Blockchain Temporality: Smart Contract Time Specifiability with Blocktime

Melanie Swan(✉)

Philosophy and Economic Theory,
New School for Social Research, New York, NY, USA
m@melanieswan.com
http://www.BlockchainStudies.org

Abstract. The aims of this paper are to (1) provide a conceptual context for smart contracts, (2) argue that blockchains are a next-generation technology enabling much larger-scale and more complex computing projects, and (3) posit blocktime as a new mode of conceiving time. Blockchains are the distributed ledger technology underlying Bitcoin and other cryptocurrencies; the payments layer the Internet never had; a mechanism for updating truth states in distributed network computing through consensus trust; and overall, a new form of general computational substrate. Blocktime is the time over which a certain number of blocks will have confirmed; and this creates an alternative event trajectory in time which can be offset against human-time or other computing clocktime regimes for arbitrage or complementary purposes. The result of this effort is to show that blocktime allows the contingency of future events to be more robustly orchestrated through temporality as a selectable smart contract feature.

Keywords: Bitcoin · Cryptocurrency · Blockchain · Temporality · Algorithmic trust · Information theory · Distributed computing · Decentralization · Network computing · Byzantine agreement

1 Introduction

1.1 Background Context

I have divided the paper into three sections toward the aims of providing a conceptual context for smart contracts, arguing that blockchains are a next-generation technology enabling much larger-scale and more complex computing projects, and positing blocktime as a new mode of conceiving time. First, I discuss computational substrates. I argue that current computational models may be limited in scope and unable to scale to address the next tiers of computing challenges. Some of these computing projects could include genome and microbiome research banks with billions of files, national property registries, searchable government records databases, astronomical data management, unified Electronic Medical Record systems, and Internet of Things (IoT) connected sensor and device coordination. I suggest that blockchain technology could be one solution for creating a next-generation computational architecture to address these kinds of larger-scale computational projects that have been impossible before. The payments layer installs a valuable functionality that allows remuneration,

© Springer International Publishing Switzerland 2016
J.J. Alferes et al. (Eds.): RuleML 2016, LNCS 9718, pp. 184–196, 2016.
DOI: 10.1007/978-3-319-42019-6_12

m@melanieswan.com

which could support an accelerated move to the automation economy in enabling a more sophisticated level of secure automated payments. For example, eventually the entire mortgage servicing industry could be outsourced to a package of smart contracts. Second, I discuss the properties of blockchain computing such as Byzantine Agreement and consensus algorithms, and new classes of blockchain applications. I suggest that blockchains comprise a new and unique form of computing system wherein trust, transparency, and entropy are reliably produced and persist over time. Third, I discuss Turing completeness and blocktime. Blockchains, and particularly smart contract platforms, are a general universal computing substrate in the Turing-complete sense, a generic computational infrastructure. I posit the idea of blocktime that makes time more malleable as a specifiable parameter of smart contracts, and offers a tool for managing the contingency of future events. Previously, the available time selections were generally restricted to human time and computing event time. Now however, there is blocktime (the time over which a certain number of blocks will have confirmed), which is a separate time regime unto itself that can be played off and against other time trajectories, and allow processes to be configured internally within the realm of blocktime. I conjecture that there is a sense of the possibility of creating “more time” by being able to access events in alternative time trajectories, like blocktime. A practical example is that I can be earning cryptocurrency with my numerous and parallel blockchain DACs (distributed autonomous corporations) that I can swap out to fiat currency to pay my physical-world obligations. The key point is starting to conceive of time in this unprecedented way as a malleable resource that can be specified in different ways as a contract feature.

1.2 Computational Substrates

A general computational substrate may be conceived as a platform upon which calculations related to information processing may be performed. Numerous computation substrates have been proposed and developed to different degrees. The most obvious and familiar in existence is the worldwide silicon chip-based computing infrastructure. Other platforms are in development such as quantum computing. In biology, there are suggestions for computational models using molecular nanotechnology, positional assembly, social network graphs, and ant colonies and other swarm-coordinated behaviors. DNA has been proposed as a miniaturized and durable means of storage and computation [1]. Organic-inorganic hybrid computational substrates have been outlined, for example Brainets (linking organic computing units (brains) to silicon computing networks) [2], and Neural Dust (thousands of 10–100 micron-sized in-brain sensor nodes providing neural recording and interface support) [3].

While the existing silicon-chip based computational infrastructure is ubiquitous in some sense, it has some challenges. First, it is not a general computational substrate upon which any program can run fungibly. There are many different kinds of machines, operating systems, languages, software versions, and installed configurations which can prevent even seemingly interoperable software programs from running in a new environment. One recent strategy designed to address this is executing software applications inside Docker containers which do not require underlying machine

m@melanieswan.com

configurations to be a certain way. Second, there is an ongoing explosion in the number and species of Internet-connected devices. 20–30 billion Internet-connected devices are estimated to be online worldwide by 2020 [4]. At the same time, there are more different kinds of computing devices. The computing world is no longer just servers and PCs; it includes drones, robots, self-driving vehicles, IoT (Internet of Things) sensors, smart phones, wearables, smart roads, and other devices. Each new species has its own processing requirements and protocols, and a different kind of infrastructure might be required to manage the traffic of all of the communication and coordination for these platforms. A third factor is the need to accommodate a higher magnitude of very large data files, for example nation-wide EMRs (electronic medical records) or million-member genome banks. In order to progress to a new era of computing that incorporates the IoT explosion, very-large data files, and interoperability, a more universal computing schema is needed, a truly general computational substrate that can handle the magnitude of IoT device messaging, that includes a remunerative payments layer, that is a truly next-generation global infrastructure, and this could be blockchain technology.

Blockchain technology is a new arrival in computational substrates. A blockchain is a software protocol and decentralized ledger for recording transactions, but more fundamentally blockchains are a global-scale computational substrate for the processing of any kind of digitized activity. Blockchains are a general base for computation [5, 6]. The first application of blockchain technology is cryptocurrencies, where the key property is being able to securely update truth states in a distributed computing network. This has been a known challenge called the Byzantine Generals Problem; e.g.; how to communicate effectively across a distributed war field of generals, not knowing which generals might be compromised [7]. One way of reaching Byzantine Agreement or Byzantine Consensus in computing systems has been needed. Several solutions have been proposed in the previous decades, and finally blockchains have a number of checks and balances such that reaching secure and accurate consensus across networks despite any failing nodes (malicious or otherwise) might be more reliably achieved. Blockchains are a software technology for updating every node in a distributed computing system with the current state of the world; a means of conferring a shared truth state in a distributed system. Blockchains are an important innovation for large-scale activity in both computing and social cohesion, where some of the social layers created are economic remuneration and distribution, and societal shared trust that is simultaneously global and local, and can facilitate human and machine interaction and collaboration.

A blockchain is like a giant interactive Google doc spreadsheet that anyone can view on-demand, where independent administrators (miners) continually verify and update the ledger to confirm that each transaction is valid. It is called a blockchain because blocks or batches of transactions are posted sequentially to a ledger, and each new block starts by referencing the prior block, so a chain of blocks is created. The result is that a secure network is created where any transaction can be independently confirmed as unique and valid without a centralized intermediary like a bank, government, or other institution. Creating trust in a distributed computing system without an intermediary (Byzantine Consensus) had been an unsolved computing problem with many other attempts at producing a workable digital cash solution failing. Blockchain

m@melanieswan.com

technology is called *trustless* in the sense of not needing to trust the counterparty but instead trusting the blockchain software system. Trust is created by using the software system, as opposed to the old model of trust, which was the need to know and trust the counterparty of the transaction. Some of the implications of trusting the software system instead of having to find and trust counterparties is that not only is there more freedom with whom one can transact (essentially any human or machine agent across the global Internet), but also there is a much larger scale of transactions that can occur. There is a worry that the extreme openness of transactability on blockchain networks enables illegal criminal activity, most notably operations like Silk Road, but more fundamentally blockchains are a technology like the Internet, which too was initially used for illicit activity (some thought it would not progress beyond pornography), but quickly became an indispensable infrastructural element for coordinating and expanding all human and machine activity. The same could be true for blockchains.

Cryptocurrencies like Bitcoin are one of the first applications using blockchain technology. Bitcoin is like 'Skype for money;' [8] performing the same transformation that Skype did for phone calls in the context of digital cash, or what email did for post office mail, which is move physical world processes with plant and materials into more efficient digital network models. Bitcoin is the first robust demonstration of blockchain technology and decentralized models more generally. Since the middle ages, hierarchical models have been the primary means of organizing large-scale activity and they work up to a point, however now decentralized models are a striking new entrant in the possibility space of the models for large-scale coordination. Further, decentralized models suggest particular traction in coordinating truly global-scale activity at a larger and more complicated level than has been previously possible. Decentralized technologies could mark the next node in the evolution of humans and computing, and are required in the contemporary big data era to orchestrate projects such as effectively-sized health data commons for research involving thousands or millions of whole human genome files beyond the mere 3,751 that have been amassed so far [5]. It is not that centralized hierarchical models would be replaced overnight; the longer-term future could be one of a coexistence of many different kinds of organizational models: centralized, decentralized, and hybrid structures, and other new forms of models (for example based on complexity), where the important dynamic becomes tuning the orchestration system to the requirements of the underlying situation.

2 Properties of Blockchain Computing

Blockchains are a universal, large-scale, global, detailed, distributed, permanent transaction record available for on-demand look-up at any future moment. It is a system wherein trust, transparency, and entropy are reliably produced and persist over time. Computation takes place in the blockchain model in different ways. The first computational area is mining, the process by which independent third parties (miners) validate and record transactions. The blockchain software system automatically packages submitted transactions (on the order of thousands) into blocks, creates a random number specific to the block, and publishes metadata about the block parameters (cryptographic difficulty, service string, nonce (32-bit number), and counter

m@melanieswan.com

(https://en.bitcoin.it/wiki/Block_hashing_algorithm)). Then anyone running the mining software performs computations and submits cryptographic guesses as to the specific parameters and nonce of the block. The mining machine with the correct guess wins the right to actually record the transactions, and receives the block rewards (transaction fee) for doing so.

2.1 Byzantine Agreement

One of the reasons that blockchain technology is such an advance is that it provides Byzantine agreement, a long-sought means of truth-state updating and trust generation in distributed computing networks. In the general space variously labeled as Byzantine fault tolerance, Byzantine consensus, and Byzantine agreement, a number of solutions have been proposed [9]. These consensus protocols are all some form of Byzantine agreement about how to arrive at secure trustable truth state updating in a consensus model in a distributed computing network. Consensus protocols can be seen in different modes of development. First there were Byzantine Agreement Protocols (BAPs) for the synchronous updating of network nodes. Beginning in the 1980s, these protocols include the Paxis algorithm for state machine replication from Lamport and Microsoft. Then Google's Chubby algorithm is a next-generation of Paxos, focused on the ability to serve strongly consistent files. Since it is not feasible to update very-large network systems of worldwide distributed nodes synchronously, more recently asynchronous models have been proposed.

Thus in the second moment of evolution, there are the different asynchronous updating algorithms proposed by blockchain technology. These include 'Nakamoto Consensus,' the proof-of-work model used with the Bitcoin blockchain, which is effective, but expensive and high latency. The proof-of-stake model is also here, which requires resource ownership, but has the risk of 'nothing-at-stake' attacks per escrow-revoking by malicious agents. There is now a third class of asynchronous Byzantine agreement consensus protocols under development for the longer-term future of cryptographic blockchain models. Some proposed models here include ARBC (Asynchronous Randomized Byzantine Consensus) from Pebble which combines traditional Byzantine Agreement Protocols with Nakamoto chains as a randomness source for faster and more-scalable decentralized networks. Other proposals are the BAR (Byzantine, altruistic, rational) protocol from the University of Texas at Austin, and the Stellar Consensus Protocol based on Quorum Slicing (trusting and updating via next-neighbor nodes, not the network as a whole). Prediction markets have been suggested as a longer-term alternative for reaching trustable truth-state consensus in distributed computing networks.

2.2 Blockchain Supercomputing

In the last two years, Bitcoin has arrived rapidly out of nowhere and created what is noted by some as being by far the world's largest and fastest computing network. While the Bitcoin blockchain is currently used to conduct necessarily wasteful

m@melanieswan.com

cryptographic trail-and-error guessing for transaction recording, it might be more widely conceived as a computational resource and deployed in applications well beyond proof-of-work mining. Some projects seeking to harness otherwise wasted mining cycles into wider computational use include Primecoin (prime number factoring) [10], GreenCoin (carbon credit offsets) [11], and Gridcoin [12] and FoldingCoin (rewarding and facilitating community computing projects) [13].

All of the worldwide computers running the Bitcoin mining operation collectively comprise the world's biggest and fastest supercomputer. Bitcoin reached 1 PetaHash per second (PH/s) of computing power/speed on September 15th, 2013. In 2015, the Bitcoin network has been routinely operating at over 350 PH/s, or over 350,000,000 GH/s [14], specifically at 380 PH/s as of August 2015 (http://www.bitcoinwatch.com/). Bitcoin's hash-rate is the total computing power of the network, defined as the number of SHA-256 cryptographic hashes (or guesses) it can compute per second. Comparisons could be made with Google, who is estimated to have 10 million servers comprising one PetaHash (Smart 2015), and the estimated 2 billion worldwide personal computers thought to comprise 20 PetaHash [15]. Another comparison is vis-à-vis supercomputers, where the world's largest supercomputer, China's Tianhe-2 (MilkyWay-2) at the National University of Defense Technology has a performance of 33.86 PetaFLOPS (quadrillions of calculations per second) [16], compared to Bitcoin's network hash rate of 4,858,117.28 PetaFLOPS.

2.3 Blockchain Consensus Algorithms

A crucial part of blockchain computing is the consensus protocol by which transactions are confirmed. The Bitcoin blockchain runs on *proof of work*, where there is a high cost to demonstrate a proof of work; the mining operation has to spend the cost and energy of doing 'real work' to make guesses at the cryptographic nonce. This is the model to deter malicious players and produce a distributed system that is trustworthy. Since cryptocurrencies involve money and financial assets, there is incentive to game the system and crypto-security must be high. However, proof-of-work mining is expensive and may not be a long-term sustainable model for consensus derivation. Therefore other less-intense mining protocols might be employed as the transaction confirmation mechanism. Another familiar proposed model is *proof of stake*, where mining participation is determined by asset ownership in the mining system (thus possibly better aligning incentives to maintain a correct and orderly system). The proof of stake miners would own a stake in the mining operation, but not necessarily be the transaction parties or otherwise connected to the transactions, so the mining operation would still be a separate and independent function from the transactions. As an example of these protocols in practice, the smart-contract platform Ethereum has launched with a proof of work model, and then envisions shifting to a proof of stake model for the system to mature into a steady-state model. Several other consensus protocols for Byzantine Agreement are in development and discussion for other means of arriving at trustable truth states in decentralized networks. For example, there is proof of existence (time-date-stamped proof of a certain document or digital asset existing in a certain state at a certain time), and proof of truth (proof of a truth state having occurred, such as an automobile being damaged in a collision). Other more abstract models for proving agent ability to participate in consensus validation and confirmation processes could include *proof of entropy*, *proof of intelligence*, *proof of reputation*, and *proof of capability*; all as a means of demonstrating some sort of trustable proof of ability to do something as a means of access-granting to systems platforms, resources, or activities. *Proof of n* as an access mechanism to digital smartnetwork assets is a futuretech concept that could be quite extensible.

2.4 New Classes of Blockchain Applications

Blockchains are a computational model in how they themselves operate, and also in the new classes of computing operations that they enable. By analogy, there are the layers of computation that facilitate the Internet's own operations as a network, and the numerous additional layers of computational applications for which the Internet is an input and infrastructure; and this could be similarly true with blockchains. The bigger endgame with blockchains is their use as a basis for many new classes of applications in areas ranging from economics and finance to legal services and governance to health and science to literacy and art. Some practical applications could be decentralized credit bureaus, open-source FICO scores, and literacy smart contracts. Another example is health data analysis, where blockchains could be used as the infrastructure and permissioning mechanism for coordinating secure access to various big health data streams from millions of persons, using blockchain pointers to secure off-chain stored files. A firm in this space is DNA.bits, offering a blockchain-based solution for the de-identified continuous sharing of genetic and correlated clinical data. A vast global-scale health commons database could be created that is decentralized and unassembled, queryable on demand. Deep-learning algorithms could then be run over this massive decentralized datastore, possibly creating the crucial large data corpora which have been established as a necessary condition for advance in artificial intelligence [17, 18], and could lead to significant medical discovery.

New theories, modes, and means of computation may be required to work with the new kinds of vastly-larger datasets that could become available and workable for worldwide processing with blockchains. Right now the epitome of computing is large centralized datastores like Google's estimated 10 million servers continually crawling the web. Now, however, blockchains invite a completely new conceptual paradigm in computing systems, one that is a completely distributed decentralized blanket of available resources that can be called upon as needed. Computing as a ubiquitous reliably-available flexible resource creates a new reality, one based on certainty, trust, and assurity; one of abundance as opposed to scarcity and constraint. This resource can be seen and experienced at different levels, from a universal computational substrate, to its higher-level applications such as a distribution mechanism for GBI (guaranteed basic income) initiatives and in the farther future, for the safely orchestrated participation in collaborative cloudminds. Philosophically, blockchains thus contribute to the constitution of a conceptually different reality, one where computing as a paradigm is pushed farther into the position of being a seamlessly available background resource like air.

m@melanieswan.com

2.5 Blockchains and Complexity

An argument can be made that blockchains are systems of general complexity, as set forth by Morin [19]. General complexity systems are those that are non-linear, emergent, open, unknowable at the outset, interdependent, and self-organizing; an accurate descriptor of blockchains in their current early evolutionary moments. What is important in systems of general complexity is the relationality between the components as opposed to the parts or the whole, or the beginnings, endpoints, and boundaries of the system. Morin's *general complexity* is distinct from *restricted complexity*, where restricted complexity the position that despite the intricate and complicated nature of complex systems, the underlying rules may become known and enumerated through scientific study. The other position, general complexity, is that an approach that is itself complexity-congruent is potentially a more accurate investigatory stance towards complex systems, especially in the case of blockchains as complex systems, including since they themselves are still evolving. Blockchains are complex systems and also generators of complexity. They reliably create randomness, indeterminacy, and entropy as it is not known or predictable ahead of time which node will 'win' the right to confirm the next block by correctly guessing the cryptographic nonce. This feature of blockchains as a robust, reliable, persistent, global source of entropy generation is being proposed for use in a number of applications.

The reason that complexity is important is that complex systems are a new kind of technology which might accommodate precisely the next phase of larger global scale projects like million-member genome banks that traditional linear hierarchical models are unable to address. One example is the idea of Blockchain Supercomputing. One of the biggest evolutionary needs in supercomputing is to address new tiers of more sophisticated computational problems, expanding beyond simple linearized parallel processing methods into situations of greater computational complexity, including with currently contemplated desktop and peer-to-peer grid computing, and beyond. Blockchains, particularly with their complexity properties, could be a model to configure new forms of non-linear supercomputing problems. The Bitcoin mining network is the biggest supercomputer we have ever built, and what does this mean? It is used for transaction confirmation and shifting balances between wallet addresses but could be used more broadly for anything.

3 Turing Completeness and Blocktime

While initial blockchain projects like Bitcoin-based cryptocurrencies are specifically not Turing-complete [20] and focus computationally on unspent transaction balances, the second generation of projects, smart contract platforms like Ethereum (launched July 2015) [21] and Eris Industries [22] are designed to be Turing-complete in the sense of running any program. Smart contract platforms accommodate more complicated validation and confirmation functionality including vast value-chain ecologies with independent truth oracles, escrow services, and multi-signature contract co-signing parties. Having Turing-complete platforms could allow a full and portable class of computing problems to be addressed, including orchestrating uncertain future

m@melanieswan.com

events. Digital cryptocurrencies could be conceived as blockchain computing 1.0, and smart contract platforms, essentially Turing-complete state-change machines as blockchain computing 2.0, and connote a completely different tier of computational complexity. Smart contracts have a number of important features related to computational complexity.

Definitionally, smart contracts are as any contract, agreements between parties, but in this case, posted to the blockchain for some sort of automated execution. Smart contracts may be (1) compliant, in accord with current legal regimes as legal contracts with the four required features of mutual assent, consideration, capacity, and legality, or (2) a-compliant, operating in a-legality outside of current regulatory mechanisms. Smart contracts are state-change machines; they are launched and await events or changes in conditions to update their states. These code-contracts (as opposed to discretionarily-enforced human contracts) will execute inexorably. They can call each other in a near-infinite complexity and be used as the architecture for autonomous entities, DAOs, Dapps, DACs, DASs, and DCOs (distributed autonomous organizations, applications, corporations, societies; distributed collaborative organizations), propelling the automation economy forward.

3.1 Temporality as a Feature

Blockchains are an important reality-making technology, a mode and means of implementing many different flavors of “crypto-enlightenment.” This includes newer, flatter, more autonomous economic, political, ethical, scientific, and community systems. But not just in the familiar human social constructs like economics and politics, possibly in physical realities too like time. Blocktime’s temporal multiplicity and malleability suggest a reality feature we have never had access to before – a way of possibly making more time. Blocktime as blockchains’ own temporality allows the tantalizing possibility of rejiggering time and making it a malleable property of blockchains. The in-built time clock in blockchains is blocktime, the chain of time by which a certain number of blocks will have been confirmed. Time is specified in units of transaction block confirmation times, not minutes or hours like in a human time system. Block confirmation times are convertible to minutes, but these conversion metrics might change over time (for example with block confirms being of the scale and frequency to convert to micro-minutes or nano-minutes).

3.2 Blocktime Arbitrage

One key point is that the notion of blocktime, as an extension of computing clocktime more generally, creates a differential. Blocktime and human time already exist as different time schemas. A differential suggests that the two different systems might be used to reinforce each other, or that the differential could be exploited, arbitraging the two time frameworks. Through the differential too is the way to ‘make more time,’ by accessing events in another time trajectory. The conceptualization of time in computer science is already different than in human time. Computing clocktime has more dimensions (discrete time, no time, asynchronous time, etc.) than human physical and biological time, which is continuous. Clocktime has always been different than human time. What is different with blocktime is that it builds in even more variability, and the future assignability of time through dapps and smart contracts. For example, MTL (machine trust language) time primitives might be assigned to a micropayment channel dapp as a time arbiter. Time has not been future-specifiable before, in the way that it can be assigned in blocktime smart contracts.

Temporality could be a standard smart contract feature. Time speed-ups, slow-downs, event-waiting, and event-positing (a true futures-class technology) could become *de rigueur* blocktime specifications. Even the blocktime regime itself could be a contract-specifiable parameter per drop-down menu, just like legal regime. Temporality becomes a feature as smart contracts are launched and await events or changes in conditions to update contract states. Time malleability could itself be a feature, arbitraging blocktime with real time. An example of a time schema differential arising could be for example, a decentralized peer-to-peer loan that is coming due in blocktime, but where there have not been enough physical-world time cycles available for generating the 'fiat resources' to repay the loan.

In blocktime, the time interval at which things are done is by block. This is the time that it takes blocks to confirm, so blockchain system processes like those involving smart contracts are ordered around the conception of blocktime quanta or units. This is a different temporal paradigm than human lived time. The human time paradigm is one that is more variable and contingent. Human time is divided and unitized by the vagaries of human experience, by parameters such as day and night; week, weekend, and holiday; seasons; and more contingently, crises, eras, and historical events. Since blocktime is an inherent blockchain feature, one of the easiest ways to programmatically specify future time intervals for event conditions and state changes in blockchain-based events is via blocktime. Arguably, it is easier, and more congruent and efficient, to call a time measure from within a system rather than from outside. It could be prohibitively costly for example, to specify an external programmatic call to NIST or another time oracle. Possibly the emerging convention could be to call NIST, including as a backup, confirmation, or comparison for blocktime. Currently, blockchain systems do not necessarily synchronize their internal clocktime with NIST, but the possibility of a vast web of worldwide smart contracts suggests the value and necessity of external time oracles, and raises new issues about global time measurement more generally. Especially since each different blockchain might have its own blocktime, there could be some standard means of coordinating blocktime synchronizations for interoperability, maybe via a time sidechain for example. The key point is starting to conceive of time in a mode which has been unprecedented; time is not a fixed given, time is a malleable resource that can be specified in different ways as a contract feature. In fact, I conjecture that the malleability of time engenders a sense of the possibility of creating "more time" by being able to access events in alternative time trajectories such as blocktime [23].

m@melanieswan.com

3.3 Computing Creates Novel Temporalities of Discontinuity and Prediction

First computing clocktime made time malleable through its different discontinuous forms. Then machine learning and big data facilitated a new temporality, one oriented to the present and future, instead of responding to just the past. There was a shift from only being able to react to events retrospectively after they had passed, to now being able to model, simulate, plan, and act in real-time as events occur, and proactively structure future events. The current change is that blockchains and particularly smart contracts add exponential power to this; they are in some sense a future reality-making technology on steroids. Whole classes of industries (like mortgage servicing) might be outsourced to the seamless orchestration of blockchain dapps and DACs in the next phases of the automation economy. While Bitcoin is the spot market for transactions in the present moment, smart contracts are a robust futures market for locking in the automated orchestration of vast areas of digital activity.

3.4 Blockchain Historicity: Computer Memory of Human Events

Blockchain logs are in a sense a human event memory server. Blockchains are already event history keepers, and now with blocktime could have even more responsibility as the memory computer of human events. It is now possible to think in terms of blockchain time sequences, in the anticipation and scoping of future events and activities, as blockchain reality unfolds, as opposed to human time scales and events. For example, there are normal human time sequences, like a one-year lease agreement. Other sequentiality is based on human-experienced conditions like ‘the park is open until dark,’ which makes little sense in a blocktime schema. There are time guidelines that vary per lived experience in human realities. Likewise, there could be analogs in lived experience in blockchain realities. Different events could mark the historicity of blockchains, for example, the time elapsed since the genesis block, and other metrics regarding number, amount, and the speed of transactions. Gesturing towards a crypto-philosophy, Hegel, Benjamin, Hölderlin, and Heidegger already have more malleable conceptions of historicity and temporality that might be instantiated in the blocktime paradigm. There is much more linkage and portability between past, present, and future (all arguably human constructions), for example, in ecstatic temporality, where the event from the future reaches back to inform the present now moment, as extended from the past [24].

4 Conclusion

In this paper, my contribution is to (1) provide a conceptual context for smart contracts, (2) argue that blockchains are a next-generation technology enabling much larger-scale and more complex computing projects, and (3) posit blocktime as a new mode of conceiving time. Blockchains are a universal general computing substrate in the Turing-complete sense: any computing problem can be formulated and run on blockchains as a universal computing platform. Not only can blockchains run any

m@melanieswan.com

program, they are an improved computational substrate because of their universality, accessibility, availability, scalability (both vertical (Merkle rooting) and horizontal (distributed network nodes)), always-on connection to the Internet, permanent record-maintaining, and auditable record-keeping. More broadly, blockchains are a new form of cryptographic software protocol and a programming paradigm for secure distributed computing. They could have a wide variety of uses in the implementation of digital currencies, financial and economic transfers; the administration, registration, and exchange of all forms of tangible and intangible assets as smart property; and the coordination of governance, legal, health, and scientific activity via smart contracts and distributed autonomous entities, ushering in a productive and trust-building era of human-machine collaboration. Computationally, blockchains provide an unprecedented fully-scalable universal worldwide computing infrastructure with built-in security and a remunerative payments layer. Blockchains could be the next evolutionary addition to the Internet by enabling a new degree of sophistication and resolution in computing. Thus there could be the start of a universal computing substrate, an always-on ubiquitous background resource, a blanket of secure processing that supports greater possibilities for human endeavor. Blockchains as a new core infrastructural tier of computational resource could prompt a reconception of computing; philosophically, mathematically, and practically. As a general computational substrate, blockchains expand the reach of computing, and this in turn expands the reach of our thinking and realizing in terms of what is possible in computing.

References

1. Connor, S.: Single DNA molecule could store information for a million years following scientific breakthrough. Independent (2015)
2. Pais-Vieira, M., Chiuffa, G., Lebedev, M., Yadav, A., Nicolelis, M.A.L.: Building an organic computing device with multiple interconnected brains. Nat. Sci. Rep. **5**, 11869 (2015)
3. Seo, D., Carmena, J.M., Rabaey, J.M., Alon, E., Maharbiz, M.M.: Neural dust: an ultrasonic, low power solution for chronic brain-machine interfaces (2013). arXiv:1307.2196 [q-bio. NC]
4. Bauer, H., Patel, M., Veira, J.: The Internet of Things: Sizing up the Opportunity. McKinsey and Co., New York (2014)
5. Swan, M.: Blockchain: Blueprint for a New Economy. O'Reilly Media, Sebastopol (2015)
6. Merkle, R.: DAOs, Democracy and Governance. Version 1.2 (2015)
7. Lamport, L., Shostak, R., Pease, M.: The Byzantine generals problem. ACM Trans. Program. Lang. Syst. **4**(3), 382–401 (1982)
8. Antonopoulos, A.M.: Mastering Bitcoin: Unlocking Digital Cryptocurrencies. O'Reilly Media, Sebastopol (2014)
9. Swan, M.: Blockchain Consensus Protocols. Bitcoin Meetup (2015). http://www.slideshare.net/lablogga/blockchain-consensus-protocols
10. Buterin, V.: Primecoin: The cryptocurrency whose mining is actually useful. Bitcoin Mag. (2013). http://primecoin.io
11. Dollentas, N.: Greencoin: carbon emissions coin. Bitcoinist (2014). http://www.grcoin.com

m@melanieswan.com

12. Cawrey, D.: 5 Global problems Bitcoin’s proof of work can help solve. CoinDesk (2014). http://www.gridcoin.us
13. Menezes, N.: Interview with the Foldingcoin team. Bitcoinist (2014). http://foldingcoin.net
14. Smart, E.: Bitcoin is 100 times more powerful than Google. Cryptocoin News (2015)
15. Gill, T.: Bitcoin hash-rate exceeds total computing power of all the world’s computers! Taran Gill Blog (2014)
16. Top 500: The List - June 2015. http://www.top500.org/lists/2015/06
17. Halevy, A., Norvig, P., Pereira, F.: The unreasonable effectiveness of data. IEEE Intell. Syst. **24**(2), 8–12 (2009)
18. Le, Q.V., Ranzato, M., Monga, R., Devin, M., Chen, K., Corrado, G.S., Dean, J., Ng, A.Y.: Building high-level features using large scale unsupervised learning (2011). arXiv:1112.6209 [cs.LG]
19. Morin, E.: Restricted complexity, general complexity. In: Gershenson, C., Aerts, D., Edmonds, B. (eds.) Worldviews, Science and Us: Philosophy and Complexity, pp. 5–29. World Scientific, Singapore (2007). Trans. by, Gershenson, C.
20. Nakamoto, S.: Bitcoin: a peer-to-peer electronic cash system (2008)
21. Liang, C.C.: A next-generation smart contract and decentralized application platform. Ethereum White paper (2016). https://github.com/ethereum/wiki/wiki/White-Paper
22. Lewis, A.: In a nutshell: Eris (Epicenter Bitcoin Interview – January 2016). Bits on Blocks Blog (2016)
23. Swan. M.: Temporality of the Future: A New Theory of Time: X-tention is Simultaneously Discrete and Continuous. Institute for Ethics and Emerging Technologies (2016). http://www.slideshare.net/lablogga/temporality-of-the-future
24. Heidegger, M.: Being and Time, pp. 1–474. Harper Perennial Modern Classics, New York (2008)

m@melanieswan.com