Securing the Decentralized Coordination of Active Distribution Grids with Blockchain

Iasonas Kouveliotis-Lysikatos

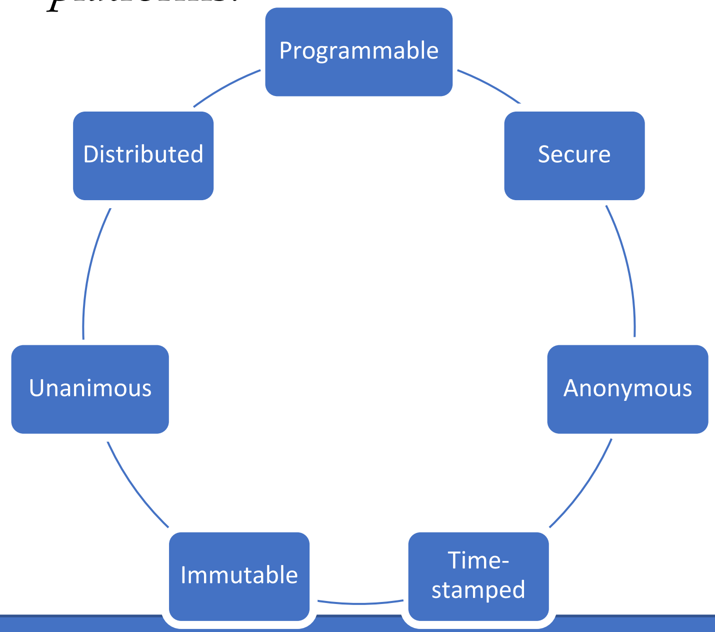
Digitization of Energy Systems, IPPS, ECE - NTUA, 25 May 2023

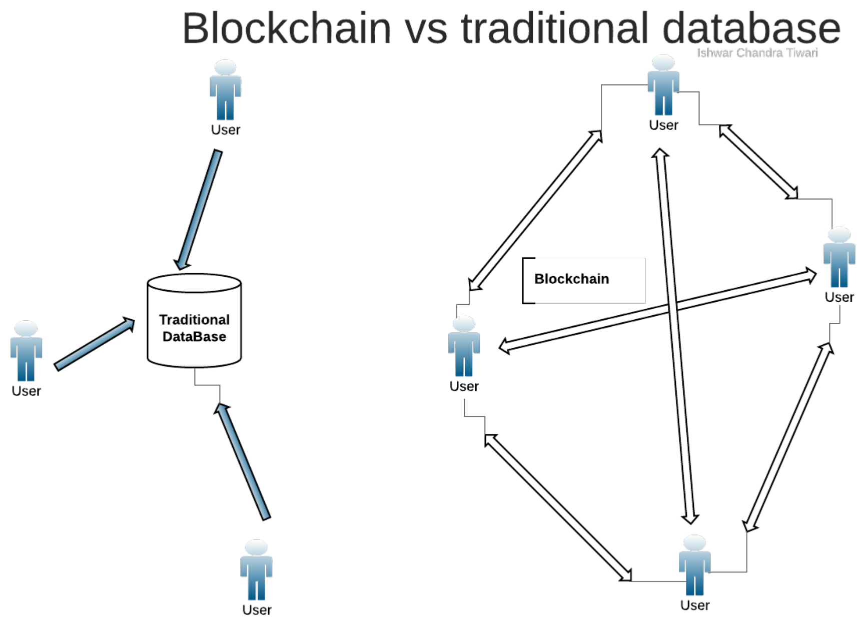
Outline

- Introduction to blockchain technologies
- The Ethereum Virtual Machine
- Decentralized energy procurement protocols
- Decentralized markets, Local Energy Communities

Introduction

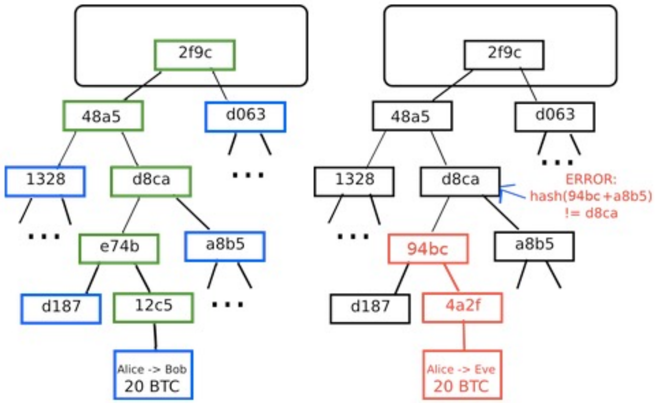
- Distributed computer network protocols that securely manage and maintain the data imported and processed by their users, without the need for a centralized authority.
- Distributed Ledger: a Distributed Data Structure.
- Interesting for applications of **automating energy transactions**: *decentralized energy management platforms*?

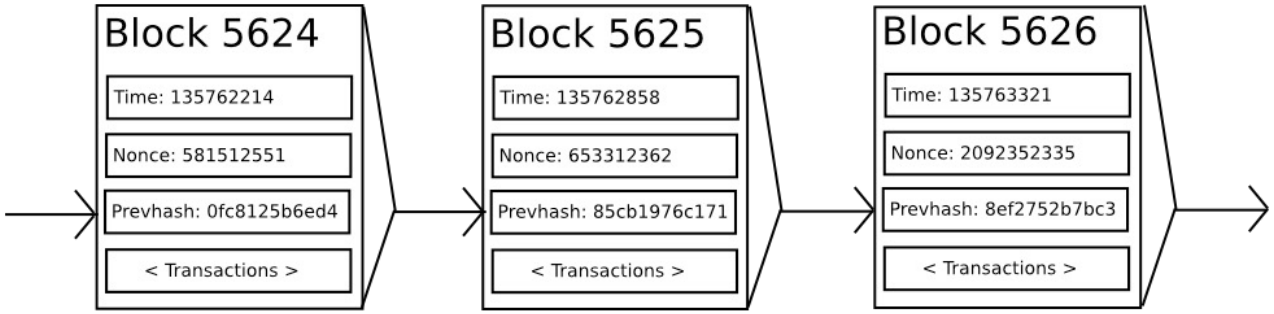




Introduction to Blockchain technologies

- Bitcoin – **peer-to-peer** protocol.
- Financial transactions without the need of a trusted third party (e.g., a bank).
- Distributed database.
- Introducing **new data** entries (in the form of blocks) requires the consensus of the network.
- Fundamental **cryptographic** primitives (hash functions, Public-key Encryption, Digital Signatures, Merkle Tree).
- Miners and clients.

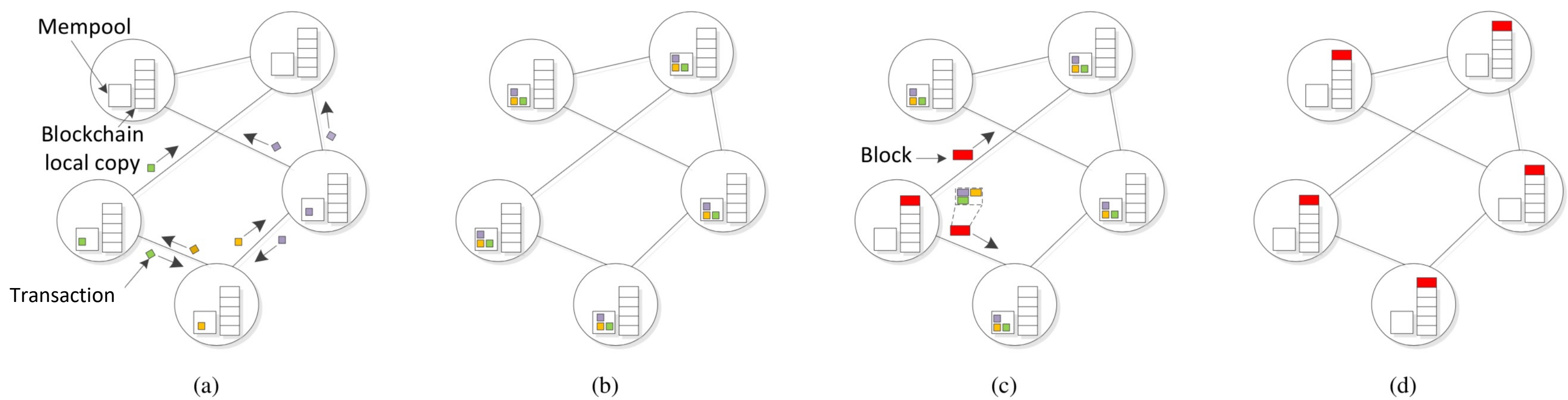




Introducing new blocks of data (Bitcoin)

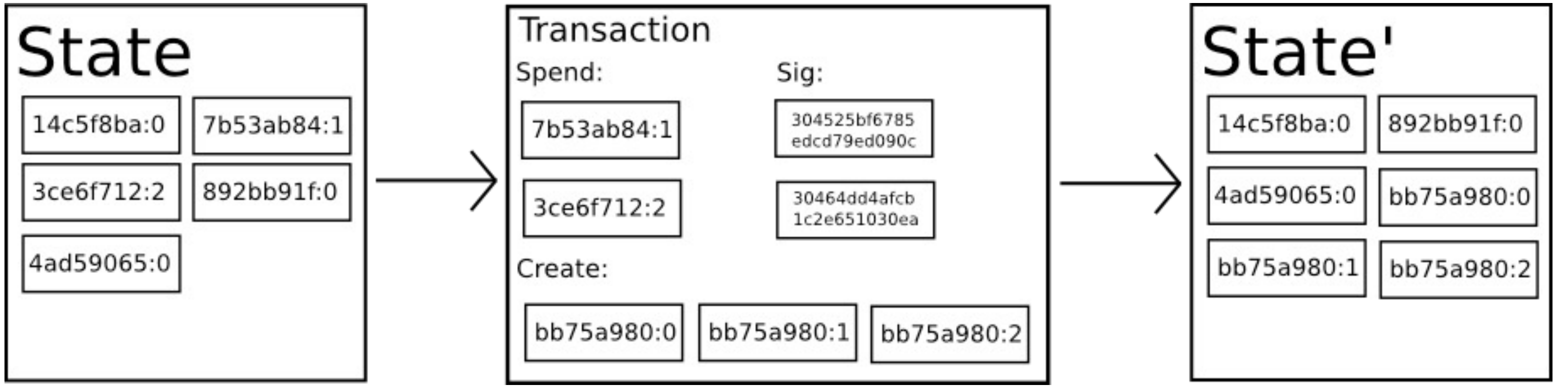
- The protocol ensures that a block is created every 10 minutes.
- The new block includes the most recent transactions that do not exist in previous blocks.
- Blocks are relayed and broadcasted throughout the network.
- A protocol procedure is followed to **verify** a new block (consensus).
- A **valid** transaction included in a valid block is **confirmed.**
- Each block is cryptographically linked to the previous one .

This linked list of blocks is called blockchain



Introducing new data ☞ Changing state

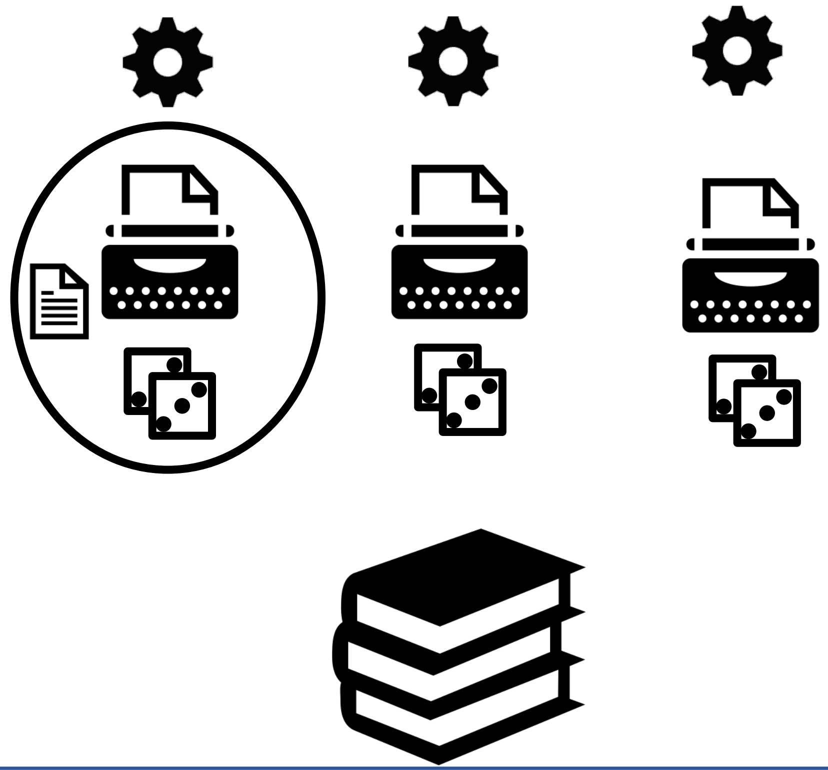
- For each input in TX:
  - (if valid input) apply state transition function: APPLY(S,TX) -> S'
  - Return S'



- **Consensus mechanisms**
  - Proof of Work,
  - Proof of Stake,
  - Practical Byzantine Fault Tolerance (PBFT),
  - Proof of Authority

Introducing new blocks of data (Bitcoin)

- All miner nodes can “write” data in the ledger (add blocks)
- Problem: which node will insert the new block (nodes compete for receiving a reward)?
  - A **probabilistic** approach is followed to grant a single node the right to add a new block by solving a **cryptographic puzzle**



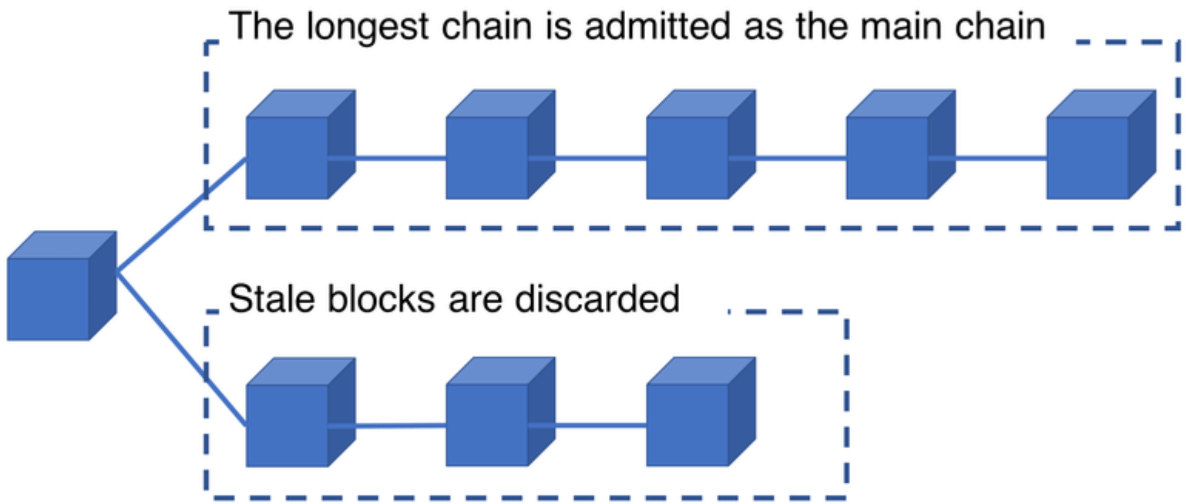
Introducing new blocks of data (Bitcoin)

- If there are different versions of the chain that contradict each other, which one is the right?

  "The right chain is the one that contains the most blocks, if there are many, pick one at random"

- The new block is broadcasted
- If valid it's adopted by the network

It turns out that: if 51% of the network nodes honestly follow the protocol, then the chance to "Modify" a transaction in a block decreases exponentially as new blocks are added!



Ethereum

- **Ethereum** provides a built-in Turing-complete language for writing **smart contracts.**
- Smart Contracts: Computer protocols that contain a set of rules under which different parties are *enforced* to interact.
- Can be thought of as autonomous agents living in the Ethereum execution environment, always executing a specific piece of code when triggered → similarities with the concept of Multi-Agent Systems (MAS) for smart grid applications.
- The advantages of the blockchain-based DApps (i.e., security and data tampering protection guarantees) are appealing to specific Smart Grids problems.

Ethereum

- A universal state transition machine:
    - Transaction-based deterministic state machine.
- A virtual machine that applies changes to global state:
    - Global state of Ethereum: **accounts** (personal & contracts).
    - They interact to each other through transactions (messages).
    - A state associated with it and a 20-byte address (160-bit identifier).
- **A global decentralized computing infrastructure.**
- **Anyone can create their own state transition functions:**
    - Smart contracts.
    - Decentralized applications.

Ethereum – Smart Contracts

- What is a smart contract: **Computer programs!**
  - The code of a smart contract cannot change.
  - The outcome of a smart contract is the same for everyone.
  - Context: Internal storage, transaction context, most recent blocks.
  - Contract code is executed by all full nodes.

When contract account is **activated**:

a. Contract code runs.

b. It can read / write to internal storage.

c. It can send other messages or create new contracts.

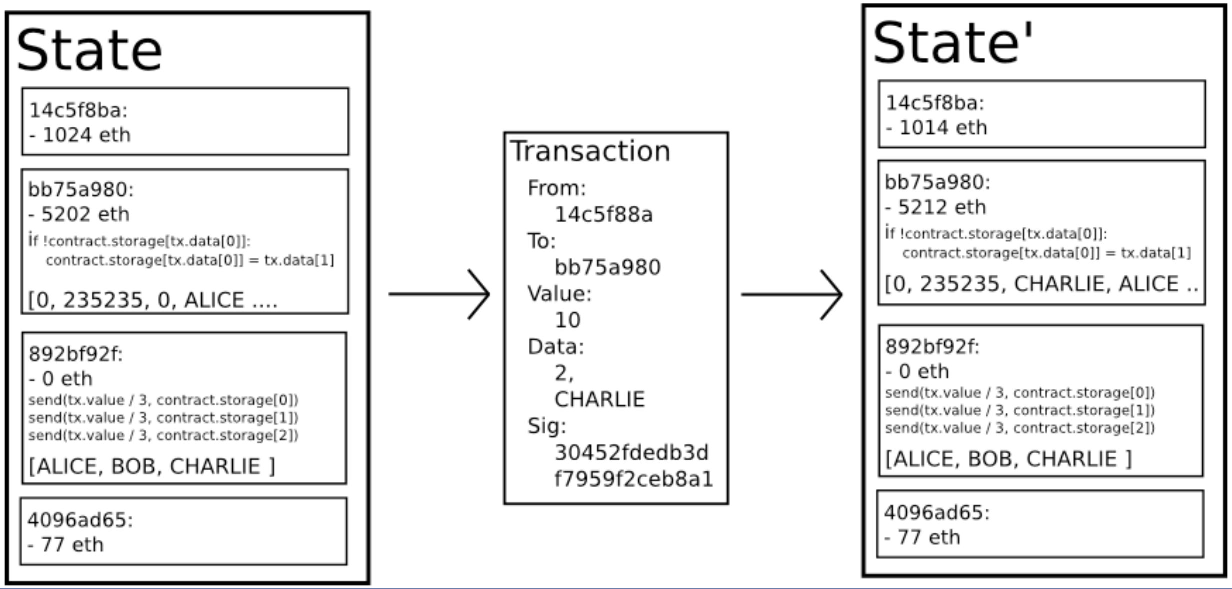
Can't initiate new transactions on their own, nan only fire transactions in response to other transactions received.

```
contract Namespace {

    struct NameEntry {
        address owner;
        bytes32 value;
    }

    uint32 constant REGISTRATION_COST = 100;
    uint32 constant UPDATE_COST = 10;
    mapping(bytes32 => NameEntry) data;

    function nameNew(bytes32 hash){
        if (msg.value >= REGISTRATION_COST){
            data[hash].owner = msg.sender;
        }
    }

    function nameUpdate(bytes32 name, bytes32 newValue, address newOwner){
        bytes32 hash = sha3(name);
        if (data[hash].owner == msg.sender && msg.value >= UPDATE_COST){
            data[hash].value = newValue;
            if (newOwner != 0){
                data[hash].owner = newOwner;
            }
        }
    }
```

Ethereum – Smart Contracts

- Every node on the network must:
  - **evaluate all transactions,**
  - **store all state.**
- Halting problem?
- Every computation step has a fee

  … which is paid in gas.
- Gas is the unit used to measure computations.



Drawbacks – Blockchain types

- Scalability – Transaction speed
- High cost of PoW consensus
- Privacy

Ethereum

Bitcoin

Public & Closed	Public & Open
• Voting • Voting records • Whistleblower	• Currencies • Betting • Video Games
Private & Closed	**Private & Open**
• Construction • National Defence • Law enforcement • Military • Tax Returns	• Supply Chain • Government financial records • Corporate earning statements

Hyperledger

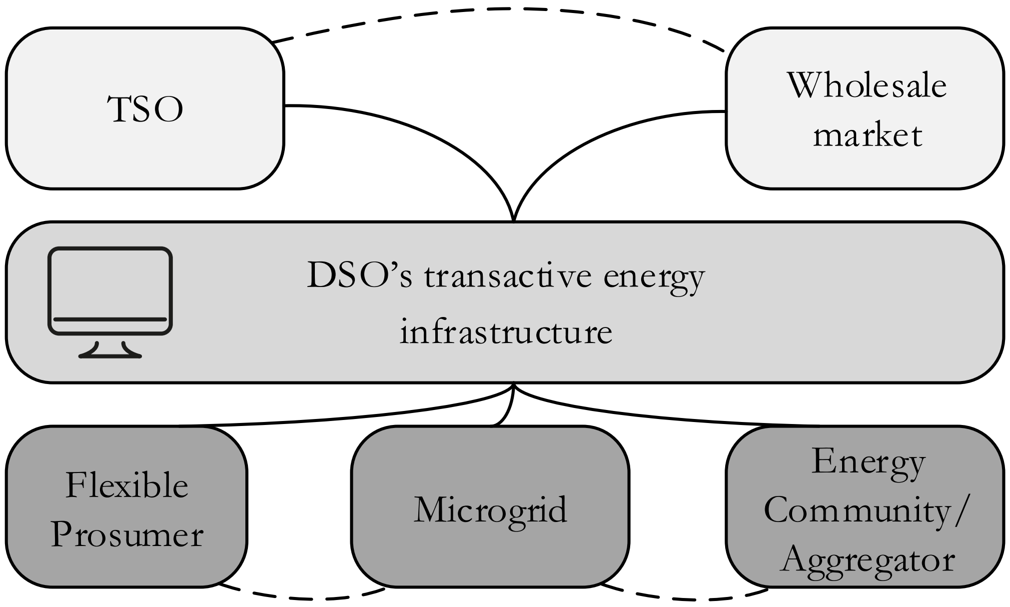
R3 Corda

ETH Price: ~$100 - Gas Price: 3 Gwei

Size	Gas	Cost
32 bytes	20.000	$0,058
1KB	724.664	$2.104
1MB	697.325.562	$2,025.03
10MB	~7.000.000.000	~$20,328
100MB	~70.000.000.000	~$203,280
1GB	~700.000.000.000	~$2,032,800

Transactive Grids & Decentralized Energy Markets

- **Transactive Energy**: energy trading and grid operation activities extended at the distribution level.
- For producing actual impacts in the distribution grid operation, a large integration of DG units and flexible consumers in necessary.
- DSO's perspective: a TG should provide a unified architecture enabling several actors: **Aggregators**, **Balancing Responsible Parties**, **consumers**, **producers**, **microgrids**, **energy communities**, etc. – to trade both *energy* and *flexibility* for balancing the grid at the lower possible cost, while satisfying the grid's constraints.



Related work

- Most implementations deal with **P2P energy trading** and **local energy markets** *(*VPPs, microgrids, energy communities or cooperatives, etc.*).*

Main technical advantage: *security of the transactions.*

- *Facilitation of the participation in wholesale market.*
- *Efficient sharing of storage systems among the members of energy communities.*
- *Sharing of renewable production according to the goals of the participants, by utilizing the smart metering infrastructure of the distribution grid.*
- *Secure and transparent issuing and management of green certificates.*

BLOCKCHAIN PROJECTS FOR THE ENERGY SECTOR

Project	*Application*	*Platform*	*Company/Location*
Energy Web	General purpose	Ethereum-based/Open Source	Switzerland
WePower	RES investment	Ethereum	Lithuania
Verv VLUX	P2P energy trading	Ethereum	UK
LO3 Energy/ Brooklyn Microgrid	Community P2P energy trading	Tendermint/ Proprietary	US
Power Ledger	P2P energy trading, green certificates, e-mobility, asset management	Ethereum	Australia
Grid+	P2P energy trading	Ethereum	US
Pylon	P2P energy trading, billing	Pylon Network/ Open Source	Spain

Practical Decentralized Application

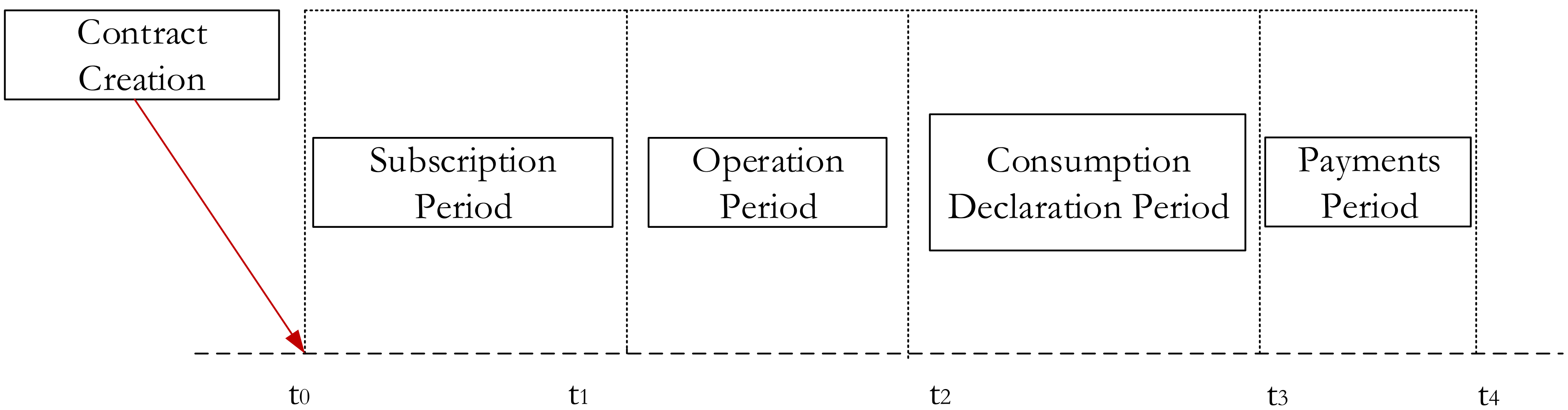
- Highly applicable in **securing** and **automating** the economic **settlement** of energy transactions (vision of transactive grids).
- …or in combination with distributed control techniques (for increasing their cybersecurity and transparency).

Although, we should keep in mind that:

- **Not applicable** to solving **large complexity algorithms** or to applications requiring **large data storage**. (E.g., optimization problems, or storing raw electrical consumption data).
- **Speed of transactions**: (in Ethereum ~3 minutes) → it excludes applications requiring quicker response times.

Electricity Supply Contracting Automation

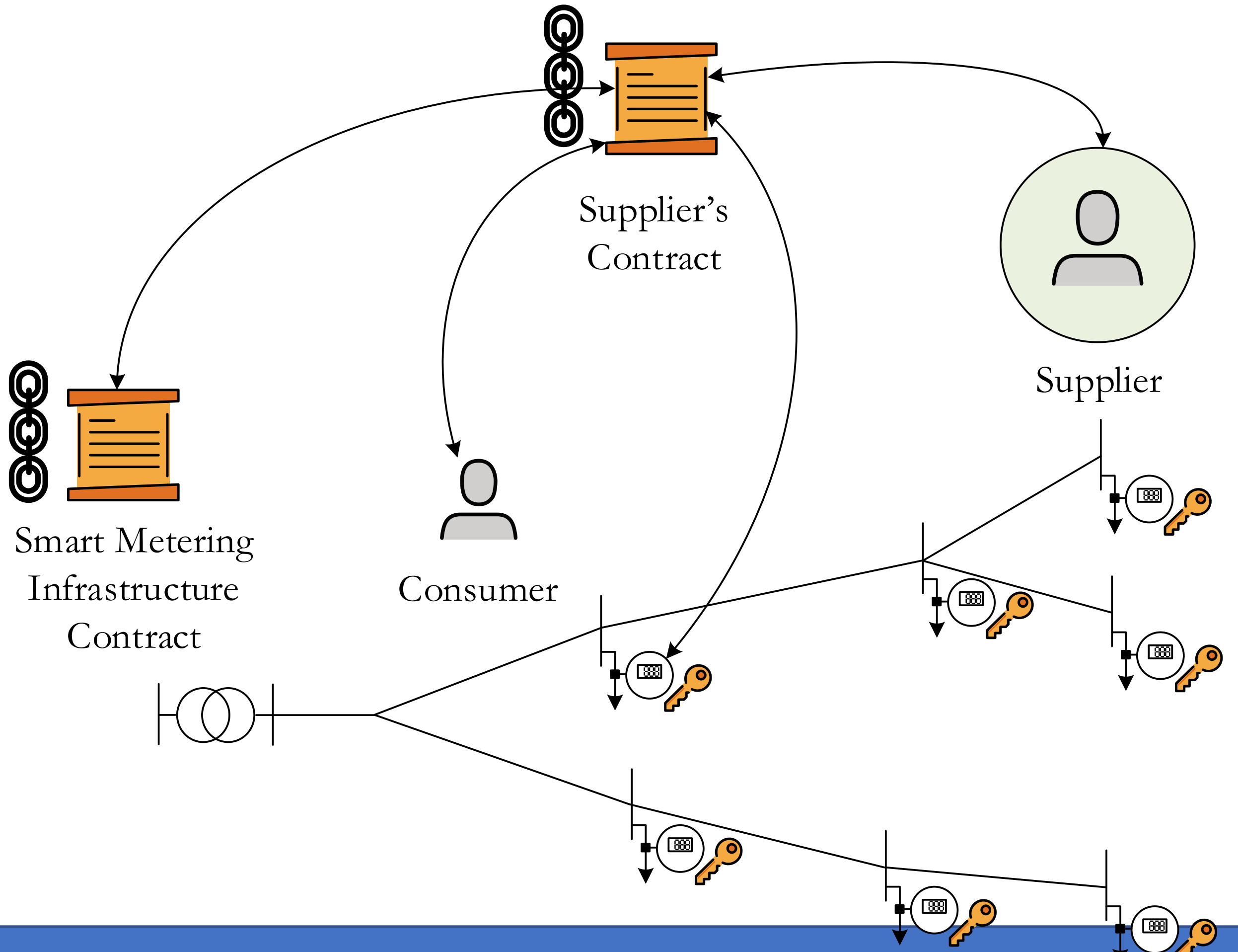
A player or an organization who would take over the operation of the local market and the interaction with the central energy market can be **replaced by a protocol** that is followed by the participants.



Protocol Modelling using Smart Contracts

Security assumptions:

- The metering infrastructure operator can be trusted;
- The smart meters can interact with the blockchain; and
- The smart meters' private keys cannot be recovered.



Protocol Modelling using Smart Contracts

Smart Contract excerpt: 1

```
contract smartMeters {

  address owner;

  struct meter {
    bool isRegistered;
  }
  mapping (address => meter) meters;

  constructor () public {
    owner = msg.sender;
  }

  function registerMeter(address meter_) public
onlyOwner {
    require(!meters[meter_].isRegistered);
    meters[meter_].isRegistered = true;
  }

  function isValidMeter(address meter_) public
view returns (bool) {
    return meters[meter_].isRegistered;
  }

  modifier onlyOwner {
    require(msg.sender == owner)
    _;
  }
}
```

Smart Contract excerpt: 2

```
import "./smartMeters.sol";

contract supplier {

  struct consumer {
    address consumer_;
    address meter;
    …;
  }
  address[] users;
  mapping (address => consumer) consumers;

  constructor (address meterProvider, uint256
t1, …, uint256 price, uint256 deposit) public {
    t0 = getBlockNumber();
    …;
    sMContract=smartMeters(meterProvider);
  }

  function subscribeUser (address meterID)
public onlySubscription payable {
    …;
  }

  function declarePeriod (address user, uint256
activeP) public onlyDeclare {
    …;
  }

  function paymentPeriod () public onlyPayment {
    …;
  }

  function finalize() public onlyOwner {
    …;
  }
  …;
}
```

Simulation results

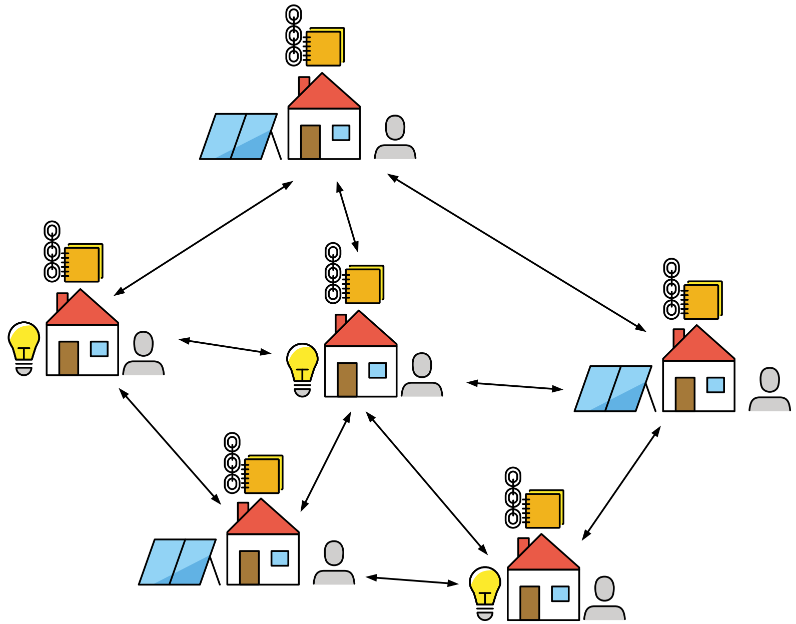
TEST RESULTS, GAS CONSUMPTION FOR EACH METHOD AND DEPLOYMENTS OF THE SMART CONTRACTS.

Methods		*Gas: 3gwei/ gas*	*Block limit: 6721975 gas 102,75 eur/eth*	
Contract	*Method*	*Average Gas*	*#calls*	*eur(avg)*
smartMeters	*registerMeter*	43900	4	0.01
	unregisterMeter	14482	4	0.00
supplier	*declarePeriod*	52656	4	0.01
	finalize	13484	1	0.00
	paymentPeriod	37281	4	0.01
	subscribeUser	125865	4	0.03
Deployments			*% of limit*	
smartMeters		524994	7.8 %	0.16
supplier		1408078	20,9 %	0.29

Deploying DApps and smart contracts requires the payment of certain **fees** (gas) reflecting the **cost/motivation** of the nodes to support the P2P network and execute the smart contracts.

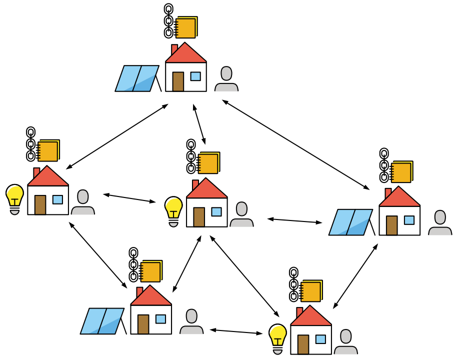
Decentralized Energy Markets – P2P Trading

- Decentralized markets can be defined by coupling the grid operation with possible business interactions:
  - ancillary services market at the distribution level,
  - markets capturing the interactions between the DSO and the TSO,
  - local energy markets.
- The DSO can be envisioned as their **facilitator**.
- Theoretical results show numerous benefits:
  - increasing the system efficiency,
  - promoting the penetration of RES,
  - reduce energy cost for consumers,
  - and increase profits for producers.



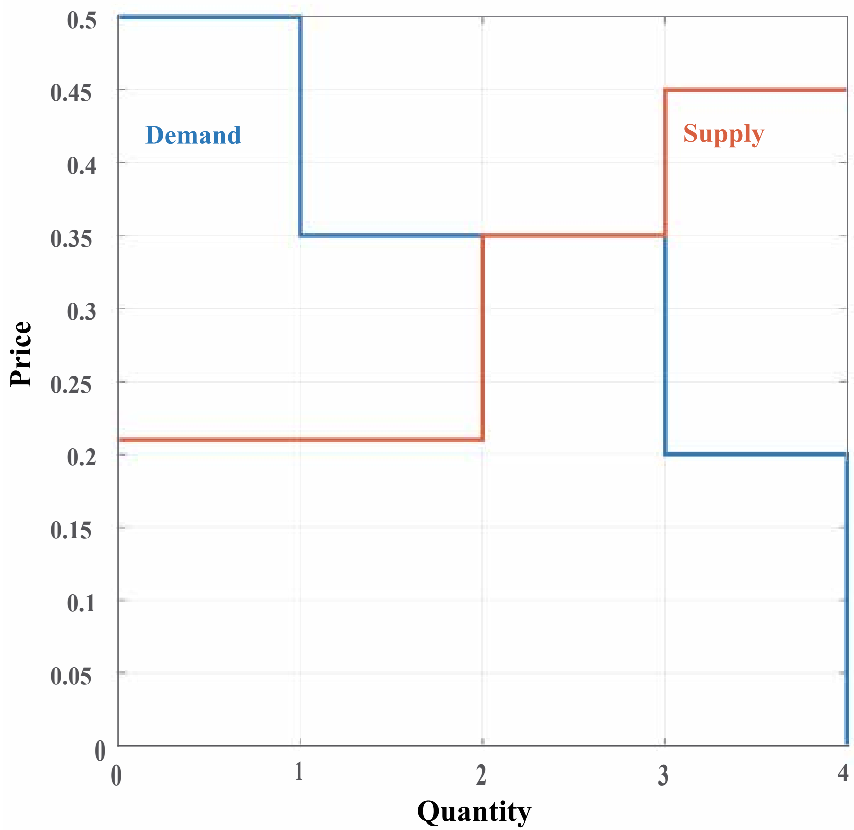
Automating and Securing the Decentralized Energy Markets of Energy Communities

- An **Ethereum-based** application has been developed for facilitating the operation of a **local energy market** of an energy community
- The energy community seeks to **optimally match local consumption with production**,
- and procure additional energy from the central market if this is required.
- The decentralized application sets the ICT platform, i.e., a layer for **processing**, **automating** and **securing** the financial transactions.

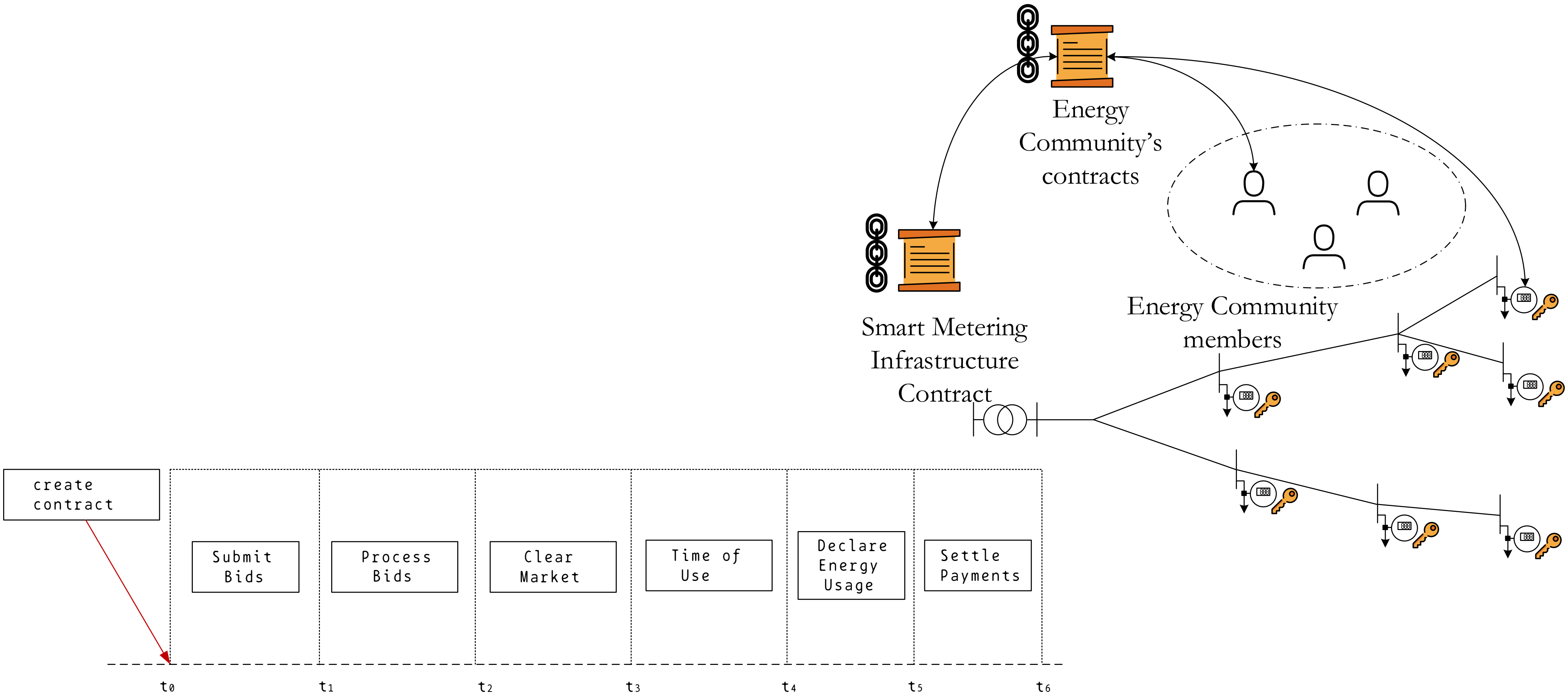


Automating and Securing the Decentralized Energy Markets of Energy Communities

- A **local electricity market** is simulated using a **double auction** negotiation, in discrete timesteps.
- Simplified bidding strategies are only considered to reduce the complexity of the simulations

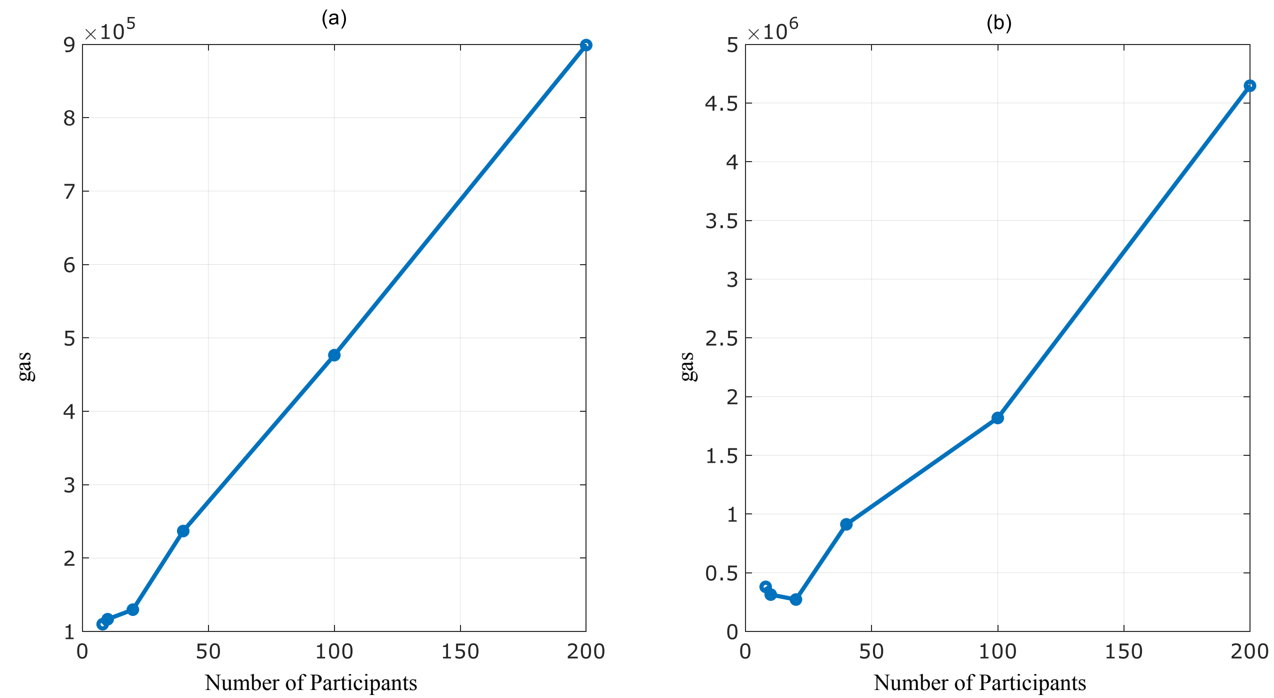


Automating and Securing the Decentralized Energy Markets of Energy Communities



Simulation results

Deploying DApps and smart contracts requires the payment of certain **fees** (gas) reflecting the **cost/motivation** of the nodes to support the P2P network and execute the smart contracts.



Thank you for your attention!
Questions?

Iasonas Kouveliotis-Lysikatos
iasonask@mail.ntua.gr