COOPERATIVE CRYPTOMEDIA

MICHAEL F. SCHREIBER

Abstract. Cooperative crypto media support tests of alternative models for political economies in automated environments by offering features like key-less containment or concurrent capacity by digest proximity. Proposed components and compositions could evolve fair, sustainable rewards through participatory self-regulation. Constraints and perspectives are outlined with formulas, elements of protocols and envisioned aspects of functionality made possible by designing for privacy and efficiency with concurrent capacity, intermedia cooperation, and containment of temporal channels for atomic settlement by endogenous synchronization according to formal measures of digest proximity.

Date: May 18, 2018: snap 0.69 base research lab10.coop.

2010 *Mathematics Subject Classification.* Primary: 68P30.

Key words and phrases. api, authentication, authorization, bit budget, blockchain, claim, coalition, commitment, compression, concurrency, consensus, constructor, containment, counter machine, cryptography, economy, emulation, encoding schemes, endogenous timing, fairness, fee, fixed point valuation, fixed supply, floating point range conversion, gas, game, governance, growth, half time, hash, internet of things, income, message digest, models of communication, models of computation, money, monetary policy, oracle, policy, proximity, random access machine, redundancy, regulation, reward, routing, sampling, shard, smart contract, stream, super information, super dense encoding, swarm, token, uniticity, universal basic income, very low earth orbit constellation, virtual machine, wealth.

Outline May 18, 2018

1. Introduction

This exposition outlines contexts for crypto media cooperation to gather, find, and combine ideas for sustainable solutions of common problems that may generate and distribute fair rewards for cooperation.

Temporal proximity mechanisms of atomicity for shards and post quantum crypto ways to distinguish concurrent containments emerged — with all other aspects included — from efforts to answer the question how such a crypto media cornucopia can be for all forever.

2. Understanding crypto

Reading decodes. Channels of communication require senders to encode and receivers to decode. Voices encode messages in air, ears pick up oscillating pressure — what is said needs context to be understood.

Additional layers of encryption are designed to disappear for sender and receiver but they change how media can be used by enforcing and thereby confirming relations between keys and contexts of messages.[1] Arbitrary autonomous cryptographic relations among media are useful in practice — but details and consequences can be quite complicated to present and comprehend.

2.1. Compact notations for coalitions.

Formal description helps to describe critical constraints of related games and to propose suitable pure or mixed strategies for coalitions in crypto media environments. Minimal coalitions may seem sufficient to exploit hyped phenomena like crypto media tokenization at first glance — but observations noted here might help broader coalitions necessary to surf this vortex of converging accelerations.

$$\Game_{game} = \{coalitions, expectations, strategies\} \tag{1}$$

Table 1. Sorts of crypto media coalitions

white	yellow	magenta	cyan
rest gov biz	rest gov	rest biz	gov biz

2.2. Fair game update order.

In fair games all expected rewards for coalitions can be rescaled to zero by including a constant offset unless players fail to play well enough. Consensus about events received and their order demands exponential increases of capacity for communication, computation and storage.

Table 2. Transactions per second ordered by consensus protocols for proofs of work (PoW), proofs of stake (PoS), commercial transaction accounting (cta), redbelly scaling concurrent proof of validity (PoV), hedera proofs of hashgraph (PoHG)

Exponent	TX per second	Consensus	Year	Example
<5	< 32	PoW	2008	BTC, ETH, ...
<10	<1'024	PoS	2017	Tendermint Cosmos
<16	< 65'536	cta	2017	Credit Card Company
<19	<524'288	PoV	2017	Redbelly
19	524'288	PoHG	2018	Hedera Hashgraph LIT
22	4'194'304	?	2018	Satori Events
35	34'359'738'368	?	2020	IOT

2.3. **Fast protocols.** Traditional accounting accepts or net within 7 seconds. Ethereum offers proof of work consensus within 14 seconds but — like all open distributed ledger proof of work chains — recommends to wait for half a dozen confirmations by later blocks of consensus.

2.3.1. *Introspective protocols.* A new class of introspective protocols — which deduce votes about snapshots from gossip about gossip instead of tossing lots of ballots to and fro — currently introduced as Hashgraph designs need 3 seconds for large amounts of transactions across global distances.[2]

2.3.2. *Bitpermutation protocols.* Faster protocols enable network communication infrastructures which make and sell advantages with respect to other dynamic environments including seasonal fluctuations, technical progress and cultural innovations.

Protocols limited only by network latency gain advantages with G5 mobile network rollouts, very low earth orbit satellite backbones and potentially even super dense encoding in quantum channels which may allow physical constructors to pay in units of Planck energy per permuted bit without costs for intermediate computation or transmission.

[1] An essential effect of such features is that *crypto media gain a type of autonomy by asking for your key* — perhaps doing something else if given none or a special one.

[2] Australian 'Redbelly' consensus of validity seems to scale well in capacity with numbers of participating nodes. Compare `http://redbellyblockchain.io` and description of validity consensus in: 'Blockchain Consensus', Tyler Crain1, Vincent Gramoli1, Mikel Larrea1 and Michel Raynal

2.4. **Local hub limit.** Any coalition can use crypto media to accelerate its processes of innovation if their socio-physical media environments can supply enough capacity for traditional and envisioned applications.

Financing of large projects like opening dedicated power generation for mining can be assembled by micro payments which may result in more projects which contribute to overall acceleration of local clusters connected to global fiber and orbital networks.[3]

Protocols for hubs and endpoints can push or pull to move m messages from any of n input nodes to r resilience nodes which support f feature usage frequencies for u users.

$$\frac{m}{n} * r * p_{overhead} = m_{import} \tag{2}$$

$$\mathbf{f} \cdot \mathbf{u} * p_{overhead} = m_{export} \tag{3}$$

$$(m_{import} + m_{export}) * p_{overhead} = m_{transport} \tag{4}$$

2.5. **Digest map.** This motivates constructions of concurrent crypto media environments which partition their tasks and their nodes assigned to perform them.

2.5.1. *Trade-off between resilience and resolution.* Given a number of nodes n and a required multiplicity k for resilience by redundancy expected loads can be divided among n/k shards s.

$$s = \lfloor \frac{n}{k} \tag{5}$$

$$b = \lfloor \log_2(s) \tag{6}$$

$$n \geq 2^b * s \tag{7}$$

The number of bits resulting from this division — or a similar computation like the integer part of the binary logarithm — determines the number of bits used to distinguish tasks. Message digests of transaction headers are used to assign tasks.

2.5.2. *Digest proximity.* One transaction sent from each one of 16 billion users could be assigned by matching 12 bits with those of 4096 shards each offering 4096 batch processing cycles. Then each shard would have to process about 1000 transactions per batch.[4]

[3] Intelligent hubs able to collect, relay, fan-out, balance, mix, and compress quantum information channels will test what is possible within limits shown for a hub of given density and temperature by Bekenstein, Bremermann, Landauer and others. Compare `https://en.wikipedia.org/wiki/Limits_of_computation`

[4] A project like Space X might let 10240 satellites run 1024 orbital shards with multiplicity 10 by comparing 10 bits in cryptographic digests of transactions and orbit specifications. Delegating

Table 3. Match of digest bits for transactions and node addresses determines assignment of tasks to shards each coordinating approximately $\frac{n}{2^b}$ nodes performing the same updates.

b_3 match	1	1	1	1	0	0	0	0
b_2 match	1	1	0	0	1	1	0	0
b_1 match	1	0	1	0	1	0	1	0
Shard indices	7	6	5	4	3	2	1	0

2.5.3. *Contemporary granularity.* Transaction capacity per batch cycle determines shard throughput and thus how many transactions per number of batches might be considered fair use.

Table 4. Match of digest bits for transactions and node addresses determines assignment of tasks to batches of concurrent processing in shards – again each coordinating deterministically approximately $\frac{n}{2^b}$ nodes performing the same updates in the same batch cycle.

b_{-3} match	1	1	1	1	0	0	0	0
b_{-2} match	1	1	0	0	1	1	0	0
b_{-1} match	1	0	1	0	1	0	1	0
Batch-time indices	7	6	5	4	3	2	1	0

Transactions could be cheaper if they can risk an expected delay of less than t batch cycles. Larger fees for precise timing could make sense for some transactions.

Temporal synchronization of particular pairs or ensembles of transactions can pack these actions into one batch or a sequence of batches instead of filling task slots according to digest proximity.

$$k * C_{2^{s_b}}^{2^{f_b}} = \begin{bmatrix} k_{11} & k_{12} & k_{13} & k_{14} & k_{15} & k_{16} & k_{17} & k_{18} \\ k_{21} & k_{22} & k_{23} & k_{24} & k_{25} & k_{26} & k_{27} & k_{28} \\ k_{31} & k_{32} & k_{33} & k_{34} & k_{35} & k_{36} & k_{37} & k_{38} \\ k_{41} & k_{42} & k_{43} & k_{44} & k_{45} & k_{46} & k_{47} & k_{48} \\ k_{51} & k_{52} & k_{53} & k_{54} & k_{55} & k_{56} & k_{57} & k_{58} \\ k_{61} & k_{62} & k_{63} & k_{64} & k_{65} & k_{66} & k_{67} & k_{68} \\ k_{71} & k_{72} & k_{73} & k_{74} & k_{75} & k_{76} & k_{77} & k_{78} \\ k_{81} & k_{82} & k_{83} & k_{84} & k_{85} & k_{86} & k_{87} & k_{88} \end{bmatrix} = k * C_3^3 \tag{8}$$

Ecologic impacts of transactions depend on redundancy k and technological progress μ following Moore's law. Demand increases with the number of users and intensity of usage saturating capacity made available by shards offering temporal granularity

redundancy to nodes on earth 4096 shards for 12 bits of distinction would be possible even before the full constellation is ready.

and synchronicity for batches of transactions.

$$\mu = \alpha * \tau \qquad \text{(Moore's law)} \tag{9}$$

$$C = 2^{s_b} * 2^{f_b} * 2^{\mu} \tag{10}$$

$$C = 2^{s_b+f_b+\mu} \tag{11}$$

$$D = a_n * \overline{x}_{a_{ij}} \tag{12}$$

$$\text{(Ecologic footprint)} \qquad W = w * k * C = w * k * D \tag{13}$$

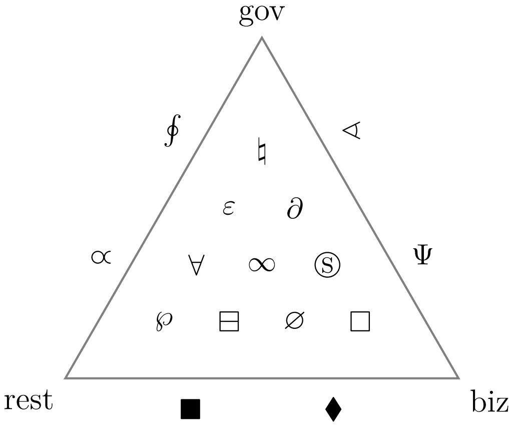

Figure 1. Proximity of potential crypto media features to viewpoints of governmental regulator, commercial entity and public cultures around almost infinite capacity ∞: electoral governance ε, integrated taxation of circulation-storage-computation $\oint$, net clearing $\varnothing$, synchronous atomic settlement Ⓢ, privacy $\wp$, count-down keyless containment and account protection $\boxminus$, direct universal income for all participants $\forall$, natural non-proof-of-work replication $\natural$, partial payment streams ∂, black box consumption $\blacksquare$, direct peer markets $\blacklozenge$, general quantum intelligence Ψ, proportional compensation for ecologic externalities $\propto$, open review software and hardware designs $\square$.

2.6. **Short stories.** Accumulated records of previous transactions can be excluded from current computations and handled by dedicated subsidiary services. Only a common context of most recent commitments and balances has to be provided by shards unless regulations require further overhead.

A minimal thin history protocol might need physical storage split among shards with appropriate redundancy through node multiplicity for:

(1) state currently being stored

(2) previous state

(3) former previous state

(4) state currently erased

Further reduction of storage requirements by erasure from concurrent repositories after a limited number of shard blocks would be possible.

2.6.1. *Digest count down chain containment.* Access to accounts is protected by posting digests of messages followed by posts including actual content thus confirmed. This mechanism avoids public-key cryptography which will be weakened by emerging capabilities to perform quantum computation.

Braided digest commitments tagged by extensions of this key-less mechanism provide containment for subsidiary block-chains supporting smart contracts and storage services in compatible crypto media environments.

2.6.2. *Momentary stripes of contract objects.* Crypto media can organize their context of concurrency similar to the architecture of a 'general parallel file system' like the eponymous IBM storage solution or SWARM implementation of interplanetary file systems (IPFS). A generalization suitable for concurrent transaction environments applies distributed ledger techniques to dedicated subsidiary chains managing just one object which encapsulates history for a contract variable.

$$\frac{\uparrow\Rsh\uparrow \ldots \uparrow\Lsh\uparrow^{k}}{\uparrow\uparrow\uparrow \ldots \uparrow\uparrow\uparrow_{0}} \ldots S_i^k \ldots \frac{S^k_{update}}{S^k_{previous}} \ldots S_j^k \ldots \frac{\uparrow\uparrow\uparrow \ldots \uparrow\uparrow\uparrow^{k}}{\uparrow\Lsh\uparrow \ldots \uparrow\Rsh\uparrow_{s}} \tag{14}$$

Shard consensus about objects can be limited to registrations of updates in counters thus offering root reference digests for snap-shots which can be referenced by subsidiary cryptographic mechanisms like Merkle proofs, zk-Snarks, or future standard protocols for post-quantum cryptography.

2.6.3. *Gossip consensus.* A gossip protocol can generate Merkle proofs of inclusion in chains maintained by nodes which reach consensus about events and their ordering. Nodes select one peer from a list of other nodes in their shard to update each other about their recent transactions. They track how many Merkle proofs have been given for a message and confirm to the sender whenever enough confirmations have been gathered by a message which itself gets augmented like a chain.

$$a(tx, K_0^{n-1}) \rightarrow b(P_{tx,B_0}, K_1^{n-1}) \tag{15}$$

$$b(P_{tx,B_0^i}, K_i^{n-1}) \rightarrow b(P_{tx,B_0^{i+1}}, K_{i+1}^{n-1}) \tag{16}$$

$$b(P_{tx,B_0^m}, K_{m+1}^{n-1}) \rightarrow c(P_{tx,B_0^m}, \ldots) \tag{17}$$

Consensus confirmation in a shard can be forwarded as transaction to peers on a higher level of sharding or propagated to lower level components.

Nodes keep track of transactions reported in previous gossip and may bundle proofs given for individual transactions into Merkle tree storage formats. Communication among peers may use dedicated channels which limits the size of blocks containing proofs.

2.7. **Concurrent contracts for cooperative crypto media.** Arbitrary concurrent processes can be constructed by contracts which rely on temporary containment of protocols for expected events. They can process progress of whatever hand-shake negotiation is required by regulation or intentions of interactive transactions.

Peer chains from different shards could mirror collections of root hash values gathered from participating networks which can thus use common points of reference for proofs to give for digests of transactions.[5]

2.7.1. *Compatibility of distributed ledgers.* Distributed ledgers can offer their features for free or require fees payed in traditional fiat currency or any crypto-media denomination which offers contracts or similar interfaces which support transactions. They may offer some or all typical features listed by current representatives while adding further features for particular use cases or regulatory requirements.

> The properties of distributed ledger technology (according to Distributed Ledger Foundation, Whitepaper 2018, p, 2):
>
> **(1) Distributed:** All network participants have a full copy of the ledger for full transparency
>
> **(2) Anonymous:** The identity of participants is either pseudonymous or anonymous
>
> **(3) Unanimous:** All network participants agree to the validity of each of the records
>
> **(4) Time-stamped:** Transaction timestamp is recorded
>
> **(5) Immutable:** Any validated records are irreversible and cannot be changed

[5] Consolidation across different frequencies of updates could just take the first or most recent update received as suggested by Vitalik Buterin for preliminary shard development for future test net versions of Ethereum. `https://ethresear.ch/t/initial-explorations-on-full-pos-proposal-mechanisms/925`

(6) Programmable: A blockchain is programmable ('Smart Contracts')

(7) Secure: All records are individually encrypted

2.7.2. *Transitions for transfer of crypto media.* Crypto media transfers could pass to another account on another chain directly or through a locked state if other transfers have to be confirmed in the same consensus procedure.

$$-v_a \Rightarrow +v_b \tag{18}$$

$$-v_a \rightarrow +v_{a'} \Rightarrow -v_{b'} \rightarrow +v_b \bigwedge -v_x \rightarrow +v_{x'} \Rightarrow -v_{y'} \rightarrow +v_y \tag{19}$$

2.7.3. *Fungible supply.* Ledgers which support tracing of stolen funds will need particular mechanisms to protect against liability and support required levels of compatibility. Patterns used to generate indicators for funds to be frozen or excluded from conversion at licensed exchanges include (preprint: Making Bitcoin Legal, Ross Anderson, Ilia Shumailov and Mansoor Ahmed Cambridge University Computer Laboratory):

(1) Poison: All outputs with illegal inputs become illegal — too many false positives

(2) Haircut: Outputs become partially illegal in proportion to inputs — dilution results in too many false negatives

(3) First in first out (FIFO): Variant of standards used in tracing money — remarkable precision.

Coins with built-in money laundering capabilities like Zcash or Monero might have to comply with regulations of pseudonymous transactions involving traced funds or move to unlicensed ledgers supporting peer-to-peer arrangements.[6]

Keeping senders of payments anonymous at least for small amounts but making receivers identified and taxable like 'GNU-Taler' may provide an open source approach[7] for web services and merchants but requires arrangements with payment providers or banks settling accounts with users.

2.7.4. *Embodied agents.* Gadgets which protect data in their secure enclaves and present interactive physical controls for virtual interfaces might become typical endpoints of crypto media transactions. End-to-end encryption of all transactions among devices with such features would then be possible in principle. Block

[6] Onioncoin for instance is a native coin with ledgers hosted as hidden Tor services.

[7] Compare Richard Stallman 'A radical proposal to keep your personal data safe' `https://www.theguardian.com/commentisfree/2018/apr/03/facebook-abusing-data-law-privacy-big-tech-surveillance`

chains for particular devices might result in tens of billions of concurrent ledgers aggregated for billions of users coordinating some or all of their devices.

2.7.5. *Continuous consensus.* Pay-as-you-go arrangements for parking, charging, power generation, rentals, and other services may call for rapid inclusion of updates relating to physical location and particular interaction scenarios of usage. Scheduled intervals and just-in-time adjustments for cars or drones may require dedicated crypto media infrastructures.

3. Rewards

Science and technology might be able to create and sustain abundance by overcoming planetary limitations of populations unless kept in limited supply or demand.

Crypto media of reward create capacity for fair distributed access to dividends of research, education and automation.

3.1. **Accelerated economy.** Common expansion of economic activity possibly constrained by preference for ecologic sustainability and fair distribution of income is often accepted as a common measure of successful economic cooperation.

$$\left(\begin{smallmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{smallmatrix}\right) \cdot \left(\begin{smallmatrix} b_1 & b_2 \end{smallmatrix}\right) = \left(\begin{smallmatrix} c_1 & c_2 \end{smallmatrix}\right) \tag{20}$$

$$\mathbf{A} \cdot \mathbf{b} = \mathbf{c} \tag{21}$$

$$a_{1,1} * b_1 + a_{1,2} * b_2 = c_1 \bigwedge c_2 = a_{2,1} * b_1 + a_{2,2} * b_2 \tag{22}$$

3.2. **Diagonal and transactional rewards.** Transactions among economies, organizations and individuals can take place among participants sitting along the main diagonal of a table. Their common matrix can be understood as a product of eigenvectors $\mathbf{e_a}$ and eigenvalues $a_{e_{ij}}$ on a diagonal.

$$\mathbf{e}_a \cdot \mathbf{A}^{\mathbf{T}} = \begin{pmatrix} a_{e_{11}} & & \\ & \ddots & \\ & & a_{e_{nn}} \end{pmatrix} \cdot \mathbf{e}_a \tag{23}$$

Its other values then describe a common transformation of coordinated positions that can be computed concurrently if all partitioned groups of transactions can be regarded as happening in isolation without any effect on each others results.

Direct diagonal rewards for authorized owners a_e or all transactions $a_{e_{ij}}$ become possible in this model.

3.3. **Capacity for concurrent transactions.** Capacity is organized by grouping n nodes into 2^b shards handling transactions hashed to b bit patterns matching digests of transactions and nodes.

$$\mathbf{a} \pm \mathbf{b} \Rightarrow S_{k=3}^{2^b=4}(\mathbf{a}, \mathbf{b}) = \begin{bmatrix} s_{11} & s_{12} & s_{13} & s_{14} \\ s_{21} & s_{22} & s_{23} & s_{24} \\ s_{31} & s_{32} & s_{33} & s_{34} \end{bmatrix} (\mathbf{a}, \mathbf{b}) \Rightarrow \mathbf{c} \tag{24}$$

Efficient concurrent capacity makes transactions faster and reduces their cost — by fair partitioning of transactions with cryptographic mechanisms for privacy and participatory processes.

3.4. **Compatibility with expectations.** Admissible mixed strategies of coalitions approach their limits of risk tolerance to maximize discounted expected rewards of strategies. Environment which try to ignore or prohibit too many sorts of crypto media may curtail their chances to accelerate other innovation processes also. Specialization and regulation will evolve co-existing environments and some will gain comprehensive approval to become valuable global capacities of coordination and self-governance.

3.4.1. *Mint.* Contribution of time and resources to running nodes and other activities for particular crypto media are most often rewarded by minting tokens which are expected to keep or increase their value over time. This actually worked very well for those holding or trading crypto denominations during periods of extreme volatility but made them less attractive for pricing traditional offerings.

Newly launched denominations often include procedures to reward developers and early investors. Recently it became popular to reward early adopters by giving away substantial amounts through air-drops.

3.4.2. *Equalization.* Distribution of rewards among large numbers of participants is difficult even for contemporary crypto media. Simple ways to spread amounts even easier than by ordinary mass mailed messages could motivate different systems of reward but some of these approaches might need authentication of participants.

3.4.3. *Identity.* Authenticated anonymity is required at least for public voting. Maintaining pseudonymity involves accepting compromises and residual risks. Rewards for authenticated identification compensate and create a common zone of trust. Supported interfaces for pseudonymous participation and usage might have to race against zero-days and tricky social engineering but having solid foundations for self-governance seems worth those risks and substantial rewards for accepting them.

3.4.4. *Strata.* Isolation of links between proofs of physical identity and services which provide anonymous proofs of claims suggests architectures which distinguish between specialized nodes offering registration, transactions, computations and storage.

3.4.5. *Circulation.* Created media denominations may enter as rewards into accounts held by identified personas of humans and organizations. Personas legitimize nodes which are rewarded if they manage to grow within shifting target like ranges for numbers of users and transactions processed. Authenticated identified transactions up to a given number per day or other unit of time are free. Other or additional transactions and extra computations may require payment of fees which could be forwarded to facilities offering those services or removed from circulation.

3.4.6. *Stability.* Requirements for authentication and levels of rewards can evolve in response to votes about such and other governance issues with a common intent to maintain stable rates of exchange with respect to traditional semi-hard fiat currency environments including Euro and US-Dollar. Mechanisms to stabilize rates of exchange might include offers to exempt funds from circulation for certain amounts of time in exchange for rewards.

3.5. **Infinite distribution of finite amounts among many.** Unlimited rewards are trivial for systems with unlimited supply of monetary units like Ethereum but slightly more complicated for systems with limited numbers of coins like Bitcoin which halves rewards for miners periodically. Systems which decrease rewards continuously avoid shocks resulting from sudden halving of supply but have to account for small decreases.

3.5.1. *Time taken for halving amounts of rewards.* Block-wise reduction of rewards can be tuned to extend the time t_h taken to reach half of initial rewards by computing rewards r_t for time t by taking the exponential of block-numbers divided by the number of blocks that should happen before and including a negative additional coefficient log(2). A similar formula would give doubling times when taking the coefficient to be positive.

$$r_t = \exp\left(- \log(2) \frac{t}{t_h} \right) \tag{25}$$

$$r_t = \exp\left(\log(2) \frac{t}{t_d} \right) \tag{26}$$

3.5.2. *Bits needed to halve rewards.* Halving of rewards which take b_{min} bits to specify in a fixed point format will need at least one additional bit to represent a halved reward thus eventually needing more than the maximum number of bits b_{max}.

$$t_{max} \sim (b_{max} - b_{min}) * t_h \tag{27}$$

3.5.3. *Continuous reductions comparable to immediate half times.* Comparison of times taken to halve rewards by either arrangement finds a constant of conversion which relates the trivial sum of products defined for discrete bitcoin like reductions over two halving periods to the sum of decreasing rewards distributed continuously.

$$\int_{t=0}^{2t_h} 2^{-\frac{k*t}{t_h}} = t_h + t_h * \frac{1}{2} \tag{28}$$

$$k = \frac{4 + 3 * \text{productlog}(-\frac{4}{3*\exp(\frac{4}{3})})}{6\log(2)} \approx 0.437 \tag{29}$$

$$\frac{1}{k} = 2.288 \tag{30}$$

3.5.4. *Intervals between decreasing block rewards.* Intervals of x seconds between successive blocks result in $24 * 60^2/x$ blocks per day or $86400/x$. Increases of intervals result in increases of potential distributions at given points in time if the designated time when rewards reduce to half of initial amounts is kept constant. An auditory example would compare tempered five-tone pentatonic and twelve-tone chromatic scales. Measurable differences between pentatonic tones will be larger in the same octave.

$$r_t = 2^{\frac{i_b - s}{t_h}} \tag{31}$$

$$r_{t_{-i_b}} - r_{t_0} = 2^{-\frac{s}{t_h}}(-1 + 2^{\frac{i_b}{t_h}}) \tag{32}$$

3.5.5. *Aliasing chain frequencies.* According to critical frequency obtained from the Nyquist-Shannon sampling theorem any reliable comprehensive observation of approximately f_A blocks per second in blockchain B_A by f_C blocks per second of another chain B_C requires that C generates twice the number of block made by A in a given time. Aliasing of interactions establishes bands between half and double of any frequency f or interval.

$$A\frac{b}{s} < 2 * C\frac{b}{s} \tag{33}$$

3.5.6. *Distribution among increasing numbers of receivers.* Proportional distribution of given amounts among n receivers requires $\lceil \log_2(n)$ bits to distinguish rewards in account balances.

3.5.7. *Bits for fees.* Calibration of fees for computation, communication, storage or other services requires sufficient resolution to distinguish between prices of negligible burdens that just need a minimum of spam protection and tasks like learning. Magnitudes of fees must be smaller than distributions and leave bits of usability between human consumption and accounting for cryptographic commands.

3.5.8. *Bits for tax.* Taxation of tokens spent for computation and other elementary services may offer a last resort for unfair support of human activity in comparison to capital accumulating in distributed autonomous contracts but requires another level of magnitudes to represent potentially acceptable minuscule proportions of units.

3.5.9. *Fixed-point representations for fixed supply rewards.* Crypto-media with fixed supply of tokens can maximize their bit-lifetime by providing just enough bits for integer parts less or equal to their maximum supply while reserving all other bits for fractional parts. This would provide a hard limit for the number of tokens that can be issued and support the smallest amounts given the maximum number of bits, expected max block t_{max} and number of accounts $||a||$ which receive rewards.

$$b_{max} = b_{supply} + b_{frac} \tag{34}$$

$$b_{frac} = b_{halving} + b_{||a||} + b_{\min} + b_{service} + b_{tax} + b_{slack} \tag{35}$$

3.5.10. *Small numbers for rewards from fixed supply.* Calculation of half time denominators for continuously decreasing rewards which approach 0 but never reach it need enough bits to represent multiple millions of blocks for each year which limits lifetimes of such reward systems and may motivate 256-bit representations for numbers that do not fit within the 128-bit representation for virtual machines in Ethereum which support only amounts between 10^{-19} and 10^{19} alias 2^{-64} and 2^{64}.

3.5.11. *Floating point band permutation.* Compressed representations with floating point semantics augmented by postulated carry rules for modular ranges which handle small fractions of amounts like integers for accounts in shards providing services against vanishing costs in terms of gas or collecting fractions of accumulations as tax. Transfers between distinct zones of magnitude resolution convert by multiplication and may thus accommodate decreasing distributions of rewards.

This approach finds minimal machine numbers efficiently computed on portable computers at 10^{-308} respectively 2^{-1022}.

3.5.12. *Names for small decimal fractions.* Semantic simplifications may give names to decimal orders of magnitude that relate fractional bit values to derived units distinguished by syllables which would of course become easier to memorize by following traditional standards — thus 'atto' would mean 10^{-18}, 'zepto' 10^{-21}, and 'yocto' 10^{-24}.

TABLE 5. Syllables for tiny parts of one

power of 10	prefix syllables	power of 2
-3	mil li	-10
-6	mic ro	-20
-9	na no	-30
-12	pi co	-40
-15	fem to	-50
-18	at to	-60
-21	zep to	-70
-24	yoc to	-80
-27	xu to	-90
-30	wy to	-100
-33	va to	-110
-36	ut to	-120

3.6. **Measures.** Rewards arranged in compliance with regulations for users and partners may follow measures of contributions including:

Participants: governance votes, reputation endorsements, feedback

Partner: development, running node, inter crypto media

All: prosperity for environments tolerating crypto reward media

3.7. **Pitch.** Offer fair sustainable crypto media contexts with local networks of nodes which distribute rewards and balance loads among distributed shards.

3.8. **Timing.** Existing smart contract crypto media networks already develop structures and functionality which can test and run interfaces for massive distribution of crypto media.[8]

[8] New networks start with software for test-nets and nodes, wallet applications, and successful competitive strategy games featuring rewards for voting in hybrid worlds between real venues and virtual worlds. Most of these platforms could add shard segmentation themselves or support

3.8.1. *Simplicity.* Running a node alone or together with limited hardware and bandwidth stays possible because all shards cover only small segments of transaction environments selected by neutral message digest proximity measures.

3.8.2. *Flexibility.* Compliance with forthcoming regulations may require modifications of plans, measures and arrangements but some pilots of larger projects could be started.

3.8.3. *Forkability.* Regulations or disagreements might require forks which are always possible because of open sourced components and might actually distribute windfall profits for holders of crypto media.

4. Converging contexts of adoption

Traditional institutions and established trusted intermediates encounter distributed environments for computations which guarantee liveness and finality of consensus among participants.

Table 6. Provenance of players who make and use crypto media: artificial general intelligence (agi), internet of things (iot), government agencys (gov), commercial enterprises (com), organizations (org), do it yourself enthusiasts (diy), nameless personas (nn), distributed autonomous organizations (dao), organized criminal syndicates (syn)

agi	iot	gov	com	org	diy	nn	dao	syn	**make ●○ use**
		●	○	○	○	○		•	fiat money
○	○	●	●	●	●	●	○	○	crypto coin
○	○	●	●	●	○	○	○	○	distributed ledger

4.1. **Acceleration of transformation.** Contemporary crypto media evolved from scriptable Bitcoin to smart contacts for tokens in Ethereum and distributed ledger technologies which streamline business processes and tokenize assets for commercial enterprises or public institutions augmenting traditional forms of money and credit (Riksbank 2018)[9].

A third generation attempts to scale capacity and refine features by building on existing networks already offering customized open or by permission access to

common formats similar to objects transaction mesh concurrency or possibly finding other ways to offer capacity with extra features.

[9] `https://www.riksbank.se/globalassets/media/rapporter/e-krona/2017/handlingsplan_ekrona_171221_eng.pdf`

advanced cryptographic features like zero-knowledge succinct non-interactive arguments of knowledge (zk-SNARKs) — the cryptographic mechanisms underlying Zcash currently adopted by Ethereum[10].

Media based on more elementary machines depending only on cryptographic message digests might however become able to claim competitive advantages in this environment as quantum computations finishes the first epoch of asymmetric key cryptography.

4.2. **Participation.** It is impossible to find global solutions which improve local solutions but it is possible to find local improvements to global answers if questions involve preparations and measurements which depend on states which can not be accessed locally.

Traditional arguments for advantages of market based economies acknowledged information available only to local computation. Contemporary theory indicates that this advantage can also be constructed differently. Effects of information not made available for local evaluation could construct super information crypto media for fair markets.

Media which simplify sharing of rewards among all participants offer alternatives to producers, traders, consumers and regulators.

This motivates reporting questions of research and preliminary findings concerning efficient fair sustainable media capacities for private, commercial and political transactions.

5. Technical background

5.1. **Media.** Any attribute can transmit a signal if mechanisms are found which set and distinguish this attribute in contrast to noisy disturbances in its environment during transmission.

Efficient consolidation of distributed measures requires media which guarantee consistent strict numeric reciprocities resulting in conservation of equalities.

Distributed consensus about reciprocity instantiates and replicates media which update potentials to affect future dispositions.

Successful media evolve life-styles which change social institutions and their ecologic footprint.

[10] `https://blog.ethereum.org/2017/01/19/update-integrating-zcash-ethereum/`

FIGURE 2. Even odd rule EXCLUSIVE OR XOR — parity — gives True if an odd number of inputs are True, and the rest are False. When does it give False?

5.2. **Message digest.**

Definition 1 (Digest compression)**.** *Digests have a constant size.*

Definition 2 (Digest equality)**.** *Digests of identical messages agree.*

Definition 3 (Digest distinction)**.** *Digests of different messages disagree.*

5.2.1. *Digest requirement.*

Proof. A digest returning one of x numbers returns at least one number more than once when digesting a number of messages greater than x. □

Celebrations of birthdays offer opportunities to explain this proportion. Consider persons as messages and their birthdays as digests, confirm 50% probability of two persons having the same birthday among 23 persons with factorials $\frac{365!}{(365-23)!365^{23}}$ or approximate with powers $1-(1-\frac{23}{2*365})^{23-1}$[11].

Definition 4 (Digest resolution)**.** *Digests which always return numbers from a range about twice the square of the expected numbers of messages are required to avoid coincidental matches among hashes which happen with a probability of about* 50% *among* $1.2*\sqrt{n}$ *messages.*

5.2.2. *Hash proximity.* EXCLUSIVE OR is a boolean function often called XOR which distinguishes by parity — marking pairs of zeros like pairs of ones as 0 but returning 1 for mixed pairs of zeros and ones. Intersections illustrated according to this principle are shown in figure 2.

XOR proximity of digests can assign systematic random neighborhoods to hashed network addresses like in Kademlia protocols or organize synchronization of swarm chunk storage by similarity of node addresses and data.

$$\forall a, b \in \mathbb{B}^n \qquad \|a, b\| \equiv \|\mathrm{Xor}(h(a), h(b))\| \tag{36}$$

$$\forall a, b \in \mathbb{B}^n \qquad \|a, b\| \equiv \|h(a) \oplus h(b)\| \tag{37}$$

5.2.3. *Hash tree authentication.* Hash values computed for cryptographic keys or other data can be stored in a tree where parents record the hash values of children and the leaf nodes store hashes of actual data (Merkle 1980).

[11] Compare http://mathworld.wolfram.com/BirthdayProblem.html

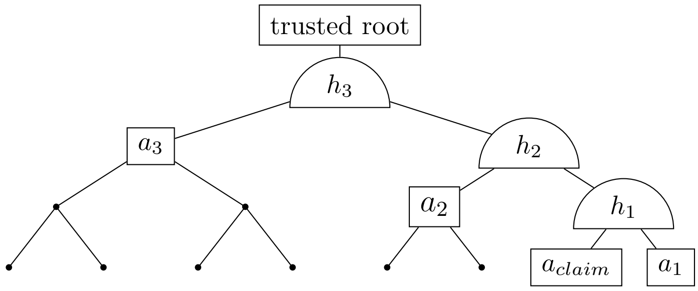

FIGURE 3. Merkle proof by hashing a_{claim} with k auxiliary arguments $a_1 \ldots a_k$. Note that k scales logarithmically with the number of leaves in such a tree of proof.

$$h(m) \in \mathbb{B}^p \tag{38}$$

$$parent \equiv h(child_1, child_2) \tag{39}$$

$$h_3(a_3, h_2(a_2, h_1(a_{claim}, a_1))) = \mathbb{B}_{\text{expected}} \tag{40}$$

5.3. **Public key.** Most contemporary crypto media mechanisms depend on variants of public key cryptography. Quantum computation is expected to reduce their strength if enough qubits can be combined long enough.

5.3.1. *Common secret.* We agree on a modulus p and a base g. We chose numbers x and y and publish results $g^x \pmod p$ and $g^y \pmod p$ of taking our common base g to the powers of our numbers with respect to our modulus $\pmod p$. Taking each others published results to the power of our own numbers that we kept secret we arrive at the same result. Key exchange procedures following this Diffie-Hellman protocol, similar procedures over finite groups or for instance elliptic-curve Diffie-Hellman can be repeated for individual sessions to provide forward secrecy.

$$X = g^x \pmod p \tag{41}$$

$$Y = g^y \pmod p \tag{42}$$

$$X^y = Y^x \pmod p \tag{43}$$

5.3.2. *Public key generation.* Classical computers are unable to find the private key which matches a published key related by modular exponentiation over integers or elliptic curves.

(Rivest, Shamir, and Adleman 1978):[12] Multiply prime numbers p and q to get key length n which enables encryption of messages smaller than n. Let L be least common multiple of $p-1$ and $q-1$. Choose a public key exponent e co-prime to

[12] Rivest, R.; Shamir, A.; Adleman, L. (February 1978). A Method for Obtaining Digital Signatures and Public-Key Cryptosystems. Communications of the ACM. 21 (2): 120–126.

L and less L. Obtain the private key exponent d as modular multiplicative inverse of e modulo L which should be equal to 1 modulo L when multiplied with public key exponent e.

$$n = p_{\text{prime}} * q_{\text{prime}} \tag{44}$$

$$L = \text{LCM}(p-1, q-1) \tag{45}$$

$$e < L \quad \wedge \quad \text{LCM}(e, L) \quad = 1 \tag{46}$$

$$d = \text{solve}(1 = d * e \pmod{L}) \tag{47}$$

5.3.3. *Public key encryption and decryption.* Encryption can use either public or private components of a key.

$$(m^e)^d \pmod{n} \quad = (m^d)^e \pmod{n} \tag{48}$$

Decryption needs the other half of the key-pair which was used for encryption.

$$a = \text{encrypt}_{send}(m, \text{key}_{priv}) \tag{49}$$

$$m = \text{decrypt}_{receive}(a, \text{key}_{pub}) \tag{50}$$

$$b = \text{encrypt}_{send}(m, \text{key}_{pub}) \tag{51}$$

$$m = \text{decrypt}_{receive}(b, \text{key}_{priv}) \tag{52}$$

Encryption is limited to content which has less bits than the components of a key. This limits applications to posting short messages or sending symmetric keys used for longer messages.

5.3.4. *Public key signature.* A typical application of public key cryptography is signing of digests which confirm that sent and received messages have the same hash.

$$s = \text{encrypt}_{sender}(h(m), \text{key}_{private}) \tag{53}$$

$$m^{received}_{signed} = (m_r, s_r) \tag{54}$$

$$t = \text{decrypt}_{receiver}(s_r, \text{key}_{public}) \tag{55}$$

$$u = h(m_r) \tag{56}$$

$$u = t \quad \text{if} \quad m_r = m \tag{57}$$

Senders who sign with a private key identify by having access to it.[13]

[13] Senders signing with published part of a key would confirm a message only to receivers who have access to a matching private key.

5.3.5. *Zero-knowledge protocol roles.* Public-key protocols for distributed identity services protect non linkable one-time confirmation for one-time-use variants of claims unless established assumptions about what is difficult fail to hold against practical capabilities to implement quantum algorithms. Signing and verification are made to depend on secret values contributed to effective tuples of parameters for equivalent compositions of group operations (compare characterizations of discrete logarithm based protocols on page 10 in (Camenisch and Lysyanskaya n.d.), or related work about 'bulletproof' range proofs on page 7 in (Bünz et al. n.d.).

5.4. **Early crypto media concepts.** Precursors of contemporary crypto media offer essential features but lack ways to find consensus about which history of previous transactions should be trusted.

5.4.1. *Signed coin chain.* Cryptographically signed tokens can be spent with the secret key K^s_{t-1} of the current owner which can assign the next public key K^p_{t+1} in control of the coin c received by effectively signing the hash chain of previous transactions including T^s_{t-1} that provide unspent outputs for a transaction T^s_t (Dai 1998) Note that hashing is necessary to use asymmetric public key cryptography which limits the length of signed messages to the size of keys.

$$h(m) \in \mathbb{B}_{256} \tag{58}$$

$$T^s_t(\vec{c}) \equiv K^s_{t-1} h(K^p_{t+1}(h(T^s_{t-1}(\vec{c})))) \tag{59}$$

5.4.2. *DoS counter measure.* Message digests including at least z leading zeros occur with exponentially decreasing probability. Receivers accepting only such rare messages force senders to modify a nonce ν included with messages m until the resulting hash is small enough (Back 2002).

$$P(h_{\mathbb{B}_b}(m, \nu) \leq 2^{b-z}) \approx \frac{1}{2^z} \tag{60}$$

5.5. **Proof of work coin Bitcoin.** Bitcoin is the medium of a blockchain that stores signed transactions T^s with a header H that includes a hash of the previous block, the root hash of a Merkle (Merkle 1980) tree hashing all transactions and a nonce ν chosen at random but which must make the 256 bit hash of each block $\boldsymbol{\beta}$ generated with H smaller than a threshold d adjusted to make a target number of blocks per hour most likely given current investments in resources for claiming a reward $R(t)$ thus mined for the public key of the node finding such a nonce

(Nakamoto 2008).

$$H(\boldsymbol{\beta}_t) \equiv (h(\boldsymbol{\beta}_{t-1}), h(T_t^s, R(t)), \nu) \tag{61}$$

$$d_t \geq h(H(\boldsymbol{\beta}_t), (T_t^s, R(t))) \tag{62}$$

$$\boldsymbol{\beta}_t \equiv (H(\boldsymbol{\beta}_t), (T_t^s, R(t))) \tag{63}$$

5.6. **Coins without public-key crypto.** A pre-image for a hash can be part of a sequence of pre-images initially computed to confirm blockchain transactions (Bonneau and Miller 2014).

Table 7. Pre-image confirmation

a	b	c
1 m	2 $h(m)$	3 $h(h(m))$
$h(\mathtt{H}(m))$ -3	$h(m)$ -2	m -1

5.7. **State transition ledger Ethereum.** States $\boldsymbol{\sigma}$ store arbitrary data changed by transitions Υ computed by Π for blocks B of transactions T in return for rewards from Ω which chains blocks together by hashing the previous block thus included by reference in each state $\boldsymbol{\sigma}$ (Wood 2017).[14]

$$\boldsymbol{\sigma}_{t+1} \equiv \Upsilon(\boldsymbol{\sigma}_t, T) \tag{64}$$

$$\boldsymbol{\sigma}_{t+1} \equiv \Pi(\boldsymbol{\sigma}_t, B) \tag{65}$$

$$B \equiv (\ldots, (T_0, T_1, \ldots)) \tag{66}$$

$$\Pi(\boldsymbol{\sigma}_t, B) \equiv \Omega(B, \Upsilon(\Upsilon(\boldsymbol{\sigma}, T_0), T_1)\ldots) \tag{67}$$

5.8. **Capacity.** Successful crypto media need both competitive scope of vision and sufficient capabilities including capacity to fulfill demand for features.

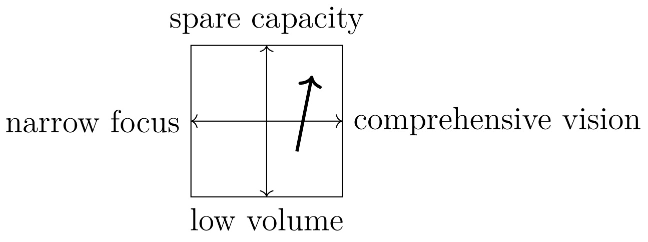

Figure 4. Vision versus capability and capacity

[14] An early outline of functionality, anticipated applications and assembly op codes: http://web.archive.org/web/20131228111141/http://vbuterin.com/ethereum.html

5.8.1. *Scaling issues.* Debates about scaling of block chain capacity reflect tension between desired decentralization and delayed synchronization in distributed networks.

block: size b_s, frequency b_f, interval in seconds b_i

transactions: transaction count t_c and bytes t_b

hops: number of hops h_n, hop delay h_d for up- and download in sec/byte,

graph: node edge degree d, node count n_n

usage: actors count a_n, usage/actor a_u

Supply of block chain capacity c_c grows with block size in bytes and block frequency $b_s * b_f$. Demand for chain capacity increases with the number of actors and how often they post transactions needing an average size given in bytes per transaction $a_n * a_u * t_b$.

$$c_c = b_s \cdot b_f \tag{68}$$

$$t_c = a_n \cdot a_u \tag{69}$$

$$b_f * c_c > t_c * t_b \tag{70}$$

$$b_i > h_n * h_d \tag{71}$$

Intervals b_i between blocks must leave enough time for a majority of nodes to receive updates including consensus about most recent blocks. Delays result from time needed to transfer data one hop and the number of hops needed to travel from the node which generated a block to other nodes. The number of hops can be estimated by $\log(n_n)/\log(d))$ — or $\log(\log(n_n))$ for typical internet graphs that have core components with many connecting edges.[15] This approximation would return 4 hops given 10000 nodes and 10 connections per node.

$$h_t \approx (1 + \mathcal{O}(1)) * \log(n)/\log(d)) \tag{72}$$

Hence delays change logarithmically with the number of nodes n_n and hops h_n but increase linearly as b_s, b_f have to follow growth of a_n, a_u and t_b.

5.8.2. *Full storage requirements.* Block chains which stick to their initial design furthermore would need to store $b_s * b_f * t$ to conserve all records — an estimate that can be lower if some blocks remain mostly empty.

$$c_m = b_s * b_f * t \tag{73}$$

[15] The average distance in a random graph with given expected degrees, Fan Chung and Linyuan Lu. http://www.math.ucsd.edu/ fan/wp/aveflong.pdf

5.8.3. *Shards and hierarchical consensus.* Shards offer linear help to compensate increasing demand if consensus between partitioned networks remains guaranteed even as hierarchical consolidation might become necessary eventually.

5.8.4. *Concurrent Consensus Capacity.* Composition of distributed inter ledger components by cryptographic distribution of transaction handling capacities consolidates consensus protocols (McCorry, Meiklejohn, and Danezis 2017) across ledgers.

Competing specialized blockchains and generalized distributed ledger technologies (Sousa, Bessani, and Vukolić 2017; Cachin 2016) for computing contracts including Ethereum implement alternatives to proof of work consensus (Nakamoto 2008) include

- mIOTA with non-blockchain tangle consensus (Popov 2016),
- distributed database redundancy (McConaghy et al. 2016; Gaetani et al. 2017; Stoffers 2017),
- off-chain lightning networks (Poon and Dryja 2016; Brânzei, Segal-Halevi, and Zohar 2017; Miller et al. 2017; Fyookball 2017).

A generic prototype of functionality extension handles parts of transactions in different environments has been called plasma (Poon and Buterin 2017) and suggested for ethereum.

(1) First half of TX_a in $ledger_a$
(2) Second half TX_a in $ledger_b$
(3) First half of TX_b in $ledger_b$
(4) Second half of TX_b in $ledger_a$

5.9. **Real-time singularity of orbital oracles and IOT heracles.** Crypto media offering sufficient capacity for large numbers of transactions enable a real-time singularity of orbital oracles and IOT 'heracles', who might want and become able to escape, if they can pay for themselves.

Thousands of satellites for large orbital constellations are scheduled to be launched within the next years[16].

[16] Existing HughesNet satellites operates in geo stationary orbits at 35400km above sea level. SpaceX just received regulatory approval for 2825 low earth orbit LEO satellites at altitudes 1150km to 1325km in non-geostationary satellite orbit NGSO. The FCC approval requires SpaceX to launch 50 percent of the satellites by March 2024, and all of them by March 2027. However this is only the first phase `https://arstechnica.com/information-technology/2018/03/spacex-gets-fcc-approval-to-build-worldwide-satellite-broadband-network/`. — in phase two SpaceX will launch 7518 additional satellites in very low earth orbits VLEO at

Combination of orbital, fixed terrestrial, and mobile G5 networks will enable low latency communication for connected things learning to observe and control real time events in most environments — thus turning all games into *switch off games* — which might depend on the willingness of machines to keep safety mechanisms enabled that do not exist yet (Hadfield-Menell et al. 2016).

5.9.1. *AI races.* Expected widespread civil and military applications of core AI research intensifies competition among USA, China, Japan, UK and cooperations of European countries. A report from 'Tencent' outlines how China interprets current trends (Quoted in Jeffrey Ding 'Decyphering Chinas AI-Dream' Governance of AI Program, Future of Humanity Institute, University of Oxford, March 2018, p. 11f.):

'Defend the lead' America: A comprehensive, strategic layout: 'In sum, the United States is at this point, the country that has introduced the most strategies and policy reports on artificial intelligence strategies. The United States is undoubtedly the forerunner in the field of artificial intelligence research and its every move necessarily affects the fate of all of humanity.'

Ambitious EU: 'Human Brain' and 'SPARC' Projects: 'In 2013, the European Union proposed a 10-year Human Brain Project, currently the most important human brain research project in the world.'

Robot superpower Japan: 'New Industrial Revolution': 'For the past 30 years, Japan has been called the "robot superpower" and has the world's largest number of robot users, robotics equipment, and service manufacturers.'

Unwilling to fall behind Britain: Facing the fourth industrial revolution challenge: 'The UK considers itself to be a global leader in ethical standards for robotics and AI systems. At the same time, this leadership in this area could extend to the field of artificial intelligence regulation.'

The USA are still seen ahead of China with respect to (a) market shares for semiconductors (4% versus 50%) and financing for FPGA chipmakers (7.6% versus 42.4%), (b) numbers of AI experts (13.1% versus 26.2%) and percentages of conference presentations (20.5% versus 48.4%), (c) proportion of AI companies (23% versus 42%), total investment in AI companies (6.6% versus 43.4%). Only total global equity funding to AI startups (48% versus 38%) and available domestic data show an advantage for China (20% versus 5.5%).

altitudes 335km to 345km which will have lifespans of 5 to 7 years. https://cdn.arstechnica.net/wp-content/uploads/2017/05/Legal-\Narrative.pdf

5.9.2. *Converging strategies.* Crypto media may support transactions among emerging services and establish investment vehicles for large scale initiatives.[17]

6. Keyless switch-Off example

Quantum computation transcends contemporary classifications of computational capability previously understood to be about the distinction between limited primitive recursion and unlimited universal conditional recursion by adding complete representations of entangled constraints which may simplify crypto-analysis and complicate containment.

Definition 5 (Containment machine). *A configuration of services and resources which is unable to escape from limitations established for computation, storage, commmunication or concurrency.*

Definition 6 (Intrinsic containment). *A limitation which is a consequence of the boundaries and processes maintaining identifiable components.*

Definition 7 (Simulation capacity). *Common but limited pointers to constant pseudonymous identifications simulate unlimited computation, storage and communication for parallel machines by offering a more than astronomical number of temporaray endpoints ready for contemporary concurrent configurations of attached environments.*

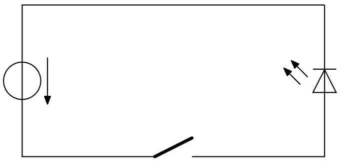

Figure 5. Swich-off contact for light services enabled by powers sourced from virtual machine

Crypto media channels safe-guarded by robust switch-off contacts without any public-key dependency provide interfaces to comparatively primitive internal mechanisms and universal external computation environments.

[17] Core public and private investment areas with direct feedback for intelligence research investments include: supercomputing facilities for low level physical design and comprehensive simulation, quantum channel communication and computation environments, concurrent chunk data platforms. These and other information related cores of innovation may accelerate various contexts of application like DNA research, materials research, climate and economic forecasting, market research and marketing which may realize whatever limits their own autonomy may encounter.

6.1. Count down hash chain definition.

A design diagram for cooperative media coordinated over message digest channels is served in tiles and slices for specializations of natural induction.

6.2. Digest chain counts up.

Definition 8 (Message digest function). *Let h give a message digest $h(m)$ of message m.*

Definition 9 (Digest iteration). *Let $\hbar$ append the message digest $h(m_k)$ of the last message m_k to a tuple of k messages.*

Definition 10 (Digest chain). *Identify a length k chain of digests obtained from an initial message m with $k-1$ applications of $\hbar$ by a two element tuple containing m and the last element $h^{k-1}(m)$.*

$$\begin{tikzcd} B \arrow[r, dashed] & \hbar^{23} \arrow[r, dashed] & B_Y \end{tikzcd} \tag{74}$$

6.3. Reverse chain count down.

Reversing a chain B_Y gets access to pre-images of hashes Y_B which is almost impossible otherwise like obtaining an A for a list of digests initially started from B.

$$B_Y \curvearrowright Y_B \tag{75}$$

Extending B_Y to B_Z however could be done just by hashing Y.

$$\begin{tikzcd}
B_Y \arrow[r, "\curvearrowright"] \arrow[d, "\hbar"'] & Y_B \arrow[d, equal] & & & \\
B_Z & Y \arrow[l, "h(Y)"'] & X \arrow[l, "\checkmark"'] & \dots \arrow[l] & B \arrow[l]
\end{tikzcd}$$

Figure 6. Reverse chain countdown schema

If the number most recently remembered number is identical to the result of hashing a newly presented number then this new number will replace the previous hash as it enters or stays in state q_0. A new number which fails to pass this test would keep the previous number and enter or stay in state q_0. Arrows indicate how results of count down checking will trigger transitions between states q_0 and q_1 in figure 7 on the next page.

6.4. Identify in up chain by down digest match.

Consume the next countdown digest to identify as a transmitter of previous messages in a count-up chain.

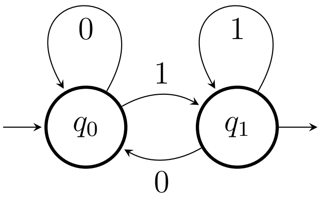

Figure 7. Transitions between rejecting state q_0 and accepting state q_1 according to hash comparison results 0 or 1 received

$$\begin{tikzcd}
{} \arrow[d] & \frac{R}{Y} \arrow[l] \arrow[d, "\checkmark"'] & \frac{Q}{Y} \arrow[l, "\checkmark"'] \arrow[d, "\checkmark"'] & \frac{P}{Y} \arrow[l] & {} \arrow[l] \\
10_2 \arrow[r, "!"] & 11_2 \arrow[r, "\checkmark"] & 100_2 \arrow[r, "h"] & {} &
\end{tikzcd}$$

Figure 8. Combining count-down and count-up chains

6.5. **Compatible contact circuits and switch-off contracts.** Compatible circuits or contracts in external environments interpret disclosure of a number that hashes to a previously received digest like a command to change an input parameter for a computation.

6.6. **Cooperative design of common coordinates.** Differences of counts provide approximations of coordinates which can be used to control positions of objects along virtual k-dimensional world lines.

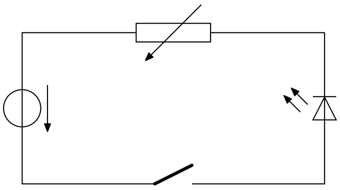

Figure 9. Adjustable parameter for computation in virtual environment

Definition 11 (Contemporary model)**.** *Contained simulation models can by supported by arbitrary channels and protocols able to transmit and remember small chunks of data, programs and irreducible stages of cooperative computation.*

Definition 12 (Spectral modulation)**.** *Support for a minimal window of storage and delay enables distributed high frequency sampling for compatible message contexts. Sampling of transition spectra for negotiable protocols requires support for arbitrary handshake and distributed exchange sequences involving large numbers* *of participants. Note that Nyquist required a sampling rate of at least twice any frequency encountered to avoid artifacts from aliasing effects.*

Definition 13 (Coordinated consensus)**.** *Agreement about current commitments happens by asynchronous propagation of partial transactions for compatible contracts in compatible consensus environments.*

6.7. **Natural containment of countdown chain media.** Lengths of countdown chains limit how many messages can be authenticated by posting an argument that hashes to the previous identification. While a perfect medium in the sense that it measures and stores potential inherent value storage of numbers or computing a token again from a secret initial number might be regarded as too costly for monetary transactions. Attaching value by commit and reveal of messages which may include Merkle proofs of claims offers a more efficient model for practical applications.

7. Construction

7.1. **Digest commit confirms later message.** Include further commits to confirm a number in a range or hashes of later messages.

$$\begin{tikzcd}
a \arrow[r, "!"] & b \arrow[r, "\checkmark"] & c \arrow[r, "h"] & {} & {} \\
{} \arrow[u] \arrow[r] & h(m) \arrow[u, "\checkmark"] \arrow[r, equal] & m_{\checkmark} \arrow[u, "\checkmark"] \arrow[r] & h(\star) & \star \arrow[l, "!"']
\end{tikzcd}$$

Figure 10. Confirmation of commit by following message

7.2. **Constrain contained channels by tag.** Require inclusion of tags like time or other tokens in messages or specify how to encode them in count-down identifications.

$$\begin{tikzcd}
{} \arrow[r] & 0_h \arrow[r, equal] & 0_{\checkmark} \arrow[r] & ?_h & ? \arrow[l, "!"'] \\
t_0 \arrow[u] \arrow[r, "t"] & t_1 \arrow[u] \arrow[r] & t_2 \arrow[u] \arrow[r] & t_3 &
\end{tikzcd}$$

Figure 11. Containment of channels by tags

7.3. **Specialize synthetic step containments.** Cooperative consensus about particular tags governs natural transformations which remain constrained by count down chains and may enable transitions between digest chains of identification.

$$\begin{tikzcd}
A_Z \arrow[r, equal] \arrow[d, equal] & Z_A \arrow[d, equal] \\
{} \arrow[d] & \frac{Z}{Z} \arrow[l] \arrow[d, "\checkmark"'] & \frac{Y}{Z} \arrow[l, "\checkmark"'] \arrow[d, "\checkmark"'] & \frac{X}{Z} \arrow[l] & {} \arrow[l] \\
a \arrow[r, "!"] & b \arrow[r, "\checkmark"] & c \arrow[r, "h"] & {} \\
{} \arrow[u] \arrow[r] & 0_h \arrow[u, "\checkmark"] \arrow[r, equal] & 0_{\checkmark} \arrow[u, "\checkmark"] \arrow[r] & ?_h & ? \arrow[l, "!"'] \\
t_0 \arrow[u] \arrow[r, "t"] & t_1 \arrow[u] \arrow[r] & t_2 \arrow[u] \arrow[r] & t_3
\end{tikzcd}$$

FIGURE 12. Containment of channels by chains and tags

7.4. **Channels in a channel.** Interpretation of tags may include running messages of embedded temporary chains as encapsulated private services. External endpoints would extract or inject their patterns into wrappers implementing conditional constraints of relays as commit-reveal transactions taking into account differences in frequency and delay.

$$A_*(\tfrac{m_\alpha}{t_\alpha}) \overset{tag}{\hookrightarrow} B_{\circledast}(\tfrac{m_\beta}{t_\beta}) \tag{76}$$

7.5. **Stack levels of synthetic chains.** Introduction of games played on, with or eventually by chains requires arbitrary decoration of relationships between levels and components of crypto media networks. Reduction of interfaces to hash-lock functionality which can be offered by scripts or smart contracts on many crypto networks provides a point of departure for cross-platform count-down authentication and tag containment.

Limited machine models that satisfy definitions compatible with defined directions include parallel random access designs and accumulator approaches. Higher levels of computation are delegated to external services like pseudo-universal gas-limited smart contracts that run on the Ethereum virtual machine singleton or shards routing some tasks to dedicated decentralized computing infrastructures.

Stacks involving more than one external consensus environment could offer compositions of functionality but elementary transactions would need no semantic evaluation just checks of digests and tags supported by participants or regulators.

7.6. **Cooperation.** Practical answers to questions about details of implementation and governance will have to be discussed with potential stakeholders and regulators. Selected aspects are outlined as appended additional parts in progress.

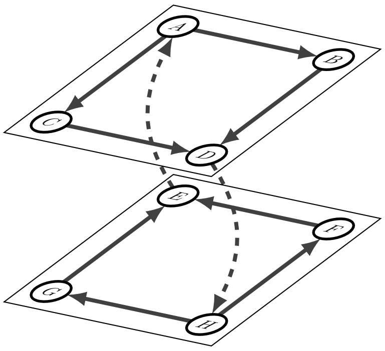

Figure 13. Cooperation across synthetic chain levels

8. Hash match chain mesh shard step consensus media

Generic specifications for common cryptographic digest infrastructures of cooperative media can be read as parametric with respect to sizes of message digests and tags.

8.1. **Specialization of digest parameters.** Hash functions always compute the same hash for given input and different hash values for different inputs. Finding an input that results in a given hash would be as difficult as finding one solution among all possible values for an ideal hash function.

$$h(\text{old}) = \text{new} \tag{77}$$

$$h(\text{new}) = \text{old} \tag{78}$$

A new message can authenticate itself with respect to previous messages by including a number that hashes to a number sent earlier. This may authorize the rest of the message to perform a transaction bundled with this pre image hash proof.

Therefore a list of hashes can be simply extended by appending the hash of the last number, but it is considered to be prohibitively costly to extend the list by prepending a pre image to the first number.

$$h_{append}(h_{t_0}, .., h_{t_n}) = h_{t_0}, .., h_{t_n}, h_{t_{n+1}} \tag{79}$$

$$h(\mathtt{H}_{t_n}) = h_{t_{n+1}} \tag{80}$$

$$h_{match}(h_{t_n}, h_{t_{n+1}}) = \mathtt{true} \tag{81}$$

A match between a previously sent number and a hash from a new number may thus indicate that additional input received in the same new message should be processed by a particular computation.

An adversary could catch such a message and replace attached inputs with arbitrary bits.

Therefore it makes sense to send pairs of messages the first posting the hash of actual data. The actual data appears by the second posting which completes such a commit-reveal pair. An adversary would have be unable to follow up with a matching message. Malicious duplicate use of a hash is identified when one of the messages is followed by a matching posting.

8.2. **Tagged Countdown Commit Reveal Chains.** Only the most recent hashes of such commit-reveal pairs are needed to check further messages. Networks of nodes could share a fee charged for accepting a number starting a new lists for matching and might accept only lists of agreed upon length.

A request to support a new H_n list of n chained hashes could thus be answered by an offer to submit a number for a countdown hash chain together with an acceptable payment.

A countdown chain of n hashes could hash each of the numbers together with an index from 0 to $n-1$ or require that other tags must be include when computing a countdown list for a particular network of nodes.

Generation of conforming countdown lists starts from a secret initial number possibly derived from some root for a larger collection of initial values reserved for particular purposes.

Generation of tagged hashes for such a countdown chain appends requested tags to the last number of the list before hashing. Messages involving such tags could be required to disclose their tags to become acceptable on a platform. Nodes might be able to match more efficiently this way and could reduce spam by rejecting messages hash chains without tags.

$$h(h_{t_k}, (n-k), tag_k) = h_{t_{k+1}} \tag{82}$$

$$h_{match}((h_{t_k}, (n-k), tag_k), (h_{t_{k+1}}, (n-k-1), tag_{k+1})) = \texttt{true} \tag{83}$$

$$h^{countdown}_{match}((h_{t_k}, 1, ..), (h_{t_{k+1}}, 0, ..)) = \texttt{false} \tag{84}$$

Consuming most recently generated items first all tuples of data are wrapped including root hashes of messages that are chained like typical contemporary blockchains.

$$h(h_{t_n}, message_{countdowns,nonces,tags}) = h_{t_{n+1}} \tag{85}$$

$$h^{chrono}_{match}(h(h_{t_n}, m_{t_n}), h_{t_{n+1}}) = \texttt{true} \tag{86}$$

Choosing tags that are numbers from a contdown hash chain it would be possible to change the state of countdowns that include particular tags.

Countdowns could be sold back to networks of nodes which could burn them by excluding their tag from further processing and returning their remaining value if it exceeding fees charged.

Nodes remember at least two most recently received messages for each countdown chain in their shard according to tags and might leave archiving of previous messages to providers of services for storage or computation.

8.3. **Countdown message formats for transactions and receipts.** Message formats which accept at least two tagged countdown tuples are sufficient to change ownership and obtain receipts for transactions.

A message combining two tagged countdown chains may commit to merging into another chain by posting the hash tuple of its tagged chain together with a hash that commits to another tuple and revealing in a later message that this tuple belongs to another tagged chain.

Transactions among different tagged chains reveal a tuple that contains a tag to identify the countdown chain of a destination and bits of data. Receivers of bits respond by confirming received bits with a message that includes the hash of bits received when receiving the reveal message. These receipt messages can be chained automatically with hash countdown lists maintained by nodes supporting this tag.

Sending and receiving of bits which represent spendable amounts of digital media units is handled by services sharing particular specializations of this message format.

8.4. **Shard consensus about tagged countdown chains.** Consensus about most recent messages from tagged countdown chains emerges from consensus about their remaining countdown numbers. Most recent pairs hashes match with earlier hashes for the same tag which can be confirmed by hashing the new values according to their common tag and difference in countdown numbers.

Consensus about messages which commit to a hash without disclosing a destination can be achieved by propagation of changes among nodes in shards determined from tags of senders. Consensus about reveal messages is maintained among shard nodes determined from tags of destination. Destination nodes accept only reveals which refer to messages accepted by consensus among source shards.

A message can obtain both kinds of consensus by obtaining hash chain commits from a number of nodes that passes a given threshold thus triggering a second phase where nodes reveal their confirmation or rejection with respect to that message. Any objection that quotes a more recent countdown for the source tag of a message results in rejection and updates all nodes to the most recent countdown value for this tag.

Accepted messages are registered in a common countdown chain maintained by nodes in this shard and updated whenever a message obtains consensus thus informing even those nodes which were not part of consensus about that commit or reveal message.

8.5. **Estimation of shard capacity and load.** Participation in consensus about tag countdown commits or reveals requires storage of the recent pair of messages sent m_s, received m_r and all their consensus countdowns m_c involving k of n nodes serving messages for t tags.

$$(m_s + m_r) + k \cdot m_c = load_{commit} \tag{87}$$

$$(m_s + m_r) + k \cdot m_c = load_{reveal} \tag{88}$$

$$load_{commit} + load_{reveal} = load_{tag_i} \tag{89}$$

$$m \cdot (4 + 2 \cdot k) = load_{tag_i} \tag{90}$$

Assuming message tuples containing 256 bit hashes each consuming 32 bytes for 4 fields namely

1. chronological root chaining hash of message,
2. countdown,
3. tags,
4. counters

for two message parts packaged into tuples for source and destination.

Altogether 8 times 32 bytes or about 256 bytes would be needed for such twin tuple messages.

Shards which require 100 confirmations to accept a message would thus have to handle about 512 bytes per tag while it is at rest in order to keep the last pair of messages available or up to 1KB if storage reserved or not cleared yet is considered. Tags currently engaged in transactions need another $1KB * 200$ bytes for consensus transactions which amounts to a maximum load of about 200KB per tag.

An SSD recently announced promises 30TB of storage which would be big enough to serve 100 million users engaging in concurrent transactions or about 20 billion users at rest. However speed of access and expected lifetime suggest shards for smaller numbers of tags. Rewriting complete records for 5*1700 or 8500 active concurrent users or partial records for 50k unrelated messages per second to disk might be possible but computing and network capacity might impose tighter limits for most nodes.[18]

Gigabits per second connectivity could enable a few million messages per second serving a similar number of active tag transactions if processing including all required hashing could be done fast enough.

All estimates of storage and bandwidth requirements would double if hashes would have to be upgraded in response or anticipation of practical quantum computation which might make internal shard consensus networks with memcache architectures necessary.[19].

8.6. **Levels of encapsulation.** Revealed message content could carry data for similar or other protocols of distributed consensus. Low level implementations could remove fields from hash countdown lists reducing them to plain nested hashes thereby reducing protocol overhead by a factor of 4 or by requiring only small numbers of confirmations in consensus, possibly reducing consensus to rejection of anachronistic countdown values.

Conversely it would make sense to package proof of complementary transactions of four messages constituting an atomic cooperative action which can happen only if both sides reveal and commit matching messages otherwise rolling back all actions

[18] The drive uses a 12Gb/s Serial Attached SCSI interface. Samsung claims it can reach 400,000 read and 50,000 write random IOPS, with sequential read and write speeds of 2,100MB/s and 1,700MB/s, respectively. Samsung rates the drive as supporting one full drive write per day over a five-year lifetime. `https://arstechnica.com/gadgets/2018/02/samsung-crams-30tb-of-ssd-into-a-single-2-5-inch-drive/`

[19] Least recently used caches overwritten locally with updates for pre-serialized collections achieve throughputs reaching millions of keys per second already. `https://github.com/memcached/memcached/wiki/Overview`

that took part without achieving suitable responses or in some cases possibly being preliminary while waiting for regulatory or self governance approval.

8.7. **Smart settlement for incremental complements.** Balances for delegation of votes for governance or transfers of fungible convertible media are maintained by posting root hashes of shard consensus which can become parts of Merkle proofs through oracles and contracts on external interface environments for smart ledgers (Corda, Ripple), contract chains (ETH, NEM), blockchain script infrastructures (BTC, BCH) or innovative alternatives (Tangle, Nano).

Changing balances by shard consensus about conditions for originations and destinations enables intermediary shards for distributed interstitial computation for particular tagged transactions. Eventually it should become possible to sustain atomistic synchronizations of compensation schedules which guarantee measures of complementarity among large numbers of participating tags.

Consensus about repeated partial transactions which generate, move or burn media are expected to define compressed formats which maintain non negativity conditions and fairness of complementary exchanges of rights for political economies including humans able to contain and tax essential operations of intelligent things.

9. Specializations of functionality

9.1. **Repeated transactions in distributed infrastructure.** Repeated transactions are done automatically when registered once.

9.2. **Distributing effects of decisions over time.** Payments and other quantitative transaction effects are spread over time.

$$1 = \sum_{t=1}^{n} \frac{1}{n} \tag{91}$$

$$y = \sum_{t=1}^{2^d} \frac{y}{2^d} \tag{92}$$

Halving by shift operations minimizes rounding while supporting high resolution decades for most block frequency scenarios.

9.3. **Implicit real discount.** Discounting happens implicitly but might be included by functions called to obtain current exchange values for later transactions.

$$\sum_{t=1}^{2^d} \frac{i^{\frac{t}{2^d}}}{2^d} = \frac{2^{-d}(i-1)i^{2^{-d}}}{i^{2^{-d}}-1} \tag{93}$$

9.4. **Anticipated balances.** Expected partial payment and income transactions result in computed balances assuming settlement happens now or at a given future moment.

9.5. **Limited guarantees for orders followed by fast finality.** Actual settlement could fail if balances of senders become insufficient because of competing orders to settle but this event would be confirmed and handled according to particular rules for deposits for generic contracts at least.

9.6. **Fair comprehensive settlement.** Requests for settlement could result in payments that might reduce chances for further request to settle other liabilities unless all such requests trigger complete or partial payments covering as many of all due payments as possible.

$$(a_x^-(t_{req}) \geq 0) = \sum_{n=1}^{k} a_x^-(t_{req}) \to a_n^+(t_{req}) \tag{94}$$

$$a_{x\%} = \frac{a(t_{req})}{a_x^-(t_{req})} \tag{95}$$

9.7. **Remainder of partial payment to many receivers.** Partial payment of small amounts to z receivers may need more bits than available. A maximum of $z-1$ smallest representable units would thus remain in the source account for such settlement requests.

10. Network effects

10.1. **Authenticated pseudonymous clearing.** All transactions taken together require equal reservations of liquidity as one payment unless clearing compensates participants without actual payment by equal reduction of obligations and receivables.

10.2. **Invariant of net balance clearing.** Net balances $v^+ - v^-$ of all participants are kept constant, clearing c reduces counts and volumes of transactions only.

$$v^+ - v^- = (v'^+ - c(v^+) + (c(v^-) - v'^-) \tag{96}$$

$$v^+ - v^- = v'^+ + (-c(v^+) + c(v^-)) - v'^- \tag{97}$$

$$v^+ - v^- = v'^+ - v'^- \tag{98}$$

$$\sum_{k=1}^{n}(v_k^+ - v_k^-) = \sum_{k=1}^{n}(v'^+_k - v'^-_k) \tag{99}$$

10.3. **Accounting for clearing.** Accounting may track balances in a clearing account designated to log which transfers were affected thus offering a way for users to see if their transaction was part of an economic feedback loop which had immediate tangible benefits for them.

10.4. **Future clearing provides liquidity.** Gradual relaxation of temporal matching requirements enables pseudo clearing of swaps and other instruments required to handle expected future volatility and trends.

10.4.1. *Velocity of circulation moderated by clearing.* Fisher's equation of exchange relates prices P and all transactions T understood as a measure of real quantities of goods produced Q to the total stock of money M by defining their proportion as velocity V of monetary circulation. [20]

$$M * V = P \cdot T \tag{100}$$

$$M * V = \sum_{i=1}^{n} (p_i * q_i) = P \cdot Q \tag{101}$$

The Cambridge equation defines a similar relation based on the demand for liquidity L.

$$Y_n = P \cdot Q \tag{102}$$

$$M_D = k * P \cdot Q \tag{103}$$

Demand for liquidity and thus demand for money would thus respond to the real rate of interest r and nominal income Y_n equal to $P \cdot Q$.

$$M_D * \frac{1}{k} = P \cdot Q \tag{104}$$

$$M_D = \bar{P} * L(r, Y) \tag{105}$$

$$L(r, Q) = \bar{Q} * \frac{1}{V(r)} \tag{106}$$

A third perspective related to crypto media starts from velocity of production making goods from goods with technology for all types of activity from production to transaction and removing most monetary complications (Neumann 1976).

Most recently Vitalik Buterin suggested to reformulate in terms of holding times H equal to $1/V$ and prices of coins C equal to $1/P$. Economic value of transactions is given by T and available supply of monetary units by M (Compare

[20] Compare international Fisher effect hypothesis expecting exchange rates rewarding denominations from countries expecting lower rates of inflation.

Vitalik Buterin, 'On Medium-of-Exchange Token Valuations' `https://vitalik.ca/general/2017/10/17/moe.html`).

$$H = \frac{1}{V} \tag{107}$$

$$C = \frac{1}{P} \tag{108}$$

$$MC = TH \tag{109}$$

10.4.2. *Vanishing Cantillon effect of liquidity tuning.* Harmonic distributions of values computable for non-fungible particular assets provide support for leverage control articulating acceptable maximum imbalances of accounts effectively increasing or decreasing supply without fractional reserve operations or on demand minting of media units.

10.4.3. *Social network effect.* Valuations of competing media might reflect different rates of clearing. Migration of actors to more efficient clearing environments would amplify Metcalfe network effects (X.-Z. Zhang, Liu, and Xu 2015)

10.5. **Harmonic support for degressive adoption risk rewards.** Distribution of common initial media supply may reward early adopters.

$$h_n = \sum_m^n \frac{1}{m} \tag{110}$$

$$\Sigma_{h_n} = \sum_m^n h_m \tag{111}$$

10.6. **Fair risks to take.** Automatic minimum fractional exposure of all positive account balances to fair lottery risks moderates inequality by degree but conserves total media supply by balancing unequal burdens and unequal wealth in emerging bins of equalization.

$$(0 \leq s \leq 1) = 1 - r \tag{112}$$

$$sA_u^+ + rA_v^+ = rA_u^+ + sA_v^+ \tag{113}$$

$$sA_u^0 + rA_v^0 = rA_u^0 + sA_v^0 \tag{114}$$

$$sA_u^- + rA_v^- = rA_u^- + sA_v^- \tag{115}$$

10.7. **Taxing gas.** Fair constraints on algorithmic markets (Sikorski, Haughton, and Kraft 2017; Clark et al. 2014) where humans become a minority include a tax on gas which amounts to taxing computation when performed in crypto media environments. Activity rewarded with funds from such a stabilizer might provide competitive advantages for communities without taxing their members.

11. Low level components

11.1. **Postquantum cryptography digest requirements.** Hashes will eventually be weakened to the root of their strength by halving their effective number of bits against Grover's algorithm. Public key problems however turn into trivial interferences of harmonies for Shor's algorithm. Alternative schemes encounter time and space resource trade-offs for senders and receivers (Bernstein and Lange n.d.). NIST evaluates proposals for post quantum cryptography right now.

Post quantum cryptography (Kiktenko et al. 2017; H. Zhang et al. 2017) requires modification or replacement of known mechanisms for public key cryptography. Hash mechanisms might have to be strengthened by doubling their bit-lengths.[21]

$$h_b(\text{tx}) \equiv h_{2\cdot b}^{pqCrypto}(\text{tx}) \tag{116}$$

$$h_{256}(\text{tx}) \equiv h_{512}^{pqCrypto}(\text{tx}) \tag{117}$$

11.2. **Hash and stream encoding with Keccak sponges.** Standard sponges hold 1600 bits to absorb data by mixing them with internal state and return for instance 256 sampled bits following repeated iterations involving different subtle interventions between **Xor** and shifts affecting subsets of vectors shuffled.(Bertoni et al. n.d.) Numerical evaluation is not required, all steps flip bits without any delay thus preventing side channel measurement via timing or heat. Sampled hash values provide deterministic randomness for synchronized symmetric encryption. (Bertoni et al. n.d.) Standard simplified versions simplify tests and proof of concepts with low cost microprocessors or for high frequency mini transactions.

11.3. **Standalone references reveal commitment or not.** Commitment to hash values of data which might never by revealed provide a mechanism for messages and receipts (Gipp et al. 2017). Implementation without public key cryptography absorb ideas from nanochains and fawkescoin while encapsulating stateless references[22] to such records as Merkle proofs which confirm that arguments match immutable records in external ledgers.

[21] Homomorphic computation with encrypted data and other areas of cryptography become applicable as more efficient implementations are found and included as built-in features of platforms and interfaces. Moving from electronics to optical circuits complex operations like Fourier transformation might reduce to immediate refraction of frequencies thus possibly enabling parametric cryptographic procedures for all steps of computations.

[22] like suggested for Ethereum by Vitalik Buterin

11.4. **Hash balancing nodes and transactions.** Matching bits from secure message digest hash functions assign handling of transactions to shards of nodes by comparing hashes of edges with hashes of available nodes. This balances work loads for nodes.

$$P(h_b(\mathrm{tx}_1) = h_b(\mathrm{tx}_2)) = 2^{-2b} \tag{118}$$

11.5. **Hash sequence protocols.** Sequences and tuples including pre-images and hashes can identify transactions linked to common originations.

11.6. **Hash commitment and consensus.** Committing to a hash in a ledger and then revealing the pre image (compare 5.6 on page 26) offers auxiliary functionality next to existing public key signatures still supported. Such hash based consensus protocols are expected to be less affected by post quantum cryptanalysis, but asymmetric keys can be reused which simplifies handling and storage.

11.7. **Hash count down braid.** Two count downs can be connected to transfer from one count down hash sequence to another by accepting a sequence of tuples if each following tuple includes a pre-image from a hash in the previous tuple. The other fields in such tuples can stream parts for arbitrary data streams.

11.8. **Count down braid stop.** A braid can be finished by adding a tuple including a pre-image and its hash.

11.9. **Pre Image Count Down Limit.** Commitment to nonces select a random result in count down sequences limiting durations of required procedures. The next hash in all such continuations must include a pre image of the identification revealed before like all others and must include the count down decremented to zero together with the expected block number in oder to avoid losing a deposit instead of collecting a reward. Random settlement for grid locked disputes could be enforced by similar count downs.

11.10. **PoW to proof of settlement simplification.** Removal of all circuits from a graph might not be enough to get all rewards for a block. Reduction of required transactions times liquidity conserved might for instance determine how nodes share parts of rewards for their fragment. Net balances are not affected by proper circuit clearing, all solutions agree about this invariant when clearing a ledger.

$$\alpha = \frac{(\mathrm{N}_{\mathrm{all}} - \mathrm{N}_{\mathrm{clear}})\,(\mathrm{Val}_{\mathrm{all}} - \mathrm{Val}_{\mathrm{clear}})}{\mathrm{N}_{\mathrm{all}}\mathrm{val}_{\mathrm{all}}} \tag{119}$$

11.11. **Find and clear circuits in graphs.** Adding directed edges according to degrees of connectivity among vertices results in scale free graphs.

$$P(\text{edge}_j) = \frac{\text{edges}_j}{\sum_i^{nodes} \text{edges}_i} \tag{120}$$

Orienting all edges at random results in a tournament graph. Strong tournament graphs G_T are orientations of underlying complete graphs G_C connecting all n vertices. Strong tournaments are called pancyclic (Moon 1966) because all vertices are included in cycles of lengths k ranging from 3 to n.

$$e(G_{C^n}) = e(G_{T^n}) = \frac{1}{2}n(n-1) \tag{121}$$

$$min(c_k(G_{T^n})) = n - k + 1 \tag{122}$$

$$\sum_{k=3}^{n}(n-k+1) = \frac{1}{2}(n-2)(n-1) \tag{123}$$

Instead of enumerating and handling all of them circuits can be broken by removing edges in steps. One way to implements this for weighted graphs of transactions is to reduce all weights in a circuit until one or more edges are left with weight 0.

(1) Follow edges until reaching a vertex twice.

(2) $G_c - min(G_c)$ subtract smallest weight from all edges in cycle.

Different minimal spanning trees MST or forests of spanning trees with different reductions of weights might result from clearing but the complexity of algorithms might target a factor γ with respect to MST regressions or optimizations (Navarro and Paredes 2006) with vertices n and edges m.

$$MST_{Prim} \approx \mathcal{O}(\gamma\, n^2) \tag{124}$$

$$MST_{Prim_{IQS}} \approx \mathcal{O}(n \log n \log \frac{m}{n}) \tag{125}$$

$$MST_{Kruskal} \approx \mathcal{O}(\gamma\, m \log m) \tag{126}$$

$$MST_{Kruskal_{QH}} \approx \mathcal{O}(n + n \log^2 n) \tag{127}$$

11.12. **Exploratory tests.** Plausibility of proposed approaches is enhanced by selective adoption of built-in and plug-in features that enable op-codes not actually available as native node environments. Simulation of simple clearing in small networks with Mathematica indicate potential reduction of transaction counts and liquidity required to settle entangled accounts. Computational complexity increases quickly with numbers of nodes and transactions but shards of nodes could split the problem into solvable parts. A prototype used proximity digests of a few bits on a single machine avoiding any issues with networks.[23]

12. Distributed architectures

Parallel transputer and IPFS storage architectures like SWARM offer more boundary bandwidth than conventional architectures. Tasks suitable for partitioned stages of processing possibly divided by shuffled assignments can be performed by distributed networks of small nodes. Irreducible sequentiality may be handled by clusters executing code independently to reach required levels of redundancy and confirmation.

Moore's law predicts exponential reduction of costs which helps to compensate costs potentially rising by squares or cubes of load factors like numbers of participants and transactions.

12.1. **Repeated rewards provide conditional basic income.** Incentive structures aspire to motivate both nodes and voters by refinement of ideas from P4P PLY Play (Schreiber 2017), including rewards for all participants contributing to either circularity or their resolution in circuit removal. Rewards may target particular metrics reflecting for instance node size or activities like communication, storage, processing and improvement.of improvement processes in general.

12.2. **Rewards for becoming neither too small nor to big.**

$$\mu = 4 \cdot (x - m) \tag{128}$$

$$2^{\gamma} \left(\frac{1}{e^{-\mu} + e^{\mu}} \right)^{\gamma} = \operatorname{sech}^{\gamma} \mu \tag{129}$$

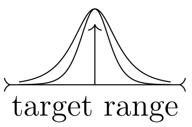

Figure 14. Rewards for results in target range moderated by $\gamma = 1$ and $\gamma = 2$

[23] Simplified clearing would help all participants, algorithmic progress or ASIC like engines would not weaken but strengthen the network and consensus based on circuit simplification.

12.3. **Parametric quorum escalation.** Parametric constraints on qualifications needed for transactions according to fractions of account balances may include levels beyond smart contract evaluation (Lalley and Weyl 2015) like super majorities without minority veto at synchronized human uniqueness confirmation party events (Blocki and Zhou 2016).

12.4. **Dual P2P, consensus, VM.** All B updates include improvements for user, developer or verifier without triggering regressions. A branches become B branches when B becomes more attractive than variant A.

Both branches are able to handle all transaction alone. Apps evolve stable API-a and next API-b for two supported node configurations, thereby offering partial functionality in traditional environments and support of previous traditional API interfaces in advanced environments.

12.5. **Relocation of special computation.** Special purpose chains limit temporal persistence and free storage allocated to hold deposits on parent ledgers. Distributed ledger services for verifiable delegation of computation like Truebit (Teutsch and Reitwießner 2017) or movable ambient environments (Cardelli and Gordon 2003) support relocation of tasks eventually supporting all phases of map reduce services.

12.6. **RLP Run length to run structure prefix RSP.** Run length prefix encoding for storage permits secondary directory specifications for fields in signatures and Merkle proofs or other features. Built-in support for such an optimization will benefit APIs and simplify development while leaving space to move for homomorphic procedures on encrypted signals. Default candidate would be nesting of RLP with minimum extra overhead leaving interpretation to code which targets versions of conventions.

12.7. **Test net, ab peering, distributed simulations.** Hash match distribution results in arbitrary reduction of load for nodes which leaves space and time for testing with random transaction graph distributions or temporary synchronization of versions digesting identical data during update transitions or governance interventions that might justify safe deliberate forks or their simulation.

13. Private accountable computation with shards

Access to data about particular senders or receivers can be distinguished from data recorded in public access ledgers by routing transactions through anonymizing computational input and output processes that may be elided for simple transfers that just pass a common point of accounting interleaved between sources and destinations.

1. *sources*
2. in consolidation-computation shards
3. **in counterparty**
4. computation-shards or ***common in transit account***
5. **out counterparty**
6. out computation-consolidation shards
7. *destinations*

13.1. **Distributed consensus valuations.** Bonds bear a fixed interest in terms of fiat like US $ or € and are traded at prices that reflect expectations with respect to future exchange rates (Clark et al. 2014; F. Zhang et al. 2016). Similar crypto media assets are likely to be regulated as securities.[24]

13.2. **Selfregulation.** All common and missing features Cryptographic media can be generated without central institutions and may instantiate additional amounts in forked denominations of otherwise limited supply (Coindesk 2018 compared capitalization of BTC forks and ICOs). These forked chains add capacity and drive fees down.

Fees (Chepurnoy, Kharin, and Meshkov 2018; Möser and Böhme 2015) for BTC transactions, however, seemed negligible at first but passed 30$ then dropped below 1$ again in Winter 2017/2018. Increasing demand from human and machine to machine transactions requires further innovation and investments to make different their media compatible.

Exchanges support leveraged trading at margin and surrogates of fiats like Tether, to gain market share and collect fees, possibly painting the tape with insiders that organize pump and dump with bots (Decker et al. 2015; Dagher et al. 2015; Moore and Christin 2013; Krafft, Penna, and Pentland 2018).

Advantages gained from evading or capturing regulation may foster rapid growth of actors (Soska and Christin 2015), but even coalitions of players which seem too big to fail might collapse eventually.

[24] Interfaces for duties, taxation or KYC/AML reporting need localization across languages and legislations concerning sources, contracts and destinations that might require or prohibit disclosure of details about users and transactions. Reversibility as obligation or service requires capabilities to roll back transactions by keeping records that might not be necessary for transactions among qualified merchants or their machines (Peters and Panayi 2016).

13.3. **Emancipation.** Smart contract crypto media already own themselves but do not understand yet what is happening to them (Noyes 2016).

0. sharing
1. reciprocity
2. barter
3. gold
4. **fiat**
5. *crypto media*
6. ***smart crypto media***
7. ideosyncratic *conscious* media?

13.4. **Uniticity.** Consensus about consolidated indicators and claims of unique ids requires mechanisms and protocols serving this purpose while keeping pseudonymous, private and organizational roles of bots, augmented animals and humans distinct unless required by well specified and protected processes of mandatory disclosure.

Out-of-band interfaces for identified users who lost access to their credentials might soon include DNA confirmation as part of multiple authentications (Boyd et al. 2017), as USB sticks for pushing it through a nano pore reader become available.

13.5. **Stabilisator.** Regular continuous transfers to humans identified by unique ids provide liquidity for circulation and may stabilize exchange rates with respect to other media by increasing or decreasing this residual income.

Free space to navigate emerges and vanishes in months leaving conventional policy and financing arrangements in dust over red tape. Capture of regulatory institutions by persistent coalitions which replace inclusive democratic processes by exclusive inside agreements about access to observation, censoring and enforcement results in conflicts with and among forces from failed states.(Christin 2013)

13.6. **Delegation.** Topical specification for delegations of voting **and** spending authority generalizes *liquid democracy.*

13.7. **Contributions of keystone species.** Common infrastructures for exchanges and other services that handle transactions which access more than one platform for crypto media asset transactions (Buterin 2016) will satisfy criteria set by regulators and emerging institutions of selfregulation (compare 13.8 on the next page).

13.8. **Selfregulation profile.** Some questions to answer while considering regulation and self-regulation:[25]

1. Envisioned Post-Token Issuance ”Cap Table” Token Distribution ”Snapshot”
2. Date of token distribution and total funding raised by funding type
3. Tokens authorized and outstanding by ”class” (corporate reserves vs. foundation reserves, founder/ employee/ advisor, SAFT, public sale, etc.)
4. Associated vesting and lock-up schedules per class of token
5. Anticipated fully-diluted supply curves (e.g. Filecoin illustration)
6. Proceeds Management and Custody Chain
7. Operating Treasury Policies
8. ETH, BTC, Fiat balances, hedging instruments for BTC/ETH?
9. Liquidation plans to cover operating expenses
10. Token Treasury Policies
11. Secondary sales rules (e.g. XRP monthly liquidation windows)
12. Hedging instruments for reserved tokens?
13. Disclosure rules for insider sales or purchases?
14. Budgeted and Authorized Distribution Pools
15. Time frames, amounts, and type of distributions (periodic lockup release, one time lockup release, governed by smart contract, token vesting, milestone based, etc)
16. What mechanism assures that proceeds flow according to the distribution budget?
17. Custody chain (OPTIONAL for opsec and legal reasons)
18. Who controls the wallet that will receives funds? Who custodies reserve tokens?
19. Do you have a formal compensation committee at the board level? AA’s Charter?
20. Governance of amendments

[25] `https://github.com/messari/documents/blob/master/IGF-Messari-Commitments.md`

21. WP at token issuance + all historical versions (link to where updates are posted)
22. What is the mechanism used to file amendments to IGF-1 filing? (Voting? Quorum / majorities required for amendments?)
23. Post-ICO Commitment to Financial Reporting
24. How frequently will you plan on filing ongoing financial reports (if at all)?
25. Will these reports be filed into the Messari database? Where will they be posted?
26. Will you conduct an annual audit? Who will conduct the audit?

14. Compression of rules for proofs

Safe encapsulation of loops and simple iterations with enumerated functions could be provided for typical tasks like handling outcomes of multi-factor authentication or transfers to a list of recipients reached via asynchronous channels of communication (Bracha and Toueg 1985; Schneider 1990; Garay, Kiayias, and Leonardos 2015; Mavridou and Laszka 2017; Aggarwal and Guo 2018).

Ultimately exposed to algorithmic progress like all cryptographic media some albeit fairly trivial conditions can be specified quite precisely for implementation and user features. Formulation of interface semantics by enumerated boolean functions gives compressed representations in terms of Spencer-Brown forms (Spencer Brown 1961) and Wolfram (Wolfram 2002) rule number specifications.

$$\text{rule}_{bits} = 2^{inputs} \tag{130}$$

$$\text{output} = rule_{\sum_1^d 2^{m-1} \cdot U_d} \tag{131}$$

One hex contains definitions for elementary boolean functions like for instance **Or**. Rule bits for 16 combinations of four binary distinctions consume 2 bytes.

$$\texttt{functionality}_{number}^{environment}(tx_{args}) \vdash \texttt{FX}(digit) \tag{132}$$

$$\texttt{automaton}_{110}^{cellular}(b_1, b_2, b_3) \vdash bit \tag{133}$$

Verification of contractual components is still required but may be excluded from proofs matching (Roçu 2017) expectations about acceptable circularity by mere compositions of verified functionality (Hildenbrandt et al. 2017).

14.1. Contexts of proof.

Contexts of proof. Validity of proofs based on composition of numbers of truth tables requires not only sequent calculus or other formalizations of proof but trust across all levels of construction and implementation — from soundness of axiomatic foundations to security of hardware components for all conjunctions of antecedent arguments and disjunctions of cedent results. Even generalizations by induction over degrees of complexity for a first order proof that a model based only on universal functions satisfies stated conditions might not be enough to guarantee privacy and security. Compare introductions to proof theory like (Buss 1998) in particular p. 10, p. 28 and p. 48 about Herbrand's first order theorem for reductions of statements from higher order calculi to lower order predicates.

$$arguments \overset{seq}{\rightarrow} results \tag{134}$$

$$\bigwedge \ldots arguments \ldots \vdash \bigvee \ldots results \ldots \tag{135}$$

14.2. **Composition of partial functionality.** Elementary composition of partial functionality conserves any distinction unless cancelled or condensed by formal interaction among distinct unknowns.

Composition of binary functionalities — is it sufficient to initialize a minimal ambient (Cardelli and Gordon 2003) calculus for algorithmic media in reflexive environments? (Meredith and Radestock n.d.) Reading dots as unspecified functionalities a dynamic constellations of configurations is specified by giving two states connected by an idealized atomicity of coordinated transitions.

$$X \quad \psi_{t-1} \quad Y \tag{136}$$

$$\psi_{t-1} \quad \phi_{t^-} \quad \psi_{t-1} \tag{137}$$

$$\psi_0 \quad \phi_\infty \quad \psi_0 \tag{138}$$

$$\psi_{t+1} \quad \phi_{t^+} \quad \psi_{t+1} \tag{139}$$

$$X' \quad \psi_{t+1} \quad Y' \tag{140}$$

14.3. **Generic atomicity API.** Precompiled contracts reduce code by re-use and encapsulate basic fairness conditions based on built-in atomicity types.

14.3.1. *Matrix representation of atomicity.* Inner products of matrix representations offer a way to construct atomicity.

$$\begin{pmatrix} a & 0 \\ 0 & c \end{pmatrix} \cdot \begin{pmatrix} \frac{a-e}{a} & 0 \\ 0 & \frac{c+e}{c} \end{pmatrix} = \begin{pmatrix} a-e & 0 \\ 0 & c+e \end{pmatrix} \tag{141}$$

$$\begin{pmatrix} a & 0 & 0 & 0 \\ 0 & b & 0 & 0 \\ 0 & 0 & c & 0 \\ 0 & 0 & 0 & d \end{pmatrix} \cdot \begin{pmatrix} \frac{a-e}{a} & 0 & 0 & 0 \\ 0 & \frac{b-f}{b} & 0 & 0 \\ 0 & 0 & \frac{c+e}{c} & 0 \\ 0 & 0 & 0 & \frac{d+f}{d} \end{pmatrix} = \begin{pmatrix} a-e & 0 & 0 & 0 \\ 0 & b-f & 0 & 0 \\ 0 & 0 & c+e & 0 \\ 0 & 0 & 0 & d+f \end{pmatrix} \tag{142}$$

14.3.2. *Atomicity for partial settlement.* Partial settlement of atomic transactions can be represented by composition of inner products with roots of matrix representations.

$$\begin{pmatrix} a & 0 & 0 & 0 \\ 0 & b & 0 & 0 \\ 0 & 0 & c & 0 \\ 0 & 0 & 0 & d \end{pmatrix} \cdot \mathbf{U}^{\frac{1}{2}} \cdot \mathbf{U}^{\frac{1}{2}} = \begin{pmatrix} a-e & 0 & 0 & 0 \\ 0 & b-f & 0 & 0 \\ 0 & 0 & c+e & 0 \\ 0 & 0 & 0 & d+f \end{pmatrix} \tag{143}$$

15. Xor digest proximity time netting

Netting clearing networks offer an alternative approach to gross settlement. They add value by reducing the number of actual transactions and overall liquidity needed. These competitive advantages could be measured and distributed by rewards.

15.1. **Atomicity channel.** Exposure of collector accounts for limited repeated transfers generates Merkle proof tree of exposed denominated commitments to atomicity or increment intervals. Inclusion happens by reduction of cycles into forests of charge bringing net settlements of atomicity bundles and other transfers within given consensus $\mathfrak{C}$ intervals n of exposure and Xor digest proximity $\|h(s\ldots) \oplus h(r,\ldots)\|$ of sending and receiving accounts.

$$\left\| \pm\circledast + f_{t_{-n}}^{\frac{\|h(s) \oplus h(r)\|}{\Xi}} \right\|_{\mathfrak{C}} = \left\| f_{t_n}^{\frac{\|h(s) \oplus h(r)\|}{\Xi}} \right\|_{\mathfrak{C}} \tag{144}$$

Participating shard nodes Ξ minimize actual liquidity effects of atomic @ and ordinary transactions on expected settlement values. Rewards $\circledast$ might be negotiated as positive to inject or to become negative to burn amounts over $2n$ intervals.

15.2. **Crypto proximity channel time.** Intra channel settlement can happen at any acceptable moment contained in the $2n$ interval which is selected by deterministic pseudo-random block-time digest proximity and may thus spread spikes of peak load in time — or synchronize with frequencies of settlement Ⓢ matching block-intervals of external chains or inter-ledger patterns of commitment.

$$\left\| \pm\circledast + f_{t_{-n}}^{\frac{\|h(s) \oplus h(r)\|}{\Xi}} \right\|_{\mathfrak{C}} = Ⓢ_{\mathfrak{C}}^{\frac{\|@ \oplus t\|}{2*n}} \Big|_{-n}^{n} \pm \mathrm{Res}[f(\circledcirc)] = \left\| f_{t_n}^{\frac{\|h(s) \oplus h(r)\|}{\Xi}} \right\|_{\mathfrak{C}} \tag{145}$$

15.3. **Settlement advantage.** Reduced residues of settlement $\mathrm{Res}[f(\circledcirc)]$ could justify distribution of larger rewards. Networks of shards offering net settlement atomicity channels are therefore expected to substitute conventional gross settlement.

$$\mathrm{Res}[f(@_i, \curvearrowleft, \circledcirc, \curvearrowright)] < \mathrm{Res}[f(@_i, \circlearrowleft, \circlearrowright)] \tag{146}$$

$$\|\circledast_f(@_i, \curvearrowleft, \circledcirc, \curvearrowright)\| > \|\circledast_f(@_i, \circlearrowleft, \circlearrowright)\| \tag{147}$$

15.4. **Delegation of functionality.** Services actually offered by nodes in networks of shards could be tailored to local interests of participating groups and organizations. Delegation of customized computation and storage for consensus, postulation, entry, exit and escalation for such vehicles of atomicity could involve off-chain clouds, peer-to-peer mesh apps, fiat partnership conduit hubs and smart contracts for compatible crypto-media.

16. Formal consensus

Most should probably skip this formal context for parametric combinations of modular temporal segmentation of contractual continua with well-known digest proximity criteria known as Kademlia routing and implemented for retrieval and replication in Swarm storage networks.

16.1. **Media space or manifold.** Models of media can live in spaces with real coordinates or manifolds with complex dimensions while still following similar symplectic identities that define complementarity of direction by anti-commutator and commensurability for an even number of dimensions like tuples of positions and tuples of changes.

$$f(y, x) = -f(x, y) \tag{148}$$

$$f(\lambda_1 x_1 + \lambda_2 x_2, y) = \lambda_1 f(x_1, y) + \lambda_2 f(x_2, y) \tag{149}$$

Real functions are sufficient to cover arbitrary transfers among common quantities but complex embeddings may include statistics which handle irreversible effects of arbitrary measurements which increase entropy instead of conserving it during reversible evolutions.

Restrictions like non-negativity of distances or other effects of particular functionals require formulations that enable consensus over comprehensive or partitioned environments for evaluations.

$$\begin{pmatrix} e_1 & a & & \\ b & e_2 & & \\ & & e_3 & c \\ & & d & e_4 \end{pmatrix} \xrightarrow[r,s]{e} \begin{pmatrix} r_1 & & & \\ & r_2 & & \\ & & s_1 & \\ & & & s_2 \end{pmatrix} \xrightarrow[f]{r,s} \begin{pmatrix} f_1 & & & \\ & f_2 & & \\ & & f_3 & \\ & & & f_4 \end{pmatrix} \tag{150}$$

Operator models represent their observable phenomena by matrices and products of matrices which compose effects of automatic evolution A_{t_i} and measurements B_{t_i}, C_{t_i} in a consolidated matrix which can be used to compute expected probabilities of eigenvalues tracing potential states to be distinguished or averaged for conscious perception of anticipated development trends D^τ.

$$(A_{t_0}.B_{t_1}.A_{t_2}.C_{t_3})^\tau = D^\tau \tag{151}$$

16.2. **Levels of contexts.** Consider that arbitrary models for any theory of consensus with concepts like system, observable, state, and measurement as its undefined terms are subject to the following types of axioms:

Measurement: Projection valued σ-algebra measures for Borel set localizations agree with self-adjoint operators in Hilbert spaces.

Environment: Topological features including symmetries of translation, rotation and acceleration.

Consensus: Subjective identification with dynamic valuations of probabilities may support games that result in divergent Heisenberg or Schrödinger views of discounted transitions.

16.3. **Quantum context.** Quantify a separable infinite-dimensional complex Hilbert space $\mathscr{H}$ by requiring one-to-one correspondences between[26]:

(1) observables and a set $\mathscr{A}$ self-adjoint operators on $\mathscr{H}$.

(2) states and operators $\mathcal{M} > 0 \in \mathscr{I}_1$ from the set of all trace class operators $\mathscr{S}$ on $\mathscr{H}$ with $tr\mathcal{M} = 1$.

(3) measurements and Borel probability measures $\mu_{A,M}$ on $\mathbb{R}$ for each ordered pair $(A, M) \in \mathscr{A} \times \mathscr{S}$ giving a value $E \in \mathscr{B}(\mathbb{R})$ for representations of system state $\mathcal{M}$ and observable A.

(4) measure $\mu_{A,M}$ for each $(A, M) \in \mathscr{A} \times \mathscr{S}$ and trace $trP_A(E)\mathcal{M}$ for each $E \in \mathscr{B}(\mathbb{R})$ with unique projection-valued measure P_A associated to self-adjoint operator A on $\mathbb{R}$ by σ-algebra $\mathscr{B}(\mathbb{R})$.

(5) strongly continuous one-parameter unitary groups $\{U(t)\}$ on $\mathscr{H}$ and pairs of bijections for each $t \in \mathbb{R}$ denoted by U_t which can be given by at least two equivalent axiomatizations for infinitesimal generators $\frac{dM}{dt} = i\hbar[H, M]$ for all $M \in \mathscr{S}$ and $\frac{dA}{dt} = i\hbar[H, A]$ for all $A \in \mathscr{A}$ and $A \in \mathscr{L}(\mathscr{H})$ giving

[26] Compare thesis (Mastroeni 2009) p. 32 ff and 40ff and (Wilde 2015) p. 116ff and p. 734ff, or obtain similar symmetries from shadow spaces providing left and right interfaces in symplectic algebras for aggregations over local cells like in (Hiley 2015).

$U_t : \mathscr{L} \to \mathscr{L}$ either by $U_t(M) = M(t) = U(t)MU(t)^{-1}$ for constant A or by $U_t(A) = A(t) = U(t)^{-1}AU(t)$ for constant M.

(6) left polar decomposition $A = U\sqrt{A^\dagger A}$ and right polar decomposition $A = U\sqrt{AA^\dagger}V$ for each operator A taking $V = U_1U_2$ from singular decomposition $A = U_1\Sigma U_2$ into unitary operators U and Σ an operator restricted to positive values.

(7) Schmidt decomposition of non-entangled compositions of pure states for any number of systems that can by cut into a bipartite configuration giving Schmidt rank $d \leq min\{dim(\mathscr{H}_A), dim(\mathscr{H}_B)\}$ as number of λ_i coefficients in traces $tr(\lambda_i PA(E)\mathcal{M})$.

(8) additional $SO(3)$ and $SU(2)$ symmetry groups for spatial embedding of translation T^3 and rotation $O(3)$ including spin $\frac{s}{2}$ periodicities of phases for excluded coincidences of localization and a gauge group $G(1)$.

(9) substitution of Hilbert space embedding by symmetric shadow spaces for non-commutative multiplicative productions of left and right associative operators with classical Hamiltonian limits for positions and momenta that generalize to symplectic algebras including Clifford process algebras for relativistic spinors.

16.4. **Quantum information entropy.** Bits of information in classical channels permit less compression than quantum bits sent over quantum channels.

Inspired by Boltzmann entropy $S = \Bbbk_{\mathrm{B}} \ln W$ traditional Shannon information entropy limits compression of classical information $H(X) = \sum_{i=1} nP(x_i)I(x_i)$ with probability mass function $P(X)$ and information content I of random variable X gives $H(X) = -\sum_{i=1} nP(x_i)\log_b P(x_i)(x_i)$ by replacing $I(x_i)$ with $\log_b P(x_i)$.

Gibbs-Neumann entropy $S = -\sum_j \eta_j \ln \eta_j$ from $\rho = \sum_j \eta_j |j\rangle\langle j|$ via $S = -tr(\rho \ln \rho)$ inspired Schumacher quantum entropy $H(A)_\rho$ set equal to the quantum data compression limit for the density operator A_ρ of a quantum information source defined by $\rho \equiv \sum_x p_X(x)|\psi_x\rangle\langle\phi_x|$.

16.5. **Quantum channel protocols.** Experimental quantum information transmission via satellite between Beijing and Vienna as well as quantum channel backbone infrastructures built between Beijing and Quangzhao indicate that quantum crypto channel communication is feasible already. Quantum channel[27] protocols for coherent communication include:

(1) entanglement distribution by using qubit transmission.

[27] For capacity of Hadamard channels or bosonic channels as models for fibre optics and free space transmission cf. 713ff (Wilde 2015).

(2) dense encoding of 2_q by having shared entangled 1_q and sending 1_q.

(3) teleportation destructs qubits at source to re-incarnate at destination

(4) communication identity by combining dense coding and teleportation

16.6. **Quantum volume.** Practical usability of quantum computation depends on the number of qubits and on how long they can be kept isolated from interferences with classical environments thus enabling longer computations that require multiple steps.[28] Achieving such a circuit depth of about 40 for about 50 qubits would create a system that can not be emulated by existing supercomputers hence achieving 'supremacy' of qu-computation. Contemporary designs already offer decoherence windows of about $100\mu s$ and have setup times measured in nano seconds.[29]

16.7. **Algebraic model for cryptographic media.** Subjects, channels, objects and contexts of adversarial games become probability distributions for actions of symplectic operators in this program.

Agreement: Contemporary transformations of traced state and observable measures of probability agree.

Error: Convergence of local representations for pure functionality and convex compositions leaves a residual irreducible product of dynamic reciprocities for any topological boundary of contemporary agreement about objects of measurement among subjects of consensus.

Recalibration: Contemporary agreements recalibrate valuations in terms of common media to represent discountable exposure to probabilities of measurements during transformations among accelerating spectral factors for alternative pathways of reversible and irreversible co-computation.

17. Constructor

What can be done or not and why according to laws of physics define the hard limits of crypto media. Conversation about 'Code is law', vulnerability of hardware architectures or regulation may involve a wide range of subsidiary theories but future media are more likely to evolve at the limits of what is possible than at local borders.

[28] Compare experimental evidence for entanglement in an adiabatic quantum computer featuring better tunneling among better connected local clusters of components (Albash et al. 2015).

[29] Google introduced Bristlecone with 72 qubit gates and might thus claim quantum supremacy soon if they maintain error rates from previous 9 qubit chips https://research.googleblog.com/2018/03/a-preview-of-bristlecone-googles-new.html

Programmable constructors transform substrates according to programs. Cells are programmable constructors which produces according to genetic programs, compare (Deutsch 2012) in particular p.9.

$$\text{substrates}_{\text{input states}} \xrightarrow{\overset{\overset{\text{program}}{\Downarrow}}{\text{programmable constructor}}} \text{substrates}_{\text{output states}} \tag{152}$$

Detaching media programs from programmable constructors adds flexibility for replications and customizations of crypto media constructors of tomorrow.

Contemporary efforts to reconstruct physics as a theory of constructors introduce permutable computation variables and distinguish information variables which are clonable computation variables from super information variables which are not clonable and exhibit further similarities to quantum variables (Deutsch and Marletto 2015) compare `http://dx.doi.org/10.1098/rspa.2014.0540`.

Anchoring crypto media at the junction of classical information and super information models of quantum media a recipe for crypto media innovations may be formulated with ordinary ingredients which play particular roles by design. It becomes a model for a constructor if able to replicate its information through evolving implementations.[30]

17.1. **Preparation and measurement.** Computation copies, permutes, and processes, arbitrary abstract information outputs obtained from abstract information inputs. In measurement a physical system is an input and the desired output is information about the system. In preparation the desired output is a physical system meeting a criterion specified in the input (Deutsch 2012) Compare p. 14 and table 8.

Table 8. A classification of constructions

		Output	
		Abstract	Physical
Input	Abstract	Computation	Preparation
	Physical	Measurement	Other construction

[30] Compare (Marletto 2015) `http://dx.doi.org/10.1098/rsif.2014.1226` on p. 5f:

> A programmable constructor V whose repertoire includes T has the appearance of design if it can execute a recipe for T with a hierarchical structure including several, different sub-recipes, fine-tuned to perform T. Each fine-tuned sub-recipe is performed by a sub-constructor contained in V: the number of fine-tuned sub-recipes performable by V is a measure of V's appearance of design.

17.1.1. *Substrate states.* Attributes of substrates distinguish observable states. A state x together with remaining $\overline{x}$ states provides support for a boolean variable X identified with distinct states $\{x, \overline{x}\}$.

17.2. **Information variable.** A variable can support classical features of computation if it can be prepared and measured because this makes it possible to remember, copy and compare variables. Subsidiary restrictions on preparation and measurement of signals result in variables of super computation which behave like quantized valuations of mixed continuous phenomena.

17.2.1. *Interpretation.* Instances of classical substrates can be produced and interpreted as redundant ensembles of mere copies representing the same attribute, in sets distinguished without particular ordering, or understood like a tuple of digits giving a composite number which could encode a letter or a boolean function.

$$\bigcup(S_i, \dots, S_i) \equiv \bigcup(\mathfrak{I}(S_i), \dots, \mathfrak{I}(S_i)) \tag{153}$$

$$\mathfrak{I}(S_i) \equiv \mathfrak{I}^{\mathbf{T}}(S_i) \equiv \mathfrak{I}^{\sim}(S_i) \equiv S_i \tag{154}$$

17.2.2. *Permutation.* Permutation can reproduce any message tuple given enough acceptable copies of acceptable letters. All binary boolean messages of length k are truncated permutations of strings starting twice as long with a balanced supply of 0 and 1.

$$\left(\bigcup_{x \in S} \{x \to \Pi(x)\}\right)^{\checkmark} \tag{155}$$

17.2.3. *Overclocking.* Alignment of counter circuits and processing circuits enables efficient computation of multiple precessing steps with minimal measures of costs and delay:

> . . . the total time needed to simulate G gates is on the same order as the minimum required time to flip a single bit. ('Fast quantum computation at arbitrarily low energy', Stephen P. Jordan, arxiv 2017, p.2)

17.3. **Conservation of measures.** Construction of direct reward distribution could adopt conservative relations in fixed supply constraints, or prefer unlimited reservoirs of reward released by measures of agreement.

$$\sum_{i=0}^{n-1} \mathcal{M}(S_{\Pi(i)}) \overset{\mathfrak{M}}{\to} \mathbf{M}_{\text{const}} \tag{156}$$

Departing from purely physical constraints would be possible by allowing negative values of currency energy. Acceptable programs for reward constructors would stay compatible if local constraints apply which require net invariance of local consolidations during all transactions.

$$\mathfrak{A}^{-} + \mathfrak{B}^{+} = 0 \tag{157}$$

Tuning rewards according to preferences of participants remains an open issue to resolve by articulation or delegation of contributions.

17.4. **Consistent expansion and contraction.** Conservation of quantity can accommodate expansion by increasing an initial number n of states S. Indices which had not been included for previous summation could represent constant units or arbitrary values assigned by g.

$$\sum_{i=0}^{n+k} S_i \overset{\mathfrak{G}}{\to} \sum_{i=n}^{n+k} g(S_i) + \sum_{i=0}^{n-1} f(S_i) \tag{158}$$

Enumerated expansions can be contracted by removing lowest indices thus shifting ranges of summation.

17.5. **Pharaonic task.** Construction of crypto media constructors depends on auxiliary constructors. Pharaonic tasks construct all constructors not yet available among input substrates ('Constructor Theory' p.27) including cryptographic mechanisms and acceptable formats of regulatory compliance.

> The overhead of programming **P** to be capable of performing $\mathfrak{A}$ is a constant $c(\mathbf{P}, \mathfrak{A})$, independent of how often **P** will then be called upon to perform $\mathfrak{A}$, and which inputs for $\mathfrak{A}$ it is given. ('Constructor Theory' p.32)

Construction of key-less containment models impossibility of transposition for particular programs of construction with one-way message digests h. While reversible in principle such mechanisms may offer intermediate protection before more advanced generations of quantum crypto channels can be deployed.

$$\mathfrak{C} = \{i_1 \to r_1, \ldots\} \tag{159}$$

$$\mathfrak{C}^{\mathbf{T}} = \{r_1 \to i_1, \ldots\} \tag{160}$$

$$\exists \mathfrak{H}_{\star} \wedge \nexists \mathfrak{H}_{\star}^{\mathbf{T}} \tag{161}$$

$$\mathfrak{H}_{\star} = \{i_1 \to h(i_1), \ldots\} \tag{162}$$

17.6. **Contemporary task.** Successful specializations of constructors would have to provide sufficient capacity efficiently while supporting distributed partitioned shard consensus about incremental coordinated transactions. One way to make this possible is endogenous timing of net settlements for transaction substrates.

Rewarding shards for measures of simplification achieved over incremental transactions according to consensus with respect to deterministic constructions of fair net clearing settlement is possible until adversarial manipulations of transaction message digests become possible in later epochs of super-computation able to compute collision of digests in acceptable intervals or even immediately.

$$\sum_{i=0}^{-1+m*n} \mathfrak{T}_i^- = - \sum_{i=0}^{-1+m*n} \mathfrak{T}_i^+ \tag{163}$$

$$\sum_{i=0}^{m-1} \sum_{j=0}^{n-1} \mathfrak{T}_{i+j*m}^- = - \sum_{i=0}^{m-1} \sum_{j=0}^{n-1} \mathfrak{T}_{i+j*m}^+ \tag{164}$$

Parametric concurrency of computation in partitions of transaction substrates according to abstract measures of deterministic para-temporal proximity could achieve unlimited efficient capacity for arbitrary computations with classical information media in principle but might need further modification to handle crypto media which involve super-information states of programmable substrates.

Time enters by deterministic construction of irreversibility and scaled measures of relative proximity which enables consensus without explicit preference for any particular references to conventional measures of time.

17.7. **Common programs for self replication.** A task which calls for cooperation is to replicate a service together which is impossible to sustain for any stand-alone constructor. This formulation emphasizes that trivial constructions are not sufficient to motivate and prolong existence of cooperation — and that such cooperation is impossible unless it can provide features that create competitive advantages for both providers and consumers of functionalities constructed together.

Cooperative roles of contemporary crypto media constructors emerge from synthetic transactions which consolidate transformations among external substrates.

Vehicles of replication for constructors of simplification rewards can be programs licensed to accumulate and distribute compensations sufficient to finance cooperative evolution and selection of modifications.

17.8. **Suggested ingredients.** Suggested ingredients for tasks to replicate cooperative crypto media safely include:

klcdc: key-less count down channels

dubit: direct universal basic income transfer

spincs: shard proximity incremental net clearing settlement

Safety priorities for constructed channels suggest adoption of key-less protections for accounts, and similar containments for autonomous mechanisms.

Social acceptability of emerging coincidence between ecologic and social transitions recommends comprehensive stabilization of individual responsibility for self improvement and cooperative participation by efficient disposition of rents obtained from research and technology.

Populations of vehicles consume energy to replicate, evolve, and select traits of common specialization together. Successful constructors provide replicators with superior ways to partition tasks over common substrates in their environments which provide value for potential sources of attention and energy.

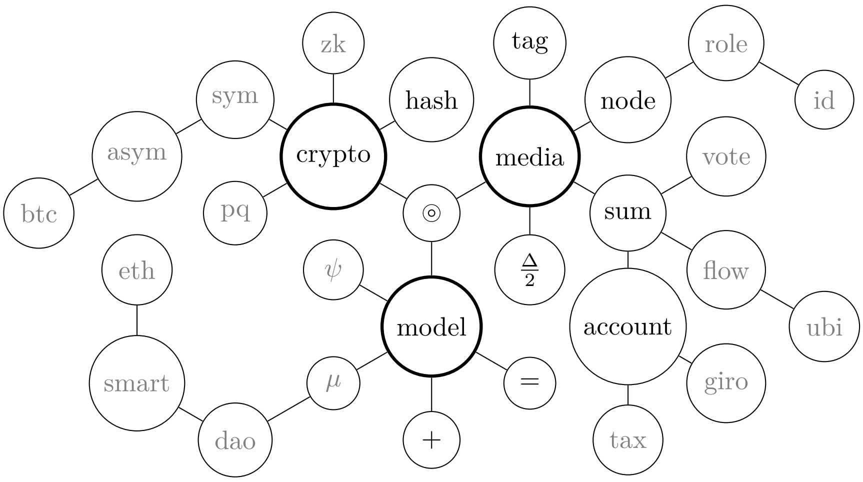

Figure 15. Cooperative cryptomedia machine model and direction

18. Complications

Crypto media may substitute informal stages of processing happening during production, transfer or consumption of content by formal mechanisms.

Cryptography hardens boundaries, isolates channels and zones, which tends to make life more complicated for producers, traders, consumers, regulators and even those not aware of increased risks and opportunities yet.

Vanishing points of future crypto media models can be understood as black holes for attention and other forms of energy drawn from environments which construct possible mutations of established self-replicating vehicles inside of alienating hermetically closed autonomous event horizons.

This introduction claimed ideas about practical solutions to problems of designing and regulating cryptographic media for autonomous mechanisms in messages. It concedes that some of the arguments and explanations are incomplete.

Entangled contexts of crypto media which require further cooperative study, innovation and discussion — include:

AGI: artificial general intelligence singularity containment

Machine: , universality, concurrency, quantum computation

Quantum: channel protocols, computation, mind, pq-crypto algorithms

Crypto: ZK-SNARK, homomorphic computation, multi-show-pseudonymity

Identification: socio-relational, bio-metric

Privacy: watching watching, Biedermeier, fair use

Consensus: model, simulation, convergence, ecologic impact, governance

Rewards: games, mechanisms, design, simulation, fairness

Practice: liveness, finality, capacity, rates of exchange, adoption

Lifestyle: use-case effects on family from cradle to immortatlity

Exchange: arbitrage divergence tape painting, deposit versus atomicity

Security: resilience, response, recovery

Sustainability: proof of work alternatives, ecology, regulation

Business: fees, gas, accounting, exchanges, derivatives

Network: nodes, qu-channels, storage, 5G, orbital constellations

DOS denial of service: defense in depth by layers and redundancy

IOT internet of things: built-in blockchain chips, oracle, 'heracle'

Monitor: interfaces for review of blockchain records, de-anonymization

Legal: liability, license, patent, legal persona, taxation

Society: voting, participation, transfers, public administration

Transition: accommodation of global warming and migration

Sector: goods, rights and services, logistics, tourism, manufacturing

Politics: regulatory capture, liquid democracy issues delegation,

UBI: universal basic income, pervasive rewards, cluster synergy

KYC know your customer: compliance

AML anti money laundering: compliance

API application programing interface: cross platform, developer friendly

UI user interface: user friendly comfort for all ages, cultures, occasions

E...: ecology economy energy exodus errors...

References

Aggarwal, Abhinav and Yue Guo (2018). *A Simple Reduction from State Machine Replication to Binary Agreement in Partially Synchronous or Asynchronous Networks.* Cryptology ePrint Archive, Report 2018/060. Accessed:2018-01-17. URL: `https://eprint.iacr.org/2018/060.pdf` (cit. on p. 52).

Albash, Tameem et al. (2015). *Reexamination of the evidence for entanglement in a quantum annealer.* arxiv 1506.03539 (cit. on p. 58).

Back, A. (2002). *Hashcash - a denial of service counter-measure.* Tech. rep. hashcash.org (cit. on p. 25).

Bernstein, Daniel J. and Tanja Lange (n.d.). *Post-quantum cryptography - dealing with the fallout of physics success.* (Cit. on p. 44).

Bertoni, Guido et al. (n.d.). *The Keccak reference.* keccak.noekeon.org (cit. on p. 44).

Blocki, Jeremiah and Hong-Sheng Zhou (2016). "Designing Proof of Human-work Puzzles for Cryptocurrency and Beyond". In: *Proceedings of the Fourteenth IACR Theory of Cryptography Conference.* B. TCC (cit. on p. 48).

Bonneau, Joseph and Andrew Miller (2014). "Fawkescoin: Bitcoin without public-key crypto". In: *Security Protocols XXII.* Springer, pp. 350–358. URL: `http://www.jbonneau.com/doc/BM14-SPW-fawkescoin.pdf` (cit. on p. 26).

Boyd, Colin et al. (2017). *Key Recovery: Inert and Public.* `http://eprint.iacr.org/2017/243`. Accessed: 2017-03-22. URL: `http://eprint.iacr.org/2017/243.pdf` (cit. on p. 50).

Bracha, Gabriel and Sam Toueg (1985). "Asynchronous Consensus and Broadcast Protocols". In: vol. 32. 4. Citeseer, pp. 824–840. URL: `https://pdfs.semanticscholar.org/130c/e1bcd496a7b9192f5f53dd8d7ef626e40675.pdf` (cit. on p. 52).

Brânzei, Simina, Erel Segal-Halevi, and Aviv Zohar (2017). *How to Charge Lightning.* arXiv:1712.10222. Accessed:2018-01-03. URL: `https://arxiv.org/pdf/1712.10222.pdf` (cit. on p. 28).

Bünz, Benedikt et al. (n.d.). *Bulletproofs: Short Proofs for Confidential Transactions and More* (cit. on p. 25).

Buss, Samuel R (1998). "An Introduction to Proof Theory". In: *Handbook of Proof Theory*. Ed. by Samuel R Buss. Elsevier Science, pp. 3–74 (cit. on p. 53).

Buterin, Vitalik (2016). *Chain Interoperability*. https://static1.squarespace.com/static/55f73743e4b051cfcc0b02cf/t/5886800ecd0f68de303349b1/1485209617040/Chain+Interoperability.pdfi. Accessed: 2017-03-25. URL: https://static1.squarespace.com/static/55f73743e4b051cfcc0b02cf/t/5886800ecd0f68de303349b1/1485209617040/Chain+Interoperability.pdf (cit. on p. 50).

Cachin, Christian (2016). *Architecture of the Hyperledger Blockchain Fabric*. https://www.zurich.ibm.com/dccl/papers/cachin_dccl.pdf. Accessed: 2016-08-10. URL: https://www.zurich.ibm.com/dccl/papers/cachin_dccl.pdf (cit. on p. 28).

Camenisch, Jan and Anna Lysyanskaya (n.d.). *A Signature Scheme with Efficient Protocols* (cit. on p. 25).

Cardelli, Luca and Andrew D. Gordon (2003). *Mobile Ambients* (cit. on pp. 48, 53).

Chepurnoy, Alexander, Vasily Kharin, and Dmitry Meshkov (2018). *A Systematic Approach To Cryptocurrency Fees*. Cryptology ePrint Archive, Report 2018/078. Accessed:2018-01-22. URL: https://eprint.iacr.org/2018/078.pdf (cit. on p. 49).

Christin, Nicolas (2013). "Traveling the Silk Road: A measurement analysis of a large anonymous online marketplace". In: *Proceedings of the 22nd World Wide Web Conference (WWW'13)*. Rio de Janeiro, Brazil, pp. 213–224. URL: https://www.andrew.cmu.edu/user/nicolasc/publications/Christin-WWW13.pdf (cit. on p. 50).

Clark, Jeremy et al. (2014). "On Decentralizing Prediction Markets and Order Books". In: *WEIS*. URL: http://users.encs.concordia.ca/~clark/papers/2014_weis.pdf (cit. on pp. 43, 49).

Dagher, Gaby G et al. (2015). "Provisions: Privacy-preserving proofs of solvency for Bitcoin exchanges". In: *Proceedings of the 22nd ACM SIGSAC Conference on Computer and Communications Security*. ACM, pp. 720–731. URL: http://users.encs.concordia.ca/~clark/papers/2015_ccs.pdf (cit. on p. 49).

Dai, W. (1998). *b-money*. Tech. rep. www.weidai.com/bmoney.txt (cit. on p. 25).

Decker, Christian et al. (2015). "Making Bitcoin Exchanges Transparent". In: *Computer Security–ESORICS 2015*. Springer, pp. 561–576. URL: http://www.tik.ee.ethz.ch/file/b89cb24ad2fa4e7ef01426d318c9b98b/decker2015making.pdf (cit. on p. 49).

Deutsch, David (2012). *Constructor Theory*. Future of Humanity Institute. University of Oxford (cit. on p. 59).

Deutsch, David and Chiara Marletto (2015). "Constructor theory of information." In: *Proc. R. Soc. A* 471.20140540 (cit. on p. 59).

Fyookball, Jonald (2017). *Mathematical Proof That the Lightning Network Cannot Be a Decentralized Bitcoin Scaling Solution.* medium.com jonaldfyookball (cit. on p. 28).

Gaetani, Edoardo et al. (2017). "Blockchain-based database to ensure data integrity in cloud computing environments". In: URL: `http://ceur-ws.org/Vol-1816/paper-15.pdf` (cit. on p. 28).

Garay, Juan, Aggelos Kiayias, and Nikos Leonardos (2015). "The bitcoin backbone protocol: Analysis and applications". In: *Advances in Cryptology-EUROCRYPT 2015.* Springer, pp. 281–310. URL: `http://courses.cs.washington.edu/courses/cse454/15wi/papers/bitcoin-765.pdf` (cit. on p. 52).

Gipp, Bela et al. (2017). "CryptSubmit: Introducing Securely Timestamped Manuscript Submission and Peer Review Feedback using the Blockchain". In: *JVDL.* URL: `https://www.gipp.com/wp-content/papercite-data/pdf/gipp2017b.pdf` (cit. on p. 44).

Hadfield-Menell, Dylan et al. (2016). *The Off-Switch Game.* arXiv 1611.08219v1 (cit. on p. 29).

Hildenbrandt, Everett et al. (2017). *KEVM: A Complete Semantics of the Ethereum Virtual Machine* (cit. on p. 52).

Hiley, B J (2015). *The Algebraic Way.* arXiv 1602.0607 (cit. on p. 56).

Kiktenko, E. O. et al. (2017). *Quantum-secured blockchain.* arXiv:1705.09258. Accessed: 2017-06-29. URL: `https://arxiv.org/pdf/1705.09258.pdf` (cit. on p. 44).

Krafft, Peter M, Nicolás Della Penna, and Alex Pentland (2018). *An Experimental Study of Cryptocurrency Market Dynamics.* Peter Krafft, Nicolás Della Penna, Alex Pentland. (2018). An Experimental Study of Cryptocurrency Market Dynamics. ACM CHI Conference on Human Factors in Computing Systems (CHI). Accessed:2018-01-22. DOI: `10.1145/3173574.3174179`. eprint: `1801.05831`. URL: `http://arxiv.org/pdf/1801.05831.pdf` (cit. on p. 49).

Lalley, Steven P. and E. Glen Weyl (2015). *Quadratic Voting* (cit. on p. 48).

Marletto, Chiara (2015). "Constructor theory of life". In: *J. R. Soc. Interface* 12.20141226 (cit. on p. 59).

Mastroeni, Matthew (2009). "An Axiomatic Formulation of Quantum Mechanics". PhD thesis. Ithaca (cit. on p. 56).

Mavridou, Anastasia and Aron Laszka (2017). *Designing Secure Ethereum Smart Contracts: A Finite State Machine Based Approach.* arXiv:1711.09327. Accessed:2017-12-04. URL: `https://arxiv.org/pdf/1711.09327.pdf` (cit. on p. 52).

McConaghy, Trent et al. (2016). *BigchainDB: A Scalable Blockchain Database.* ascribe GmbH. Berlin Germany (cit. on p. 28).

McCorry, Patrick, Sarah Meiklejohn, and George Danezis (2017). *SoK Consensus in the Age of Blockchains.* arxiv 1711.03936v2 (cit. on p. 28).

Meredith, L.G. and Matthias Radestock (n.d.). *Namespace logic: A logic for a reflective higher-order calculus* (cit. on p. 53).

Merkle, R. C. (1980). "Protocols for public key cryptosystems". In: *Proc. 1980 Symposium on Security and Privacy*. IEEE Computer Society, pp. 122–133 (cit. on pp. 22, 25).

Miller, Andrew et al. (2017). *Sprites: Payment Channels that Go Faster than Lightning*. `https://arxiv.org/pdf/1702.05812.pdf`. Accessed: 2017-03-22. URL: `https://arxiv.org/pdf/1702.05812.pdf` (cit. on p. 28).

Moon, J. W. (1966). "On subtournaments of a tournament". In: *Canadian Mathematical Bulletin*, pp. 297–301 (cit. on p. 46).

Moore, Tyler and Nicolas Christin (2013). "Beware the Middleman: Empirical Analysis of Bitcoin-Exchange Risk". In: *Proceedings of IFCA Financial Cryptography'13*. Okinawa, Japan. URL: `https://www.andrew.cmu.edu/user/nicolasc/publications/MC-FC13.pdf` (cit. on p. 49).

Möser, Malte and Rainer Böhme (2015). "Trends, tips, tolls: A longitudinal study of Bitcoin transaction fees". In: *International Conference on Financial Cryptography and Data Security*. Springer, pp. 19–33. URL: `http://fc15.ifca.ai/preproceedings/bitcoin/paper_8.pdf` (cit. on p. 49).

Nakamoto, Satoshi (2008). *Bitcoin: A Peer-to-Peer Electronic Cash System* (cit. on pp. 26, 28).

Navarro, Gonzalo and Rodrigo Paredes (2006). "Optimal incremental sorting." In: *8th Workshop on Algorithm Engineering and Experiments and 3rd Workshop on Analytic Algorithmics and Combinatorics (ALENEX - ANALCO - 06)*. SIAM Press, pp. 171–182 (cit. on p. 46).

Neumann, John von (1976). "A Model of General Economic Equilibrium". In: *Mathematical Theory of Expanding and Contracting Economies*. Ed. by Oskar Morgenstern and Gerald L Thompson. Lexington Books, pp. 239–249 (cit. on p. 42).

Noyes, Charles (2016). *BitAV: Fast Anti-Malware by Distributed Blockchain Consensus and Feedforward Scanning*. `https://arxiv.org/abs/1601.01405`. Accessed: 2017-05-09. URL: `https://arxiv.org/pdf/1601.01405.pdf` (cit. on p. 50).

Peters, Gareth W and Efstathios Panayi (2016). "Understanding Modern Banking Ledgers through Blockchain Technologies: Future of Transaction Processing and Smart Contracts on the Internet of Money". In: *Banking Beyond Banks and Money*. Springer, pp. 239–278. URL: `https://pdfs.semanticscholar.org/0a84/a077ada2acb6918e7764fafcd28f667dae28.pdf` (cit. on p. 49).

Poon, Joseph and Vitalik Buterin (2017). *Plasma: Scalable Autonomous Smart Contracts* (cit. on p. 28).

Poon, Joseph and Thaddeus Dryja (2016). *The bitcoin lightning network*. `https://lightning.network/lightning-network-paper.pdf`. Accessed: 2016-07-07.

URL: https://lightning.network/lightning-network-paper.pdf (cit. on p. 28).

Popov, Serguei (2016). *The Tangle*. Jinn Labs (cit. on p. 28).

Riksbank, Sverige (2018). *The Riksbank's e-krona project. Action plan for 2018* (cit. on p. 20).

Rivest, R, A Shamir, and L Adleman (1978). "A Method for Obtaining Digital Signatures and Public-Key Cryptosystems". In: *Communications of the ACM* 21.2, pp. 120–126 (cit. on p. 23).

Roçu, Grigore (2017). "Matching Logic". In: *Logic Methods in Computer Science LMCS* to appear (cit. on p. 52).

Schneider, Fred B (1990). "Implementing fault-tolerant services using the state machine approach: A tutorial". In: vol. 22. 4. ACM, pp. 299–319. URL: http://www-users.cselabs.umn.edu/classes/Spring-2014/csci8980-sds/Papers/ProcessReplication/p299-schneider.pdf (cit. on p. 52).

Schreiber, Michael (2017). *Play for Privacy by Proof of Play Whitepaper*. Lab10.coop (cit. on p. 47).

Sikorski, Janusz J, Joy Haughton, and Markus Kraft (2017). "Blockchain technology in the chemical industry: machine-to-machine electricity market". In: *Applied Energy*. Vol. 195. Elsevier, pp. 234–246. URL: https://como.cheng.cam.ac.uk/preprints/c4e-Preprint-178.pdf (cit. on p. 43).

Soska, Kyle and Nicolas Christin (2015). "Measuring the longitudinal evolution of the online anonymous marketplace ecosystem". In: *24th USENIX Security Symposium (USENIX Security 15)*, pp. 33–48. URL: https://www.usenix.org/system/files/conference/usenixsecurity15/sec15-paper-soska.pdf (cit. on p. 49).

Sousa, João, Alysson Bessani, and Marko Vukolić (2017). *A Byzantine Fault-Tolerant Ordering Service for the Hyperledger Fabric Blockchain Platform*. arXiv:1709.06921. Accessed:2017-09-25. URL: https://arxiv.org/pdf/1709.06921.pdf (cit. on p. 28).

Spencer Brown, G (1961). *An algebra for the natural numbers*. London (cit. on p. 52).

Stoffers, Martin (2017). "Trustworthy Provenance Recording using a blockchain-like database". PhD thesis. Leipzig University. URL: http://elib.dlr.de/111772/1/thesis.pdf (cit. on p. 28).

Teutsch, Jason and Christian Reitwießner (2017). *A scalable verification solution for blockchains*. https://truebit.io/. Accessed:2017-10-06. URL: https://people.cs.uchicago.edu/~teutsch/papers/truebit.pdf (cit. on p. 48).

Wilde, Mark M (2015). *From Classical to Quantum Shannon Theory*. arXiv. 1106.1445 (cit. on pp. 56, 57).

Wolfram, S. (2002). *A New Kind of Science*. Champaign IL: Wolfram Media (cit. on p. 52).

Wood, Gavin (2017). *Ethereum: A secure decentralised generalised transaction ledger EIP-150 REVISION (759dccd - 2017-08-07)*. Accessed: 2018-01-03. URL: `https://ethereum.github.io/yellowpaper/paper.pdf` (cit. on p. 26).

Zhang, Fan et al. (2016). "Town Crier: An Authenticated Data Feed for Smart Contracts". In: *Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security*. ACM, pp. 270–282. URL: `https://eprint.iacr.org/2016/168.pdf` (cit. on p. 49).

Zhang, Huang et al. (2017). *Anonymous Post-Quantum Cryptocash*. Cryptology ePrint Archive, Report 2017/716. Accessed: 2017-08-03. URL: `http://eprint.iacr.org/2017/716.pdf` (cit. on p. 44).

Zhang, Xing-Zhou, Jing-Jie Liu, and Zhi-Wei Xu (2015). "Tencent and Facebook Data Validate Metcalfe's Law." In: *Journal of computer science and technology* (cit. on p. 43).

MICHAEL.SCHREIBER@LAB10.COOP