**1. Introduction**

비트코인 – 제 3자의 도움 없이 거래를 할 수 있다. 블록체인 기술 적용

비대칭 암호 알고리즘과 분산 합의 알고리즘으로 보안과 원장의 일관성을 구현

블록체인 – 비중앙화, 지속성, 익명성, 감사성

블록체인은 변경할 수 없다(immutable), 일단 거래가 블록에 들어가면 변조할 수 없다

Single point of failure을 피할 수 있다 – P2P 구조이기 때문에

잠재력이 큰 기술이지만 여러 어려움에 직면하고 있다

확장성 – 비트코인을 예로 들면 비트코인의 한 블록은 1MB로 제한되어 있지만 약 10분마다 채굴된다. 그리고 초당 최대 7개의 거래만 처리할 수 있다.

그러나 블록을 크게 하면 많은 저장 공간이 필요하고, 네트워크에서의 전파가 느려진다.

몇몇 광부들이 이기적인 채굴 전략을 통해서 더 많은 수익을 얻는 것이 증명되었다. 자신이 채굴한 블록을 숨김으로써 – 블록체인의 발전을 저해

공개키 암호와 거래를 하는 블록체인에서도 개인정보 유출이 발생할 수 있다는 것이 보여졌다.

현재 잘 알려진 합의 알고리즘들은 몇몇 심각한 문제에 직면했다

**2. Blockchain Architecture**

블록체인은 완료된 거래 리스트를 가지고 있는 블록의 연속적인 모임이다

첫 번째 블록은 genesis block이라고 부르고, 부모 블록이 없다

**2-A. Block**

블록은 블록 해더와 블록 바디로 이루어져 있다

블록 헤더의 구성요소

- Block Version: 블록이 어떤 유효성 검사 규칙을 따랐는지 표시

- Merkle Tree Root Hash: 이 블록 내의 모든 거래의 해시 값

- Time stamp: 현재 시간

- nBits: 유효 블록 해시의 target threshold(임계값)

- Nonce: 4바이트 필드, 일반적으로 0으로 시작해서 해시 계산 때 마다 1씩 증가

- Parent Block Hash: 이전 블록을 가리키는 256비트 해시 값

블록 바디는 transaction counter와 transactions로 구성

블록체인은 비대칭 암호 매커니즘을 이용해서 거래의 유효성을 검증(Digital Signature)

**2-B. Digital Signature**

각각의 유저는 공개키와 개인키를 가지고 있다

개인키: 비밀 유지, 거래 서명에 사용

서명된 거래는 네트워크 전체에 broadcast된다

전형적인 디지털 서명은 서명 단계와 검증 단계로 구성, 예를 들어 Alice가 Bob에게 메시지를 전송한다고 가정

1. 서명 단계에서 Alice는 그녀의 개인키로 데이터를 암호화, 그 후 Bob에게 암호화된 데이터와 원본 데이터를 함께 전송

2. 검증 단계에서 Bob은 Alice의 공개키로 받은 암호화된 데이터를 검증, 이렇게 하면 Bob은 데이터의 변조 여부를 쉽게 확인 가능

**2-C. Key Characteristics of Blockchain**

- Decentralization: 일반적인 중앙 집중형 거래 시스템은 각 트랜잭션은 중앙의 신뢰기관을 통해 검증 받아야 할 필요성이 있어서 불가피하게 중앙 서버의 비용 증가 및 병목 현상이 발생하게 된다. 블록체인은 더 이상 제3자가 필요하지 않다. 블록체인의 합의 알고리즘은 분산된 네트워크에서 데이터 일관성을 유지하기 위해 사용된다.

- Persistency: 일단 거래가 블록체인에 포함되면 지우거나 롤백하는 것은 거의 불가능하다. 유효하지 않은 거래가 포함된 블록들은 즉시 발견될 수 있다.

- Anonymity: 각 사용자는 생성된 주소로 블록체인과 상호작용을 할 수 있으며 이는 사용자의 진짜 정체를 드러나지 않게 해준다. 하지만 블록체인의 본질적인 제약 때문에 완전한 개인정보 보전을 보장할 수는 없다.

- Auditability: 비트코인은 Unspent Transaction Output(UTX-O) 모델에 기초해서 사용자 잔액에 대한 데이터를 저장한다. 어떤 거래이건 이전의 몇몇 소비되지 않은 트랜잭션들을 참조해야 한다.

**2-D. Taxonomy of Blockchain Systems**

현재 블록체인은 세 가지로 분류되어 있다: public blockchain, consortium blockchain, private blockchain

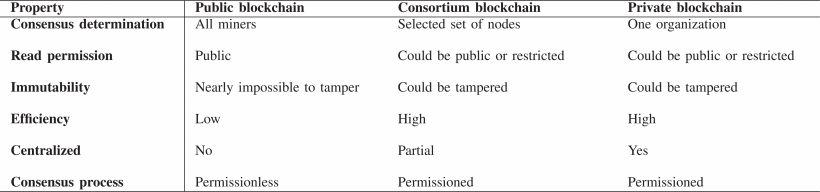
**Public** blockchain에서는 모든 기록은 대중에게 공개되고, 누구나 합의 과정에 참여할 수 있다

**Consortium** blockchain에서는 사전에 선택된 그룹만이 합의 과정에 참여할 수 있다.

**Private** blockchain에서는 특정 조직으로부터 오는 node만이 합의 과정에 참여할 수 있다

Private blockchain은 하나의 조직에 의해 완전히 컨트롤되기 때문에 중앙집중식 네트워크로 간주된다

Consortium blockchain은 몇 개의 조직으로 구성되므로 부분적으로 분산이 된다.



Public blockchain은 전 세계에 개방되어 있기 때문에 많은 유저를 모을 수 있고, 커뮤니티가 활발하다. 많은 public blockchain들이 출현하고 있다

Consortium blockchain은 사업 어플리케이션에 적용할 수 있을 것이다

**3. Consensus Algorithms**

**3-A. Approaches to Consensus**

**PoW(Proof of Work)**는 비트코인에서 사용되는 합의 전략

분산 네트워크에서는 거래를 기록하기 위해 누군가를 선택해야 한다

가장 쉬운 방법은 랜덤 선택이지만 그렇게 하면 공격에 취약해 진다

따라서 어떤 node가 블록을 추가하고자 하는 경우, 네트워크를 공격할 가능성이 없다는 것을 입증해야 한다

일반적으로 컴퓨터 계산을 이용

PoW에서는 각 node들이 블록 헤더의 해시 값을 계산

다른 해시 값을 얻기 위해 nonce를 변경할 수 있다

합의는 계산된 값이 특정 주어진 값보다 같거나 작아야 한다고 요구

하나의 node가 목표 값에 도달하면 해당 블록을 다른 node로 broadcast

모든 다른 node들은 해시 값의 정확성을 상호 확인해야 한다

그 블록이 검증된다면 이 새로운 블록을 블록 체인에 추가

비트코인에서 해시 값을 계산하는 node들을 miner(채굴자), PoW 절차를 mining(채굴)이라고 한다

여러 node들이 적절한 nonce값을 거의 동시에 찾을 때 유효한 블록이 동시에 생겨날 수 있다

이렇게 되면 블록 체인에 branch가 생기게 된다

PoW에서는 branch된 이후에 더 길어지는 체인을 진짜 체인이라고 판단한다

채굴자는 PoW에서 많은 컴퓨터 계산을 한다 -> 너무 많은 자원을 낭비

그래서 PoW를 개선한 여러 프로토콜이 탄생

**PoS(Proof of Stake)**는 PoW의 에너지 절약 버전이다

PoS에서는 자신이 얼마나 많은 코인을 가지고 있는지 증명해야 한다 -> 많은 코인을 가지고 있으면 공격할 가능성이 낮다고 가정

하지만 이 방법은 가장 부유한 한 사람이 우세할 수 밖에 없기 때문에 불공평 -> 많은 대안이 나옴

PoW와 비교해 PoS는 더 많은 에너지를 절약하고 더 효율적

많은 블록체인이 PoW에서 PoS로 이동하는 추세

**PBFT(Practical Byzantine Fault Tolerance)**는 byzantine faults를 허용하는 복제 알고리즘이다

Hyperledger Fabirc에서는 최대 1/3의 악의적인 byzantine replica들을 처리할 수 있기 때문에 PBFT를 합의 알고리즘으로 사용

새로운 블록은 라운드로 결정, 각 라운드에는 규칙이 존재

현재 라운드는 세 단계로 구분: pre-prepared, prepared, commit

각 단계에서 2/3이상의 node들에게 투표를 받을 경우 다음 단계로 진입

그래서 PBFT는 모든 node가 네트워크에 대해 알 것을 요구

**DPOS(Delegated Proof of Stake)**와 PoS의 가장 큰 차이점은 Pos가 직접 민주주의인 반면 DPOS는 대표자 민주주의(간접)라는 것이다

코인을 가지고 있는 사람들은 블록을 생성하고 검증하기 위해서 대표자를 선출

블록 유효성을 확인하는 node가 상당히 적기 때문에 블록을 더 빨리 확인할 수 있고, 더 빠른 거래 확인으로 이어진다

블록 크기나 블록 간격은 대표자들이 조정한다

유저들은 쉽게 그들을 탄핵할 수 있기 때문에 대표자의 부정을 걱정할 필요가 없다

**Ripple**은 대규모 네트워크 내에서 공동 신뢰의 하위 네트워크를 활용하는 합의 알고리즘이다

이 네트워크에서는 node들을 두 가지 타입으로 구분

* 합의 과정에 참여하는 server
* 자금 거래만 하는 client

각 server에는 Unique Node List(UNL)이 존재, server에 매우 중요

원장에 거래를 넣을지 넣지 않을지 결정할 때, server는 UNL에 그 node가 있는지 조회한다

수신된 승인이 80% 이상이면 그 거래를 원장에 포함한다

**Tendermint**는 byzantine consensus algorithm이다

새로운 블록이 한 라운드 안에 결정

이번 라운드에 확인되지 않은 블록을 broadcast하기 위해 proposer가 선택된다

3단계로 구분: prevote, precommit, commit

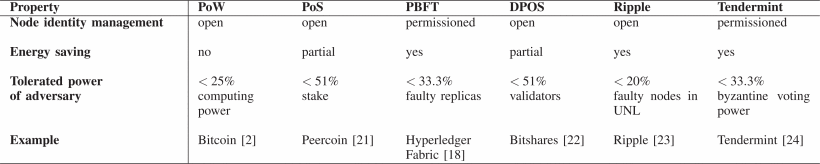
Prevote: 검증자가 제안된 블록에 대한 사전 투표에 대한 broadcast를 할지 안 할지 결정

Precommit: 제안된 블록에 2/3이상이 투표를 했다면, 그 블록에 대한 precommit을 broadcast 한다. 2/3이상의 precommit을 수신했을 경우, commit단계로 진입

Commit: node는 블록을 검증하고 블록에 대한 commit을 broadcast 한다. 2/3이상의 commit을 수신했을 경우 블록을 포함

PBFT와의 다른 점은 검증자가 되기 위해서는 코인이 있어야 한다

**3-B. Consensus Algorithms Comparison**



Node identity Management: PBFT는 매 라운드마다 예비 선거 선택을 위해 각 광부의 신원을 알 필요가 있고, Tendermint는 각 라운드의 제안자를 선택하기 위해 검증자를 알아야 한다

Energy saving: PoW는 해시 값을 계산해 목표 값을 찾기 위해 계속적인 계산을 해야 하고, 결과적으로 많은 양의 전력량이 소모된다. PoS와 DPOS에서도 여전히 해시 값 계산을 해야 하지만 찾을 공간이 제한되어 있어서 어느 정도 절약할 수 있다

Tolerated Power of Adversary: 일반적으로 해시 파워의 51%는 네트워크 제어권을 얻기 위한 임계치로 가정된다. 그러나 이기적인 채굴 전략으로 PoW에서는 25%만 가지고 있어도 더 많은 이득을 취할 수 있다. PBFT와 Tendermint는 최대 1/3의 결함이 있는 node들을 처리하게 디자인 되어 있다. Ripple은 UNL에 결함이 있는 node가 20% 미만일 경우 정확성을 유지하는 것으로 입증 되었다

PoW, PoS -> public blockchain

PBFT, DPOS, Ripple, Tendermint -> consortium, private blockchain

**3-C. Advances on Consensus Algorithms**

좋은 합의 알고리즘 -> 효율성, 안정성, 편의성을 의미

블록체인의 구체적인 문제들을 해결하기 위한 새로울 합의 알고리즘이 고안되고 있다

**4. Challenges & Recent Advances**

몇 가지 주요 문제점을 설명

**4-A. Scalability**

현재 거래는 계속해서 늘어나고 있고, 각 node들은 현재 거래가 미사용인지 아닌지를 체크해야 하기 때문에 모든 거래를 저장해야 한다

비트코인은 새로 생성되는 블록의 제한으로 초당 7건의 거래만 처리할 수 있어 수백만 건의 거래를 실시간으로 처리할 수 없다

또한 블록의 용량이 매우 작아서 채굴자들이 수수료가 높은 거래 위주로 처리하면 소규모 거래는 지연될 수 있다

이런 문제를 해결하기 위한 방안이 제안되었는데 두 가지 타입으로 분류

- Storage Optimization of Blockchain

Node가 원장 전체를 조작하는 것은 어렵기 때문에 Bruce는 오래된 거래를 지우는 새로운 암호화폐 스키마를 제안

Account tree라는 이름의 데이터 베이스는 모든 비어있지 않은 address의 잔액을 보관하는데 사용

- Redesigning Blockchain

Bitcoin-NG(Next Generation)이 제안

블록은 두 개의 파트로 나누자: 리더 선택을 위한 key block,. 거래 저장을 위한 microblock

이 프로토콜은 시간을 epoch로 나눈다

각 epoch에서 채굴자는 key block을 만들기 위해 해시를 해야 한다

일단 키 블록이 생성되면 그 node는 microblock 생성을 담당하는 리더가 된다

**4-B. Privacy Leakage**

블록체인에서 각 공개키에 대한 모든 거래 및 잔액의 값이 공개되어 있기 때문에 블록체인이 거래 프라이버시를 보장할 수 없다는 것이 나타났다

유저의 비트코인 거래를 연결해 유저의 정보를 얻는 것이 가능

NAT나 방화벽이 있더라고 유저의 필명(pseudonyms)과 IP를 연결할 수 있다

익명성을 개선하기 위한 방법이 제안

- Mixing

유저의 정보를 알아낼 수 있는 이유 중 하나는 유저가 하나의 주소로만 거래를 하기 때문이다

Mixing은 복수의 입력 주소에서 복수의 출력 주소를 사용해 익명을 보장하는 서비스

중재자를 사용해야 한다 -> 믿을 수 없는 중개자면 유저 정보를 드러낼 수도 있고, 중간에 자금을 가로챌 수도 있다

- Anonymous

제로코인에서는 zero-knowledge proof를 사용

채굴자는 디지털 서명이 된 거래는 검증할 필요가 없지만, 코인은 유효 코인 리스트에 속해 있다

**4-C. Selfish Mining**

블록체인은 이기적인 광부들의 담합 공격에 취약

Selfish mining 전략에서, 이기적인 채굴자는 그들이 채굴한 블록을 broadcast하지 않고 가지고 있고, 특정 조건이 만족되면 이 private branch를 공개한다

이러면 public보다 private branch가 더 길기 때문에 이를 다른 광부들도 승인해 주어야 한다

이러면 원래 public branch를 채굴하던 정직한 채굴자들은 필요 없는 곳에 자원을 낭비하게 된 것이다

이런 문제를 해결하기 위해 Heilman은 정직한 광부들이 어떤 branch를 따를지 선택하는 새로운 방식을 제안 -> 하지만 이러면 timestamps를 위조하는 것에 취약

ZeroBlock은 블록을 생성하면 일정 시간 내에 네트워크에 수용되어야 한다