The Rise of Enforceable Business Processes from the Hashes of Blockchain-Based Smart Contracts

Sara Migliorini[1(✉)], Mauro Gambini[1], Carlo Combi[1], and Marcello La Rosa[2]

[1] Department of Computer Science, University of Verona, Verona, Italy
{sara.migliorini,mauro.gambini,carlo.combi}@univr.it
[2] School of Computing and Information Systems, The University of Melbourne, Melbourne, Australia
marcello.larosa@unimelb.edu.au

Abstract. Over the past few decades, the automation of inter-organizational processes has been the focus of several research efforts that have produced a broad spectrum of design methods and technologies. Recently, some experiments have shown how in principle a Decentralized Autonomous Process-Aware Information System (DAIS) can be implemented by means of Blockchain-based Smart Contracts (BSCs). In this paper, we cast a shadow on this novel approach by arguing that such kind of contracts cannot be considered an optimal abstraction to specify inter-organizational process models. We base our analysis on contractual incompleteness, a pivotal concept in widely accepted economic theories. We identify the main weakness in the conflict between the immutability-by-default of the BSCs and the nature of inter-organizational processes. As a result of this analysis, we introduce the concept of *enforceable business process* that is more in line with the original idea of smart contract and extends it to better match the essential requirements of a DAIS.

Keywords: Blockchain · Smart Contracts · Network coalition · Process-Aware Information System · Enforceable Business Process

1 Introduction

Modern life is embodied in a striking complex socio-economic infrastructure. In many countries, such infrastructure is effective enough that people can take for granted the availability of a wide range of products and services, safely ignoring the complexity of the underlying processes and means of production. The socio-economic infrastructure is made of people that takes part in several emergent or consolidated forms of organization. We can identify at least three ideal forms that we call here *competitive market*, *hierarchical firm* and *network coalition* [8,9,12].

S. Migliorini and M. Gambini—Member of the IEEE Blockchain Technical Community.

I. Reinhartz-Berger et al. (Eds.): BPMDS 2019/EMMSAD 2019, LNBIP 352, pp. 130–138, 2019.
https://doi.org/10.1007/978-3-030-20618-5_9

In their essence, the competitive market is an emergent form of coordination, the hierarchical firm is a designed form of cooperation and the network coalition is a concerted form of cooperation, where cooperation is intended here as the process of coordinating a group of actors with the explicit purpose of achieving a common goal. Supply chains, strategic business alliances, joint ventures, and also decentralized cryptocurrencies can be good examples of coalitions. Conversely, consortia are rarely considered coalitions, because their inter-organizational processes are related to the optimal allocation of shared resources, not necessarily to the achievement of a common goal.

This paper focuses on the problems that coalitions have to face in automating inter-organizational processes: due to their cooperative and exogenous nature, these processes can be neither delegated to the market nor managed entirely inside the boundary of a single hierarchical firm. To fix these concepts, we define an inter-organizational business process, or more briefly an *Exogenous Business Process (XBP)*, as a cooperative stable process, managed by a coalition of two or more mutually independent organizations, in order to pursue a common goal. By postulating its stability, we are emphasizing that the process is repetitive in nature and its identity is preserved over time, while being mutually independent implies the lack of an established central authority. An XBP often arises from repetitive casual market interactions, a typical example is a firm that acquires from the market a specific component on a regular basis. In order to reduce uncertainty and lower transaction costs, such firm can steadily choose the same supplier, building up a trustworthy relation that can be eventually formalized by a contract. Broadly speaking, any organization is established through a contract expressed in more or less formal terms. Here we consider a contract any agreement on a collection of mutual obligations that after being accepted by the parties, can be enforced in some way, for instance by social pressure or law.

The aim of this paper is to investigate how XBPs can be designed, enacted and monitored by a network coalition. In particular, we examine the limits in automating such processes by means of *Blockchain-based Smart Contracts* (*BSCs*) provided by the existing blockchain technology [1,3]. In doing this, we will attempt to face the challenges exposed by Mendling et al. in their exploratory work about how blockchain-based systems can support inter-organizational processes [6]. As emphasized by the authors, there is no general acceptance on how these processes can be managed. Several technologies can be deployed to streamline and automate them, but no one seems able to offer a complete solution. They also speculate that smart contracts and the blockchain technology can provide a new way to overcome these obstacles. Unfortunately, this new approach comes with its own challenges: it not clear how XBPs should be modeled as BSCs and then interpreted by an effective trustless Decentralized Autonomous Process-Aware Information System (DAIS).

In this paper, we argue that BSCs are too limited and not very suitable to automate XBPs, because they neglect renegotiation. The Caterpillar project [3] shows how in principle process models can be mapped to one or more BSCs and executed on a blockchain platform, but this mapping does not change the underlying semantics of such contracts. The lack of support for renegotiation in the current BSCs does not dismiss the potentiality of the blockchain technology, but this problem has to be carefully stated before proposing new solutions. We identify in the immutability-by-default of BSCs one of their major drawbacks that does not fit very well with the nature of network coalitions which are rooted on long-term incomplete contracts. Contractual incompleteness is a pivotal concept of several economic theories that can be traced back to the seminal work of Williamson [11], Hart and Holmström [5]. To the best of our knowledge, this is the first time that contractual incompleteness is used for shedding new light on the inter-organizational process automation. We use such concept to characterize XBPs, analyze the limits of BSCs and capture the essential requirements of a DAIS. In order to scale up, network coalitions require not only enforceable and verifiable, but also easily renegotiable contracts that can foster a continuous improvement of their contractual terms. We capture these features with the notion of *Enforceable Business Process* (*EBP*) which is essentially an XBP model archetype representing a renegotiable smart contract. The notion of EBP could be a good starting point to design an effective DAIS.

2 Incomplete Contracts

In an ideal world with a costless and flawless legal system, opportunistic behavior can be prevented by drawing up a complete contract that precisely dictates, for any future eventuality, which actions the parties should take, with the related incentives and penalties and no trust needs to be factored in. Unfortunately, law enforcement can be very expensive and actual contracts are far from being complete. Contracts are often poorly written, ambiguous and even purposely silent in many respects, to the point that virtually any contractual dispute brought before the courts concerns an incompleteness problem [4,5].

In the last few decades, prominent economists have recognized that many relevant economic phenomena can be explained in terms of contractual incompleteness, making the notion of *incomplete contract* a pivotal concept of several theories, such as Transaction Cost Economics and Contract Theory [5,11]. Contractual incompleteness can have manifold origins, for instance (1) the impossibility to predict all the relevant eventualities, (2) the costs to describe all the identified eventualities in advance, (3) the impossibility to observe certain actions of the contracting parties, and (4) the difficulty to make the observable actions verified by a trusted third-party. During the contracting process, the parties evaluate costs and benefits of including additional contractual terms. In light of these costs, even if in principle the parties would be able to draw up a complete formal contract, they can rationally choose to omit several details and leaving out many unlikely eventualities. Given these facts, it is reasonable to assume that any contract, even the most formal one, could be *intentionally* incomplete. Short-term contracts representing occasional relationships in a market setting do not escape this logic: an ideal contract of this kind should include a formal specification of the offered product, but this practice can be unacceptably expensive for any nontrivial product. These problems are exacerbate in case of long-term contracts on which network coalitions are grounded. In long-term interactions, purposely incomplete contracts can mitigate renegotiation costs, but at the expense of a higher dependence on trust.

3 Szabo's Smart Contracts (SSCs)

The original concept of *smart contract* has been introduced by N. Szabo to ease the definition of legal contracts and reduce the related transaction costs [10]. A *Szabo's Smart Contract (SSC)* is a computerized transaction protocol that formalizes and secures a set of relationships over computer networks. It is an agreement between two or more parties that can be automatically enforced without the need for a trusted third-party intermediary. An SSC should communicate the protocol semantics to the parties through good visual representations of the contract elements, so that each party has a clear understanding about the contract content. We generalize this requirement saying that any SSC realization should foster *intelligibility* as a primary concern.

The essential property that makes a contract *smart* w.r.t. a traditional one is its *automatic enforceability*, namely the execution of an SSC can be forced without relying on a trusted intermediary, because replaced by a trustworthy machine or network. Besides intelligibility and automatic enforceability, an SSC should ensure three other important properties: observability, verifiability, and privity. *Observability* is the ability of each party to observe the actions performed by others, or similarly to prove the execution of some actions to other parties. *Verifiability* is the ability of a party to prove to an adjudicator that a contract has been poorly performed or breached. Finally, *privity* states that no third-party has control over the enforcement of a contract, except for the appointed intermediaries or adjudicators but only in case of a dispute. These properties can substantially reduce the transactional costs associated to a contract.

The original work of Szabo [10] is focused on protocol design for algorithmically specifiable relationships and no specific SSC feature seems to directly address contractual incompleteness. In our context, XBPs are grounded on long-term contracts, more or less formalized in legal terms, and in this setting, we consider contractual incompleteness and renegotiation two primary concerns.

4 Blockchain-Based Smart Contracts (BSCs)

A *blockchain* is essentially a temporally ordered list of permanent data blocks. Blocks are considered immutable because the effort needed to revert them could be quite expensive and the probability of observing a replaced block decreases over time as new blocks are added in front of it. The key innovation of the blockchain technology is a decentralized emergent consensus protocol that enables a group of agents to reach an agreement about a global state by accepting data transmitted across an open byzantine Peer-to-Peer (P2P) network. The blockchain technology appeared for the first time in the implementation of the Bitcoin protocol [7] as a innovative solution to the double-spending problem that does not require a trusted central authority. In Bitcoin, each block contains a set of transactions representing a transfer of tokens from a source to a destination account address. BSC platforms extend this basic functionality by supporting the execution of general-purpose on-chain stored procedures.

The core idea behind BSCs can be found in the Bitcoin scripting mechanism, but a fully working BSC implementation has been offered by successive systems, like Ethereum [1]. The Ethereum platform can run general-purpose scripts encoding arbitrary state transition functions that are automatically enforced when a certain event occurs, for instance when a transaction is scheduled. Every smart contract deployed on Ethereum is unique and no modification is allowed.

BSCs are sufficiently expressive to create new cryptocurrencies and to establish novel network coalitions in the form of Decentralized Autonomous Organizations (DAOs). However, immutability-by-default of BSCs can generate subtle problems when used for organizational purposes. A clear evidence is provided by the history of The DAO project that was one of the first attempts to found a large network coalition on the Ethereum platform. The BSCs encoding its governance rules were not able to capture the actual intents of the parties and all the future eventualities, namely they were fatally incomplete. In June 2016, a bug, or a feature for the immutability-by-default proponents, has allowed an attacker to subtract about USD 50M from The DAO project. The ad-hoc updating procedure included by The DAO developers in their BSCs was not sufficient to fix the breach. The problem caused these contracts was partially solved outside The DAO authority by forking the entire Ethereum platform.

5 Enforceable Business Process (EBP)

In the previous sections, we argued that cost-optimized contracts could be intentionally incomplete and network coalitions are generally established by long-term contracts which should be considered inherently renegotiable. We also explained why BSCs are poor abstractions for modeling XBPs. In this section, we outline an alternative notion that would better match the requirements of a DAIS. We define an *XBP specification* as a potentially incomplete, renegotiable, procedural contract. A contract is said to be procedural if all its completely specified parts can be directly interpreted by a DAIS without additional details. Following the SSC intelligibility principle, we also define an *XBP model* as an XBP specification built using graphical constructs with a clearly stated semantics. Visually, an XBP model can be represented as a common business process model enhanced with banking abstractions and temporal constraints [2]. This additional features are required to make the XBP activities effectively enforceable in a decentralized way by means of incentives and deadlines. We can now introduce an evolution of the SSC concept that should better match the discussed XBP traits. An *Enforceable Business Process (EBP)* is an XBP model archetype, characterized by five primary life cycle phases or states (S1–S5), four enforcement modes (E1–E4) and eight fundamental contractual properties (P1–P8). To ease the discussion, phases and enforcement modes are depicted in Fig. 1.

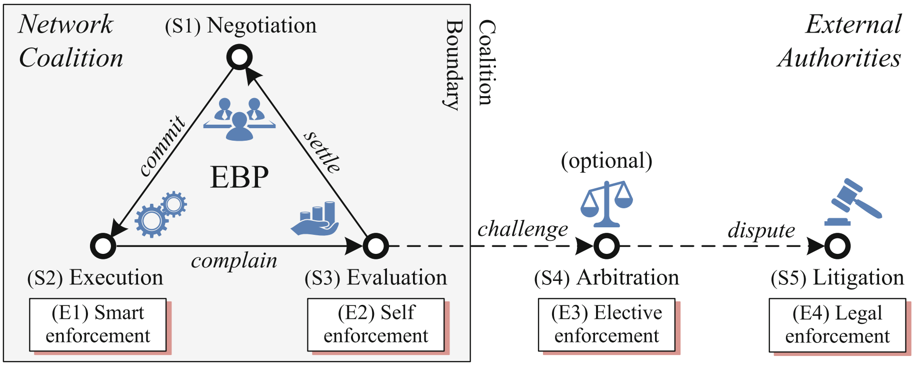

Fig. 1. The life-cycle of the Enforceable Business Process (EBP).

The EBP life cycle is made of three recurring phases (S1–S3) that are managed inside a coalition and two external phases (S4–S5) that could lead to the process disposal or even to the coalition dissolution. In the *negotiation* phase (*S1*) the parties discuss a new EBP or refine an existing one until a good compromise is reached, namely until a specific XBP model is committed. When an EBP is ready, it can be instantiated several times on demand during the *execution* phase (*S2*). The contracting parties will follow the agreed protocol by performing the required tasks. This phase can terminate if a party rises a complain. In the *evaluation* phase (*S3*), the EBP performance is discussed. One or more parties may not be satisfied and may opt for a compensation, opening a dispute. If an agreement is reached, the EBP enters in a new negotiation phase, where it can be improved; otherwise, it can enter into a phase of arbitration or litigation. The *arbitration* phase (*S4*) is useful when the parties are not able to settle a dispute, but they can agree on a trusted third-party arbitrator, see (E3) below. Finally, in the *litigation* phase (*S5*) the EBP and the related tamper-proof logs will be brought before a court for adjudication, see (E4) below.

An EBP can be subject to four different kinds of enforcement, two of them can be implemented by a DAIS, while the other two should be supported as fail-safe methods to settle a dispute. *Smart* (*E1*) is an automatic enforcement mode that can be obtained by running the EBP instances in a trustless system, similarly to what happens for BSCs. *Self* (*E2*) is an autonomous enforcement mode triggered by one or more parties to complain about the overall EBP performance. A voting mechanism may be used to solve the raised conflicts. *Elective* (*E3*) is an external enforcement mode that can be applied when the contracting parties are not able to settle a dispute on their own, but they can reach an agreement about a trusted third-party arbitrator. Finally, *legal* (*E4*) is an external enforcement mode that brings the dispute before the court. The blockchain logs can be in principle used by an adjudicator as evidences to solve a conflict.

An EBP shall fulfill the following properties (P1–P8) that should be expressed by its executable semantics and the related run-time support. *Observability (P1)*: the execution of an EBP shall be observable by the parties, unless otherwise stated in the contract. Process monitoring and mining techniques can be adapted to examine the running EBP instances and their logs. *Verifiability (P2)*: the performance of an EBP shall be verifiable by a third-party. In principle, the blockchain property of being irreversible, together with process mining and auditing techniques, could be used by a designated third-party to identify the divergent behaviors and act appropriately. *Enforceability (P3)*: an EBP shall support the four kinds of enforcement discussed above. In particular, a DAIS shall support the first two enforcement modes and provide tamper-proof logs, making the other external methods a viable alternative. *Privity (P4)*: an EBP shall affect only the contracting parties and only such parties have the rights to interpret it and disclosure its content whenever necessary. A third-party should interfere with the process only when explicitly solicited. *Intelligibility (P5)*: an EBP shall be as clear as possible for the end-users. User-friendly interfaces, graphical notations and simulation techniques could be applied to enhance comprehensibility. *Underspecifiability (P6)*: a DAIS shall support an EBP with a best-effort execution where incomplete parts are manually handled by the contracting parties. For instance, an override mechanism is necessary for any task without an explicitly stated deadline. *Renegotiability (P7)*: an EBP shall always be renegotiable by the contracting parties even when it is explicitly stated otherwise: in line with contractual incompleteness and privity, a non-negotiable statement may not match the actual aims of the parties that can agree on changing it. *Durability (P8)*: an EBP represents a long-term contract and its execution shall persist over time, its logs shall be recoverable in case of a system failure.

Example – The program committee of a conference can be considered a network coalition whose common business goal is selecting the most valuable contributions, improving the quality of the conference at every edition. The actual process is refined during the negotiation phase (S1) through which the parties decided important details such as the prescribed deadline for the revisions, the number of reviewers for each paper, the minimum evaluation for the acceptance, and so on. When an agreement is reached, the process enters into the execution phase (S2) during which the automatic enforcement (E1) of some tasks may happen. For instance, the automatic assignment of some papers to a reviewer that does not promptly communicates any preference. Moreover, during the execution some complains may arise, for instance some reviewers may not be able to meet the deadline or a reviewer may disagree with the evaluation performed by another one. An evaluation phase (S3) can start during which the program committee can decide to modify the original process, for instance by changing the deadline or by requiring an additional revision. This phase may involve a self enforcement (E2) eventually implementing a voting mechanism. In case the parties are not able to reach an agreement, the process can enter into an arbitration phase (S4) during which the conference chairs unilaterally decide the actions to perform (E3). The parties can be subject to some form of incentives and penalties, for example can be encouraged with some discount on the conference price.

6 Conclusion

In principle, a broad uptake of the blockchain technology can induce a paradigm shift in the way network coalitions design and automate their inter-organizational processes. In this paper, we have investigated what are the limits of the BSCs in supporting such kind of processes. We have based our analysis on contractual incompleteness, a key concept of quite a few economic theories. We have seen that XBPs are grounded in potentially incomplete, inherently renegotiable, long-term contracts. In contrast, BSCs are intended to be immutable, relegating renegotiability as a secondary class feature. We found in this mismatch a major concern that hinders the development of an effective DAIS. To characterize this problem, we have proposed the notion of EBP, an evolution of the original SSC concept that better captures the requirements of a DAIS. In a foreseeable future, a new generation of decentralized information systems built around the notion of EBP could substantially lower the contractual costs, opening the path to innovative large-scale network coalitions and changing the actual socio-economic landscape.

Acknowledgment. This work is partially supported by the Italian National Group for Scientific Computation (GNCS-INDAM) and by "Progetto di Eccellenza" of the Computer Science Department, University of Verona, Italy.

References

1. Buterin, V.: A Next-Generation Smart Contract and Decentralized Application Platform (2014). http://github.com/ethereum/wiki/wiki/White-Paper. Accessed 04 2019
2. Combi, C., Gambini, M., Migliorini, S., Posenato, R.: Representing business processes through a temporal data-centric workflow modeling language: an application to the management of clinical pathways. IEEE Trans. Syst. Man Cybern. Syst. **44**(9), 1182–1203 (2014)
3. García-Bañuelos, L., Ponomarev, A., Dumas, M., Weber, I.: Optimized execution of business processes on blockchain. In: Carmona, J., Engels, G., Kumar, A. (eds.) BPM 2017. LNCS, vol. 10445, pp. 130–146. Springer, Cham (2017). https://doi.org/10.1007/978-3-319-65000-5_8
4. Hart, O.: Incomplete contracts and control. Am. Econ. Rev. **107**(7), 1731–1752 (2017)
5. Hart, O., Holmström, B.: The Theory of Contracts. Econometric Society Monographs, pp. 71–156. Cambridge University Press, New York (1987)
6. Mendling, J., et al.: Blockchains for business process management - challenges and opportunities. ACM Trans. Manage. Inf. Syst. **9**(1), 4:1–4:16 (2018)
7. Nakamoto, S.: Bitcoin: a peer-to-peer electronic cash system (2008). http://www.bitcoin.org/bitcoin.pdf. Accessed 04 2019

8. Powell, W.W.: Neither market nor hierarchy: network forms of organization. Res. Organ. Behav. **12**, 295–336 (1990)
9. Simon, H.A.: Organizations and markets. J. Econ. Persp. **5**(2), 25–44 (1991)
10. Szabo, N.: Formalizing and securing relationships on public networks. First Monday **2**(9) (1997). https://doi.org/10.5210/fm.v2i9.548. Accessed Apr 2019
11. Williamson, O.E.: The Economic Institutions of Capitalism: Firms, Markets, Relational Contracting. The Free Press. A division of Macmillan, Inc. New York (1985)
12. Williamson, O.E.: Comparative economic organization: the analysis of discrete structural alternatives. Adm. Sci. Q. **36**(2), 269–296 (1991)