Journal of
Risk and Financial Management

MDPI

Concept Paper

Blockchain Economical Models, Delegated Proof of Economic Value and Delegated Adaptive Byzantine Fault Tolerance and their implementation in Artificial Intelligence BlockCloud

Qi Deng [1,2,3]

[1] Accounting and Finance Group, International Business School Suzhou, Xi'an Jiaotong-Liverpool University, Suzhou 215123, China; qi.deng@cofintelligence.ai
[2] Cofintelligence Financial Technology Ltd., Hong Kong, China
[3] Cofintelligence Financial Technology Ltd., Shanghai 201106, China

Received: 12 October 2019; Accepted: 21 November 2019; Published: 25 November 2019

Abstract: The Artificial Intelligence BlockCloud (AIBC) is an artificial intelligence and blockchain technology based large-scale decentralized ecosystem that allows system-wide low-cost sharing of computing and storage resources. The AIBC consists of four layers: a fundamental layer, a resource layer, an application layer, and an ecosystem layer (the latter three are the collective "upper-layers"). The AIBC layers have distinguished responsibilities and thus performance and robustness requirements. The upper layers need to follow a set of economic policies strictly and run on a deterministic and robust protocol. While the fundamental layer needs to follow a protocol with high throughput without sacrificing robustness. As such, the AIBC implements a two-consensus scheme to enforce economic policies and achieve performance and robustness: Delegated Proof of Economic Value (DPoEV) incentive consensus on the upper layers, and Delegated Adaptive Byzantine Fault Tolerance (DABFT) distributed consensus on the fundamental layer. The DPoEV uses the knowledge map algorithm to accurately assess the economic value of digital assets. The DABFT uses deep learning techniques to predict and select the most suitable BFT algorithm in order to enforce the DPoEV, as well as to achieve the best balance of performance, robustness, and security. The DPoEV-DABFT dual-consensus architecture, by design, makes the AIBC attack-proof against risks such as double-spending, short-range and 51% attacks; it has a built-in dynamic sharding feature that allows scalability and eliminates the single-shard takeover. Our contribution is four-fold: that we develop a set of innovative economic models governing the monetary, trading and supply-demand policies in the AIBC; that we establish an upper-layer DPoEV incentive consensus algorithm that implements the economic policies; that we provide a fundamental layer DABFT distributed consensus algorithm that executes the DPoEV with adaptability; and that we prove the economic models can be effectively enforced by AIBC's DPoEV-DABFT dual-consensus architecture.

Keywords: blockchain; BlockCloud; Artificial Intelligence; consensus algorithms

1. Introduction

After the outburst of the 2008 financial crisis, Satoshi Nakamoto publishes a paper titled "Bitcoin: A Point-to-Point Electronic Cash System," symbolizing the birth of cryptocurrencies (Nakamoto 2008). Vitalik Buterin (Buterin 2013) improves upon Bitcoin with a public platform that provides a Turing-complete computing language, the Ethereum, which introduces the concept of smart contracts, allowing anyone to author decentralized applications where they can create their own arbitrary rules for ownership, transaction formats, and state transition functions. Bitcoin and Ethereum

are the first batch of practical blockchains that make use of distributed consensus, decentralized ledger, data encryption and economic incentives afforded by the underlying blockchain technology. Essentially, the blockchain technology enables trustless peer-to-peer transactions and decentralized coordination and collaboration among unrelated parties, providing answers to many challenges unsolvable by the traditional centralized institutions, including not but limited to, low efficiency, high cost, and low security.

Bitcoin, the pioneer of the blockchain's distributed ledger and distributed database revolution, is widely regarded as "Blockchain 1.0." "Blockchain 2.0" is represented by Ethereum, which adds a smart contract mechanism to the Bitcoin foundation. The blockchain is entering its 3.0 era: it seeks to create ecosystems with a proliferation of application scenarios with no apparent scope limitation. It has the potential to become the low-level protocol of the "Internet of Everything," and is particularly friendly to applications that require process management, such as supply chain finance, transportation and logistics, property right certification, charity and donation management, etc.

"Blockchain 3.0" is not without challenges. To begin with, as of today, there are only a very few choices of "proven" blockchain consensus algorithms, yet there are literally unlimited number of blockchain applications. To make things worse, each and every existing blockchain employs only one (predetermined) consensus algorithm. As a result, a vast majority of applications would have to rely upon consensus algorithms that are not optimized for them, greatly reducing their efficiency and effectiveness.

Furthermore, while the blockchain starts out as a bottom-layer technology, its true promise goes beyond technology. As "Blockchain 3.0" seeks to create ecosystems with numerous interconnected applications that perform at the highest level collectively; these ecosystems are thus "economies" in the digital world, and applications their "agents." Therefore, a well-designed "Blockchain 3.0" implementation needs to include economic models that provide "rules" to the ecosystem it creates, especially a macroeconomic monetary policy that enforces real-time synchronization between "economic growth (within the ecosystem)" and "money (token) supply". To that end, none of the existing self-claimed "Blockchain 3.0" solution is yet successful.

We propose the Artificial Intelligence BlockCloud (AIBC), an Artificial Intelligence (AI) based blockchain ecosystem. The AIBC is an attempt at addressing the aforementioned challenges "Blockchain 3.0" faces. Anchored on the principles of decentralization, scalability, and controllable cost, the AIBC seeks to provide a perfect platform for Distributed SOLutions (DSOLs) by leveraging the basic blockchain technology and sharing computing power and storage space system-wide.

The AIBC emphasizes on ecosystem expansion. Our goal is to build a cross-application distributed and trusted ecosystem. Based on our economic model, the upper layer Delegated Proof of Economic Value (DPoEV) incentive consensus enables connections among diverse computing, data and information entities. The value in the AIBC is essentially the knowledge that existed in and accumulated by participating entities. The entities then participate in exchanges of values through resource sharing activities, facilitated by token (unit of economic value) transfers. The benefits of the AIBC are then value creation and exchange across entities.

The AIBC also stresses on application support. It provides a flexible technical support infrastructure of distributed services for large business scenarios. Its AI-based fundamental layer Delegated Adaptive Byzantine Fault Tolerance (DABFT) distributed consensus allows individualized real-time customization of protocols. Thus application scenarios in the AIBC ecosystem can be optimized according to differentiated requirements of multiple entities on a public chain that provides common bottom-layer services.

Our contribution is four-fold: that we develop a set of innovative economic models governing the monetary, trading and supply-demand policies in the AIBC; that we establish an upper-layer DPoEV incentive consensus algorithm that implements the economic policies; that we provide a fundamental layer DABFT distributed consensus algorithm that executes the DPoEV with adaptability;

and that we prove the economic models can be effectively enforced by AIBC's DPoEV-DABFT dual-consensus architecture.

The rest of the paper is organized as follow: Section 2 provides an overview of the AIBC, Section 3 surveys the existing proven blockchain consensus algorithms, Section 4 presents the AIBC economic models, Sections 5 and 6 give the details of the DPoEV and DABFT consensus algorithms, and Section 7 concludes the paper.

2. AIBC Overview

2.1. AIBC Key Innovation

The AIBC is an Artificial Intelligence and blockchain technology based decentralized ecosystem that allows resource sharing among participating nodes. The primary resources shared are the computing power and storage space. The goals of the AIBC ecosystem are efficiency, fairness, and legitimacy.

The key innovation of the AIBC is separating the fundamental (blockchain) layer distributed consensus and the application layer incentive mechanism. The AIBC implements a two-consensus scheme to enforce upper-layer economic policies and achieve fundamental layer performance and robustness: The DPoEV incentive consensus to create and distribute award on the application and resource layers; the DABFT distributed consensus for block proposition, validation and ledger recording on the fundamental layer.

The DPoEV consensus is derived from a model of cooperative economics (macroeconomics, microeconomics, and international trade). It uses the knowledge map algorithm (a branch of Artificial Intelligence) to accurately assess the economic value of digital assets (knowledge).

The DABFT is the fundamental layer distributed consensus algorithm. It improves upon the ADAPT algorithm (Bahsoun et al. 2015) and uses deep learning (a branch of Artificial Intelligence) techniques to predict and dynamically select the most suitable Byzantine Fault Tolerant (BFT) algorithm for the current application scenario in order to achieve the best balance of performance, robustness and security. The DABFT is currently the most adaptive distributed consensus solution that meets various technical needs among public chains.

2.2. AIBC Architecture

The AIBC consists of four layers: a fundamental layer conducts the essential blockchain functions, a resource layer that provides the shared services, an application layer that initiates a request for resources, and an ecosystem layer that comprises physical/virtual identities that own or operate nodes:

1. The fundamental layer (or blockchain layer) that conducts the essential blockchain functions, namely distributed consensus-based block proposition, validation, and ledger recording. The nodes delegated to perform these fundamental blockchain services are the super nodes.
2. The resource layer that provides the essential ecosystem services, namely, computing power and storage space. The AIBC ecosystem is based on the concept that resources are to be shared, and these resources are provided by the computing nodes and storage nodes. While their functions are different, the computing nodes and storage nodes can physically or virtually be collocated or coincide.
3. The application layer that requests resources. Each application scenario is initiated by a tasking node. In the AIBC ecosystem, tasking nodes are the ones that have needs for computing power and storage space, thus it is their responsibility to initiate tasks, which in turn drive the generation of economic value.
4. The ecosystem layer that comprises physical/virtual entities that own or operate the nodes. For example, a tasking node can be a financial trading firm that needs resources from a number of computing nodes, which can be other trading firms or server farms that provides computing power.

Figure 1 illustrates the AIBC layer structure and corresponding consensus algorithms.

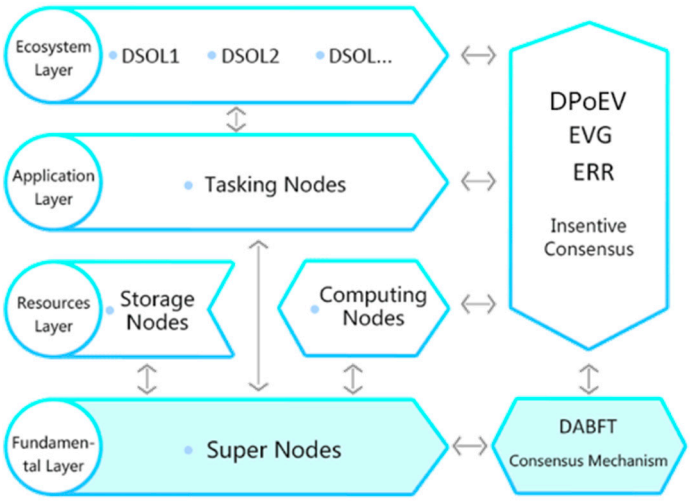

Figure 1. AIBC Layer Structure and Consensus Algorithms. The AIBC consists of four layers: A fundamental layer conducts the essential blockchain functions, a resource layer that provides the shared services, an application layer that initiates a request for resources, and an ecosystem layer that comprises physical/virtual identities that own or operate nodes.

2.3. AIBC Two-Consensus Implementation

The AIBC layers have distinguished responsibilities and thus performance and robustness requirements. For example, once a task is initiated, the application and resource layers are primarily concerned with delivering resources and distributing reward. Therefore, these layers need to follow the economic policies strictly and run on a deterministic and robust protocol, but not necessarily a high-performance one (in terms of speed). On the other hand, the fundamental layer is the workhorse providing basic blockchain services such as consensus building, block proposition and validation, transaction tracking, and ledger recording. Therefore it needs to follow an adaptive protocol with high throughput without sacrificing robustness.

As such, the AIBC implements a two-consensus approach: the DPoEV incentive consensus to create and distribute awards on the application and resource layers, and the DABFT distributed consensus responsible for blockchain functions on the fundamental layer. The DPoEV is deterministic and does not necessarily require high-performance as most of the application scenarios do not demand real-time reward distribution. On the other hand, the DABFT has to be real-time and adaptive, as block validation and record bookkeeping need to be done quickly and robustly.

The two-consensus implementation is a distinguishing feature of the AIBC. It enforces upper-layer economic policies and bottom-layer consensus building, a perfect combination for resource-sharing application scenarios. On the other hand, most of the existing and proposed public chains adopt one-consensus schemes, which do not provide flexibility in performance and robustness tradeoff and are vulnerable against risks such as 51%-attacks.

3. Review of Literature and Practice on Major Consensus Algorithms

Most of the existing blockchains adopt one-consensus schemes, we survey the ones that have actually been used in "mainstream" blockchains in the section. The majority of these consensus algorithms are proof-based (PoX), and some of them are vote-based. While most of the vote-based consensus algorithms are flavors of the Byzantine Fault Tolerance (BFT), a noticeable few utilize the Crash Fault Tolerance (CFT) approach.

3.1. Proof-Based Consensus Algorithms

3.1.1. PoW (Proof of Work) Workload Proof Consensus

The PoW consensus behind Bitcoin plays the zero-sum game of SHA256 hash for the miners to win ledger recording privilege. With the increased level of difficulty on block mining, the PoW consumes a tremendous amount of computing power (and electricity) with a great reduction of throughput. Even worse, the higher number of miners, the higher level of difficulty of mining, and the lower level of probability for a miner to win ledger recording privilege, which induces a yet higher level of energy consumption and longer latency. This is the key reason why Ethereum has long considered the use of the PoS (Proof-of-Stake) algorithm, Casper, instead of the PoW. Therefore, from the perspective of mining speed and cost, the PoW is not conducive to long-term and rapid development of blockchain based ecosystems. Other mainstream PoW-based blockchains include the Litecoin (LTC 2018).

3.1.2. PoS (Proof of Stake) Equity Proof Consensus and DPoS

The PoS consensus measures the amount and age of wealth in the ecosystem in order to grant ledger recording privilege (Buterin 2013). PeerCoin (King and Nadal 2012), NXT (NXT 2015), as well as the Ethereum's Casper implementation (Buterin 2014), adopt the PoS. Although the PoS consumes a much lower level of energy than the PoW, it amplifies the impact of accumulated wealth, as such, in a PoS ecosystem, participants with a higher level of wealth can easily monopolize ledger recording. In addition, block confirmations are probabilistic, not deterministic, thus in theory, a PoS ecosystem may have exposure to other attacks. Therefore, from the perspective of miner composition, the PoS is not conducive to the interests of participants in the ecosystem.

The DPoS is derived from the PoS, and is being used by EOS (EOS 2018). The main difference is that, in the DPoS regime, all asset holders elect a number of representatives, and delegate consensus building to them. The regulatory compliance, performance, resource consumption, and fault tolerance of the DPoS are similar to that of the PoS. The key advantage of the DPoS is that it significantly reduces the number of nodes for block verification and ledger recording, thus is capable of achieving consensus in seconds. However, the DPoS inherits the PoS's major shortcomings, that it is probabilistic and does not acknowledge monopolization.

3.1.3. PoI (Proof of Importance) Importance Proof Consensus

The PoI introduces the concept of account importance, which is used as a measure to allocate ledger recording privilege (NEM 2018). The PoI partly resolves the wealth monopolization dilemma of the PoS. However, it exposes to a nothing-at-stake scenario, which makes cheating rather low cost. Therefore, the PoI deviates from the AIBC goal of legitimacy and the DPoEV requirement of "rule of relevancy."

3.1.4. PoD (Proof of Devotion) Contribution Proof Consensus

The PoD introduces the concept of contribution and awards ledger recording privilege according to contributions of accounts (NAS 2018). However, the PoD uses otherwise meaningless pseudo-random numbers to determine ledger recording privilege among participants, which is not consistent with the concept of utilizing resources only for meaningful and productive endeavors. Moreover, due to the limitation of design, the PoD cannot achieve desired level of efficiency.

3.1.5. PoA (Proof of Authority) Identity Proof Consensus

The PoA is similar to the PoS (VET 2018). However, unlike the POS, the PoA nodes are not required to hold assets to compete for ledger recorder privilege, rather, they are required to be known and verified identities. This means that nodes are not motivated to act in their own interest. The PoA is cheaper, more secure and offers higher TPS than the PoS.

3.1.6. PoET (Proof of Elapsed Time) Sample Size Proof Consensus

The PoET (Intel 2017a) is used in Intel's Hyperledger Sawtooth blockchain, it utilizes a "trusted execution environment" to improve on the efficiency and reduce the power consumption of the PoW. The PoET stochastically elects individual nodes to execute requests at a given target rate. These nodes sample an exponentially distributed random variable and wait for an amount of time dictated by the sample. The node with the smallest sample wins the election.

3.1.7. PoSpace (Proof of Space) Disk Space Proof Consensus

The PoSpace (Park et al. 2015) was proposed to improve the inefficient mining of the PoW and inexpensive mining of the PoS, and is used in the SpaceMint blockchain. To mine blocks in a PoSpace blockchain, miners invest disk space instead of computing power, and dedicating more disk space yields a proportionally higher expectation of successfully mining a block.

3.2. Vote-Based Consensus Algorithms

3.2.1. BFT Distributed Consistency Consensus Algorithms

All the above proof-based consensus algorithms are susceptible to a variety of attacks, especially variations of the 51% attack, which can be partially addressed by vote-based consensuses.

The BFT provides $F = \lfloor (N-1)/3 \rfloor$ fault tolerance. The possible solution to the Byzantine problem is that consistency can be achieved in the case of $N \geq 3F + 1$, where N is the total number of validators, and F is the number of faulty validators. After information is exchanged between the validators, each validator has a list of information obtained, and the information that exists in a 2/3 majority of validators prevails. The BFT advantage is that consensus can be reached efficiently with safety and stability (Lamport et al. 1982; Driscoll et al. 2003). The disadvantages of the BFT are that, when one third or more of the validators stop working, the system will not be able to provide services; and that when one third or more of the validators behave maliciously and all nodes are divided into two isolated islands by chance, the malicious validators can fork the system, though they will leave cryptographic evidence behind. The decentralization level of the BFT is not as high as the other consensuses, thus it is more suitable for multi-centered application scenarios.

A high-performance variant of the BFT, the PBFT (Practical BFT), can achieve a consensus delay of two to five seconds, which satisfies the real-time processing requirements of many commercial applications (Castro and Liskov 2002). The PBFT's high consensus efficiency enables it to meet high-frequency trading needs. The PBFT is a "permissioned" blockchain among a set of known, identified participants; it provides a way to secure the interactions among a group of entities that have a common goal but do not fully trust each other, such as businesses that exchange funds, goods, or information. Thus it may not be suitable for public blockchains. Also, the PBFT PBFT is a network-intensive algorithm, thus not scalable to large networks. Hyperledger Fabric utilizes the PBFT (Androulaki et al. 2018).

The DBFT (Delegated BFT) improved upon the PBFT, and is to select the validators by their stake in the ecosystem, and the selected validators then reach consensus through the BFT algorithm (BTS 2018; NEO 2018). The DBFT has many improvements over the BFT. It updates the BFT's client/service architecture to a peer-node mode suitable for P2P networks. It evolves from static consensus to dynamic consensus that validators can dynamically enter and exit. It incorporates a voting mechanism based on the validators' stakes for ledger recording. It also introduces the usage of a digital certificate, which resolves the issue of validator identity authentication. The DBFT has many desirable features, such as specialized bookkeepers, tolerance of any type of error, and no bifurcation. Just as with the BFT, when one third or more of the validators behave maliciously and all nodes are divided into two isolated islands by chance, the malicious validators can fork the system, though they will leave cryptographic evidence behind.

The Ripple (Schwartz et al. 2014) and its newer version, the XRP (Chase and MacBrough 2018), are proposed to reduce latency of the more basic BFT algorithms, while still maintain robustness in the face of Byzantine failures. The Ripple/XRP is used in the Ripple blockchain.

3.2.2. CFT Distributed Consistency Consensus Algorithms

The Raft (Ongaro and Ousterhout 2014) is a leader-based consensus algorithm. It defines an election process whereby a leader is established and recognized by all followers. Only one nod (leader) publishes blocks, which are then validated and agreed on by the other nodes in the network (the followers).

The Raft is a Crash Fault Tolerant (CFT) algorithm, i.e., it is not a BFT. It continues to make progress as long as a majority of its nodes are available. However, the Raft only guarantees safety and availability under non-Byzantine conditions, which makes it ill-suited for networks that require BFT. It is implemented as Sawtooth Raft in Intel's Hyperledger Sawtooth as one of the consensus engines (Intel 2017b).

3.3. Flaws of Existing Consensus Algorithms

In this section we survey a number of the most popular consensus algorithms. As the blockchain is a very dynamic field, more comprehensive surveys are available for interested readers (e.g., Nguyen and Kim 2017; Wang et al. 2019)

All existing consensus algorithms function well as standalone protocols. However, all the blockchains that are built on these consensus algorithms do not offer balanced (high) performance and (resilient) robustness. The reason is simple, that different blockchain layers have conflict performance measures that cannot be satisfied by any single consensus algorithm.

For example, the AIBC layers have distinguished responsibilities and thus performance and robustness requirements. Once a task is initiated, the application and resource layers are primarily concerned with delivering resources and distributing reward. Therefore, these layers need to follow the economic policies strictly and run on a deterministic and robust protocol, but not necessarily a high-performance one (in terms of speed). On the other hand, the fundamental layer is the workhorse providing basic blockchain services such as consensus building, block proposition and validation, transaction tracking, and ledger recording. Therefore it needs to follow a protocol with high throughput without sacrificing robustness. As such, a multi-protocol AND adaptive approach is necessary, which gives rise to AIBC's DPoEV-DABFT dual-consensus architecture.

While academic literature addressing the above concerns is still lacking, there are some noticeable efforts, mainly on explaining the mechanisms of blockchains from a scholarly perspective. (Tschorsch and Scheuermann 2016) studies the Bitcoin protocol, as well as its building blocks and its applications, for the purpose of establishing academic research directions. (Herlihy 2018) provides a tutorial on the basic notions and mechanisms underlying blockchains, stressing that blockchains are not mirrored images of distributed computing. (Sultan et al. 2018) aim to address the gap and presents an overview of blockchain technology, identifying blockchain's key characteristics with discussions on blockchain applications.

Some academic studies provide alternatives of consensus algorithms. (Bonneau et al. 2015) seek to identify key components of Bitcoin's design that can be decoupled to enable a more insightful analysis of Bitcoin's properties and future stability. Other academic literature provides guidance on how to design or improve blockchains in order to use their respective consensus algorithms more effectively. (Li et al. 2018) conduct a study on security threats to blockchains, survey the corresponding real attacks by examining popular blockchains, and review security enhancement solutions that could be used in future blockchain development. (Belotti et al. 2019) come up with a "vademecum" guiding developers to the right decisions on when, which and how to adopt blockchains and consensus algorithms.

Other scholars focus on impact of business logics on blockchain implementation. (Governatori et al. 2018) analyze on how concepts pertinent to legal contracts can influence certain aspects of their digital implementation through distributed ledger technology and smart contracts.

Again, academic research on dual-consensus architecture is warranted.

3.4. Multi-Protocol Consensus Algorithms

While a multi-protocol and adaptive approach is necessary, it has not been studied thoroughly in the academic circle. Neither such an approach is readily available in any of today's blockchains.

There are "dual-token" blockchains. Ontology (ONT 2017) is a public blockchain for a peer-to-peer trust network. It has two tokens: ONT is tied to its consensus algorithm (VBFT[1]) and is the token that represents "ownership" of a stake of the chain itself; and ONG is the measure of payment for services. LuckyBlock (LKC 2018) is a blockchain designed specifically for decentralized gaming with two tokens: LKC follows the consensus algorithm (a mix of PoW and PoS) that functions as the stakeholder of the chain, and LuckyS is issued and used as the measure of payment on a number of sidechains. Other similar dual-token blockchains include Crysto, XMT, etc. These dual-token blockchains merely separate ownership and measure of payment in order to achieve higher level of convenience in terms of ecosystem management. They do not offer multi-protocol consensus algorithms that seek to balance the network level requirements of performance and robustness.

To the author's best knowledge, there has been only one instance of "true" multi-protocol blockchains in practice at the time of this draft[2]. VeriCoin (VRC) and Verium (VRM) Reserve (Pike et al. 2015) are dual blockchain protocols pairing a digital currency with a digital commodity using a protocol called "binary chain." It utilizes a variation of PoW on the VeriCoin side (PoWT—Proof of Work Time) and a variation of PoS (PoST—Proof of Stake Time) on the Verium side, trying to be both a fast currency and a secure store of value. The VeriCoin/Verium essentially serves a rather narrow purpose of providing a faster Bitcoin alternative, thus, unlike the AIBC, it dose not seek to address the performance and robustness challenge Blockchain 3.0 faces. That is, the VeriCoin does not offer an economic model that is attack-proof, and the Verium does not offer a consensus protocol that is adaptive.

Table 1 compares the mainstream blockchains and their consensus algorithms to the AIBC.

---

[1] VBFT is proposed by Ontology. It is a consensus algorithm that combines PoS, VRF (Verifiable Random Function), and BFT.
[2] November, 2019.

Table 1. Comparisons of Blockchains and Their Consensus Algorithms.

Blockchain	Dual Token	Multi Consensus	Consensus Token	Payment Token	Blockchain Consensus	Incentive Consensus	Consensus Family
Bitcoin (Nakamoto 2008).	No.	No.	BTC.		PoW.		Proof-based.
Ethereum (Buterin 2013).	No.	No.	ETH.		PoW/PoS.		Proof-based.
Litecoin (LTC 2018).	No.	No.	LTC.		PoW.		Proof-based.
PeerCoin (King and Nadal 2012).	No.	No.	ATOM.		PoS.		Proof-based.
NXT (NXT 2015).	No.	No.	XTZ.		PoS.		Proof-based.
EOS (EOS 2018).	No.	No.	EOS.		DPoS.		Proof-based.
Bitshares (BTS 2018).	No.	No.	BTS.		DPoS.		Proof-based.
NEM (NEM 2018).	No.	No.	NEM.		PoI.		Proof-based.
NAS (NAS 2018).	No.	No.	NAS.		PoD.		Proof-based.
VET (VET 2018).	No.	No.	VET.		PoA.		Proof-based.
Hyperledger Sawtooth PoET (Intel 2017a).	No.	No.	Platform.		PoET.		Proof-based.
SpaceMint (Park et al. 2015).	No.	No.	Space.		PoSpace.		Proof-based.
Hyperledge Fabric (Androulaki et al. 2018).	No.	No.	Platform.		PBFT.		Vote-based: BFT.
NEO (NEO 2018).	No.	No.	NEO.		DBFT.		Vote-based: BFT.
Ripple (Schwartz et al. 2014; Chase and MacBrough 2018).	No.	No.	XRP.		Ripple/XRP.		Vote-based: BFT.
Hyperledger Sawtooth Raft (Intel 2017b).	No.	No.	Platform.		Raft.		Vote-based: CFT.
LuckyBlock (LKC 2018).	Yes.	No.	LKC.	LuckyS.	PoW-PoS mix.		Proof-based.
Ontology (ONT 2017).	Yes.	No.	ONT.	ONG.	VBFT.		Vote-based: BFT.
VeriCoin/Verium (Pike et al. 2015).	Yes.	Yes.	VRC.	VRM.	PoWT.	PoST.	Proof-based.
Artificial Intelligence BlockCloud (AIBC).	Yes.	Yes.	DSOL.	CFTX.	DPoEV.	DABFT.	Vote-based: Adaptive BFT.

The table compares mainstream blockchains and aspects of their consensus algorithms that we are interested against the AIBC.

4. AIBC Economic Models

The AIBC ecosystem is essentially a closed economy, of which the operations run on a set of carefully designed economic models. These economic models, at the minimum, must include a macroeconomic model that governs monetary policy (token supply), a trade economic model that enforces fair trade policy, and a microeconomic model that manages supply and demand policy.

4.1. Economic Model Overview

The most important economic model is the macroeconomic model that provides tools to govern the monetary policy, which principally deals with money (token) supply.

Before the birth of modern central banks, money was essentially in the form of precious metals, particularly gold and silver. Thus, money supply was basically sustained by physical mining of precious metals. Paper money in the modern sense did not come to existence till after the creation of the world's first central bank, Bank of England in 1694. With the creation of the central banks, the modern monetary policy was born. Initially, the goal of monetary policy was to defend the so-called gold standard, which was maintained by their promise to buy or sell gold at a fixed price in terms of the paper money (Kemmerer 1994). The mechanism for the central banks to maintain the gold standard is through setting/resetting the interest rates that they adjust periodically and on special occasions.

However, the gold standard has been blamed for inducing deflationary risks because it limits money supply (Keynes 1920). The argument gains merit during the great depression of the 1920's and 1930's, as the gold standard might have prolonged the economic depression because it prevented the central banks from expanding the money supply to stimulate the economy (Eichengreen 1995; American Economic Association 1936). The "physical" reason behind the gold standard's deflationary pressure on the economy is the scarcity of gold, which limits the ability of monetary policy to supply needed capital during economic downturns (Mayer 2010). In addition, the unequal geographic distribution of gold deposits makes the gold standard

disadvantageous for countries with limited natural resources, which compounds the money supply problem when their economies are in contrarian mode (Goodman 1981).

An obvious way to combat the gold standard's natural tendency of devaluation risk is to issue paper money that is not backed by the gold standard, the so-called fiat money. A fiat money has no intrinsic value and is used as legal tender only because its issuing authority (a central government or a central bank) backs its value with non-precious metal financial assets, or because parties engaging in exchange agree on its value (Goldberg 2005). While the fiat money seems to be a good solution for the devaluation problem, central governments have always had a variety of reasons to oversupply money, which causes inflation (Barro and Grilli 1994). Even worse, as the fiat money has no intrinsic value, it can become practically worthless if the issuing authorities either are not able or refuse to guarantee its value, which induces hyperinflation. Case in point is the Deutsche Mark hyperinflation in the Weimar Republic in 1923 (Board of Governors of the Federal Reserve System 1943).

Therefore, neither the gold standard nor the fiat currency can effectively create a "perfect" monetary policy that closely matches the money supply with the state of the economy. After the breakdown of the Bretton Woods framework, all economies, developed and developing alike, still, struggle with choices of monetary policy instruments to combat money supply issues de jour. In addition, because of the physical world's "stickiness (of everything)," all money supply instruments (e.g., central bank interest rates, reserve policies, etc.) lag behind the economic reality, making real-time policy adjustment impossible.

Therefore, eradication of deflation and inflation will always be impractical, unless a commodity money with the following properties can be found or created:

1. That it has gold-like intrinsic value but not its physical scarcity.
2. That it can be mined at the exact pace as the economic growth.
3. And that it can be put into and taken out of circulation instantaneously and in sync with economic reality.

Such a commodity does not exist in the physical world. However, things might be different in the digital world, if digital assets can be monetized into digital currencies.

There have been discussions about a "Bitcoin standard." For example, (Weber 2015) with the Bank of Canada explores the possibility and scenarios that central banks get back to a commodity money standard, only that this time the commodity is not gold, but Bitcoin. However, just like gold, Bitcoin faces a scarcity challenge in that its quantity is finite, and just like gold it needs to be mined at a pace that may lag far behind economic growth (Nakamoto 2008). As such, other than that Bitcoin resides in the digital worlds, it does not offer obvious and significant benefits over gold as the anchor for a non-deflationary commodity money standard.

However, such a digital currency can be created, which instantaneously satisfies the requirement that it can be put into and taken out of circulation instantaneously and in sync with economic reality.

The requirements that the digital currency must have gold-like intrinsic value but not its physical scarcity and that it must be mined at the exact pace as the economic growth are not trivial. First of all, there must be an agreement that digital assets are indeed assets with intrinsic value as if they were physical assets. While such an agreement is more of a political and philosophical nature, and therefore beyond the scope of our practicality-oriented interest, it is not a far stretch to regard knowledge as something with intrinsic value, and since all knowledge can be digitized, it thus can form the base of a digital currency with intrinsic value. This is what we call the "knowledge is value" principle.

Based on our "knowledge is value" principle, there is some merit to Warren Buffett's argument that Bitcoin has no intrinsic value, "because [Bitcoin] does not produce anything (Buffett 2018)." Warren Buffett's remarks refer to the facts that during the Bitcoin mining process, nothing of value (e.g., knowledge) is actually produced, and that holding Bitcoin itself does not produce returns the way traditional investment vehicles backed by physical assets do (i.e., through value-added production processes that yield dividends and capital appreciation).

Therefore, again, based on our "knowledge is value" principle, a digital currency that forms the base for a commodity money standard must have intrinsic value in and unto itself; thus not only it is knowledge, it also produces knowledge. This is the fundamental thesis upon which a digital ecosystem that uses a quantitative unit of knowledge as value measurement, thus currency, can be built.

In a digital ecosystem, there is both knowledge in existence and knowledge in production. If the value of knowledge in existence can be directly measured by a quantitative and constant unit, then the unit itself can be regarded as a currency. Furthermore, the value of knowledge in production can also be measured by the constant unit (currency) in an incremental manner, thus expansion of knowledge is in sync with the expansion of currency base. Effectively, the value measurement system is an autonomous monetary policy that automatically synchronizes economic output (knowledge mining) and money supply (currency mining), because the currency is not a stand-alone money, but merely a measurement unit of the value of knowledge. Thus, this digital currency simultaneously satisfies the requirements that it must have gold-like intrinsic value but not its physical scarcity and that it must be mined at the exact pace as the economic growth, as the currency (measurement unit) and the economic growth (knowledge) are now one and the same; they are unified. In the next section, we discuss how to develop the measurement unit.

The trade economic model provides tools to enforce fair trade policy among participants in a "globalized" economic environment. In a conventional open and free trade regime with no restrictions, it is quite likely that a few "countries" over-produce (export) and under-consume (import), thus they accumulate vast surpluses with regard to their trading partners. These countries will eventually appropriate all the wealth in the global economy, reducing their trade partners to an extreme level of poverty. Therefore, there must be a fair trade policy, enforced by a collection of bilateral and multilateral trade agreements, which penalizes the parties with unreasonable levels of surplus, and provides incentives to the parties with unreasonable levels of deficit. The penalization can be in the form of tariff levy, and other means to encourage consumption and curb production. The incentives can be tariff credit to encourage production and curb consumption. They are essentially wealth rebalancing devices that a "World Trade Organization (WTO)" like body would deploy to guarantee that trades should be both free and fair (WTO 2015).

The microeconomic model provides tools to help manage supply and demand policy in order to set market-driven transaction prices between participants. When there are multiple products simultaneously competing for consumers, the price of a product is set at the point of supply-demand equilibrium. The supply and demand policy discourages initially high-value products to dominate production capability and encourages initially low-value products to be produced. Therefore, consumers can find any product that serves their particular need at reasonable price points.

4.2. Economic Model Implementation Overview

Because of the physical world's "stickiness (of everything)," all monetary policy instruments (e.g., central bank interest rates, reserve policies, etc.), fair trade devices and supply-demand balancing tools lag behind the economic reality. This means these economic models can never dynamically track economic activities and adjust economic policies accordingly on a real-time basis. To make things more complicated, because all economic policies are controlled by centralized authorities (central banks, WTO, etc.), they may not necessarily reflect the best interests of majority participants in economic activities.

The Internet, however, provides a leveling platform that makes real-time economic policy adjustment practical. This is because the digital world can utilize advanced technological tools in order not to suffer from the reality stickiness and policy effect lag that are unavoidable in the physical world, as well as the potential conflict of interest that cannot be systematically eliminated with centralized authorities. The most important tool of them all, in this sense, is the blockchain technology, which provides a perfect platform for a decentralized digital economy capable of real-time economic policy adjustment.

On the upper-layer, the AIBC ecosystem is an implementation of the "knowledge is value" macroeconomic model through a DPoEV incentive consensus algorithm. The DPoEV consensus establishes a digital economy, in which a quantitative unit that measures the value of knowledge, the CFTX token, is used as the media of value storage and transactions. Since the token issuance and the knowledge expansion are unified and therefore always in-sync on a real-time basis, no deflation and inflation exist in the ecosystem by design. Along with the trade and microeconomic models, the AIBC provides a framework of decentralized, consensus-based digital economy with real-time policy adjustment that enables resource sharing.

On the bottom layer, the AIBC implements a DABFT distributed consensus algorithm that enforces the upper-layer DPoEV policies. It combines some of the best features of the existing consensus algorithms and is adaptive, capable of selecting the most suitable consensus for any application scenario. The DABFT is the blockchain foundation upon which the AIBC ecosystem is built.

5. Delegated Proof of Economic Value (DPoEV)

5.1. DPoEV Overview

Inside the AIBC ecosystem, all activities create (or destroy) economic value. Therefore, there is a need for a logical and universal way to assess the economic value of an activity, measured by the community's value storage and transaction medium, the CFTX token. The DPoEV incentive consensus algorithm is to create and distribute award to participating nodes in the AIBC ecosystem. The DPoEV, in turn, is established upon an innovative Economic Value Graph (EVG) approach, which is derived from the knowledge graph algorithm (a branch of Artificial Intelligence and deep learning). The EVG is designed to measure the economic value ("wealth") of the ecosystem in a dynamic way. The EVG will be explained in the next sub-section.

The implementation of DPoEV is as follow:

1. At the genesis of the AIBC, the EVG mechanism accurately assesses the economic value, or initial wealth ("base wealth") of the knowledge in the entire ecosystem (all participating nodes: super nodes, tasking nodes, computing nodes and storage nodes to be explained in the next few sections), and comes up with a system-wide wealth map. The DPoEV then issues an initial supply of CFTX tokens according to the assessment.
2. Afterward, the EVG updates the wealth map of the entire ecosystem on a real-time basis, with detailed wealth information of each and every node in the ecosystem. In the AIBC ecosystem, wealth generation is driven by tasks. The EVG assesses the incremental wealth brought about by a task, and the DPoEV issues fresh tokens accordingly. This enables the ecosystem to dynamically adjust the money (token) supply to prevent any macroeconomic level deflation and inflation in a very precise manner. Essentially, the DPoEV supervises monetary policy in a decentralized ecosystem.
3. The DPoEV monitors the real-time transactions among participating nodes of a task and manages the token award mechanism. After an amount of tokens is created for a task, the DPoEV distributes tokens to nodes that participate in the task. The number of tokens awarded to each node, as well as transaction costs (gas) attributed to the node, depending on that node's contribution to the task.
4. In an open and free trade economic system with no restrictions, it is quite likely that a few resource nodes will accumulate a tremendous level of production capability (computing power and storage space) and experience (task relevancy), who may then be given a majority of tasks/assignments due to a "rule of relevancy" ranking scheme. This would accelerate wealth generation for these dominating nodes in a speed that is unfair to other resource nodes. This is where a "rule of wealth" scheme comes in as a counter-balance, as the DPoEV can elect to grant assignments to nodes with lowest levels of wealth. If, however, there are simply not enough low wealth level resource nodes, which renders the "rule of wealth" ineffective, a "rule of fairness" scheme then comes to play. The "rule of fairness" imposes tariff levies on the dominating nodes, which are then distributed to

resource nodes with a low probability of winning assignments. Thus, the DPoEV also functions as a "world trade organization" that enforces fair trade in a decentralized ecosystem.

5. When there are multiple tasks on the ecosystem simultaneously competing for limited resources, the DPoEV decides on a real-time basis whether and how to adjust the value of each task, based on factors such as that how many similar tasks have been initiated and completed in the past and the historical values of these tasks. This prevents initially high-value tasks dominating the limited resources and encourages initially low-value tasks to be proposed. Thus, on the microeconomic level, the DPoEV dynamically balances the supply and demand of tasks. If, in rare cases, the aggregated outcome of task value adjustments is in conflict with the macroeconomic level goal (no inflation or deflation), a value-added tax (VAT) liability (in case of inflation) or a VAT credit (in case of deflation) can be posted on a separate ledger, of which the amount can be used to adjust the next round of macroeconomic level fresh token issuance. Thus the DPoEV provides a central-bank-like open market operations service in a decentralized ecosystem.
6. Finally, the DPoEV conducts periodical true-up as an extra layer of defense for housekeeping purposes. One of the key activities during true-up is for the DPoEV to "burn" surplus tokens that have been created, however, have not been awarded to participating nodes because of economic policy constraints. This is somewhat equivalent to central banks' action of currency withdrawal, which is a macroeconomic tool to destroy currency with low circulation efficiency.
7. The DPoEV is essentially conducted by the super nodes to ensure performance and efficiency in the ecosystem, this is where the "D (Delegated)" in DPoEV comes from.

5.2. Economic Value Graph (EVG) Overview

Up to this point, we still have not answered the question of how the value of knowledge is actually measured. The pursuit of a public blockchain is to create an ecosystem that supports a variety of application scenarios, and one of the challenges is to define a universal measurement of economic value.

We propose an innovative Economic Value Graph (EVG) mechanism to dynamically measure the economic value ("wealth") of knowledge in the AIBC ecosystem. The EVG is derived from the knowledge graph algorithm, which is very relevant in the context of the AIBC.

5.2.1. Knowledge Graph Overview

A knowledge graph (or knowledge map) consists of a series of graphs that illustrate the relationship between the subject's knowledge structure and its development process. The knowledge graph constructs complex interconnections in a subject's knowledge domain through data mining, information processing, knowledge production and measurement in order to reveal the dynamic nature of knowledge development and integrate multidisciplinary theories (Watthananona and Mingkhwanb 2012).

A knowledge graph consists of interconnected entities and their attributes; in other words, it is made of pieces of knowledge, each represented as an SPO (Subject-Predicate-Object) triad. In knowledge graph terminology, this ternary relationship is known as Resource Description Framework (RDF). The process of constructing a knowledge graph is called knowledge mapping. Figure 2 illustrates Knowledge Graph Subject-Predicate-Object triad.

Figure 2. Knowledge Graph Subject-Predicate-Object Triad. A knowledge graph consists of interconnected entities and their attributes; in other words, it is made of pieces of knowledge, each represented as an SPO (Subject-Predicate-Object) triad. In knowledge graph terminology, this ternary relationship is known as Resource Description Framework (RDF). The process of constructing a knowledge graph is called knowledge mapping.

The knowledge graph algorithm is consistent with the EVG. There are two steps in knowledge mapping for an ecosystem: realization of initial knowledge, and dynamic valuation of additional knowledge.

For an ecosystem, at the realization of initial knowledge stage, the knowledge graph algorithm assesses ith node's initial economic value of knowledge, which is a combination of explicit and implicit economic values of all relevant knowledge pieces at and connected to that node. The total economic value of the entire ecosystem is thus the sum of all node level economic values.

$$v0_i = \prod_{j=1}^{M} P(v0_{i,j}|v0_{i,j-1})\ v0_{i,j} \tag{1}$$

$$V0 = \sum_{i=1}^{N} v0_i\ ,\ i = 1,\ldots,N,\ j = 1,\ldots,M \tag{2}$$

where $v0_{i,j}$ is the initial economic value of the jth knowledge piece, and $P(v_{i,j}|v_{i,j-1})$ is the probability of $v0_{i,j}$ given all knowledge pieces prior to the jth, at the ith node, and Π is a Cartesian product.

Once the initial economic value of the ecosystem is realized, in a task-driven ecosystem, as the tasks start to accumulate, a collection of knowledge graphs of the tasks is then created to assess the incremental economic value of the new knowledge. Finally, the knowledge graph of the entire ecosystem is updated. This dynamic valuation of additional knowledge requires automatic extraction of relationships between the tasks and participating nodes, as well as relationship reasoning and knowledge representation realization. The total economic value of the entire ecosystem is thus the sum of all node level updated economic values.

$$t1_i = \prod_{k=1}^{K} P(t1_{i,k}|t1_{i,k-1})t1_{i,k} - \prod_{k=1}^{K} C(t1_{i,k}|t1_{i,k-1})t1_{i,k} \tag{3}$$

$$T1 = \sum_{i=1}^{N} t1_i \tag{4}$$

$$V1 = V0 + T1 = \sum_{i=1}^{N}(v0_i + t1_i),\ i = 1,\ldots,N,\ k = 1,\ldots,\ K \tag{5}$$

where $t0_{i,k}$ is the incremental economic value of the kth knowledge piece of the task, $P(t1_{i,k}|t1_{i,k-1})$ is the probability of $t1_{i,k}$ given all knowledge pieces prior to the kth, $C(t1_{i,k}|t1_{i,k-1})$ is the covariance of $t1_{i,k}$ given all knowledge pieces prior to the kth, at the ith node, and Π is Cartesian product.

5.2.2. EVG Implementation

The essence of EVG is "knowledge is value," and it accesses the entire ecosystem's economic value dynamically.

At the genesis of the AIBC ecosystem, there are no side blockchains, as no task has been initiated yet, and the EVG mechanism just simply depicts a knowledge graph of each and every node (super, tasking, computing and storage node) in the blockchain. The EVG then aggregates the knowledge graph of all nodes and establishes a global knowledge graph. At this juncture, the EVG has already assessed the original knowledge depository of the entire ecosystem. Furthermore, in order to quantify this original wealth, the EVG equates it to an initial supply of CFTX tokens, issued by the DPoEV consensus. This process establishes a constant measurement unit of economic value (token) for the future growth of the ecosystem. The EVG then creates a credit table, which contains all nodes in the ecosystem, and their initial economic values. When a new node joins the ecosystem, the EVG appends a new entry to the credit table for it, with its respective initial economic value. The credit table resides in all super nodes, and its creation and update need to be validated and synchronized by

all super nodes by the fundamental layer DABFT distributed consensus algorithm. The DABFT will be discussed in the next section.

The wealth generation is driven by tasks, and the super nodes are the ones that are responsible for handling them. As the tasks continue to be initiated, side chains continue to grow and accumulate from the super nodes. These side chains are the containers of the incremental knowledge, and the EVG measures the economic value of this incremental knowledge with the measurement unit (token). Upon the acceptance of every task, the DPoEV consensus issues a fresh supply of CFTX tokens proportional to the newly created economic value to ensure that the money supply is in sync with the economic growth in order to avoid macroeconomic level inflation or deflation.

Each task is tracked by a distinguished task blockchain, which is a side chain with the root block connected to its handling super node. Each block in the task blockchain tracks the status of the task. The root block contains information including the initial estimation of the economic value of the task. Each subsequent block provides updated information on contributions from the task validation, handling, and resource nodes. When the task blockchain reaches its finality, the EVG has a precise measure of economic value generated by this task. Furthermore, the blocks contain detailed information on contributions from participating nodes, and transactions. Thus, the EVG can accurately determine the size of the reward (amount of tokens) issued to each participating node. The DPoEV then credits a respective amount of tokens to each participating node, which is recorded in the credit table validated by the DABFT consensus.

The EVG enables the DPoEV to manage the economic policy of the ecosystem on a real-time basis through the credit table. The DPoEV can dynamically determine the purchase price of a task, which covers the overall cost paid to the super and resource nodes. It can also set the transaction cost for each assignment. The overall effect is that all macroeconomic, microeconomic and trade policies are closely monitored and enforced. Table 2 is an example of the EVG Node Credit Table.

Table 2. EVG Node Credit Table.

Economic Value. (CFTX Token)	Total Economic Value	Initial Economic Value	Incremental Economic Value-Task 1	Incremental Economic Value-Task k	Incremental Economic Value-Task K
Super Node 1.	1,250,000.	1,000,000.	1000.	750.	500..
. . . .	. . . .	. . . .	. . . .	. . . .	. . . ..
Super Node NS.	2,100,000.	2,000,000.	2000.	1500.	1000..
Tasking Node 1.	75,000.	50,000.	50.	30.	25..
. . . .	. . . .	. . . .	. . . .	. . . .	. . . ..
Tasking Node NT.	125,000.	75,000.	75.	60.	50..
Computing Node 1.	200,000.	100,000.	100.	90.	75..
. . . .	. . . .	. . . .	. . . .	. . . .	. . . ..
Computing Node NC.	300,000.	250,000.	250.	175.	100..
Storage Node 1.	200,000.	150,000.	150.	80.	25..
. . . .	. . . .	. . . .	. . . .	. . . .	. . . ..
Storage Node NST.	350,000.	300,000.	300.	175.	75..

The EVG (Economic Value Graph) enables the DPoEV to manage the economic policy of the ecosystem on a real-time basis through the credit table. The DPoEV can dynamically determine the purchase price of a task, which covers the overall cost paid to the super and resource nodes. It can also set the transaction cost for each assignment. The overall effect is that all macroeconomic, microeconomic and trade policies are closely monitored and enforced.

5.3. Economic Relevancy Ranking (ERR)

While the EVG measures the economic value of knowledge created by task, it does not assess the validation, handling, computing, and storage capabilities of participating nodes, as these capabilities are not necessarily based on knowledge. This can be fatal because the DPoEV assigns tasks to super nodes and resource nodes first and foremost with a "rule of relevancy" ranking scheme. This issue is resolved by the Economic Relevancy Ranking (ERR) mechanism.

The ERR ranks tasks as well as the super node and resource nodes (collectively known as "service nodes"). Based on the ERR rankings, the DPoEV provides a matchmaking service that pairs tasks and service nodes.

The ERR assesses each newly created task by the following factors:

1. Time criticalness: How much time a task takes before a task timer expires.
2. Computing intensity: How much computing power is required to complete the task and associated assignments.
3. The Frequency of transactions: Higher transaction frequency improves liquidity, which further increases transaction frequency. Higher transaction frequency allows a faster growth of wealth, however, brings higher demand to the network and database framework.
4. The Scale of transactions: Larger transaction scale improves liquidity, which further increases the transaction scale. Larger transaction scale allows a faster growth of wealth, however, brings higher demand to the network and database framework.
5. Required propagation: Stronger propagation in terms of bandwidth means improved liquidity, which improves transaction frequency and scale. Stronger propagation allows a faster growth of wealth, however, brings higher demand to the network and database framework.
6. Optional data requirement: What and how much data is required to complete the task and associated assignments, and more importantly, where the data is stored.

The ERR ranking score of a task is thus given as:

$$TR_{ERR} = \sum_{i=1}^{N} \frac{w_i TR_i}{n_i}, \sum_{i=1}^{N} w_i = 1 \quad (6)$$

where TR_i is ranking score of the ith factor, w_i is that factor's weight, and n_i is the factor's normalization coefficient.

As tasks start to accumulate, they are ranked by the above criteria. The ERR then creates a task ranking table, which contains the addresses of all tasks (root blocks of side chains) and their ranking scores. When a new task is initiated, the ERR appends a new entry to the task ranking table for it, with its respective ranking score. The task ranking table resides in all super nodes, and its creation and update need to be validated and synchronized by all super nodes based on the DABFT consensus. Table 3 gives an example of the ERR Task Ranking Score Table.

Table 3. ERR Task Ranking Score Table.

Economic Relevancy Ranking (ERR Score)		**Task 1**	…	…	**Task i**	…	…	**Task N**
		0.39			**0.59**			**0.46**
Time Criticalness	Score	100	…	…	50	…	…	75
	Weight	0.15			0.05			0.25
	Normalization Coefficient	100			100			100
Computing Intensity	Score	25	…	…	75	…	…	100
	Weight	0.15			0.25			0.15
	Normalization Coefficient	100			100			100
Frequency of Transactions	Score	5,000	…	…	250,000	…	…	100,000
	Weight	0.25			0.25			0.15
	Normalization Coefficient	1,000,000			1,000,000			1,000,000
Scale of Transactions	Score	5	…	…	8	…	…	3
	Weight	0.25			0.25			0.15
	Normalization Coefficient	10			10			10
Required Propagation	Score	350	…	…	500	…	…	150
	Weight	0.15			0.15			0.15
	Normalization Coefficient	1,000			1,000			1,000
Data Requirement	Score	50	…	…	75	…	…	25
	Weight	0.05			0.05			0.15
	Normalization Coefficient	100			100			100

The ERR (Economic Relevancy Ranking) creates a task ranking table, which contains the addresses of all tasks (root blocks of side chains) and their ranking scores. When a new task is initiated, the ERR appends a new entry to the task ranking table for it, with its respective ranking score. The task ranking table resides in all super nodes, and its creation and update need to be validated and synchronized by all super nodes based on the DABFT consensus.

In parallel, the ERR assesses the capabilities of the service nodes based on the same criteria. It then creates a service node ranking table, which contains the addresses of all service nodes and their ranking scores. When a new service node joins, the ERR appends a new entry to the service node ranking table for it, with its respective ranking score. The service node ranking table resides in all super nodes, and its creation and update need to be validated and synchronized by all super nodes with the DABFT consensus.

The ERR algorithm has three major properties:

1. Consistency. A ranking score, once recorded, cannot be altered through paying more cost by the tasking node (for task ranking) or the service node (for service node ranking). However, the ranking score does change as both the tasking and service nodes do evolve. Adjustment to the ranking score can only be conducted by DPoEV through the DABFT consensus.
2. Computability. The ERR ranking scores need to be retrieved by the DPoEV instantly, thus the ERR algorithm requires low computational complexity.
3. Deterministicness. The ERR algorithm should produce identical results on all nodes for the same service node.

The ERR ranking score of a service node is thus given as:

$$SNR_{ERR} = \sum_{j=1}^{M} \frac{w_j SNR_j}{n_j}, \sum_{j=1}^{M} w_j = 1 \tag{7}$$

where SNR_j is the ranking score of the jth property, w_j is that property's weight, and n_j is that property's normalization coefficient.

Based on the ERR ranking scores of tasks and service nodes, the DPoEV provides a matchmaking service that pairs tasks with service nodes with the closet ranking scores. Thus the "rule of relevancy" in service node selection is observed, and service nodes with the highest rankings cannot dominate task handling and assignment. Rather, they have to be "relevant" to the tasks for which they compete. In addition, the "rule of wealth" and "rule of fairness" are used to enforce economic principles. Table 4 is an instance of the ERR Service Node Ranking Score Table.

Table 4. ERR Service Node Ranking Score Table.

Economic Relevancy Ranking (ERR Score)		**Super Node 1**	…	…	**Computing Node 1**	…	…	**Service Node 1**
		0.50			**0.35**			**0.70**
Consistency	Score	50	…	…	25	…	…	35
	Weight	0.50			0.70			0.25
	Normalization Coefficient	100			100			100
Computability	Score	50	…	…	60	…	…	75
	Weight	0.30			0.20			0.30
	Normalization Coefficient	100			100			100
Deterministicness	Score	50	…	…	50	…	…	85
	Weight	0.20			0.10			0.45
	Normalization Coefficient	100			100			100

Based on the ERR ranking scores of tasks and service nodes, the DPoEV provides a matchmaking service that pairs tasks with service nodes with the closet ranking scores. Thus the "rule of relevancy" in service node selection is observed, and service nodes with the highest rankings cannot dominate task handling and assignment. Rather, they have to be "relevant" to the tasks for which they compete. In addition, the "rule of wealth" and "rule of fairness" are used to enforce economic principles.

The service node selected (out of N service nodes) given a task j is follows the following equation:

$$SN_T = min[\bigcup_i^N (SN_{ARR,i} - TR_{ERR,j})] \quad (8)$$

where $SN_{ERR,i}$ is the ERR ranking score of the ith service node, and $TR_{ERR,j}$ is the ERR ranking score of the jth task.

It is important to notice that, unlike the EVG, the ERR does not measure the economic value of tasks and service nodes. Rather, it ranks them based on their requirements and capabilities, which are not the bearers of economic value, but its producers. As such, the ERR has no role in money supply policy in the DPoEV framework.

5.4. DPoEV Advantages

The DPoEV incentive consensus algorithm creates and distributes award to participating nodes in the AIBC ecosystem in the form of CFTX tokens. It eliminates the possibility of macroeconomic level inflation and deflation, enforces free and fair trade, and balances microeconomic level supply and demand.

With the EVG and ERR, by design, the DPoEV enforces the economic policies and the "rules of relevancy, wealth and fairness." It thus guarantees that no tasking nodes can dominate task initiation, no super nodes can dominate task handling, and no resource nodes can dominate task assignment.

A key benefit of the DPoEV is that it effectively eliminates the possibility of 51% attack based on the number of efforts (like Proof-of-Work in Bitcoin), or wealth accumulation (like Proof-of-Stake in Ethereum). As a matter of fact, it has the potential to eliminate 51% attack of anything.

6. Delegated Adaptive Byzantine Fault Tolerance (DABFT)

While the DPoEV algorithm provides the application layer incentive consensus, it needs to work with a high-performance fundamental layer distributed consensus protocol that actually provides blockchain services. This bottom layer consensus is the "real" blockchain enabler.

Therefore, unlike most of the existing public chains, the AIBC establishes a two-consensus approach: on the application layer, the DPoEV consensus is responsible for economic policy enforcement, and on the fundamental layer, a Delegated Adaptive Byzantine Fault Tolerance (DABFT) distributed consensus algorithm is responsible for managing each and every transaction in terms of block generation, validation, and ledger recording. While the DPoEV does not need to be real-time as most of the application scenarios do not demand real-time reward distribution, the DABFT has to be real-time, as block validation and ledger recording need to be done quickly and robustly. The goal of DABFT is to achieve up to hundreds of thousands of TPS (Transactions per Second).

6.1. DABFT Design Goals

The DABFT implements the upper-layer DPoEV economic policies on the fundamental layer and provides the blockchain services of block generation, validation, and ledger recording. It focuses on the AIBC's goals of efficiency, fairness, and legitimacy. Unlike the dominant consensus algorithms (e.g., PoW) that waste a vast amount of energy just for the purpose of winning ledger recording privilege, the DABFT utilizes resources only for meaningful and productive endeavors that produce economic value.

6.2. DABFT Adaptive Approach

In Section 3 we survey all the existing blockchain consensus algorithms. In view of the advantages and disadvantages of the existing consensus algorithms, we conclude that, although some of them offer useful features, none of them alone can fully meet the AIBC goals of efficiency, fairness, and legitimacy, and hence, do not comply with the DPoEV economic models.

We thus propose the DABFT, which combines the best features of the existing consensus algorithms. Conceptually, the DABFT implements certain PoS features to strengthen the legitimacy of the PoI, and certain PoI features to improve the fairness of PoS. It also improves the PoD's election mechanism with the BFT algorithm.

In addition, the DABFT is further extended by a feature of adaptiveness. The DABFT is a delegated mechanism with a higher level of efficiency and is essentially a more flexible DBFT that is capable of selecting BFT flavors most suitable for particular (and parallel) tasks on the fly. The adaptiveness is achieved by deep learning techniques, that real-time choices of consensus algorithms for new tasks are inferred from trained models of previous tasks.

Therefore, the DABFT is the perfect tool to build the efficient, legit and fair AIBC ecosystem that conducts only meaningful and productive activities.

6.3. DABFT Algorithm Design

6.3.1. New Block Generation

Upon the release of a new task, a subset of super nodes that are most relevant to the task is selected as the representatives (task validators), who then elect among themselves a single task handler responsible for managing the task. The task handler then selects a number of resource nodes that are the most relevant to the task, and distribute the task to them. Upon successful release of the new task, the task handler proposes a new block that is then validated by the task validators. A new block is thus born.

Because of the "rule of relevancy," it is highly likely that each new task is assigned a completely different set of task validators and task handler. However, once the task handler and validators are selected, they manage the task from inception to completion (from the root block to the final block of the side chain). Therefore, there is no need for the periodical system-wide reelection of representatives.

The key benefit of this arrangement is that no dynasty management is required, which reduces the system's complexity and improves its efficiency.

The real-time selection of task validators and handler for a new task based on the "rule of relevancy" means the DABFT has a built-in "dynamic sharding" feature, which will be explained in a later subsection.

6.3.2. Consensus Building Process

After a task handler proposes a new block, the task validators participate in a round of BFT voting to determine the legitimacy of the block.

At present, none of the mainstream BFT algorithms is optimal for all tasks. The DABFT utilizes a set of effectiveness evaluation algorithms through AI based deep learning to determine the optimal BFT mode for the task at hand. The flavors of BFT algorithms for the DABFT to choose from include, but not limited to, DBFT and PBFT (Practical BFT), as well as Q/U (Abd-El-Malek et al. 2005), HQ (Cowling et al. 2006), Zyzzyva (Kotla et al. 2009), Quorum (Guerraoui et al. 2010), Chain (Guerraoui et al. 2010), Ring (Guerraoui et al. 2011), and RBFT (Redundant BFT) (Aublin et al. 2013), etc. Figure 3 shows the consensus process for several mainstream BFT algorithms.

Through the machine learning prediction, DABFT dynamically switches the system to the optimal BFT consensus of the present task. The DABFT improves upon the ADAPT (Bahsoun et al. 2015) and is similar to it in several ways. Like the ADAPT, the DABFT is a modular design and consists of three important modules: BFT System (BFTS), Event System (ES), and Quality Control System (QCS). The BFTS is essentially an algorithms engine that modularizes the aforementioned BFT algorithms. The ES collects factors that have a significant impact on performance and security in the system, such as a number of terminals, requests, sizes, etc., and sends task information to the QCS. The QCS drives the system through either static (Shoker and Bahsoun 2013), dynamic, or heuristics mode, and evaluates a set of Key Performance Indicators (KPI) and Key Characteristics Indicator (KCI) to select the optimal BFT flavor for the task at hand.

The QCS computes the evaluation scores of the competing BFT protocols for a particular task and then selects the protocol with the highest score. For a given task t and protocol $p_i \in BFTS$ that has an evaluation score $E_{i,t}$ $(element\ of\ Matrix\ E)$, the best protocol p_t is given as:

$$p_t = p_i,\ s.t.\ E_{i,t} = \underset{1 \le j \le n}{max} E_{j,t} \tag{9}$$

$$where \begin{cases} E = C \circ P \\ C = \left\lfloor \frac{1}{a} \cdot (A \dot{\vee} (e_n - U)) \right\rfloor \\ P = B^{\pm} \cdot (V \circ W) \end{cases} \tag{10}$$

where C is the KCI matrix and P the KPI matrix; matrix A represents the profiles (i.e., the KCIs) of the protocols; Column matrix U represents the KCI user preferences (i.e., the weights); column matrix e_n is a unit matrix used to invert the values of the matrix U to $-U$. The use of $1/a$ within the integer value operator $\lfloor\ \rfloor$ rules out protocols not matching all user preferences in matrix U. Matrix B represents KPIs of the protocols, one protocol per row. Column matrix V represents the KPI user-defined weights for evaluations. Column matrix W is used in the heuristic mode only, with the same constraints as matrix V. The operator "$\circ$" represents Hadamard multiplication, and the operator "$\dot{\vee}$" represents Boolean multiplication.

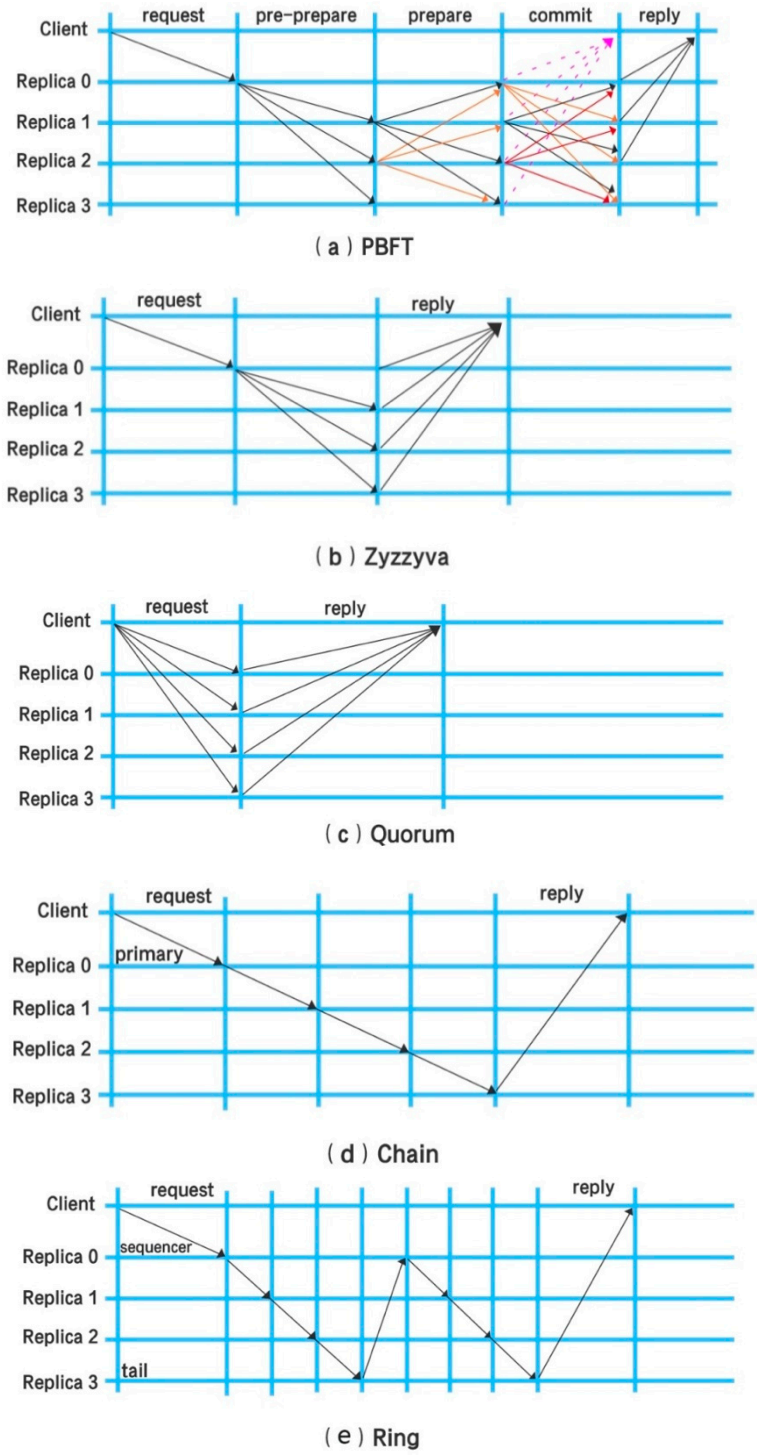

Figure 3. Mainstream BFT Algorithm Consensus Processes.

There is one major shortcoming in the ADAPT design. The ADAPT employs the Support Vector Regression (SVR) method (Smola and Schölkopf 2004) with a five-fold cross-validation to predict the KPI parameters for elements in matrix B. There are six fields in the dataset: number of clients, request size, response size, throughput, latency, and capacity. While the methodology is useful in BFT settings in and of itself, it is not designed for the highly complex blockchain application scenarios in which there are many interactions between participants, thus it is not particularly effective for them. For example, in the AIBC context, at any given time there are multiple tasks (handled by different handlers) that compete for resources. As such, the ever-increasing number of tasks and interactions between them

affect the key KPI parameters (throughput, latency, and capacity) for individual tasks continuously (time series), for the purpose of achieving the best performance on the system-level. As such, it is necessary to introduce a mechanism that incorporates time-varying conditional correlations across tasks in order to adjust the KPI parameters on the fly. What sets the DABFT apart from the ADAPT is that the DABFT has such a function built in.

The DABFT implements the time-varying conditional correlation mechanism in the QCS. First of all, for task t, the QCS trains on the existing data to come up with the initial matrix $\hat{B}_t$ (basically matrix B in the ADAPT, but specifically for task t). It then calculates a residual matrix E_t as follows[3]:

$$E_t \ = \ \hat{B}_t \ - \ B_t \tag{11}$$

where B_t is the "real" KPI parameter matrix derived from empirical tests.

The specification with time-varying multi-dimensional correlation matrix for task t is thus given as[4]:

$$\begin{gathered} E_t|\Psi_{t-1} \sim N(0, \Omega_t = H_t P_t H_t) \\ H_t^2 = H_0^2 + \mathrm{K} E_{t-1} E_{t-1}^T + \Lambda H_{t-1}^2 \\ P_t = O_t^* O_t O_t^* \\ \Xi_t = H_t^{-1} E_t \\ O_t = (1-a-b)\overline{O} + a\Xi_t \Xi_{t-1}^T + bO_{t-1} \\ a+b<1 \end{gathered} \tag{12}$$

where:

1. E_t is the conditional residual vector at time t given the previous state Ψ_{t-1}.
2. Ω_t is the conditional covariance matrix of E_t.
3. P_t is the conditional correlation matrix of E_t.
4. H_t is the normalization matrix for P_t.
5. K and Λ are diagonal coefficient matrices for H_t.
6. Ξ_t is the standardized residue vector of E_t.
7. O_t and O_t^* are estimator matrices for P_t.
8. $\overline{O}$ is the unconditional correlation matrix of E_t.

It's worth mentioning that Equations (11) and (12) only propagate from task t back to task $t-1$ for the purpose of reducing computation complexity.

Finally, the predicted KPI matrix for task t, $\overline{B}_t$, is given as:

$$\overline{B}_t \ = \ \hat{B}_t \ + \ \Omega_t \tag{13}$$

From this point onward, DABFT is similar to the ADAPT, and proceeds to select the BFT protocol with the evaluation highest score based on Equations (9) and (10). For any BFT choice, the DABFT provides fault tolerance for $F = \lfloor (N-1)/3 \rfloor$ for a consensus set consisting of N task validators. This fault tolerance includes security and availability and is resistant to general and Byzantine faults in any network environment. The DABFT offers deterministic finality, thus a confirmation is a final confirmation, the chain cannot be forked, and the transactions cannot be revoked or rolled back.

Under the DABFT consensus mechanism, it is estimated that a block is generated every 0.1 to 0.5 s. The system has a theoretical sustainable transaction throughput of 30,000 TPS, and with proper

---

[3] The $\hat{B}_t$ and B_t are full matrices made of row vectors for individual BFT flavors, while E_t is actually a column matrix. The mathematical representation in this subsection is simplified just to illustrate the analysis process without losing a "high-level" accuracy.

[4] Essentially, this is a Dynamic Conditional Correlation (DCC) for multivariate time-series analysis with a DCC (1,1) specification (Engle and Sheppard 2001; Engle 2002).

optimization, has a potential to achieve 100,000 TPS and beyond, making the AIBC ecosystem capable of supporting high-frequency large-scale commercial applications.

The DABFT has an option to incorporate digital identification technology for the AIBC to be real name based, making it possible to freeze, revoke, inherit, retrieve, and transfer assets under judicial decisions. This feature makes the issuance of financial products with compliance requirement possible.

6.3.3. Fork Selection

The DABFT selects the authority chain for each task with a block score at each block height. Under the principle of fairness and legitimacy, the forked chain of blocks with the highest economic value is selected to join the authority chain. The economic value of a forked chain is the sum of the economic value of the forked block and the descendants of that block. This is achievable because all tasks are tracked by their corresponding side chain blocks that will eventually reach finality.

6.3.4. Voting Rules

In order to defend against malicious attacks to the consensus process, the DABFT borrows Casper's concept of minimum penalty mechanism to constrain task validators' behavior. The voting process follows the following basic rules:

1. The consensus process of a single block has a strict sequence. Only after the total number of votes in the first stage reaches 2/3 majority, can the next stage of consensus start.
2. The consensus of a subsequent block does not need to wait until the consensus of the current block is concluded. The consensuses of multiple blocks can be concurrent, however not completely out of order. Generally, after the consensus of the current block is 2/3 completed, the consensus of a subsequent block can start.

6.3.5. Incentive Analysis

The task validators (including the task handler) participating in the DABFT of a task receive rewards in the form of CFTX tokens according to the DPoEV incentive consensus. The total number of tokens awarded to the task validators is a percentage of the overall number of tokens allocated to the task and is shared by all participating task validators and handler. The number of tokens awarded to the task handler and each task validator is determined by its contribution to the completion of the task. These numbers are dynamically determined by the DPoEV, particularly its EVG engine.

6.4. Attack-Proof

There are several attacks of particular interest in distributed consensus, and three of the most analyzed ones are double spending attack, short-range attack and 51% attack. In the DPoEV-DABFT two consensus AIBC ecosystem, by design, none of the attacks have a chance to succeed.

A double spending attack happens when a malicious node tries to initiate the same tokens through two transactions to two distinguished destinations. In a delegated validation regime (e.g., DPoS or DBFT), for such an attack to succeed, the malicious node must first become a validator through the election (with deposit paid) and then bribe at least one-third of other validators in order for both transactions to reach finality. It is impossible to succeed in double spending in the DPoEV-DABFT two consensus AIBC ecosystem. The reasons are that the validators (super nodes) are chosen by their relevancy to tasks but not their deposits; that the validators are not allowed to initiate tasks; and that the validators are rewarded based on their levels of contribution, not by other validators. Essentially, conditions for the double spending attack to occur do not exist.

A short-range attack is initiated by a malicious node that fakes a chain (A-chain) to replace the legitimate chain (B-chain) when the H+1 block has not expired. In a delegated regime for this attack to be successful, the attacker needs to bribe the validators in order to make the block A1 score higher than B1. Thus, essentially, the short-range attack is very much like a double-spending attack at the

A1/B1 block level, which has no chance to succeed for the same reason that makes the double-spending attack futile.

In the PoW, a 51% attack requires a malicious node to own 51% of the total computing power in the system, in the PoS 51% of the deposit, and in the PoD 51% of the certified accounts. In the DPoEV-DABFT two consensus AIBC ecosystem, restrained by the economic model, there is no possibility for any node to own more than 51% of the economic value. More importantly, since the validators are not allowed to initiate tasks (thus transactions), a validator with bad intention must bribe its compatriots to even launch such an attack. However, the validators are rewarded based on their levels of contribution, not by other validators. Essentially, conditions for the 51% attack to occur do not exist either.

6.5. Dynamic Sharding

One of the challenges the mainstream blockchains face is scalability, which is key to performance improvement. Ethereum seeks to resolve the scalability issue with the so-called sharding approach, in which a shard is essentially "an isolated island" (Rosic 2018; Buterin 2018). The DABFT, by design, has a built-in dynamic sharding feature.

First of all, the AIBC ecosystem is a 2D BlockCloud with super nodes that track the status of tasks through side chains. Once a task is initiated, a set of task validators are then selected according to the "rule of relevancy". A task handler is then chosen among the task validators to handle the task. The task handler and validators manage the task from the beginning to the end with no dynasty change. Thus, effectively, from a task's perspective, the task validators form a shard that is responsible for managing it, with the task handler being its leader.

In addition, due to the "rule of relevancy," it is highly likely that each new task is assigned a different set of task validators from the previous task, although overlapping is possible, especially when the number of super nodes is small. Therefore, once a task is completed and its associated task reaches finality, its shard dissolves automatically. Therefore, in the AIBC, no periodic "re-sharding" is necessary. Such fluidity affords the AIBC a "dynamic" sharding feature.

The dynamic sharding feature makes the so-called single-shard takeover attack against the AIBC impossible to succeed. First off, shards are directly formed by tasks in a highly random fashion due to the unpredictable nature of the "rule of relevancy." Second, shards have very short lifespans as they only last till tasks are completed. Practically, malicious nodes never have a chance to launch attacks.

The AIBC also maintains a 1D "main chain" at each super node, with the blocks of side chains of shards intertwined. A Merkle tree structure of the 1D blockchain makes it topologically identical to the 2D BlockCloud.

7. Conclusions

The AIBC is an Artificial Intelligence and blockchain technology based large-scale decentralized ecosystem that allows system-wide low-cost sharing of computing and storage resources. The AIBC consists of four layers: a fundamental layer, a resource layer, an application layer, and an ecosystem layer.

The AIBC layers have distinguished responsibilities and thus performance and robustness requirements. The application and resource layers need to follow the economic policies strictly and run on a deterministic and robust protocol; the fundamental layer needs to follow an adaptive protocol with high throughput without sacrificing robustness. As such, the AIBC implements a two-consensus approach: the DPoEV incentive consensus to create and distribute awards on the upper layers, and the DABFT distributed consensus responsible for blockchain functions on the fundamental layer. The DPoEV is deterministic and does not necessarily require high-performance as most of the application scenarios do not demand real-time reward distribution. The DABFT is real-time and adaptive, as block validation and record bookkeeping need to be done quickly and robustly.

The DPoEV follows a set of economic policy (especially the macroeconomic policy that governs the monetary policy, thus token supply), and uses the knowledge map algorithm to accurately assess the economic value of digital assets. The DABFT uses deep learning techniques to predict and select

the most suitable BFT algorithm in order to enforce the economic policies on the fundamental layer, as well as to achieve the best balance of performance, robustness, and security. In addition, by design, the DABFT has a built-in dynamic sharding feature, which affords the AIBC scalability, while at the meantime eliminates the possibility of single-shard takeover.

With the DPoEV-DABFT dual-consensus architecture, the AIBC has a theoretical sustainable transaction throughput of 30,000 TPS, and with proper optimization, has a potential to achieve 100,000 TPS and beyond, making the AIBC ecosystem capable of supporting high-frequency large-scale commercial applications. In addition, the dual-consensus architecture, by design, makes the AIBC attack-proof against risks such as double-spending, short-range and 51% attacks.

Our contribution is four-fold: that we develop a set of innovative economic models governing the monetary, trading and supply-demand policies in the AIBC; that we establish an upper-layer DPoEV incentive consensus algorithm that implements the economic policies; that we provide a fundamental layer DABFT distributed consensus algorithm that executes the DPoEV with adaptability; and that we prove the economic models can be effectively enforced by AIBC's DPoEV-DABFT dual-consensus architecture.

Funding: This research was funded by Cofintelligence Financial Technology Ltd. (Hong Kong and Shanghai, China).

Conflicts of Interest: The authors declare no conflict of interest.

References

Abd-El-Malek, Michael, Gregory R. Ganger, Garth R. Goodson, Michael K. Reiter, and Jay J. Wylie. 2005. Fault-scalable Byzantine Fault-Tolerant Services. *Association for Computing Machinery* 39: 59–74. [CrossRef]

American Economic Association. 1936. The Elasticity of the Federal Reserve Note. *The American Economic Review* 26: 683–90.

Androulaki, Elli, Artem Barger, Vita Bortnikov, Christian Cachin, Konstantinos Christidis, Angelo De Caro, David Enyeart, Christopher Ferris, Gennady Laventman, Yacov Manevich, and et al. 2018. Hyperledger fabric: A distributed operating system for permissioned blockchains. Paper presented at the Thirteenth EuroSys Conference, Porto, Portugal, April 23–26; Available online: https://arxiv.org/pdf/1801.10228.pdf (accessed on 24 November 2019).

Aublin, Pierre-Louis, Sonia Ben Mokhtar, and Vivien Quéma. 2013. RBFT: Redundant Byzantine Fault Tolerance. Paper presented at the 2013 IEEE 33rd International Conference on Distributed Computing Systems, Philadelphia, PA, USA, July 8–11.

Bahsoun, Jean-Paul, Rachid Guerraoui, and Ali Shoker. 2015. Making BFT Protocols Really Adaptive. Paper presented at the 2015 IEEE International Parallel and Distributed Processing Symposium, Hyderabad, India, May 25–26; pp. 904–13. [CrossRef]

Barro, Robert, and Vittorio Grilli. 1994. *European Macroeconomics*. chp. 8, Figure 8.1. Beijing: Macmillan, p. 139. ISBN 0-333-57764-7.

Belotti, Marianna, Nikola Božić, Guy Pujolle, and Stefano Secci. 2019. A Vademecum on Blockchain Technologies: When, Which and How. *IEEE Communications Surveys & Tutorials*. Available online: https://hal.sorbonne-universite.fr/hal-01870617/document (accessed on 24 November 2019). [CrossRef]

Board of Governors of the Federal Reserve System. 1943. *Banking and Monetary Statistics 1914–1941*. Washington: Board of Governors of the Federal Reserve System, p. 671.

Bonneau, Joseph, Andrew Miller, Jeremy Clark, Arvind Narayanan, Joshua A. Kroll, and Edward W. Felten. 2015. SoK: Research Perspectives and Challenges for Bitcoin and Cryptocurrencies. Paper presented at the 2015 IEEE Symposium on Security and Privacy, San Jose, CA, USA, May 17–21; Available online: https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=7163021 (accessed on 24 November 2019). [CrossRef]

BTS. 2018. The BitShares Blockchain. Available online: https://www.bitshares.foundation/papers/BitSharesBlockchain.pdf (accessed on 24 November 2019).

Buffett, Warren. 2018. Available online: https://www.cnbc.com/2018/05/07/warren-buffett-on-bitcoin-it-doesnt-produce-anything.html (accessed on 24 November 2019).

Buterin, Vitalik. 2013. What Proof of Stake Is and Why It Matters. Available online: https://bitcoinmagazine.com/articles/what-proof-of-stake-is-and-why-it-matters-1377531463 (accessed on 24 November 2019).

Buterin, Vitalik. 2014. Ethereum: A Next-Generation Smart Contract and Decentralized Application Platform. Available online: https://github.com/ethereum/wiki/wiki/White-Paper (accessed on 24 November 2019).

Buterin, Vitalik. 2018. On Sharding Blockchains. Available online: https://github.com/ethereum/wiki/wiki/Sharding-FAQs#what-is-the-basic-idea-behind-sharding (accessed on 24 November 2019).

Castro, Miguel, and Barbara Liskov. 2002. Practical Byzantine Fault Tolerance and Proactive Recovery. *ACM Transactions on Computer Systems* 20: 398–461. [CrossRef]

Chase, Brad, and Ethan MacBrough. 2018. Analysis of the XRP Ledger Consensus Protocol. Available online: https://arxiv.org/pdf/1802.07242.pdf (accessed on 24 November 2019).

Cowling, James, Daniel Myers, Barbara Liskov, Rodrigo Rodrigues, and Liuba Shrira. 2006. HQ Replication: A Hybrid Quorum Protocol for Byzantine Fault Tolerance. Paper presented at the 7th USENIX Symposium on Operating Systems Design and Implementation, Seattle, WA, USA, November 6–8; pp. 177–90.

Driscoll, Kevin, Brendan Hall, Håkan Sivencrona, and Phil Zumsteg. 2003. Byzantine Fault Tolerance, from Theory to Reality. In *International Conference on Computer Safety, Reliability, and Security*. Berlin/Heidelberg: Springer, pp. 235–48. ISSN 0302-9743. [CrossRef]

Eichengreen, Barry J. 1995. *Golden Fetters: The Gold Standard and the Great Depression, 1919–1939*. Oxford: Oxford University Press, ISBN 0-19-510113-8.

Engle, Robert F., and Kevin Sheppard. 2001. *Theoretical and Empirical Properties of Dynamic Conditional Correlation Multivariate GARCH*. Cambridge: National Bureau of Economic Research.

Engle, Robert. 2002. Dynamic conditional correlation: A simple class of multivariate Generalized Autoregressive Conditional Heteroscedasticity models. *Journal of Business and Economic Statistics* 20: 339–50. [CrossRef]

EOS. 2018. EOS White Paper. Available online: https://github.com/EOSIO/Documentation/blob/master/TechnicalWhitePaper.md (accessed on 24 November 2019).

Goldberg, Dror. 2005. Famous Myths of Fiat Money. *Journal of Money, Credit and Banking* 37: 957–67. [CrossRef]

Goodman, George Jerome Waldo. 1981. *Paper Money*. New York: Dell Pub Co., pp. 165–66.

Governatori, Guido, Florian Idelberger, Zoran Milosevic, Regis Riveret, Giovanni Sartor, and Xiwei Xu. 2018. On legal contracts, imperative and declarative smart contracts, and blockchain systems. *Artificial Intelligence and Law* 26: 377–409. [CrossRef]

Guerraoui, Rachid, Nikola Knežević, Vivien Quéma, and Marko Vukolić. 2010. The next 700 bft protocols. Paper presented at the 5th European Conference on Computer Systems, Paris, France, April 13–16; pp. 363–76.

Guerraoui, Rachid, Nikola Knezevic, Vivien Quema, and Marko Vukolic. 2011. *Stretching BFT*. Lausanne: EPFL.

Herlihy, Maurice. 2018. Blockchains from a Distributed Computing Perspective. *Communications of the ACM* 62: 78–85. [CrossRef]

Intel. 2017a. PoET 1.0 Specification. Available online: https://sawtooth.hyperledger.org/docs/core/releases/latest/architecture/poet.html (accessed on 24 November 2019).

Intel. 2017b. Hyperledger Sawtooth Raft Documentation. Available online: https://sawtooth.hyperledger.org/docs/raft/nightly/master/ (accessed on 24 November 2019).

Kemmerer, Edwin Walter. 1994. *Gold and the Gold Standard: The Story of Gold Money Past, Present and Future*. Princeton: McGraw-Hill Book, Company, Inc., p. 134. ISBN 9781610164429.

Keynes, John Maynard. 1920. *Economic Consequences of the Peace*. New York: Harcourt, Brace and Howe.

King, Sunny, and Scott Nadal. 2012. PPCoin: Peer-to-Peer Crypto-Currency with Proof-of-Stake. Available online: https://peercoin.net/assets/paper/peercoin-paper.pdf (accessed on 24 November 2019).

Kotla, Ramakrishna, Lorenzo Alvisi, Mike Dahlin, Allen Clement, and Edmund Wong. 2009. Zyzzyva: Speculative Byzantine Fault Tolerance. *ACM Transactions on Computer Systems* 27: 7. [CrossRef]

Lamport, Leslie, Robert Shostak, and Marshall Pease. 1982. The Byzantine Generals Problem. *ACM Transactions on Programming Languages and Systems* 4: 382–401. [CrossRef]

Li, Xiaoqi, Peng Jiang, Ting Chen, Xiapu Luo, and Qiaoyan Wen. 2018. A survey on the security of blockchain systems. *arXiv*. [CrossRef]

LKC. 2018. LuckyBlock Whitepaper. Available online: https://luckyblock.com/static/whitepaper.pdf (accessed on 24 November 2019).

LTC. 2018. Comparison between Litecoin and Bitcoin. Available online: https://litecoin.info/index.php/Comparison_between_Litecoin_and_Bitcoin (accessed on 24 November 2019).

Mayer, David A. 2010. *Gold standard at Google Books the Everything Economics Book: From Theory to Practice, Your Complete Guide to Understanding Economics Today (Everything Series)*. New York: Simon and Schuster, pp. 33–34. ISBN 978-1-4405-0602-4.

Nakamoto, Satoshi. 2008. Bitcoin: A Peer-to-Peer Electronic Cash System. *arXiv*.

NAS. 2018. NAS White Paper. Available online: https://nebulas.io/docs/NebulasTechnicalWhitepaper.pdf (accessed on 24 November 2019).

NEM. 2018. NEM White Paper. Available online: https://nem.io/wp-content/themes/nem/files/NEM_techRef.pdf (accessed on 24 November 2019).

NEO. 2018. NEO White Paper. Available online: https://github.com/neo-project/docs/blob/master/en-us/index.md (accessed on 24 November 2019).

Nguyen, Giang-Truong, and Kyungbaek Kim. 2017. A Survey about Consensus Algorithms Used in Blockchain. *Journal of Information Processing Systems* 14: 101–28.

NXT. 2015. NXT Whitepaper (Blocks). Revision 4, Nxt v1.2.2. Available online: https://bravenewcoin.com/assets/Whitepapers/NxtWhitepaper-v122-rev4.pdf (accessed on 24 November 2019).

Ongaro, Diego, and John Ousterhout. 2014. In Search of an Understandable Consensus Algorithm. Paper presented at the 2014 USENIX Annual Technical Conference (USENIX ATC '14), Philadelphia, PA, USA, June 19–20; pp. 305–19.

ONT. 2017. Ontology Technical W. Available online: https://github.com/ontio/Documentation/blob/master/Ontology-technology-white-paper-EN.pdf (accessed on 24 November 2019).

Park, Sunoo, Krzysztof Pietrzak, Joël Alwen, Georg Fuchsbauer, and Peter Gazi. 2015. Spacecoin: A Cryptocurrency Based on Proofs of Space. Available online: https://eprint.iacr.org/2015/528 (accessed on 24 November 2019).

Pike, Douglas, Patrick Nosker, David Boehm, Daniel Grisham, Steve Woods, and Joshua Marston. 2015. Proof-of-Stake-Time. Available online: https://vericonomy.ams3.cdn.digitaloceanspaces.com/documents/VeriCoinPoSTWhitePaper10May2015.pdf (accessed on 24 November 2019).

Rosic, Ameer. 2018. What Are Ethereum Nodes and Sharding? Available online: https://blockgeeks.com/guides/what-are-ethereum-nodes-and-sharding/ (accessed on 24 November 2019).

Schwartz, David, Noah Youngs, and Arthur Britto. 2014. The Ripple Protocol Consensus Algorithm. Available online: https://ripple.com/files/ripple_consensus_whitepaper.pdf (accessed on 24 November 2019).

Shoker, Ali, and Jean-Paul Bahsoun. 2013. BFT Selection. Paper presented at the International Conference of Networked Systems, Marrakech, Morocco, May 2–4.

Smola, Alex J., and Bernhard Schölkopf. 2004. A tutorial on support vector regression. *Statistics and Computing* 14: 199–222. [CrossRef]

Sultan, Karim, Umar Ruhi, and Rubina Lakhani. 2018. Conceptualizing Blockchains: Characteristics & Applications. Paper presented at the 11th IADIS International Conference Information Systems 2018, Lisbon, Portugal, April 14–16; Available online: https://arxiv.org/pdf/1806.03693v1.pdf (accessed on 24 November 2019).

Tschorsch, Florian, and Björn Scheuermann. 2016. Bitcoin and Beyond: A Technical Survey on Decentralized Digital Currencies. *IEEE Communications Surveys & Tutorials* 18: 2084–123. [CrossRef]

VET. 2018. VET White Paper. Available online: https://github.com/vechain/thor (accessed on 24 November 2019).

Wang, Wenbo, Dinh Thai Hoang, Peizhao Hu, Zehui Xiong, Dusit Niyato, Ping Wang, Yonggang Wen, and Dong In Kim. 2019. A Survey on Consensus Mechanisms and Mining Strategy Management in Blockchain Networks. *IEEE Access* 7: 22328–70. [CrossRef]

Watthananona, Julaluk, and A. Mingkhwanb. 2012. Optimizing Knowledge Management using Knowledge Map. *Procedia Engineering* 32: 1169–77. [CrossRef]

Weber, Warren E. 2015. Available online: http://www.bankofcanada.ca/wp-content/uploads/2015/12/bitcoin-standard-lessons.pdf (accessed on 24 November 2019).

WTO. 2015. Understanding the WTO, Handbook. Available online: https://www.wto.org/English/Thewto_E/whatis_e/tif_e/understanding_e.pdf (accessed on 24 November 2019).