

# Atomic Appends: Selling Cars and Coordinating Armies with Multiple Distributed Ledgers<sup>\*</sup> <sup>\*\*</sup>

Antonio Fernández Anta<sup>1</sup>, Chryssis Georgiou<sup>2</sup>, and Nicolas Nicolaou<sup>3</sup>

<sup>1</sup> IMDEA Networks Institute, Madrid, Spain [antonio.fernandez@imdea.org](mailto:antonio.fernandez@imdea.org)

<sup>2</sup> Dept. of Computer Science, University of Cyprus, Nicosia, Cyprus  
[chryssis@cs.ucy.ac.cy](mailto:chryssis@cs.ucy.ac.cy)

<sup>3</sup> Algolysis Ltd, Cyprus [nicolas@algolysis.com](mailto:nicolas@algolysis.com)

**Abstract.** The various applications using Distributed Ledger Technologies (DLT) or blockchains, have led to the introduction of a new “market-place” where multiple types of digital assets may be exchanged. As each blockchain is designed to support specific types of assets and transactions, and no blockchain will prevail, the need to perform *interblockchain* transactions is already pressing.

In this work we examine the fundamental problem of interoperable and interconnected blockchains. In particular, we begin by introducing the *Multi-Distributed Ledger Objects* (MDLO), which is the result of aggregating multiple *Distributed Ledger Objects* – DLO (a DLO is a formalization of the blockchain) and that supports append and get operations of records (e.g., transactions) in them from multiple clients concurrently. Next we define the *AtomicAppends* problem, which emerges when the exchange of digital assets between multiple clients may involve appending records in more than one DLO. Specifically, AtomicAppend requires that either *all* records will be appended on the involved DLOs or *none*. We examine the solvability of this problem assuming *rational and risk-averse* clients that may *fail by crashing*, and under different client *utility* and *append* models, *timing models*, and client *failure scenarios*. We show that for some cases the existence of an intermediary is *necessary* for the problem solution. We propose the implementation of such intermediary over a specialized blockchain, we term *Smart DLO* (SDLO), and we show how this can be used to solve the AtomicAppends problem even in an asynchronous, client competitive environment, where all the clients may crash.

---

<sup>\*</sup> Partially supported by the Spanish Ministry of Science, Innovation and Universities grant DiscoEdge (TIN2017-88749-R), the Comunidad de Madrid grant EdgeData-CM (P2018/TCS-4499), the NSF of China grant 61520106005, and research funds from the University of Cyprus (CG-RA2019). We would like to thank Kishori Konwar, Michel Raynal, and Gregory Chockler for insightful discussions.

<sup>\*\*</sup> A conference version of this work has been presented at Tokenomics 2019: International Conference on Blockchain Economics, Security and Protocols, May 2019, Paris, France.