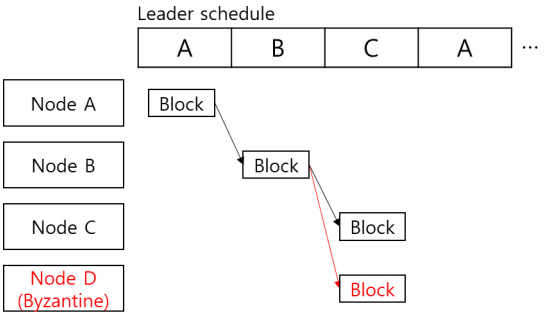

그림 3 Solana 블록 합의 예시

리 성능과 400ms의 블록 생성 시간을 달성한다.

Solana의 노드들은 끊임없이 SHA256 해시 함수를 수행하여 연속된 해시 체인 형태의 PoH를 생성한다. Solana는 GCP n1-standard 하드웨어 및 xeon e5-2520 v4 기준으로 200만 번의 해시 함수를 수행하면 1초가 되는 것으로 설정했다[16]. Solana의 리더 노드는 자신의 상태값(State)과 논리 시간(Count) 및 트랜잭션(Input)을 해시하고 해시값을 다시 입력값으로 해시하는 형태를 반복하여 PoH를 생성하며, 생성된 PoH를 통해 트랜잭션들의 순서 및 발생 시점을 증명할 수 있다.

그림 2는 PoH의 생성 예시를 보여준다. state(7c028c4f5...)와 count(420311237266)를 입력값으로 다음 state(6b390bdf2...)를 만들고, state와 그때의 count(420312471338)와 input(fde3e8bd0...)을 입력으로 다음 state(2a16f352e...)를 만든다. 이렇게 만들어진 PoH는 state(6b390bdf2...) 시점에 이벤트 input(fde3e8bd0)가 발생한 것을 증명한다.

Solana는 노드들이 예치한 담보금을 기준으로 리더 노드 순서를 결정하고 정해진 순서대로 한 노드씩 리더 노드가 된다. 이때 한 리더 노드가 지속되는 시간 주기를 슬롯이라고 한다. 정해진 리더 목록이 지속되는 시간은 1epoch라고 하며, 1epoch은 432,000슬롯 동안 지속된다. Solana의 블록 생성 간격이 약 400ms임을 고려했을 때 1epoch는 약 2~3일 동안 지속된다.

나머지 노드들은 검증 노드가 되어 리더가 생성한 블록을 검증하며, 검증 노드들의 지분의 합이 2/3를 초과하면 블록 검증이 완료된다. 만약 자신이 리더가 아닐 때 블록을 생성한다면 PoH에 의해 발각되고 담보금을 빼앗기게 된다.

그림 3은 Solana의 블록 합의 예시를 보여준다. 4개의 노드 A, B, C, D(D는 비잔틴 노드)가 존재하며 리더 스케줄링을 통해 리더 노드 순서가 A-B-C-A로 결정됐다고 가정한다.

첫 번째 리더인 노드 A는 자신의 슬롯 시간에 블록을 생성한다. 블록은 슬롯 시간 동안 수행한 트랜잭션이 포함된 PoH를 의미한다. 이때 노드 B, C, D도 마찬가지로 트랜잭션이 포함되지 않은 PoH를 생성하면서 자신이 리더가 될 때까지 기다린다. 생성된 블록은 다른 검증자 노드들로부터 2/3 이상의 검증을 받아 확정된다.

두 번째 슬롯에서 리더가 된 노드 B는 노드 A가 보낸 PoH의 마지막 state를 포함하여 자신의 트랜잭션을 포함한 PoH를 생성한다. 이는 노드 A가 생성한 트랜잭션 이후에 노드 B의 트랜잭션이 생성

되었음을 검증 가능하도록 한다.

마찬가지로, 세 번째 슬롯이 되면 리더 노드인 노드 C는 노드 B의 PoH의 마지막 state를 포함하여 자신의 트랜잭션을 포함한 PoH를 생성한다.

만약 비잔틴 노드 D가 세 번째 슬롯에서 노드 B의 블록 이후에 자신의 블록을 생성한다면, 이는 PoH에 의해 자신이 리더가 아닐 때 블록을 생성했음이 밝혀진다. 만약 비잔틴 노드 D가 네 번째 슬롯에서 노드 B의 블록 이후에 자신의 블록을 생성하더라도 노드 C의 PoH가 이미 전파되었기 때문에 노드 D의 블록은 무효화된다.

Solana는 리더 노드 및 검증 노드가 미리 결정되고 공개되어 있기 때문에 탈중앙성이 낮으며 네트워크 공격에 취약하다. 실제로 Solana는 2021년 9월 DDoS 공격으로 인한 17시간 동안의 네트워크 중지 문제가 있었다.

Ⅲ. 탈중앙성 중심 합의 알고리즘

1. Algorand

Algorand[10]는 블록체인의 트릴레마인 탈중앙화, 확장성, 안전성의 3중 딜레마 해결을 목표로 미국 MIT 교수인 Silvio Micali가 개발한 블록체인 플랫폼이다. Algorand의 분산 합의 알고리즘은 무허가형(Permissionless) 순수지분증명(PPoS: Pure Proof of Stake)으로 블록체인 네트워크에 참여한 노드들 중에서 VRF를 통해 합의체(블록 생성자 및 위원회)를 선정한다. 그 후, 선정된 합의체는 BFT 기반의 BA★(Byzantine Agreement★) 알고리즘을 수행하여 블록을 합의한다. 이를 통해 Algorand는 900TPS의 트랜잭션 처리 성능과 약 10~22초의 블록 생성 시간을 달성한다.

Algorand는 시빌어택(Sybil Attack) 방지, 수백만 노드를 수용 가능한 확장성 제공, DDoS(Distributed Denial of Service)와 같은 합의체 대상 공격 방지의 해결책을 제시한다.

첫 번째, 시빌어택을 방지하기 위해 블록체인 네트워크의 2/3 이상이 정직한 노드라고 가정한 상황에서 노드의 지분을 기반으로 가중치를 부여하고 가중치를 기반으로 블록 합의에 참여하는 블록 생성자 및 위원회를 선정한다.

두 번째, 확장성을 제공하기 위해 모든 노드가 합의에 참여하는 것이 아니라 노드의 가중치 기반으로 선정된 일부 노드가 합의를 진행한다.

마지막으로 합의체 대상 공격 방지를 위해 합의 노드를 특정할 수 없도록 합의 과정의 각 세부 단계마다 합의 노드를 교체한다.

Algorand의 블록 합의는 1) VRF(검증 가능 함수) 기반의 암호학적 추첨(Cryptographic Sortition)을 통해 블록 생성자 및 위원회를 선정하는 과정, 2) 선정된 블록 생성자가 후보 블록을 생성하는 과정, 3) 선정된 위원회가 다수의 후보 블록 중에서 하나의 블록에 대한 합의를 도출하는 BA★ 과정으로 구성된다.

블록 생성자 및 위원회를 선정하는 과정은 그림 4의 추첨(Sortition) 알고리즘에 의해 수행된다. 모든 노드는 자신의 지분에 따른 가중치 w를 부여받으며, 추첨을 통해 예상되는 선정 수를 나타내는 상수값 τ와 가중치 합 W를 통해 계산된 p=τ/W 확률로 블록 생성자 또는 위원회에 선정될 확률을 갖

$$< hash, \pi > \leftarrow VRF_{sk}(seed||role)$$
$$p \leftarrow \tau/W$$
$$j \leftarrow 0$$
$$\textbf{while } \frac{hash}{2^{hashlen}} \notin \left[\sum_{k=0}^{j} B(k;w,p), \sum_{k=0}^{j+1} B(k;w,p)\right) \textbf{ do}$$
$$j++$$
$$\textbf{return } j$$

출처 Reproduced from [10], CC-BY 4.0.

그림 4 합의체 추첨(Sortition) 알고리즘

는다.

각 노드는 VRF에 사용될 자신의 공개키와 비밀키 쌍을 가지고 있으며, 비밀키 sk를 이용하여 만들어진 VRF의 출력값 hash와 증거값 π를 공개한다. 그리고 hash, p, w를 이용한 조건식을 통해 j값 만큼의 투표권을 행사하는 블록 생성자 또는 위원회 자격(Role)을 부여받는다. 공개된 hash, π는 다른 위원회 노드에 의해 해당 노드가 부여받은 role이 정당한지 검증하는 데 사용된다.

이후, 추첨 알고리즘을 통해 블록 생성자 자격을 획득한 모든 노드는 후보 블록을 생성하고 이를 모든 노드에 전파한다. 이때 생성된 후보 블록과 블록을 생성한 노드의 가중치 j가 함께 전파된다.

Algorand에서 블록 생성자는 확률적으로 선정되기 때문에 복수 개의 블록 생성자가 선정될 수 있다. 그러므로 블록 생성자의 선정 확률이 중요하다. 블록 생성자의 개수가 확률적으로 작은 경우에는 최악의 경우 하나의 블록도 생성되지 않을 수 있으며, 개수가 많은 경우에는 합의 메시지 교환 비용과 최종 블록 선정의 복잡성이 증가해 합의 시간이 지연되는 문제가 있다. 이를 위해 Algorand는 블록 생성자의 수가 1개 이상 70개 미만일 확률이 $1-10^{-11}$을 보장할 수 있도록 추첨 알고리즘에 사용되는 τ값을 설정한다(50,000노드 기준 τ=26).

마지막으로, BA★ 알고리즘을 통해 생성된 여러 후보 블록 중에서 하나의 블록에 대한 합의를 도출한다. BA★ 알고리즘은 Reduction과 BinaryBA★로 구성되며 각각은 다시 세부 단계로 구성된다. 이때 각 세부 단계마다 새로운 위원회가 선정되며, 선정된 위원회는 후보 블록에 투표권을 행사하여 전파하고 수신된 투표권을 집계하여 설정된 투표수($T\cdot\tau$) 이상을 획득한 하나의 후보 블록을 결정한다.

Reduction은 2개의 세부 단계로 구성되며 전파된 후보 블록 중에서 투표수 $T\cdot\tau$ 이상을 획득한 하나의 블록을 결정한다. 이때 결정된 블록이 없을 경우 빈 블록으로 결정한다.

BinaryBA★는 Reduction에서 결정된 후보 블록에 대해 최종 합의에 이르기까지 3개의 세부 단계를 반복적으로 수행한다. 이때 해당 블록에 대해 합의가 이루어지지 않은 경우에는 빈 블록에 대한 합의를 수행한다.

만약 위원회에 다수의 비잔틴 노드가 포함된다면 비잔틴 노드들이 자신에게 유리한 블록 생성을 유도할 수 있다. 이를 방지하기 위해 Algorand는 각 세부 단계별로 합의 안전성 보장 확률 $5\cdot10^{-9}$를 제공하기 위한 위원회의 크기 τ와 정족수 비율 T를 정의한다. 이때 선정되어야 하는 위원회의 크기는 각 세부 단계에서는 50,000노드 기준 2,000노드이며, 최종 단계에서는 10,000노드이다.

Algorand는 블록체인 트릴레마를 해결하기 위해 제안되었으나 여전히 일부 한계점을 갖고 있다. 첫째, 블록 확정성을 제공하고자 했으나 합의 알고리즘상 여전히 포크의 가능성이 존재하여 주기적인 포크 해결 과정을 수행해야 한다. 둘째, 확장성을 제공하고자 했으나 중간 세부 단계에서는 2,000노드, 최종 단계에서는 10,000노드가 참여해야 한다. 이로 인해 합의 메시지 교환 복잡도가 증가하고 합의 시간이 지연되는 한계가 있다. 마지막으로 합의 단계에서 충분한 수의 투표수를 획득한 블록이 결정되지 않으면 세부 단계를 무한히 반복하는 경우도 확률적으로 존재한다.

2. PoN+BADA

PoN+BADA는 ETRI에서 제안한 블록체인 분산 합의 알고리즘이다[11]. PoN+BADA는 탈중앙화 합의체 선정을 위한 PoN 알고리즘과 선정된 합의

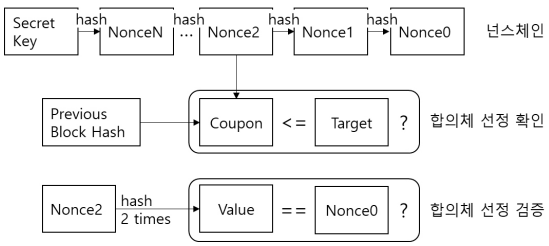

그림 5 PoN 합의체 선정 과정

체 간 블록을 합의하기 위한 BADA 알고리즘으로 구성되어 있다. PoN 알고리즘은 $N \geq 3f+1$(f는 비잔틴 노드 수)을 만족하는 합의체(N)를 매 블록마다 합의 안전성 보장 확률이 $5 \cdot 10^{-12}$ 이하로 제어되도록 선정하며, 선정된 합의체는 BADA 알고리즘을 통해 $O(n)$의 메시지 복잡도로 BFT 기반 블록 합의를 수행한다. PoN+BADA는 30노드 기준 7,500TPS 트랜잭션 처리 성능과 1초의 블록 생성 시간을 달성한다.

그림 5는 PoN을 통한 합의체 선정 과정 예시를 보여준다. 합의에 참여하고자 하는 모든 노드는 자신의 넌스체인을 생성해야 한다. 넌스체인을 생성하기 위해 각 노드는 자신의 비밀키(Secret Key)를 해시하여 해시값을 계산하고 얻은 해시값을 한 번 더 해시하고 이를 반복함으로써 해시값들의 체인인 넌스체인을 생성한다. 현재 PoN+BADA에서 사용하는 해시체인은 1080의 길이를 갖는다. 이는 10초에 1번 블록이 생성된다고 가정했을 때 3시간 동안 사용 가능한 길이이다.

이후 넌스체인을 생성한 모든 노드는 넌스체인의 마지막 넌스값(그림 5에서 Nonce0)을 다른 노드들에 공개하고 참여 노드가 된다. 결과적으로 블록체인 네트워크 내 모든 참여 노드는 다른 참여 노드들의 마지막 넌스값을 가지고 있다. 참여 노드들은 블록이 생성될 때마다 자신이 다음 블록의 합의체인지 아닌지 확인한다.

합의체 선정 확인을 위해 노드들은 자신의 넌스체인에서 매 블록마다 넌스값을 뒤에서부터 하나씩 가져온다(Nonce1, Nonce2, …, NonceN 순서). 그리고 넌스값을 이전 블록의 해시값(Previous Block Hash)과 함께 해시하여 쿠폰값(Coupon)을 계산한다. 계산된 쿠폰값이 목표값(Target)보다 같거나 낮으면 합의체로 선정된다. 이후 자신이 합의체로 선정되었다는 사실을 다른 참여 노드들에 알리기 위해 쿠폰값 계산에 사용한 넌스값(그림 5에서 Nonce2)을 합의체 선정 메시지를 통해 합의체 중 대표 노드인 체어 노드에 공개한다.

합의체 선정 메시지를 받은 체어 노드는 전달받은 넌스값(그림 5에서 Nonce2)이 올바른 값인지 검증한다. 전달받은 넌스값 NonceN을 N번 해시하여(그림 5에서는 2번 해시) 미리 공개된 마지막 넌스값 Nonce0가 나오는지 확인한다. 넌스값 검증이 완료되면 체어 노드는 해당 노드를 다음 블록에 대한 합의체로 추가하고 이 정보를 블록에 피기백(Piggyback)하여 싣는다.

합의체 선정 과정에서 비잔틴 노드는 자신이 합의체로 선정되지 않았음에도 합의에 참여하기 위한 시도를 할 가능성이 있다. 하지만 모든 참여 노드는 합의체가 되기 위해 자신의 마지막 넌스값을 다른 모든 참여 노드에 미리 공개했기 때문에 비잔틴 노드는 합의체 선정을 통과하기 위한 임의의 넌스값을 만들어낼 수 없다. 이는 해시 함수의 비가역성을 전제로 한다.

또한 합의체 선정 확인을 위해 이전 블록의 해시값이 필요하기 때문에 모든 노드는 자신이 언제 합의체가 되는지 미리 알 수 없다. 이는 어떤 노드의 넌스체인이 노출되더라도 비잔틴 노드는 해당 노드가 언제 합의체가 될지 예측할 수 없기 때문에 미리 알고 공격하기 어려워 구조적으로 안전하다.

선정된 합의체(Congress)는 BADA 합의 알고리

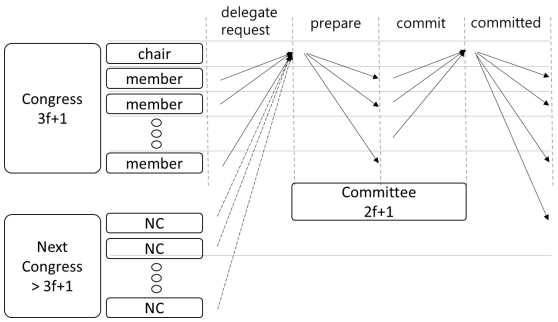

그림 6 BADA 블록 합의 과정

즘을 통해 블록을 합의한다. 이때 합의체 노드(N=3f+1)는 1개의 체어 노드와 3f개의 멤버 노드로 구성된다. 그림 6은 BADA 블록 합의 과정을 보여준다. BADA는 BFT 기반 합의 알고리즘을 수행함과 동시에 합의체 선정 메시지 전달을 함께 수행한다. BADA는 다음과 같은 4가지 단계로 나뉜다.

- Delegate Request: 3f+1개의 모든 멤버 노드(체어 노드 포함)는 자신의 멤풀(Mempool)에 있는 모든 트랜잭션을 체어 노드에 전달한다. 이때 멤풀은 각 노드가 클라이언트로부터 받은 유효한 트랜잭션들을 모아 놓은 저장소이다. 이와 동시에 PoN을 통해 자신이 다음 블록의 합의체(NC: Next Congress)인지 확인한 다음 만약 맞다면 합의체 선정 메시지를 체어 노드에 전달한다.
- Prepare: 체어 노드는 2f+1개의 멤버 노드(체어 노드 포함)로부터 Delegate Request 메시지를 받을 때까지 기다렸다가 2f+1개의 메시지 중에서 f+1개의 메시지에 공통으로 포함된 트랜잭션을 선정하여 후보 블록을 만든다. 이와 동시에 어떤 노드가 어떤 트랜잭션을 보냈는지 나타내는 Evidence를 만들고 후보 블록과 함께 피기백하여 Delegate Request를 전달한 2f+1개의 멤버 노드에 전달한다. 이때 2f+1개의 노드를 위원회(Committee)라고 한다.
- Commit: 체어 노드로부터 Prepare 메시지를 받은 위원회 노드들은 전달받은 후보 블록을 검증한다. 검증을 위해 전달받은 Evidence를 확인하여 자신이 전달한 트랜잭션이 맞는지, 포함된 트랜잭션이 f+1개의 멤버 노드가 공통으로 제안한 트랜잭션이 맞는지 확인한다. 검증이 완료되면 위원회 노드는 블록에 대한 서명을 체어 노드에 전달한다.
- Committed: 체어 노드는 2f+1개의 위원회 노드로부터 Commit 메시지를 받으면 자신이 생성한 후보 블록에 모든 위원회 노드들의 서명을 합해서 다중서명을 만들고 다중서명을 블록에 추가하여 블록을 확정한다. 체어 노드는 확정된 블록에 다음 합의체 정보를 피기백하여 모든 참여 노드에 전달한다. Committed 메시지를 받은 참여 노드들은 위 4가지 단계를 반복한다.

PoN+BADA는 탈중앙화를 달성하기 위해 매 블록마다 전체 참여 노드 중에 확률적으로 임의의 합의체를 뽑는 방식을 사용한다. 또한 선정된 합의체들은 1개의 블록만을 합의한다. 이는 확장성 중심 합의 알고리즘에서 선정된 합의체가 N개의 블록을 합의하는 것과 대조적이다. 이러한 방식은 미리 선정된 합의체를 집중적으로 공격하는 방식의 네트워크 공격을 방지하고 더 많은 노드가 합의에 참여할 수 있도록 함으로써 탈중앙성을 높인다는 장점이 있다.

PoN+BADA는 Algorand와 비교했을 때 더 높은 탈중앙성을 달성한다. PoN+BADA는 50,000개의 노드에서 862개의 합의체를 뽑았을 때 합의체 노드 중 비잔틴 노드의 비율이 33% 이하가 될 확

률(안전성 보장 확률)이 $5 \cdot 10^{-9}$가 되는 것을 증명하였다. 이 확률은 Algorand가 50,000개의 노드에서 2,000개의 합의체를 뽑았을 때 달성 가능한 수치이며, 이는 동일한 수의 참여 노드가 존재할 때 Algorand에서 뽑아야 하는 합의체 수의 43%의 합의체만 뽑더라도 같은 수준의 안전성을 달성함을 의미한다.

또한 기존 IBFT를 비롯한 BFT 기반 합의 알고리즘이 블록 합의를 위해 노드 간 주고받는 메시지 복잡도가 $O(n^2)$인 반면, PoN+BADA는 다중서명을 사용하여 메시지 복잡도를 $O(n)$로 낮추었다.

Ⅳ. 결론

본고에서는 최근 주목받고 있는 고성능 분산 합의 알고리즘들에 대해 소개했다. 비트코인과 이더리움은 기존 중앙화된 시스템의 한계를 해결하기 위해 등장하였으나, 이들은 PoW의 자원 낭비 및 성능 한계로 인해 다양한 산업 분야 적용에 한계가 있었다.

이를 해결하기 위해 다양한 분산 합의 알고리즘들이 등장하였으며, 특히 확장성과 탈중앙성을 둘 다 보장하기 위해 두 가지 이상의 합의 알고리즘을 동시에 사용하는 하이브리드 방식의 고성능 분산 합의 알고리즘이 등장했다. 본고는 이를 성능에 집중한 합의 알고리즘과 탈중앙화에 집중한 합의 알고리즘으로 구분하고 네 가지 대표 합의 알고리즘을 소개했다.

현재도 다양한 블록체인 시스템이 등장하고 있는 추세이며, 이에 따라 다양한 고성능 분산 합의 알고리즘들도 소개되고 있다. 앞으로 블록체인 기술이 다양한 산업 분야에 실질적으로 적용됨에 따라 각 산업 분야에 특화된 고성능 분산 합의 알고리즘 연구가 등장할 것으로 예상하며, 확장성, 탈중앙성, 안전성을 모두 달성하여 블록체인 트릴레마를 해결하는 고성능 분산 합의 알고리즘 연구를 위한 노력 또한 계속될 것으로 전망한다.

약어 정리

BA★	Byzantine Agreement★
BADA	Byzantine Agreement among Decentralized Agents
BFT	Byzantine Fault Tolerance
DDoS	Distributed Denial of Service
DPoS	Delegated Proof of Stake
IBFT	Istanbul Byzantine Fault Tolerance
NC	Next Congress
PBFT	Practical Byzantine Fault Tolerance
PoH	Proof of History
PoN	Proof of Nonce
PoS	Proof of Stake
PoW	Proof of Work
PPoS	Pure Proof of Stake
TPS	Transaction per Second
TTP	Trusted Third Party
VRF	Verifiable Random Function

참고문헌

[1] S. Nakamoto, "Bitcoin: A peer-to-peer electronic cash system," Decentralized Bus. Rev., 2008, article no. 21260.

[2] T Hewa, M Ylianttila, and M Liyanage, "Survey on blockchain based smart contracts: Applications, opportunities and challenges," J. Netw. Comput. Appl., vol. 177, 2021, article no. 102857.

[3] S. Ruoti et al., "SoK: Blockchain technology and its potential use cases," arXiv preprint, CoRR, 2019, arXiv: 1909.12454.

[4] V. Buterin, "Ethereum white paper," GitHub Repository, vol. 1, 2013, pp. 22-23.

[5] Q. Zhou et al., "Solutions to scalability of blockchain: A survey," IEEE Access, vol. 8, 2020, pp. 16440-16455.

[6] M.S. Ferdous et al., "Blockchain consensus algorithms: A survey," arXiv preprint, CoRR, 2020, arXiv: 2001.07091.

[7] D.P. Oyinloye et al., “Blockchain Consensus: An overview of alternative protocols,” Symmetry, vol. 13, no. 8, 2021, article no. 1363.

[8] 서상민 외, “클레이튼 블록체인 플랫폼의 고성능 합의 알고리즘,” 한국통신학회지(정보와 통신), 제37권 제3호, 2020, pp. 28-36.

[9] A. Yakovenko, “Solana: A new architecture for a high performance blockchain v0.8.13,” Whitepaper, 2018.

[10] Y. Gilad et al., “Algorand: Scaling byzantine agreements for cryptocurrencies,” in Proc. Symp. Oper. Syst. Princ., (Shanghai, China), Oct. 2017, pp. 51-68.

[11] J. Oh et al., “Algorithm based on Byzantine agreement among decentralized agents(BADA),” ETRI J., vol. 42, no. 6, 2020, pp. 872-885.

[12] The ZILLIQA team, “The ZILLIQA Technical Whitepaper,” version 0.1, Aug. 2017, Available from: https://docs.zilliqa.com/whitepaper.pdf

[13] E. Buchman, “Tendermint: Byzantine fault tolerance in the age of blockchains,” M.S. thesis, Engineering Systems and Computing, University of Guelph, Canada, Ontario, 2016.

[14] X. Brent et al., “Eos: An architectural, performance, and economic analysis,” Retrieved June, vol. 11, 2018.

[15] H. Moniz, “The Istanbul BFT consensus algorithm,” arXiv preprint, CoRR, 2020, arXiv: 2002.03613.

[16] https://github.com/solana-labs/solana