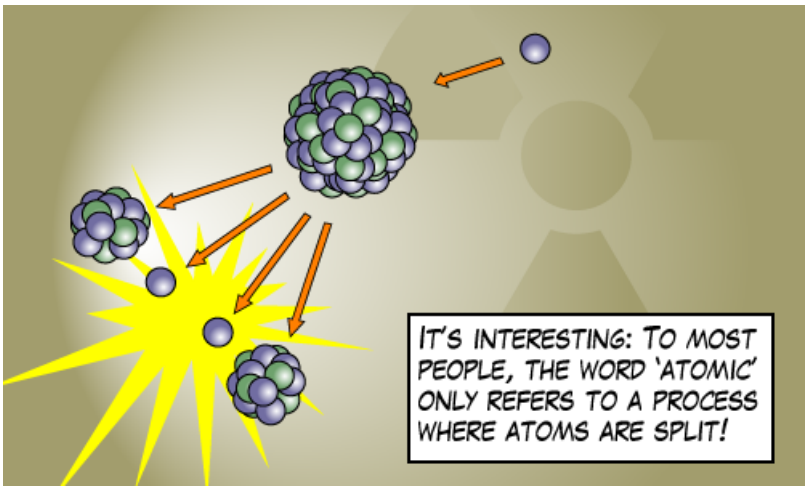
ETH
Eidgenössische Technische Hochschule Zürich
Swiss Federal Institute of Technology Zurich

Distributed Computing

Prof. R. Wattenhofer

Smart Contract Aggregation

In blockchain systems like Ethereum, users call smart contract functions by sending transactions mostly one at a time. These transactions are included in a block by miners/validators based on the fees paid by the user. If the same user as many transactions to execute, would it help the user to batch these into a single atomic transaction? Would that help save fees? Does it help prevent frontrunning of some of the internal transactions? Does it matter if all the transactions into the same block or not?



In this project, we investigate what kind of popular smart contract transactions can be batched/aggregated/made-atomic without changing functionality, *while* gaining something in return. Does the batching system work as a non-blockchain server, which converts multiple calls into one call? Or does it have to be a smart contract itself? If it is a server, how can it convince its users that it is not doing anything other than batching (like maliciously frontrunning its users' transactions itself)?

Requirements: This project involves understanding Ethereum, smart contract design, popular smart contracts like Uniswap, and being able to understand and program in Solidity.

Interested? Please contact us for more details!
Contact: Tejaswi Nadahalli: tejaswin@ethz.ch, ETZ G97