

# A survey of blockchain consensus algorithms performance evaluation criteria

Seyed Mojtaba Hosseini Bamakan, Amirhossein Motavali, Alireza Babaei Bondarti,  
Expert Systems with Applications, Volume 154, 2020

<https://doi.org/10.1016/j.eswa.2020.113385>

# 大まかな内容

---

- ・ コンセンサスアルゴリズムのサーベイ論文
- ・ 既存のサーベイの分析手法の欠点を洗い出し, 改良
- ・ **スループット・マイニングの収益性・分散化の度合い・安全性** の4つの基準からなる分析フレームワークを提唱
  - ▶ それを用いて, 幅広いコンセンサスアルゴリズムを分析した

# 1. Introduction

---

## ブロックチェーンの種類

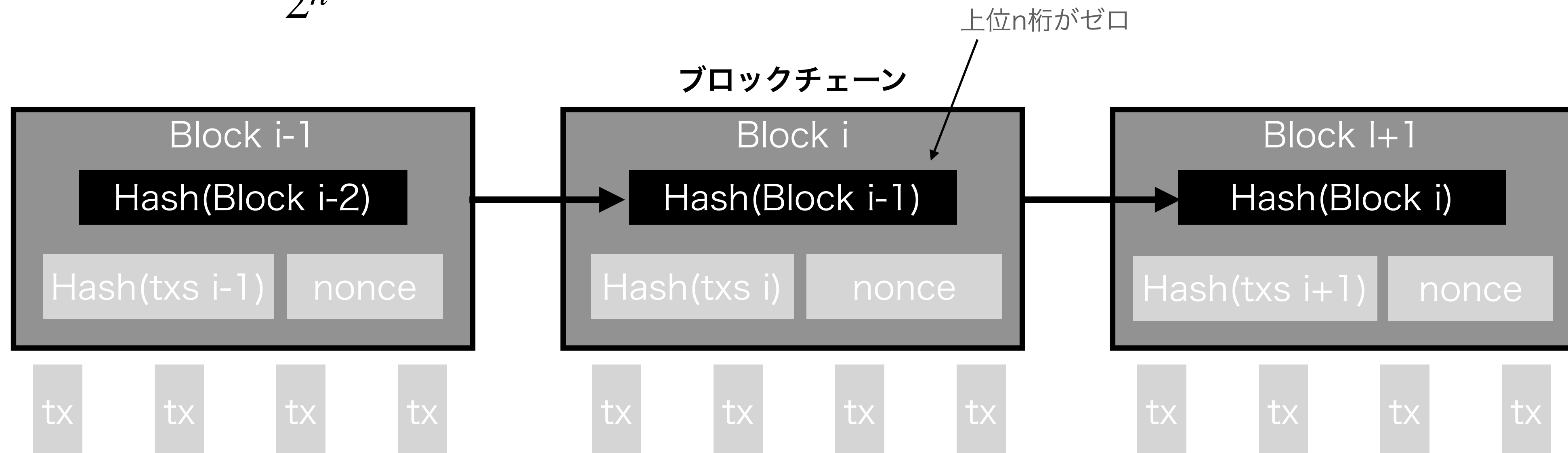
- Public blockchains
- Consortium blockchains
  - ▶ 情報はパブリックでも, 情報の更新は決められたグループのみが可能
- Private blockchains
  - ▶ 特定のグループのみが閲覧・更新可能. より中央集権的.
- Public blockchainではすべてのマイナーがコンセンサスを決定するが, その他は選択されたノード群あるいは一つの組織が決定することが可能

# 3. Blockchain consensus algorithms

## 3.1 Proof of Work (PoW)

- ・ **マイニング**: ブロックの1領域である **nonce** にランダムな値を入れて, 上位n桁が0になるようなハッシュ値を見つけようゲーム

- ・ 見つかる確率は  $\frac{1}{2^n}$  なので, 大体計算を  $2^n$  回やれば見つかる → 莫大な計算量



# 3. Blockchain consensus algorithms

---

## 3.1 Proof of Work (PoW)

- 長所
  - ▶ 安全性が高い
  - ▶ スケーラブル(最初から参加者の数がわかっていなくてよい)
- 短所
  - ▶ パズルを解くのに時間がかかる (Transaction per secが下がる)
  - ▶ エネルギー効率が悪い → 環境問題に

# 3. Blockchain consensus algorithms

---

## 3.2 Proof of Stake (PoS)

- 次のブロックの作成者を, Stakeの供給量, ランダムな順番など様々なパラメータによって選択
- スケーラビリティを改善
- マイナーは取引手数料のみ受け取る
- 多くのステークを持つノードに依存してしまい, 中央集権的になる
- Nothing at stake: ステークがなければ, 悪さをしてしても失うものがない  
“無敵の人” 状態

# 3. Blockchain consensus algorithms

---

## 3.2 Proof of Stake (PoS)

- ・ 長所

- ▶ 高速なブロック生成時間
- ▶ 高スループット
- ▶ エネルギー効率がいい
- ▶ スケーラビリティ
- ▶ ASICとかに依存しない

- ・ 短所

- ▶ 中央集権化の危険性
- ▶ 無敵の人対策を考えないといけない

# 3. Blockchain consensus algorithms

---

## 3.3 Delegated Proof of Stake (DPoS)

- PoSの改良
- ノードの投票によって代表を決定
- より中央集権的なアプローチ



# 3. Blockchain consensus algorithms

---

## 3.4 Proof of Elapsed Time (PoET)

- ネットワーク全体でランダムにマイナー選択をする
- 中央集権的なタイマーはない
- 各ノードのTrusted Execution Environment(TEE)でタイマーを実行
  - ▶ Hyperledgerの実装はIntel SGXで動く
- マイナーは, ちゃんと指定時間待ちましたよという証明を添付する

# 3. Blockchain consensus algorithms

---

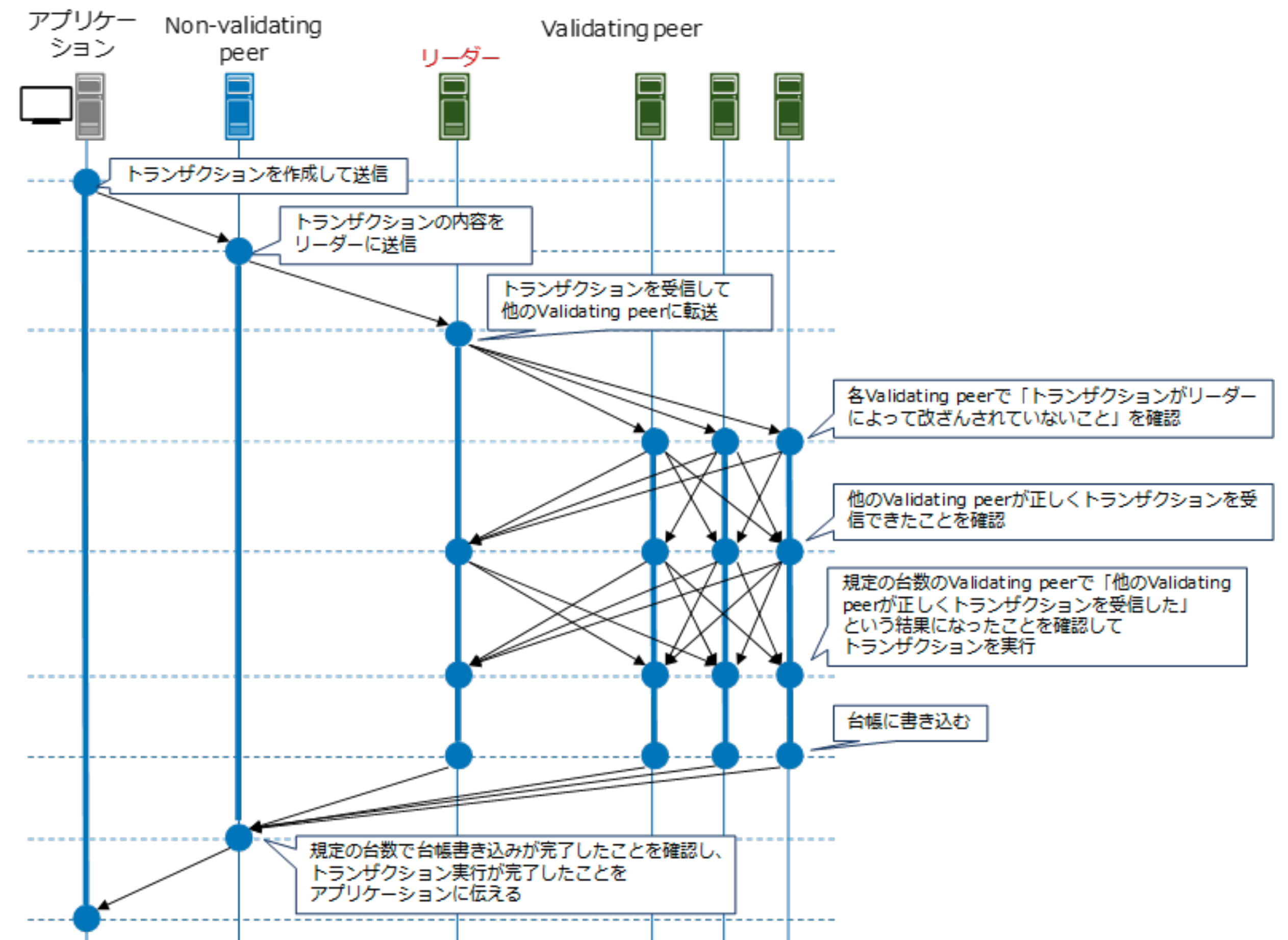
## 3.4 Proof of Elapsed Time (PoET)

- 長所
  - ▶ 選ばれるまでは寝ていればいいので, 大変エコ
- 短所
  - ▶ Intel(TEE作ってる会社)への依存
    - SGXは脆弱性どんどん見つかっている...

# 3. Blockchain consensus algorithms

## 3.5 Practical Byzantine Fault Tolerance (PBFT)

- 次のブロックを追加するために、すべてのノードが投票に参加する
  - ▶ 2/3がOKすればコンセンサスが成立
  - ▶ 1/3の悪意あるノードに耐えられる
- 全体数がわからないと投票が成り立たないので、Publicチェーンには向かない



<https://gaiax-blockchain.com/pbft>

# 3. Blockchain consensus algorithms

---

## 3.4 Proof of Elapsed Time (PoET)

- 長所
  - ▶ ファイナリティが得られる
  - ▶ マイニングいらないのでスループットが高速
- 短所
  - ▶ 参加ノードが増えると2次関数的に通信が増えるので増やしづらい

# 3. Blockchain consensus algorithms

---

## 3.6 Delegated Byzantine Fault Tolerance (dBFT)

- PBFTの改良
- 投票プロセスに全ノードが参加しなくてもよい
- いくつかのノードを代表者として選出され, 彼らが投票をする

# 3. Blockchain consensus algorithms

---

## 3.7 Proof of Weight (PoWeight)

- 各ユーザーに重み付けをする方式の総称?
  - ▶ どれだけ資金があるか
  - ▶ どれくらいIPFSのデータをPinしているか(Filecoin)
  - ▶ などなど
- PoSはステーク量による重み付けだったけど, PoW系は他の重み付けをしている



# 3. Blockchain consensus algorithms

---

## 3.8 Proof of Burn (PoB)

- PoBコインとPoWコインの2枚を作る
- マイナーは, すでに所有している暗号資産(PoWコイン)の一部を燃やす必要がある
  - ▶ 燃やす: Eater addressに送金する. 誰も二度と使うことができない.
- マイナーは, 長期的な報酬(採掘報酬)を受け取るために, 短期的な損失を引き受ける
- PoBコインの価値がPoWコインより高くなる前提?

# 3. Blockchain consensus algorithms

---

## 3.4 Proof of Elapsed Time (PoET)

- 長所
  - ▶ PoSと違って分散化が進む
- 短所
  - ▶ Burnするの, エネルギーの浪費では? という話もある



# 3. Blockchain consensus algorithms

---

## 3.9 Proof of Capacity (Proof of Space)

- マイナーはHDDの空きスペースを利用して採掘する
- 計算が遅いハッシュ関数を利用 ← 計算するより, 事前に計算したnonceをHDDから探したほうが早い
- プロットイング: マイナーのIDを含むデータをハッシュ化することを繰り返す
  - HDDにいろいろなnonceを書き込みまくる(HDDの空き領域を全部nonceで埋める)

# 3. Blockchain consensus algorithms

---

## 3.4 Proof of Elapsed Time (PoET)

- 長所
  - ▶ 特殊なハードウェアに依存しない(HDDがあればいい)
  - ▶ 電力消費を抑えられる (ASIC依存のPoWの30倍)
- 短所
  - ▶ HDD不足を引き起こす

# 3. Blockchain consensus algorithms

---

## 3.10 Proof of Importance

- PoSの改良
- 最近の取引回数, 取引相手の多さ, 保有コインの多さなどに応じてユーザーのスコアを決定
- スコアに応じてブロックを追加できるチャンスが高まる

# 3. Blockchain consensus algorithms

---

## 3.4 Proof of Elapsed Time (PoET)

- ・ 長所

- ▶ 高速で電力効率に優れる
- ▶ スコアリングシステムによって分散化, スケーラビリティ, 安全性が高まる (ほんまか?)
- ▶ 特別なハードウェアは必要ない

- ・ 短所

- ▶ HDD不足を引き起こす

# 3. Blockchain consensus algorithms

---

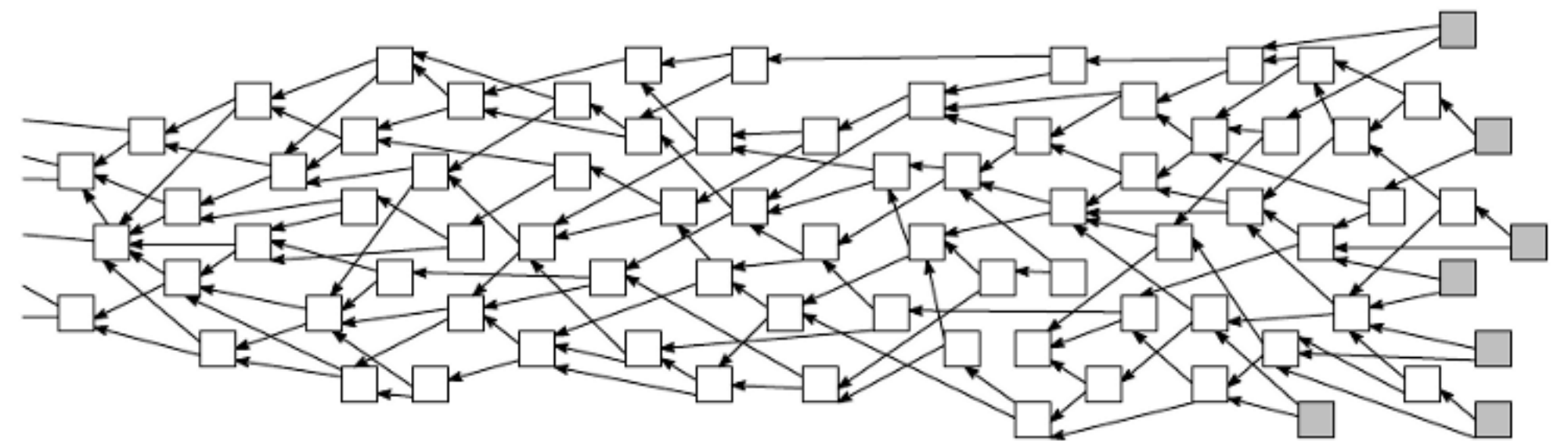
## 3.1.1 Proof of Activity

- PoWとPoSを組み合わせた
- マイニングはPoWでやり, その後PoSに切り替わる
- ランダムなバリデータが選ばれ, 検証後署名する
- 全コインの51%と採掘量の51%を同時に達成しないと乗っ取れない

# 3. Blockchain consensus algorithms

## 3.12 Directed Acyclic Graphs (DAG)

- コンセンサスアルゴリズムではないけど, 本論文に追加しました(?)
- ブロックチェーンとは別のデータ形式
  - ▶ 一本のチェーンにブロックをつなぐのではなく, 分岐ありの非循環・有向グラフにTXを繋いでいく
  - ▶ ユーザーが勝手に検証したTXの次にどんどん追加していく
- ブロックタイムが必要ないので, 超高速
- 幅が広がりやすい. どうするの.





# 4. Consensus algorithms evaluation criteria

## 評価基準の決定

- 過去のサーベイ論文で触れられていた評価基準

**Table 1**  
Summary of different sets of performance evaluation criteria used in blockchain consensus literature.

Authors	Year	Performance evaluation criteria
(Croman et al., 2016)	2016	1- Maximum throughput, 2- Latency, 3-Bootstrap time, 4-Cost per Confirmed Transaction, 5-Transaction validation, 6-Bandwidth, 7-Storage
(Baliga, 2017)	2017	1-Transaction finality, 2-Transaction rate, 3-Token needed, 4-Cost of participation, 5-Scalability of the peer network, 6-Trust model, 7-Adversary Tolerance
(Mingxiao et al., 2017b)	2017	1-Byzantine fault tolerance, 2-Crash fault tolerance, 3-Verification speed, 4-Throughput (TPS), 5-Scalability
(Xu, Luthra, Cole, & Blakely, 2018)	2018	1-Architecture (Accounts, Transactions, and Contracts, State Management, Execution Environment), 2-Fault tolerance, 3- Economic Systems Analysis, 4-Block Size, 5-Block Time, 6-Transactional Throughput, 7-Block Throughput, 8-CPU Usage, 9-Transaction Size
(Nguyen & Kim, 2018)	2018	1-Energy efficiency, 2-Modern hardware, 3-Forking, 4-Double spending attack, 5-Block creating speed, 6-Pool mining
(Wang et al., 2018a)	2018	1-Origin of Hardness, 2-Implementation description, 3-ZKP Properties, 4-Simulation of random function, 5-Features of puzzle design, 6-Virtual mining, 7-Simulating Leader election
(Tang et al., 2019)	2019	1-Basic technology, 2-Applicability, 3-TPS, 4-Market capitalization, 5-Number of forks, 6-Total commits in GitHub, 7-Ranking in GitHub, 8-Team activity
(Alsunaidi & Alhaidari, 2019)	2019	1-Node Identity management, 2-Data Model, 3-Electing miners method, 4-Energy saving, 5-Tolerated power of the adversary, 6-Transaction fees, 7-Block reward, 8-Verification speed, 9-Throughput, 10-Block creation speed, 11-Scalability, 12-Extendible 13-51% Attack, 14-Double Spending, 15-Crash Fault Tolerance, 16-
Byzantine fault tolerance (Hasanova et al., 2019)	2019	1-Double spending attack, 2-51% attack, 3-Private key security,4-Noting at stake, 5-criminal problem, 6-selfish mining, 7-block withholding, 8-Bribery attack, 9-DDos/DoS, 10-Sybil attack, 11-Routing attack,12-Time jacking attack
(Bano et al., 2019)	2019	1-Committee configuration, 2-Transaction censorship resistance, 3-DoS resistance, 4-Adversary model, 5-Throughput, 6-Scalable, 7-Latency, 8-Experimental setup

# 4. Consensus algorithms evaluation criteria

- 8名の専門家にアンケートを実施し、重要度を計算
- 4%を超えている項目を、性能評価のための重要な指標と位置づけた

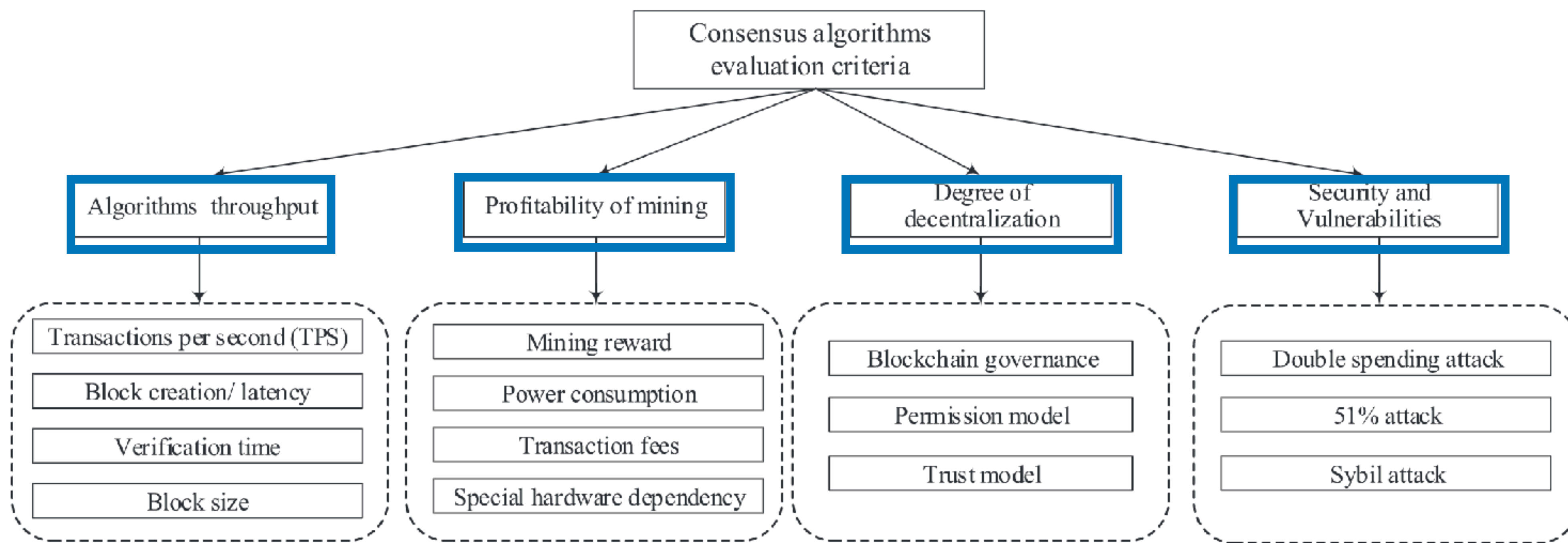
**Table 4**  
The importance of identified criteria.

Performance evaluation criteria	Weight%	Performance evaluation criteria	Weight%
Transactions per second	8.68	Hardware dependency	4.41
Transaction fees	4.57	Time jacking attack	2.54
51% attack	8.03	Double spending attack	8.85
Latency	5.29	Number of forks	2.29
Governance	4.24	Verification time	5.37
Virtual mining	2.29	Permission model	5.48
Block size	4.18	Power consumption	7.52
Routing attack	2.38	Block withholding	2.87
Trust model	5.75	Mining reward	4.32
Crash Fault Tolerance	2.93	Sybil attack	7.99



# 4. Consensus algorithms evaluation criteria

- Table 4の結果をさらに分析(?), 基準を4カテゴリに分けた



**Fig. 4.** A performance evaluation framework for blockchain consensus algorithms.

# 4. Consensus algorithms evaluation criteria

---

## 4.1 Throughput

- 最大スループット: ブロックチェーンが取引を確認できる最大速度
  - ▶ 最大ブロックサイズと, inter-block timeのトレードオフ

# 4. Consensus algorithms evaluation criteria

## 4.1 Throughput

- Transaction per second (**TPS**)
  - ▶ コンセンサスアルゴリズムに強く依存している

**Table 5**

Comparison of consensus algorithms based on their representative cryptocurrencies ([coincheckup, 2019](#)).

Row	Consensus algorithms	Cryptocurrencies	Algorithm	Genesis Block	Rank	Market CAP ( \$ )	TPS	Block Time Minutes	Mining reward
1	PoW	Bitcoin	SHA256	January 3, 2009	1	180,207,092,238	7	10	12.5 BTC
		Ethereum	Ethash (KECCAK256)	July 30, 2015	2	22,757,000,420	15	0.25	2
		Litecoin	Scrypt	October 8, 2011	5	4,587,952,794	28	2.3	25
		Monero	Cryptonight	April 18, 2014	11	1,268,871,523	30	2	4.9
		Zcash	Equihash	October 28, 2016	28	348,443,197	27	2	10
2	PoS	Waves (LPoS)	LPoS	June 12, 2016	55	100,304,755	100	1	Non-mineable
		Qtum	POS 3.0	December 26, 2016	36	202,601,750	70	2	Non-mineable
		Nxt	SHA256	November 24, 2013	175	16,162,355	100	1	Non-mineable
		Blackcoin	Scrypt	February 24, 2014	500	4,569,548	0	1	Non-mineable
		Nano	Blake2b	February 29, 2016	45	123,741,646	7000	Instant	Non-mineable
3	DPoS	EOS	DPoS	July 1, 2017	7	3,641,735,649	4000	0.5	Non-mineable
		Cardano	Ouroboros (DPoS)	December 26, 2017	12	1,266,573,741	257	0.33	Non-mineable
		TRON	DPoS	August 28, 2017	13	1,186,299,015	2000	0.05	32 TRON
		Lisk	DPoS	January 30, 2016	47	118,714,644	3	0.284	Non-mineable
		BitShares	DPoS	July 19, 2014	58	91,575,735	100000	0.05	Non-mineable
4	PBFT	Ripple	N/A	April 11, 2013	3	12,010,477,031	1500	0.06	Non-mineable
		Stellar	N/A	April 6, 2016	10	1,410,189,643	1000	0.08	Non-mineable
		Zilliqa	Keccak	January 12, 2018	79	59,022,911	0	45s to 4 m	Non-mineable
5	PoC	Burst	Shabal256	August 11, 2014	190	14,417,212	80	4	460
6	DAG	IOTA	Curl-P	October 21, 2015	17	788,711,735	1000	Instant	Non-mineable
		Byteball (Obyte)	DAG	September 5, 2016	262	17,301,594	10	0.5	Non-mineable
		Travelflex	DAG	December 2, 2017	1374	163,648	3500	1	30.00 TRF
7	PoA (Hybrid PoW/PoS)	Dash	X11	January 19, 2014	16	850,165,302	56	2.5	2.09
		Decred	BLAKE256	December 15, 2015	32	233,089,579	14	5	18.22
		Komodo	Equihash	September 1, 2016	67	80,699,867	100	1	3.00 KMD
		Peercoin	SHA-256	August 19, 2012	373	7,844,163	0	10	37.36 PPC
		Espers	HMQ1725	April 28, 2016	1026	625,199	0	5	5000
8	dBFT	NEO	RIPEMD160	October 17, 2016	20	650,866,809	1000	0.25	Non-mineable
9	PoI	NEM (XEM)	Ed25519	March 31, 2015	26	403,570,701	10000	1	Non-mineable
10	PoB	Slimcoin	Dcrypt	May 07 2014	2661	16,195	0.00003	1.5	50.00 SLM

# 4. Consensus algorithms evaluation criteria

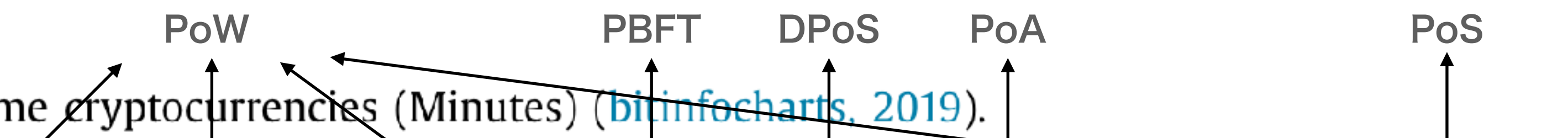
## 4.1 Throughput

- **Blocktime** / latency

- ▶ トランザクションを検証し、ブロックに配置する際の遅延
- ▶ あるトランザクションがネットワークに提示されてから、コンセンサスが得られるまでの時間のこと

**Table 6**  
Average block times of some cryptocurrencies (Minutes) ([bitinfocharts, 2019](#)).

Cryptocurrency	Bitcoin	Ethereum	Litecoin	Ripple	EOS	Dash	Dogecoin	Blackcoin
Min	7.40	0.22	2.10	0.00	0.00	2.56	1.02	1.01
Max	14.50	0.34	3.09	0.00	0.00	2.69	1.06	1.19





# 4. Consensus algorithms evaluation criteria

## 4.1 Throughput

- **Block verification time**

- ▶ ブロックが検証され, 正統なものと確認されるまでの時間

**Table 7**

Block confirmation time in different cryptocurrencies.

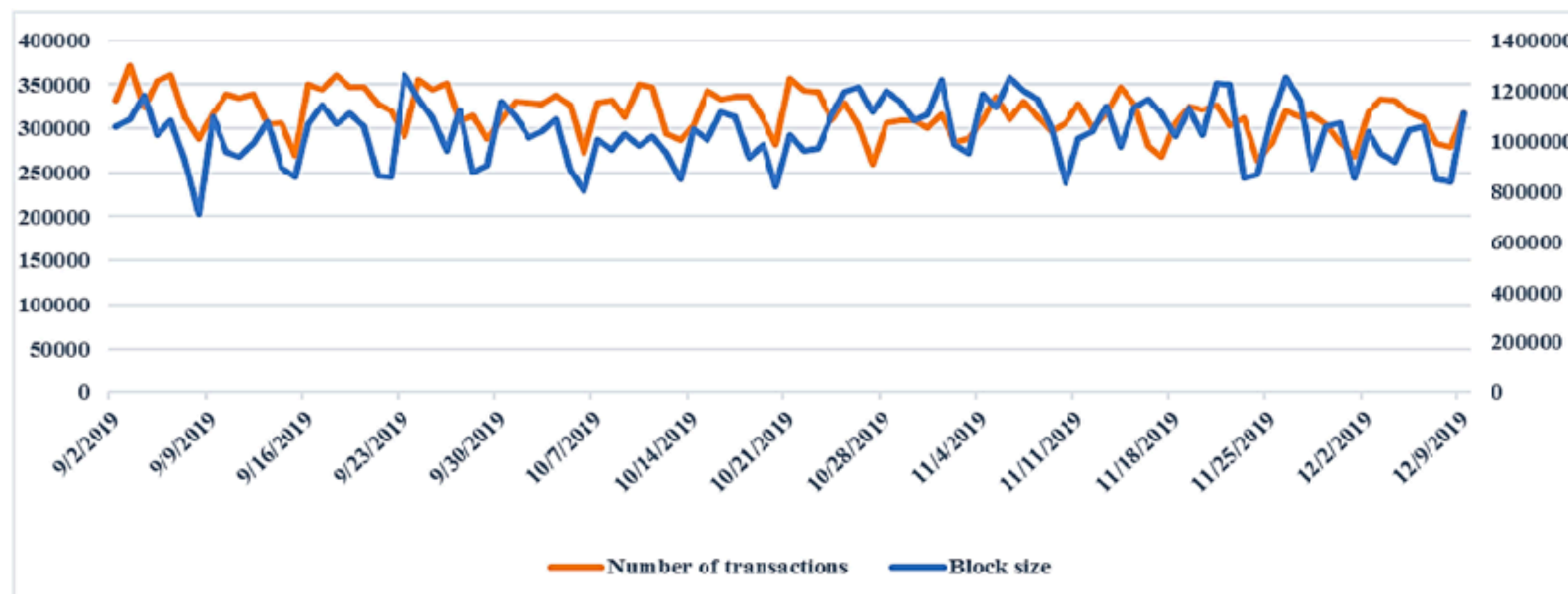
Cryptocurrency	Consensus algorithm	# of Confirmations	Validation Time (Minutes)
Augur (REP)	PoW	30	6
Bitcoin (XBT)		6	60
Bitcoin Cash (BCH)		15	150
Dogecoin (XDG)		20	20
Ether (ETH)		30	6
Litecoin (LTC)		12	30
Monero (XMR)		15	30
Zcash (ZEC)		24	60
Cosmos (ATOM)	PoS	N/A	Near-instant
Qtum (QTUM)		24	60
Dash (DASH)	PoA	6	15
Ripple (XRP)	PBFT	N/A	Near-instant
Stellar Lumens (XLM)		N/A	Near-instant
Cardano (ADA)	DPoS	15	10
Eos (EOS)		N/A	Near-instant

# 4. Consensus algorithms evaluation criteria

## 4.1 Throughput

- Block size

- ▶ 1ブロックに詰め込めるデータサイズ
  - Bitcoin: 1MB
- ▶ TX数はBlock sizeに依存する



- ▶ ブロックサイズが小さいと…
  - ネットワークが遅くなる
  - TX数が増加すると詰まる
  - 取引手数料が高くなる
- ▶ ブロックサイズが大きいと…
  - セキュリティが低下する

# 4. Consensus algorithms evaluation criteria

---

## 4.2 Profitability of mining

- ・ マイニングの収益性に影響を与える要因は様々
  - ▶ プロセスの複雑さ
  - ▶ マイニング報酬
  - ▶ 電力消費
  - ▶ 取引手数料
  - ▶ 特定のハードウェアへの依存度

# 4. Consensus algorithms evaluation criteria

---

## 4.2 Profitability of mining

- Mining rewards
  - ▶ マイナーが取引の検証プロセスに参加するインセンティブ
  - ▶ 参加者がインセンティブに従って行動するという前提のもと、ネットワークの安定性がもたらされている
  - ▶ 難易度が高いと、マイニングプールに参加することが合理的になってしまう(分散性をそこなう)



# 4. Consensus algorithms evaluation criteria

---

## 4.2 Profitability of mining

- Power consumption
  - ▶ PoWは国家レベルの電力を消費する
    - ほとんどは中国の火力発電所から… → 地球温暖化
  - ▶ 採掘効率: 1秒あたりのハッシュ数を消費電力で割った比率
    - より採掘効率のよいコンセンサスアルゴリズムを採用したいよね

# 4. Consensus algorithms evaluation criteria

---

## 4.2 Profitability of mining

- Transaction fee
  - ▶ 高い!
  - ▶ 取引手数料をなくすためのアプローチ
    - Developerへの課金
    - 取引手数料用のトークンを分ける
    - ユーザーに検証させる (IOTAなどDAG系)
    - プラットフォームへの補助金(?)

# 4. Consensus algorithms evaluation criteria

---

## 4.2 Profitability of mining

- Special hardware dependency
  - ▶ ASICを持っていないとマイニングに参加しづらい…
  - ▶ PoS, dPoS, PoA系はASIC耐性を高める設計をしているものがある
  - ▶ PoET: Intel SGXに依存する代わりに電力消費を下げている

# 4. Consensus algorithms evaluation criteria

---

## 4.3 Decentralization levels

- Governance
  - ▶ コンセンサスアルゴリズムとは直接関係しないが, 分散化の度合いを定義する上で大切
  - ▶ インフラによるガバナンスと, インフラのガバナンス
    - 特にインフラのガバナンスは色々問題がある(テクノクラート優位)

# 4. Consensus algorithms evaluation criteria

## 4.3 Decentralization levels

- Permission model

- ▶ Public/Private以外にも,  
Permissionless/Permissionedの  
カテゴリがある

- Public/Private: 検証者の匿名性  
の度合い

- Permissionless/Permissioned:  
バリデータの信頼度合い

**Table 9**

comparison of some cryptocurrencies in terms of permission models.

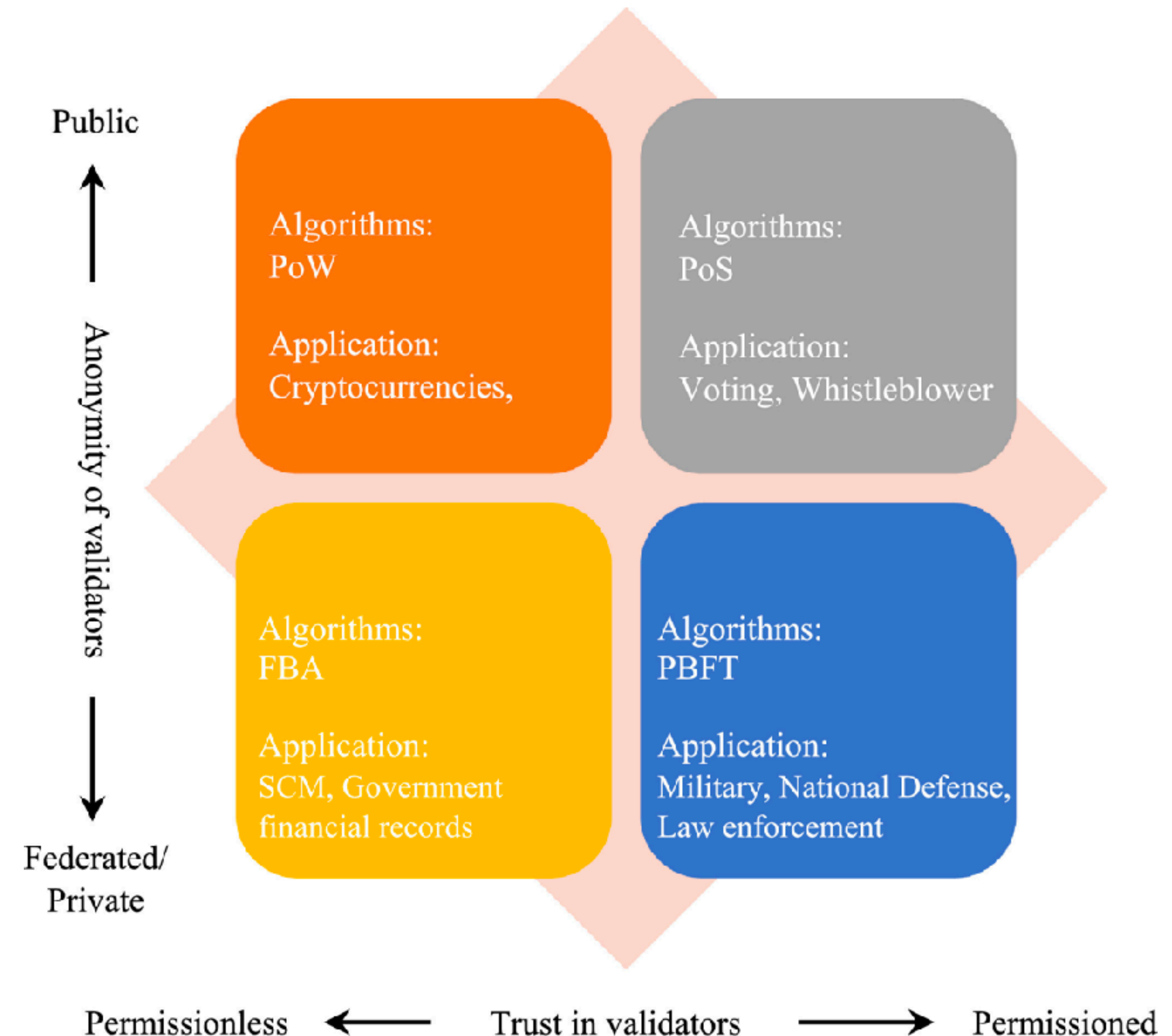
Permission Model	Trust Model	Cryptocurrency	Consensus Model
Permissioned	non-trustless	Neo	dBFT
	non-trustless	Icon	LFT (Loop Fault Tolerance)
	non-trustless	WTC	Hybrid PoW/PoS
	non-trustless	EOS	DPoS
	non-trustless	Ark	DPoS
	non-trustless	Lisk	DPoS
	non-trustless	VeChain	Proof-of-Authority
	non-trustless	Nuls	Proof-of-Credits
	trustless	Hashgraph	asynchronous BFT
Permissionless	trustless	Bitcoin	PoW
	trustless	Ethereum	PoW
	trustless	Litecoin	Pow
	trustless	Dogecoin	PoW
	trustless	Monero	PoW
	non-trustless	Stellar	PBFT
	non-trustless	XRP	N/A
	trustless	Nano	PoS
	trustless	Cardano	DPOS
	trustless	Decred	Hybrid PoW/PoS
	trustless	Zilliqa	PBFT
	trustless	Elastos	DPoS
	trustless	IOTA	DAG



# 4. Consensus algorithms evaluation criteria

## 4.3 Trust model

- Trust model
  - ▶ すべてのブロックチェーンがトラストレスというわけではないよ
    - リップルとか...



# 4. Consensus algorithms evaluation criteria

---

## 4.4 Blockchain consensus vulnerabilities

- Double spending attack
  - ▶ 不正者が, 2重支払いを含むブランチを伸ばそうと頑張る
  - ▶ 理論上起こりうる

# 4. Consensus algorithms evaluation criteria

---

## 4.4 Blockchain consensus vulnerabilities

- 51% attack
  - ▶ PoW, PoSなど
  - ▶ PoAは、攻撃者が全コインの51%と同時に採掘力の51%の両方を持たないといけないので、よりコストが高まる



# 4. Consensus algorithms evaluation criteria

---

## 4.4 Blockchain consensus vulnerabilities

- Sybil attack
  - ▶ たくさんのユーザーIDを作成することで投票数・影響力をかさ増し
  - ▶ 正直なノードを取り囲んで、台帳に届く情報を遮断するなど
  - ▶ 対処法
    - ノード作成のコストを上げる(ある程度のコインを保持させる)
    - 認証を求める
    - ノードへの重み付け(PoWeight)

# 5. Discussion and challenges

## Summary of comparison

**Table 10**

Summary comparison of blockchain consensus algorithms.

Consensus algorithms	Designing Goal	Decentralization level	Permission model/ Node Identity Management	Electing Miners/ verifiers Based on	Energy efficiency	Scalability	%51 Attack	Double Spendingattack	Hardware dependency	speed
PoW	Sybil-proof	Decentralized	Permissionless	Work (Hash)	No	Strong	Vulnerable	Vulnerable	Yes	Slow
PoS	Energy efficiency	Semi-centralized	Permissionless	Stake	Yes	Strong	Vulnerable	Difficult	No	Fast
DPOS	Organize PoS effectively	Semi-centralized	Both	Vote	Yes	Strong	Vulnerable	Vulnerable	No	Fast
PBFT	Remove software errors	Decentralized	Both	Vote	Yes	Low	Safe	Safe	No	Slow
PoC	Less energy than PoW	Decentralized	Permissionless	Work (Hash)	Fair	Strong	Vulnerable	Vulnerable	Yes	Slow
DAG	Speed and Scalability	Decentralized	Permissionless	N/A	Yes	Strong	Safe	Safe	No	Fast
PoA	Benefits of both Pos and PoW	Decentralized	Permissioned	Vote and work	No	Strong	Safe	Vulnerable	Yes	Fair
dBFT	Faster PBFT	Semi-centralized	Permissioned	Vote	Yes	Medium	Vulnerable	Vulnerable	No	Slow
PoI	Improve PoS	Decentralized	Permissionless	Importance scores	Yes	Strong	Safe	Safe	No	Fast
PoB	N/A	Decentralized	Permissionless	Burnt coins	No	Medium	Vulnerable	vulnerable	No	Fast

# 感想

---

- いろいろなコンセンサスアルゴリズムがあるんだな
- あらゆるチップに(脆弱性のない)Trusted Execution Environmentが搭載される未来がきたら, PoETとても良さそう
  - ▶ そんな未来くるか?
  - ▶ もうすこし勉強したい
- DAG面白そう
  - ▶ ほんまにPermissionlessか? (IOTAなどはコーディネーターが権力を握っているはず)
  - ▶ DAGをPermissionlessにできたらアツそう