Hyperledger Fabric Smart Grid Communication Testbed on Raspberry PI ARM Architecture

Andrija Goranović*, Marcus Meisel*, Stefan Wilker* and Thilo Sauter*†
*TU Wien – Institute of Computer Technology, 1040 Vienna, Austria
Email: {first.lastname}@tuwien.ac.at
†Danube University Krems – Center for Integrated Sensor Systems, 2700 Wiener Neustadt, Austria
Email: {first.lastname}@donau-uni.ac.at

***Abstract*—Necessary energy policy targets trying to limit climate change effects are very ambitious. New technologies and combinations of available technologies are needed to achieve the goals by 2030/2050. A promising technology being explored in smart grids communications is the Blockchain, allowing peer-to-peer contractual-like or monetary-like interactions without the need for trust for an intermediary. This paper describes a Smart Grid Controller Blockchain Platform, allowing to simulate Virtual Power Plants communication along load profiles or smart meter measurement data. The communication requirements of these recent advances are similar to industrial cyber-physical systems, hence, applicable in other research domains.**

***Index Terms*—Blockchain, Hyperledger Fabric, Virtual Power Plant, Smart Grid, Communication Platform, Smart Contracts**

I. Introduction

Rising energy consumption and the shift from fossil fuels to renewables in hope of countering climate change, pose global challenges affecting every sector. To deal with fluctuations of weather, day/night cycles, and seasons, concepts such as virtual power plants (VPP), aggregating distributed mixtures of loads and sources are highly promising. However, essential is the available communication technology enabling many, intelligent, connected systems, the smart grid. Digitization trends by consumers, using energy management capable devices, managing actuate-able loads, help maturing sensors and actors in general. One promising technology explored is the blockchain technology. In the energy industry, the blockchain is experimented with to realize processes in smart grids, which are characterized by a decentralized structure, of which the best-known application is local P2P energy trading.

This research takes a look at the actual realization of a small scale VPP communication platform making use of blockchain technology for the integration of renewable energy into the energy system, focusing on local energy communities. The small scale and use of Off-The-Shelf (OTS) hardware components allows for a reproducible lab setting with racks of 14 blockchain nodes for less than 700 €, a Smart Grid Controller Blockchain Platform (SGCBP). This scalable solution allows evaluation of the technology in settings close to reality, allows to design network structures and influence the connection quality. This mimicking of, e.g., companies or consumers situations participating in the electricity market, enables simulation, testing, evaluation, and improvement for robustness of various future blockchain business models.

II. State of the Art and Related Work

This section provides a quick introduction to smart grids, virtual power plants, and blockchain technology.

A. Smart Grid

The traditional electricity grids carried the power unidirectionally from power plants to the customers. Due to the unidirectional information flow, gathering information about the grid in real-time was not possible. To avoid blackouts, grids had to be built to withstand rarely occurring maximum demand peaks. This lead to an inefficient utilization of the grid, which is still present today with many opportunities to increase the hosting capacity. The biggest problem for utilities is the distribution network itself, which is like a blackbox. Therefore, solutions have to be developed, to further integrate information and communication technology (ICT) into the grid. Especially the distribution network and the possibility for demand-side management (DSM) need to be extended [1].

The bidirectional flow of electricity and information (see early definition of Smart Grids by [2]) in every level of the grid, enables the possibility to optimize quality, reliability, and efficiency. Smart Grids are expected to have self-healing propperties, being able to respond to events, e.g., failures or demand peaks. By supporting all energy generation and storage possibilities, consumer involvement in the operation of the grid and energy market is rising.

B. Virtual Power Plant

Increase of decentralized energy resources (DER) introduces new challenges to power systems. First, most DER units, e.g., PVs, are weather-dependent and therefore intermittent, which means that their output is fluctuating and only partially controllable. Second, the participation of DERs on the energy market is allowed at a certain size and reliability only. Finally, most DER units are operated isolated to only satisfy local needs, without contributing to the grid. To overcome these challenges, multiple DER units are aggregated and actively controlled in a VPP [3].

An Energy Management System (EMS) [4] controls the generation units, storage units, and controllable loads according to specific targets, such as the maximization of profits. Therefore, a bidirectional communication is necessary. The EMS receives information, such as generation forecasts from distributed units and send back commands. It can be realized as a centralized system with a control center or as a decentralized system, where every unit is controlled by a separate local controller or as a mixed system, as is done here.

C. Blockchain

2008 in the Bitcoin white paper [5] introduced the Blockchain with the aim to establish a peer-to-peer (P2P) currency without the need for a trusted third party, banks. It can be described as a distributed ledger, which stores records in chained together blocks by referencing the previous block using tamper-proof cryptographic hashes. Every node in the blockchain network has a copy of the blockchain and a public/private key pair. The public key is used to address the node and the private key to sign transactions. When making a transaction, it is first signed and then broadcasted to the neighboring nodes. These nodes then validate the transaction and forward it only if it is valid. Repetitively, nodes group received validated transactions into a block to be added to the blockchain. These blocks can differ from each other, e.g., due to propagation of the transactions through the network, which leads to different copies of the blockchain in the network and different views on the current state. Therefore, a consensus mechanism is needed to achieve a common view within the network. Multiple consensus mechanisms with different approaches exist. Public networks mainly adopt Proof-of-Work (PoW) or Proof-of-Stake (PoS) as consensus mechanism. PoW is solving cryptographic puzzles by brute-forced cryptographic hashing. The node, which solves this puzzle, is rewarded and its block is added to the blockchain. The biggest PoW criticism is its high energy consumption. In contrast, in PoS, where the node's chance to create the next block is proportional to its balance, the number of computations is negligible [6].

A consensus on the classification of blockchain types does not exist[1]. One approach is to differentiate blockchains into public, private, and consortium blockchains. In public blockchains everyone can send transactions, read transactions, and participate in the consensus process. Consortium blockchains are under control of a group, e.g., a group of factories. The consensus process is restricted to selected participants and the permission to read can be restricted too. Compared to public blockchains, consortium blockchains are faster and offer more privacy. In private blockchains the permission to write is restricted to one entity, e.g., an organization. Further, the permission to read may be restricted[1]. Another approach is to differentiate blockchains into permissioned and permissionless blockchains[2].

Smart contracts were first mentioned in 1994 in [7] by Nick Szabo with the vision to realize legal contracts as computer code. Running atop of the blockchain, smart contracts can be used to reach and enforce agreements automatically according to specified rules, such as energy trading based on a smart contract, leading to cost reduction, especially for low-value transactions, and trust-free agreements[3].

The use of the blockchain technology in the energy system related areas is currently being extensively researched. The most prominent use case of the blockchain technology is P2P energy trading, where energy trading between two parties is processed directly via the blockchain, without the involvement of third parties, as can be compared with [8]. Researched topics related to P2P energy trading are further transactive energy auctions in [9] and energy trading for industrial IoT [10]. A technical overview of projects can be found in [11] and [12].

III. Blockchain Communication Platform Functions

To select a blockchain for the SGCBP, Ethereum, Quorum, Hyperledger Fabric, Hyperledger Sawtooth, and Corda have been compared based on following requirements criteria: light energy-footprint, modular, permissioned, privacy mechanisms, resource-aware, and smart contracts capable.

A. Hyperledger Fabric selection

While most criteria can be evaluated directly, for the light energy-footprint and resource consumption, the consensus mechanism has been considered as indicator. Based on the available information and comparison of the presented blockchain implementations, Hyperledger Fabric (HLF) presented itself as the best choice for the implementation along the SGCBP criteria.

HLF is a general-purpose distributed ledger platform implemented by the Linux Foundation. As a permissioned private blockchain it supports a Membership Service Provider for managing the network members. Its modular design features pluggable consensus mechanisms, MSPs, and encryption mechanisms. Further, access control lists (ACL) can be used as an additional permission layer. Smart contracts in HLF are realized as chaincodes. Transactions are operations invoked on a chaincode. The chaincode execution is separated from transaction ordering, thereby increasing scalability and performance. Currently chaincodes can be written using the programming languages Go and NodeJS, and are executed by validating nodes inside a Docker container.

To reach consensus in HLF, the transaction order and the transaction validity have to be fulfilled. In HLF, nodes are differentiated by their roles and associated tasks for reaching consensus, however, a single server can run multiple nodes with the different roles, clients, peers, and orderers. While clients act on behalf of the end-user and peers maintain the ledger, orderers are responsible for the transaction order. In HLF, privacy and confidentiality can be achieved with channels and will be focus in future testbed developments.

B. Hyperledger Fabric Transaction Flow

A crucial part of HLF is its transaction workflow, which starts with an application constructing a transaction proposal by using the HLF SDK. This transaction proposal is sent to

[1] Blockchains & Distributed Ledger Technologies https://blockchainhub.net/blockchains-and-distributed-ledger-technologies-in-general/ (accessed: 3/19)

[2] Private Vs. Public and more http://blocktonite.com/2017/06/27/private-vs-public-and-permissioned-vs-permission-less/ (accessed: 3/19)

[3] Smart contracts https://blockchainhub.net/smart-contracts (accessed: 3/19)

specific endorsing peers. Each endorsing peer inspects the received transaction proposal. If all checks succeeded, each endorsing peer invokes the chaincode according to the transaction proposal. Instead of updating the ledger, the resulting read-write set is signed by the endorsing peer and returned as proposal response to the application. The application checks if the endorsement policy has been satisfied and sends a transaction containing the transaction proposal and the proposal responses to the orderer for updating the ledger.

The orderer has to order these and bundle them into blocks for each channel. A created block is then sent to the channel's peers including the endorsing peers. The peers independently handle the block's transactions according to their order in the block. For each transaction the peer checks, if the endorsement policy is satisfied, and if the transaction is valid. In case of several transactions updating the same asset, only the first update would be valid and applied. While valid transactions, in specific their write sets, are applied to the peer's ledger, invalid transactions are only kept for auditing. If an application is registered for events of a channel, it will receive event notifications for block events and transaction events.

IV. Platform Implementation

The system can be divided into the Intra-VPP level and the Inter-VPP level. Whereas, the Intra-VPP level focuses only on one VPP and its members and processes, the Inter-VPP level considers the processes between multiple VPPs.

A. System Model

A VPP can include any number of participants with different configurations, e.g., different combinations of devices such as PV systems, battery storages, EVs, and household appliances. The platform enables VPP participants to track their energy production and consumption at participant level.

The recorded energy surpluses and demands are then matched each time-slot by the platform by creating transactions. This means, that the available energy surpluses of different VPPs owners satisfy the energy demands of other VPPs, as far as possible. Thereby, the highest energy surplus is matched to the highest energy demand. If after the matching energy is left, it is sold or bought to or from the grid.

B. Hyperledger Fabric Components for ARMv7

The RPI is based on the ARM architecture, which HLF does not support out of the box. Further, HLF does not support 32-bit operating systems. This means that the HLF Docker images and binaries have to be created for the RPI. Therefore, the HLF code needs to be modified, which can be found on the three GitHub repositories[4].

For the paper, HLF version 1.1 was used, which corresponds to the tag *1.1.0* on the GitHub repositories. In case of the RPI 3 model B, previous HLF versions have been successfully ported onto[5,6]. Furthermore, the Docker Hub frbrkoala[7] contains HLF v1.1 Docker images for the ARMv7 architecture. It was tried to use these images for the SGCBP. But running the basic-network example from the fabric-samples[8] with these images failed due to a runtime error in the fabric-peer container. Regarding the repository fabric-baseimage, the images from the Docker Hub frbrkoala function properly. Therefore, these were used as basis and only the repositories fabric and fabric-ca were modified to fit the 32-bit ARMv7 architecture.

[4]GitHub repository HLF (fabric-baseimage, fabric, fabric-ca) https://github.com/hyperledger (accessed: 3/19)

C. Chaincodes

The Intra-VPP chaincode sgcbpintra is responsible for processes inside a VPP and the Inter-VPP chaincode sgcbpinter for processes between VPPs. The Intra-VPP-chaincode sgcbpintra focuses on the participant's energy production and consumption. The Inter-VPP chaincode sgcbpinter handles the energy surplus or demand of the VPPs and the transactions between them. Therefore, it contains a method for the matching of energy surpluses and demands of the VPPs for a given timeslot. In contrast to the sgcbpinter chaincode, the sgcbpintra chaincode integrates encryption and decryption for the amount and the type of a participant's energy production or consumption. The chaincodes have been written in Golang.

D. Applications

The applications for the platform and the simulation, which operate on top of the HLF network, can be divided into following three categories:

- API: acts as connector to the HLF network by providing endpoints for the chaincode functions.
- Simulation: simulates the functionalities of the participant, the controller or the master.
- Dashboard: displays different data depending on the role in the platform, e.g., participant's energy consumption and production.

All applications were written in NodeJS. Further, to deploy the applications onto the RPIs, the applications were built into Docker images.

E. Smart Grid Controller Container Services

The IBM Container Services (ICS)[9] are usually used to launch an IBM Blockchain network on the IBM Blockchain Platform. For the deployment onto the RPIs, a modified version of the ICS was created, called Smart Grid Controller Container Services (SGCCS). Before deploying the components to the RPIs, the chaincodes, the HLF configuration files and the simulation datasets had to be setup. All components of the SGCBP, including the HLF network entities and the applications, can be deployed by using the SGCCS. The

[5]HLF on RPI 3 https://stackoverflow.com/questions/45800167/hyperledger-fabric-on-raspberry-pi-3/45804324 (accessed: 3/19)

[6]HLF V1.0 on a RPI http://www.joemotacek.com/hyperledger-fabric-v1-0-on-a-raspberry-pi-docker-swarm-part-2/ (3/19)

[7]frbrkoala's Profile https://hub.docker.com/u/frbrkoala/ (accessed: 3/19)

[8]HLF Samples github.com/hyperledger/fabric-samples (accessed: 3/19)

[9]IBM Blockchain Platform for Developers on IBM Container Service https://github.com/IBM-Blockchain/ibm-container-service (accessed: 3/19)

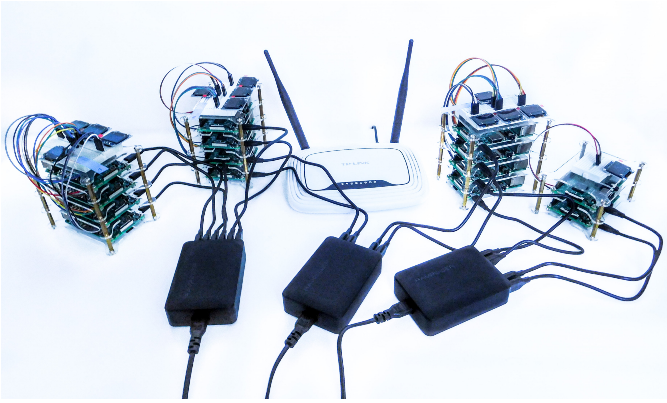

Fig. 1. SGCBP Raspberry PI cluster setup

SGCCS utilizes kubectl, which is a command line interface for managing Kubernetes clusters. It enables to create Kubernetes objects, such as pods and services, by providing a YAML-file with object specifications. The SGCCS JSON configuration file contains necessary information for deployment, namely IP addresses, hostnames, chaincodes, and channels.

V. Testbed Setup and Preliminary Results

The SGCBP functionality was successfully tested by simulating the three diverse VPPs, balancing consumers, producers, and commercial for days. The VPP producers have energy sources such as a PV farm, a biogas power plant, or a small wind power plant. The VPP commercial group consists of a retail store, a bakery, and a large PV plant. The VPP consumers are represented as a residential street with one general household, one household equipped with a hot water tank, and one household with a non-switchable electric vehicle charging station. For this setup, see Fig. 1, 14 RPIs were used (kubemaster RPI, master RPI, three controller RPIs, nine participant RPIs). This diversity mix and normalized profiles enable thorough test of the platform operation, especially the matching algorithm. The *Verband der Elektrizitätswirtschaft* and *Energie-Control Austria für die Regulierung der Elektrizitäts- und Erdgaswirtschaft* [13] standard load profiles are used for consumers in forecasting the balancing energy demand of balance groups. All profiles include quarter-hourly values for a year, that are normalized to an annual energy consumption of 1,000 kWh.

VI. Discussion & Outlook

The paper presents a functional blockchain-based Smart Grid Controller for local and regional consumer driven decentralized virtual power plants.

A novelty for the development of the platform was to port the HLF components onto the Raspberry PI 3B ARMv7 architecture. For the platform processes, e.g., protocolling and matching, the chaincodes sgcbpintra and sgcbpinter have been created. The SGCCS consists of a modified version of the IBM Container Services. Based on a configuration file, the HLF network and platform applications are deployed onto RPIs.

The code is available in the GitHub repositories[10]. All created Docker images are located on the Docker Hub goranovic. The created HLF components and the SGCCS can be used to deploy a HLF network onto a Kubernetes cluster of Raspberry PIs. Further, the SGCBP includes all necessary applications, such as API for connection to the HLF network, dashboard for querying the ledger, visualizing the data, and simulations for writing data to the ledger. The platform can be used as basis to study the effect of blockchain communication in the smart grid domain and their development.

For blockchain technology the realization of applications has just begun. It will need to mature and overcome several challenges, technological, but also regulatory. As most projects focus on P2P energy trading, these applications will likely be the first to be market-ready. Nevertheless, other use cases should be considered.

References

[1] Farhangi, Hassan: The path of the smart grid. In: IEEE power and energy magazine 8 (2010), Nr. 1

[2] Lugmaier, Andreas ; Fechner, Hubert ; Pruggler, Wolfgang: National technology platform-smart grids austria. (2008)

[3] Pudjianto, Danny ; Ramsay, Charlotte ; Strbac, Goran: Virtual power plant and system integration of distributed energy resources. In: IET Renewable power generation 1 (2007), Nr. 1, S. 10-16

[4] Saboori, H ; Mohammadi, M ; Taghe, R: Virtual power plant (VPP), definition, concept, components and types. In: Power and Energy Engineering Conference (APPEEC), 2011 Asia-Pacic IEEE, 2011, S. 1-4

[5] Nakamoto, Satoshi: Bitcoin: A peer-to-peer electronic cash system. (2008)

[6] Tschorsch, Florian ; Scheuermann, Bjorn: Bitcoin and beyond: A technical survey on decentralized digital currencies. In: IEEE Communications Surveys & Tutorials 18 (2016), Nr. 3, S. 2084-2123

[7] Szabo, Nick: Smart contracts. (1994). - http://www.fon.hum.uva.nl/rob/Courses/InformationInSpeech/CDROM/Literature/LOTwinterschool2006/szabo.best.vwh.net/smart.contracts.html (accessed: 1/2019)

[8] Sabounchi, Moein ; Wei, Jin: Towards resilient networked microgrids: Blockchainenabled peer-to-peer electricity trading mechanism. In: EI2, 2017, S. 1-5

[9] Hahn, Adam ; Singh, Rajveer ; Liu, Chen-Ching ; Chen, Sijie: Smart contractbased campus demonstration of decentralized transactive energy auctions. In: Power & Energy Society ISGT, 2017 IEEE, S. 1-5

[10] Li, Zhetao ; Kang, Jiawen ; Yu, Rong ; Ye, Dongdong ; Deng, Qingyong ; Zhang, Yan: Consortium blockchain for secure energy trading in industrial internet of things. In: IEEE transactions on industrial informatics 14 (2018), Nr. 8, S. 3690-3700

[11] Goranovic, Andrija ; Meisel, Marcus ; Fotiadis, Lampros ; Wilker, Stefan ; Treytl, Albert ; Sauter, Thilo: Blockchain applications in microgrids an overview of current projects and concepts. In: IECON 2017 - 43rd Annual Conference of the IEEE, 2017, S. 6153-6158

[12] Pichler, Mario ; Meisel, Marcus ; Goranovic, Andrija ; Leonhartsberger, Kurt ; Lettner, Georg ; Chasparis, Georgios ; Vallant, Heribert ; Marksteiner, Stefan ; Bieser, Hemma: Decentralized Energy Networks Based on Blockchain: Background, Overview and Concept Discussion. In: BIS, Springer, 2018, S. 244-257

[13] E-Control Load Profiles. - https://www.e-control.at/documents/20903/-/-/90219f4d-ba76-41d6-948f-3d82c82a42e3 (accessed: 3/19)

[10] GitHub rep. goranovic (fabric, fabric-ca, sgcbp, sgcbp-container-services, sgcbp-simulation) https://github.com/goranovic (accessed: 3/19)