https://doi.org/10.33778/kcsa.2020.20.1.003

허가형 블록체인 합의알고리즘의 성능평가항목 연구

민 연 아*

요 약

블록체인은 중앙 집중 시스템 형태에서 벗어난 탈중앙화 형태의 데이터 관리를 통하여 데이터 투명성과 보안성을 높일 수 있다. 블록체인 플랫폼 중 허가형 블록체인은 신뢰기반의 허가된 노드만이 분산 네트워크에 참여할 수 있다. 허가형 블록체인의 특징을 고려하였을 때 합의 알고리즘 선정을 위한 조건으로 네트워크 통신 속도 및 거래내역의 최종성 합의, 안정성 등의 고려가 필요하다. 허가형 블록체인 환경의 합의 알고리즘은 PoA, PBFT, Raft 등 다양하지만 합의 알고리즘 선정을 위한 다양한 평가요소가 존재하지 않는다. 본 논문에서는 허가형 블록체인의 각 합의 알고리즘의 특징을 분석하고 네트워크를 구성하는 사용자 환경의 특징을 고려한 효율적 합의 알고리즘 선정을 위하여 다양한 성능평가항목을 제안하였다. 제안한 성능평가항목은 신뢰를 전제로 한 노드 간 네트워크 속도, 안정성, 최종성 합의의 적합성 등을 고려할 수 있으며 이를 통하여 보다 효율적인 블록체인 네트워크 환경을 구성할 수 있다.

A Study on Performance Evaluation Factors of Permissioned Blockchain Consensus Algorithm

Min Youn A*

ABSTRACT

Blockchain can enhance data transparency and security through decentralized data management that is out of the centralized system. permissioned blockchain of the blockchain platform, only trust-based authorized nodes can participate in the distributed network. Considering the characteristics of the permissioned blockchain, it is necessary to consider the network communication speed, transaction finality agreement, and stability as a condition for selecting the consensus algorithm. The consensus algorithms of the permissioned blockchain environment are diverse such as PoA, PBFT, Raft, etc., but there are no various evaluation factors for selecting consensus algorithms. In this paper, various performance evaluation factors are proposed to analyze the characteristics of each consensus algorithm of the permissioned blockchain and to select an efficient consensus algorithm considering the characteristics of the user environment that composes the network. The proposed performance evaluation factor can consider the network speed, stability, and consensus of the finality agreement between nodes under the premise of trust. Through this, a more efficient blockchain network environment can be constructed.

Key words : Blockchain, Consensus Algorithm

접수일(2020년 03월 05일), 수정일(1차: 2020년 03월 16일), 게재확정일(2020년 03월 25일)

* 한양사이버대학교/응용소프트웨어공학과

1. 서 론

블록체인(Blockchain) 기술은 참여하는 모든 노드를 통하여 거래내역(Transaction)을 합의하고 공유하며 그러한 과정을 통하여 데이터의 정확성과 투명성이 보장된다. 블록체인은 클라이언트를 통한 거래내역 생성 및 거래내역 확인이 가능하며 각 노드를 통하여 거래내역의 승인 및 분산 합의가 가능하다.

블록체인은 발표 초기에 비트코인(Bitcoin) 중심의 암호화폐 기술이 주를 이루었으나 2008년 이더리움(Ethereum)의 개발과 함께 스마트 계약 등의 기술이 적용되며 금융, 식품의 이력 관리, 물류 관리 등 데이터의 공유와 이력 관리 및 인증 등이 필요한 다양한 곳에 블록체인 기술이 적용되고 있다[1][2].

블록체인의 가장 큰 특징은 기존 대표적인 거래 방식인 중앙 집중 시스템 방식을 탈피하여 탈중앙화 방식으로 노드를 연결하는 것이다. 블록체인 네트워크에 연결된 모든 노드를 통하여 거래내역이 검증되고 합의된다[2]. 블록체인은 합의된 데이터만을 블록에 저장하여 관리하기 때문에 블록 생성을 위한 합의 알고리즘이 매우 중요하다.

블록체인의 합의 알고리즘(Consensus Agorithm)은 플랫폼 또는 사용자 환경에 따라 다양하다. 적절한 합의 알고리즘 적용에 따라 블록체인 적용 환경의 효율성이 달라질 수 있다. 현재 합의 알고리즘 선정을 위한 성능평가 항목이 다양하지 않다. 블록체인 기술 적용사례의 증가를 고려할 때 합의 알고리즘 선정을 위한 다양한 성능평가항목은 매우 중요한 전제조건이며 효율적인 합의 알고리즘을 선정할 수 있도록 사용자 환경을 고려한 다양한 성능평가항목의 연구가 필요하다.

2. 연구 배경

2.1 블록체인 기술

블록체인은 임의의 다수가 분산 네트워크에 참여하여 분산된 노드 모두 거래내역을 공유하는 퍼블릭 블록체인(Public blockchain)과 허가된 일부만이 참여하는 허가형 블록체인(permissioned Blockchain)으로 구분할 수 있다.

퍼블릭 블록체인은 불특정 다수 누구나 네트워크의 노드로 참여하여 블록 생성 및 합의 과정에 참여할 수 있다[1][3].

퍼브릭 블록체인은 동시에 여러 개의 블록이 생성되었을 때 대부분의 경우 가장 긴 블록을 가진 노드의 블록 생성을 선택한다. 만일 동시에 생성된 블록이 비슷한 블록의 길이를 가진 경우에는 노드들이 분산하여 생성된 여러 개 블록에 분산 연결될 수 있다. 이러한 경우를 포크(fork)라고 하며 포크가 발생되었을 경우 네트워크의 분산장부는 일치하지 않을 수 있다.

프라이빗 블록체인 중 허가형 블록체인이라고 불리며 허가된 노드들만이 네트워크에 참여하는 방식으로 운영되기 때문에 퍼블릭 블록체인과는 전혀 다른 합의 알고리즘 적용이 가능하다[1][4].

허가형 블록체인의 합의과정은 비교적 단순하다.

권위 있는 노드를 정해놓고 해당 노드들로 구성된 위원회가 블록 생성의 자격을 가질 수 있다. 이러한 구성원에 의하여 합의된 블록은 네트워크를 구성하는 전체 노드에게 전달된다[1][9][10].

최근 하이퍼레져(hyper ledger)와 같은 컨소시움 형태의 허가형 블록체인 형태가 소개되며 엔터프라이즈(enterprise)기반의 활용이 활발해지고 있다[1][9].

2.2 블록체인 합의 알고리즘

블록체인 네트워크에서 노드들이 동일한 데이터를 가질 수 있도록 데이터를 검증하고 처리하는 과정을 합의 과정이라 하며 해당 과정에 사용되는 알고리즘을 합의 알고리즘이라 한다[1].

블록체인 합의 알고리즘은 플랫폼과 사용자 환경을 구분하여 적용된다.

퍼블릭 블록체인과 같이 누구나 참여하는 비허가형 분산 네트워크의 경우 모든 노드에 대한 불신을 전제로 하기 때문에 정확한 거래내역 저장 및 관리를 위하여 블록을 생성하려는 노드에게 과도한 컴퓨팅 파워나 노드의 지분율 등을 요구한다. 과도한 노력을 기울여 블록을 생성한 노드는 자신의 노력을 유지하기 위하여 블록 유지에 많은 관심을 가지게 된다[3][4].

허가형 블록체인과 같이 허가된 노드만이 참여하는 분산 네트워크의 경우 참여하는 노드의 신뢰를 기반

으로 하기 때문에 퍼블릭 블록체인과 달리 인증과 합의 과정이 비교적 단순하다[4].

허가형 블록체인의 합의 알고리즘 장애가 발생(Fail)하면 더이상 상태변화가 일어나지 않는 Fail -Stop 방식과 메세지 도착 여부와 도착한 메세지의 정상 여부 그리고 검증해야만 하는 모든 장애에 대한 방식과 그 장애를 처리할수 있는 경우까지 처리하는 BFT(Byzantine Fault Tolerance)으로 구분할 수 있다. 이 중 Fail-stop 방식을 사용하는 대표적 합의 알고리즘은 Paxos, Raft이며 BFT를 가정한 것은 PBFT(Practical Byzantine Fault Tolerance)이다[5][6].

2.2.1 허가형 블록체인 합의 알고리즘

본 절에서는 대표적인 허가형 블록체인의 합의 알고리즘인 Paxos, Raft, PBFT에 대하여 알아본다.

① Paxos

Paxos는 합의 알고리즘 처리를 위하여 ProPoser와 Accepter, Learner의 역할로 구분되며 여러 개의 값을 제안하고 투표하며 제안된 내용을 학습하는 과정으로 구성되며 프로토콜의 동작과 연산이 복잡하기 때문에 활발하게 사용되지 않는다[6].

② Raft

Raft는 Paxos의 복잡한 연산을 쉽게 수정하여 고안된 알고리즘이다.

Raft는 클라이언트의 블록 생성 요청에 대하여 하나의 선출된 리더(Leader)가 요청을 처리하고 처리 내용을 로그(Log)에 업데이트한다. 리플리카들은 업데이트된 로그의 내용을 반영하여 동작한다.

만일 리더에 문제가 있을 경우 후보자 중 선출 프로토콜에 의해 새롭게 리더 선출되는 과정을 가진다. Raft는 Leader, Follower, Candidate 세 가지 상태를 가진다[5].

③ PBFT

BFT(Byzantine Fault Tolerance) 계열의 프로토콜은 악의의 노드가 있음을 감안하여 안전한 노드의 수의 비율을 감안하여 합의되는 방식으로 작동되며 이러한 작동 방식의 특징으로 인하여 시스템이 안정적으로 동작된다[6].

PBFT는 전체 노드들 중 51%가 악의의 노드라는 가정을 가진 비잔틴 장군 문제의 해결을 위하여 제안된 알고리즘이며 BFT계열 프로토콜이다[7].

PBFT는 총 노드 N개 중 F개의 오류가 있을 경우 N=3F+1의 환경에서 정상적으로 동작이 보장된다[8].

PBFT는 다수의 리플리카 중 리더(Primary)를 선출하여 리더의 주도하에 명령 수행한다. 만일 리더가 고장 등의 오류가 발생하면 view change를 통하여 리더를 변경한다. 여러 개의 리플리카 노드와 하나의 리더로 구성된 PBFT는 여러 번의 브로드캐스트 방식의 합의와 인증을 통하여 합의를 이룬다[6][7].

2.2.2 전통적인 블록체인 성능평가요소

블록체인은 속도 위주의 성능평가가 이루어지고 있다. 블록체인의 속도를 결정하는 대표적 요소는 Tansaction per Second, 블록생성시간, 블록생성확정시간이다[9].

TPS의 경우 소트트웨어 및 하드웨어의 설계와 네트워크의 성능과 거래내역의 종류에 따라 측정방법이 상이하다. 따라서 TPS는 블록체인의 성능평가 지표가 아닌 참고지표로 활용되는 경우가 많다.

블록 생성시간이란 블록이 생성되는데 걸리는 시간으로 흔히 네트워크의 Latency와 같다. 블록생성시간은 확장성 및 처리속도에 큰 영향을 미치기 때문에 블록체인 성능에 영향을 미친다.

블록체인 확정시간이란 내가 공유하는 거래내역이 최신 블록체인에 저장되어있음을 보증해주는 시간으로 신각 거래소의 신뢰도에 비례하므로 블록체인 성능을 평가하기 위한 절대적 요소가 될 수 없다[9].

블록체인의 사용자 및 플랫폼 환경이 다양해짐에 따라 속도 뿐 아니라 합의 알고리즘의 안정성 측면에 대한 평가요소 고려가 필요하다.

3. 블록체인 성능평가를 위한 요소

블록체인의 합의 알고리즘은 다양하며 어떤 합의 알고리즘을 선정하였는가에 따라 블록체인 운영방식 및 안정성이 달라질 수 있다.

전술한 바와 같이 대다수의 블록체인 플랫폼의 경우 성능평가항목으로 TPS와 블록생성시간, 블록확정시간 등 속도측면의 평가항목을 고려한다[8][9]. TPS의 경우 사용자의 컴퓨터 성능이나 구성된

네트워크 환경에 따라 다르게 계산될 수 있으며 블록 확정시간은 거래소의 신뢰도를 반영하기 때문에 성능 평가를 위한 절대적 기준이 될 수 없다. 따라서 범용적 환경 고려가 가능한 다양한 성능평가항목의 제안이 필요하다.

퍼블릭 블록체인의 경우 성능평가를 위하여 확장성, 안정성, 속도 등 다양한 항목의 평가가 필요하다. 본 논문에서는 허가된 노드들만 참여하는 허가형 블록체인 환경으로 국한하여 효율적인 합의 알고리즘을 선정하기 위한 성능평가항목으로 다음의 요소를 제안한다.

- 권위를 가진 연결 노드의 수
- A와 수정된 TPS를 고려한 합의 안정성

3.1 권위를 가진 연결 노드의 수

합의 알고리즘에서 전체 노드 수 대비 합의 과정에 참여하는 노드의 비율에 따라 네트워크 통신비용 및 처리속도가 달라질 수 있다.

본 논문에서는 허가형 블록체인의 특징을 고려하여 네트워크에 연결된 노드들의 역할을 확인자, 참여자로 구분하여 활동 횟수를 체크하여 허가형 블록체인의 사용자 환경에 대한 신뢰를 평가할 수 있도록 한다. 전체 노드의 수를 N이라 할 때 노드 중 권위 있는 노드의 비율 a라 가정 한다. a의 증가에 따른 네트워크의 신뢰를 평가할 수 있으며 a를 고려한 신뢰를 기반으로 하며 제안하는 성능평가수식에 a를 추가 수식 변수로 활용한다.

3.2 합의의 안정성

본 논문에서는 권위 있는 노드의 비율을 고려하여 빠르게 합의를 도출하고 블록을 생성하기 위하여 블록체인 생성 시간의 수식을 변형하여 합의 가능성 수식을 제안하였다.

해당 수식은 권위 있는 노드의 비율에 따라 안정성이 증가 또는 감소할 수 있다.

본 연구에서는 성능평가방법을 위하여 기존 연구의 수식[2]을 수정하여 <표 1>의 수식을 제안하였으며 p는 정직한 노드가 블록을 생성할 확률이며 q는 악의의 노드가 블록을 생성할 확률이다.

<표 1> 수정된 블록 생성 확률

$$p = 1 - \sum_{k=0}^{z} (\lambda^k e^{-\lambda}/k!)(1-(q/p)^{z-k})) \times a \times TPS$$

- TPS : (S_B/S_T)*(1/t)*a*Δ
- 블록생성시간: t
- 블록 사이즈: S_B
- 트랜잭션의 크기 S_T
- 블록의 평균생성시간: Δ
- z: 확정을 기다리는 블록들의 수
- λ = z * (q / p)
- a = 권위를 가진 노드의 비율

Paxos, Raft, PBFT 모두 BFT 가능하나 일반적으로 총 노드 수 N과 악의적 의도를 가진 노드 수 F를 고려하였을 때 PBFT는 33.3%, Paxos와 Raft는 51%의 공격감내(Fault Tolerance)가 가능하다.

본 논문에서는 위에서 제안한 성능평가항목의 타당성 검토를 위하여 각 합의 알고리즘의 공격감내비율(Fault Tolerance Rate)을 수식에 활용한다.

4. 성능평가항목의 타당성 검토

본 논문에서는 제안한 성능평가항목의 적합성을 평가하기 위하여 노드 수를 다양하게 제시하여 적합한 합의 알고리즘을 선정하는 과정을 제시한다.

본 논문에서는 기존의 수식을 기반으로 변형된 수식을 기반으로 타당성 검토를 진행하였기 때문에 별도의 실험환경은 필요하지 않으며 검토를 위한 네 개의 사례가 동일한 환경이라는 가정을 기반으로 한다.

적합성 평가에 사용되는 변수로 N은 총 노드 수를 나타내며 f는 악의적 의도를 가진 노드 수, α는 권위를 가진 노드의 비율, TPS는 <표 1>의 식을 적용한 (S_B/S_T)*(1/t)*a*α의 식을 갖는다.

<표 2>는 3장에서 제시한 성능평가항목을 기반으로 제안한 수식들을 적용하여 각 합의 알고리즘의 특징에 맞도록 수정한 것이다. 권위 있는 노드의 연결 수를 A라 하고 수정된 TPS를 고려한 합의 안정성을 B라고 한다. 성능분석은 총 노드의 수가 많은 경우와 적을 경우 / 노드 수 대비 권위 있는 노드의 비율이 높은 경우와 그렇지 않은 경우를 샘플로 측정한다.

<표 2> 성능평가항목

ITEM	Paxos	PBFT	Raft
A	α=(N+1)/2	α=(2N+1)/3	α=(N+1)/2
B	β * (N / 2) + 1 time +TPS	β * N times +TPS	β * (N / 2) + 1 time + TPS

○ 성능분석 사례 1 : 총 노드가 최소한의 개수로 구성된 경우로 N은 4, f는 1

(그림 1) 사례 1의 분석

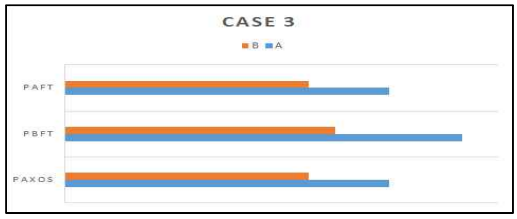
전체노드가 4개 이하의 최소 노드인 경우 네트워크의 돌발 상황이 발생하지 않는다는 전제하에 해당 사례에는 PBFT가 안정성측면에서 우수하므로 안정성을 우선으로 여기는 경우 PBFT가 적합하다.

○ 성능분석 사례 2 : 총 노드가 최대한의 개수로 구성된 경우로 N은 100, f는 25

(그림 2) 사례 2의 분석

전체 노드가 100개인 경우를 계산하였을 경우 네트워크의 돌발 상황이 발생하지 않는다는 전제하에 해당 사례에는 PBFT알고리즘의 적용이 가장 적합함을 알 수 있다. 해당 결과의 경우 네트워크의 변동상황 등의 돌발상황을 고려하지 않았기 향후 네트워크의 안정성 등을 고려한다면 M_TPS의 수치변동이 가능하며 이에 따른 평가 수정이 필요할 수 있다.

○ 성능분석 사례 3 : 총 노드 수 대비 권위를 가진 노드 수가 적을 경우로 N은 4, f는 2



(그림 3) 사례 3의 분석

전체노드 수 및 권위 있는 노드 수가 모두 적을 경우 통하여 네트워크의 돌발 상황이 발생하지 않는다는 전제하에 해당 사례에는 PBFT알고리즘의 적용이 가장 적합함을 알 수 있다.

○ 성능분석 사례 4 : 총 노드 수 대비 권위를 가진 노드 수가 많을 경우로 N은 100, f는 1

(그림 4) 사례 4의 분석

전체 노드 수 대비 권위 있는 노드의 비율이 높은 경우 네트워크의 돌발 상황이 발생하지 않는다는 전제하에 해당 사례에는 PBFT 알고리즘의 적용이 가장 적합함을 알 수 있다.

본 논문에서 제안한 성능평가항목의 적합성 평가를 위하여 노드의 수를 달리하여 4가지 경우에 대하여 합의 알고리즘 선정 과정을 제시하였으며 다양한 사용자 환경에서 제안한 성능평가항목을 적용함으로써 효율적 합의 알고리즘 선택이 가능하다.

5. 결 론

신뢰기반의 허가된 기관들의 허가형 블록체인 플랫폼 활용 사례가 증가하고 있다. 허가된 노드들의 경우

악의적 노드의 위협으로부터 보안을 유지하고자 하는 노력이 필요 없기 때문에 퍼블릭 블록체인과 같이 과도한 작업이나 지분을 증명할 필요가 없다. 또한 사용자 환경을 고려하여 권위를 가진 일부의 노드에 의한 책임 있는 인증도 가능하다.

본 논문에서는 블록체인 기반 효율적 합의 알고리즘 선정을 위한 성능평가항목을 권위를 가진 연결 노드의 수, 수정된 초당 거래내역 처리량을 고려한 합의 안정성의 2가지를 제시하였다.

제안한 평가요소의 적합성 평가를 위하여 다수의 사례에 제안하는 성능평가항목을 적용하여 효율적인 합의 알고리즘 선정을 위한 수식을 제안하였다,

총 노드 수(N) 대비 악의를 가진 노드(f)의 비율이 적은 경우 권위를 가진 연결 노드 수(A)와 합의 안정성(B) 대체적으로 높았으며 해당 경우에 노드의 수가 최소일 경우 PBFT의 성능이 근소하게 우수하였으며 노드의 수가 최대일 때 PBFT의 성능이 월등하게 높았으며 특히 B항목이 우수하였다.

본 논문에서는 네트워크 상황에서 발생 가능한 돌발 상황 등에 대해서는 고려하지 않았으나 향후 대표적 돌발 상황을 고려한 추가 연구를 진행할 예정이다.

참고문헌

[1] Yim J C, Yoo H K, Kwak J Y, Kim S M. "Blockchain and Consensus Algorithm" Telecommunication Trend Analysis VOL.33 NO.1, pp. 45-56, 2018

[2] Satoshi Nakamoto, “Bitcoin: A Peer-to-Peer Electronic Cash System,” http://bitcoin.org/bitcoin.pdf

[3] Jung, G.S, Kim, D.W.“Blockchain Industry Status and Foreign Policy Trend“, NIPA,Issue Report, No 38, 2019

[4] Yim, J.C. , Yoo, H.K. , Kwak, J.Y. , Kim, S.M. "Blockchain and Consensus Algorithm", Electronics and telecommunications trends, Vol 33, No 1, 2018

[5] L. Lamport, "Paxos Made Simple", ACM SIGACT News, VOL.32, NO.4, pp. 18-26, 2001

[6] Castro M , Liskov B, "Practical Byzantine Fault Tolerance", Operating systems review VOL.33/SPI1, pp. 173-186, 1998

[7] Leslie Lamport,Robert Shostak,Marshall Pease, "The Byzantine Generals Problem", ACM transactions on programming languages and systems, VOL.4, NO.3, pp. 382-401, 1982

[8] https://www.usenix.org/legacy/events/osdi99/full_papers/castro/castro_html/castro.html

[9] Do Gyun Kim, · Jin Young Choi,Kiyoung Kim,Jintae Oh,J. Soc. ‘Performance Improvement of Distributed Consensus Algorithms for Blockchain through Suggestion and Analysis of Assessment Items’ Korea Ind. Syst. Eng Vol. 41, No. 4m pp. 179-188, 2018

[10] Lee, H.G, Won, D.H, Lee, Y.S,"Blockchain massive computing attack protection technology", KCSA,Vol 19, No 2, pp. 11-19, 2019

〔저자소개〕

민 연 아 (Min Youn A)
2002년 2월 동국대학교 컴퓨터교육학과 석사
2013년 2월 동국대학교 컴퓨터공학과 박사
2020년 ~ 현재 한양사이버대학교 응용소프트웨어공학과 조교수
email : yah0612@hycu.ac.kr