Published in: 2022 IEEE International Conference on Communications Workshops (ICC Workshops)
DOI: 10.1109/ICCWorkshops53468.2022.9814557

Predictive 5G Uplink Slicing for Blockchain-driven Smart Energy Contracts

Fabian Kurtz, Robin Wiebusch, Dennis Overbeck, Christian Wietfeld
Communication Networks Institute (CNI), TU Dortmund University, Otto-Hahn-Strasse 6, 44227 Dortmund
Email: {fabian.kurtz, robin.wiebusch, dennis.overbeck, christian.wietfeld}@tu-dortmund.de

***Abstract*—The energy grid is facing a paradigm shift away from traditionally centralized electricity generation towards distributed renewable energy resources. These so-called Smart Grids (SGs) require a mechanism for balancing power consumption and generation. In this context, Blockchain (BC)-based Smart Contracts (SCs) have emerged as a means to facilitate distributed transactions without requiring trust among the involved parties. Yet, resulting communication traffic loads need to be considered. Here, 5G network slicing promises to enable the coexistence of such mission critical services on a single shared physical communication infrastructure. Nevertheless, challenges in terms of latencies and resource efficiency exist. As static slicing mechanisms can be inefficient, we propose a predictive Machine Learning (ML)-driven approach to Resource Block (RB) scheduling by harnessing the Configured Grant (CG) mechanism in the 5G uplink. The developed solution is evaluated on the particularly challenging example of an energy grid driven by SCs. Based on an energy model derived from a real-world setup, we generate corresponding SC communication traffic. For this, predictive 5G slice radio resource allocation is employed to demonstrate significant improvements in terms of latency and spectrum usage efficiency. Thus, ML-enabled 5G network slicing for mission critical SCs is evaluated within large-scalable SGs.**

I. Introduction

Modern societies increasingly depend on services provided by so-called Critical Infrastructures (CIs), including energy, water, health, transportation, public safety and communication systems. In this context a wide variety of use cases for Blockchain (BC)-driven mechanisms have emerged to support decentralized services such as accounting or facilitating transactions. Nevertheless, their associated traffic loads and criticality require a high level of reliability and performance, not always attainable using traditional communication technologies. Here, 5G enables network slicing, which promises to enable the parallel operation of mission critical applications by providing virtually dedicated networks on a shared physical communication infrastructure. In this work, the Configured Grant (CG) mechanism introduced by 5G is employed to circumvent the associated slice scheduling delays. However, this requires the timing and size of transmissions to be known in advance. By anticipating future communications via Machine Learning (ML), base stations can grant access to User Equipments (UEs) without the latter having to request Resource Blocks (RBs). In contrast to established solutions, this may reduce latencies and increase radio spectrum usage efficiency. For evaluation purposes, we select the particularly challenging application scenario of a Smart Contract (SC)-enabled Smart Grid (SG). Specifically, we employ SCs based on the highly scalable Hyperledger Fabric (HLF) [1] to drive grid operation. This enables the balancing of energy generation and demand on a local level, thus establishing a cellular energy grid topology. In turn, the need for additional power transmission lines [2], otherwise necessary to support the transition towards renewable electricity generation, is reduced. This setup mirrors cellular communication networks as shown by Fig. 1. To achieve realistic results, a detailed energy load/generation model based on real-world data [3, 4] is employed. This drives the highly dynamic and realistic creation of HLF-based SCs within a physical testbed setup. The resulting transmissions are first used to train a Long Short-Term Memory (LSTM)-based ML model, which then serves as input for the predictive scheduling of 5G network slices, performed via a previously developed simulation framework [5]. In summary, this work harnesses ML to predict the resulting mission critical SC traffic, which serves to achieve a reduction in 5G uplink scheduling latency and increase slice resource usage efficiency.

This paper is structured as follows: Sec. II discusses selected related works to highlight this papers's key contributions relative to the state-of-the-art. Next, Sec. III provides insights on the developed dynamic predictive 5G network slicing approach. The cellular grid model, the derived SC traffic generation and the ML-based approach to communication load prediction are presented in Sec. IV. Next, Sec. V details the evaluation scenario, with results discussed in Sec. VI. Finally, a conclusion and an outlook are given.

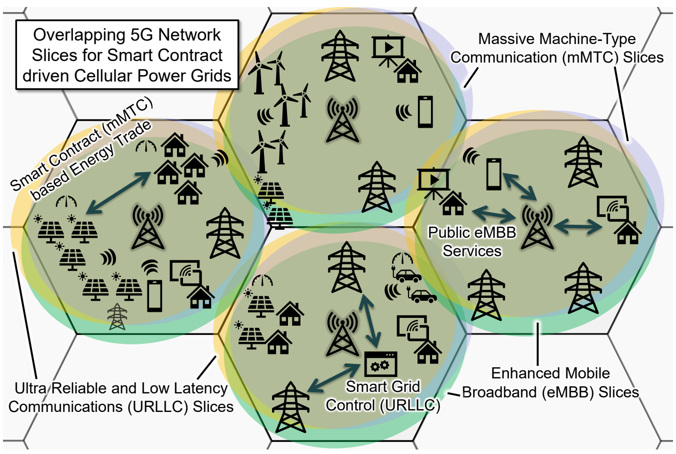

Figure 1: Balancing Cellular Power Grids via Blockchain-based Smart Contracts over Corresponding 5G Network Slices

II. Related Work

This section provides an overview of related work regarding applications of Blockchain-based Smart Contracts for energy load balancing as well as the topic of machine learning driven resource allocation in the context of 5G network slicing.

The authors of [6] identify potential issues of 5G and beyond in areas of security, automated management requirements as well as availability and outline approaches to address these challenges via decentralized Blockchain technologies. In contrast, [7] analyzes Blockchain based decentralization approaches for enhancing the National Institute of Standards and Technology (NIST) conceptual smart grid model. The work of [8] compares HLF as a key representative for private Blockchains and Ethereum as an established public variant in the context of 5G communications. The authors demonstrate HLF to be a sound choice for systems with computational and radio resource constraints, such as those of cellular communication networks. A decentralized Peer-to-Peer (P2P) electricity trading scheme for private consumers and prosumers with Photovoltaic (PV) systems is proposed by [9]. The analysis focuses on the fairness and efficiency of the trading scheme, with HLF serving as platform. In contrast, a field deployment of a HLF-based P2P Smart Contract trading network is given in [10]. Reduction of electrical peak loads as well as individual energy bills are achieved and validated on the example of a real-world microgrid containing four households. The deployment of HLF and Ethereum for energy trading use cases, i.e. akin to the former's role in this work, is analyzed by [11] and [12]. They conclude that HLF outperforms Ethereum in terms of scalability of the transaction throughput, yet compromises in terms of decentralization. The influence of communication delays on the transaction flow of HLF are studied by [13] and [14]. Both show a reduction in the Blockchains' throughput, with the latter observing system breakdowns in the worst case. Hence, hard service guarantees are crucial to ensure reliable operation of future cellular power grids based on Smart Contracts.

Several approaches of implementing Artificial Intelligence (AI) and ML for improved resource allocation in 5G exist in literature. An overview of ML assisted applications for Radio Access Network (RAN) slicing and automated Radio Access Technology (RAT) selection is given by [15], who also discuss corresponding research challenges. A Deep Reinforcement Learning (DRL) based model for the automated management of communication, computing and caching resources is presented by [16]. Another approach to DRL is used in [17], minimizing RB allocation in a 5G RAN. The authors achieve high RB utilization while fulfilling the requirements of relevant network slices. In [18] various AI based strategies for 5G resource management are considered. It focuses on long-term slice orchestration. Instead of selecting Recurrent Neural Networks (RNNs), traditionally deployed for time-dependent regression, the authors develop a Convolutional Neural Network (CNN) which allows for the integration of data traffic and geographical dependencies. In contrast, this paper focuses on enabling hard service guarantees at low latencies for critical SC traffic as generated by the SG. Thus reliable grid operation is ensured while providing efficient use of shared radio resources in a sliced 5G network.

III. Dynamic Predictive 5G network slicing

This section provides an overview of the RB scheduling mechanisms employed for implementing dynamic predictive 5G network slicing via ML. Also, the overall framework developed for scheduling uplink radio resources in a SC-enabled cellular grid is introduced. In this work we employ our purpose-developed network slicing framework [5], designed to simulate static, dynamic reactive and dynamic predictive slicing, c.f. Fig. 2. It adheres to the 3GPP's 5G specifications [19] and supports the CG mechanism for assigning uplink RB to UEs. Specifically, CGs allow circumventing the traditional process of using Scheduling Request Occasions (SROs) to request radio resources by defining fixed allocations in the future. The challenge associated with this method arises as the precise packet sizes and timing (on the order of ms) need to be predicted to derive the amount of RBs required in each slice. In case too few RBs are allocated proactively, transmissions of the mission critical SC and grid control slices have to be re-scheduled. Such mistakes increase end-to-end latencies, counteracting this key goal. Conversely, if too many RBs are assigned, radio resource efficiency is reduced at the expense of less critical network slices. A perfect model (i.e. $100\,\%$ prediction accuracy) would reduce the scheduling latency to $0\,\mathrm{ms}$ for any given data transmission, and maximize spectral efficiency. As energy generation and consumption are highly dynamic, it is particularly challenging but also mission critical to predict this traffic. It results from SCs, as created by customers within a cellular grid.

The upper part of Fig. 2 depicts the traditional reactive approach to allocating uplink resources. While RBs are used highly efficiently, as only requests for those which are actually required are sent, a delay is incurred until any uplink transmission can occur. Alternatively, static slicing can be used, shown in the middle. Here, resources are scheduled ahead of time, but in fixed intervals and sizes. This approach is thus

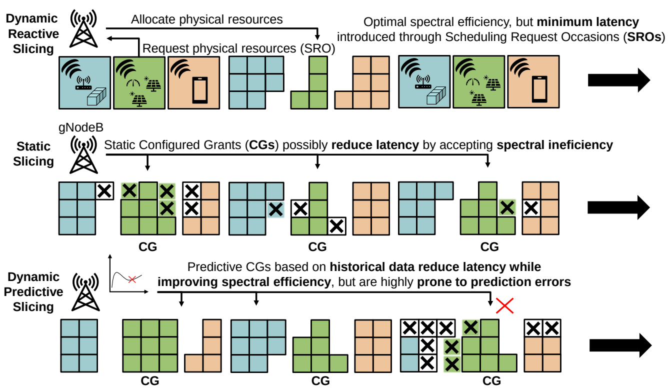

Figure 2: Comparison of Traditional Static / Reactive and the Proposed Predictive 5G Slice Resource Allocation Schemes

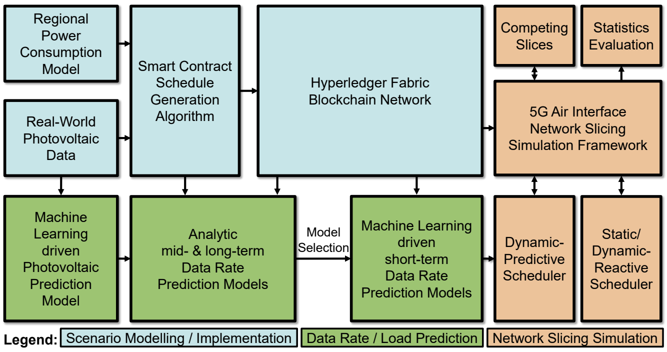

Figure 3: Framework Developed for Predictive 5G Slice Scheduling in Smart Contract enabled Cellular Power Grids

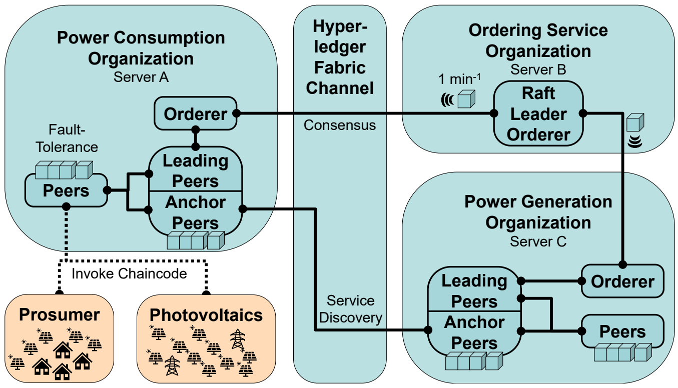

Figure 4: Hyperledger Fabric Framework for Enabling Smart Contracting in Cellular Power Grids

particularly well suited for demanding traffic flows with highly regular characteristics in terms of timing and transmission sizes. Otherwise, inefficiencies and/or additional delays arise, as caused by reactive scheduling for irregular traffic. Finally, the proposed dynamic predictive slicing aims to avoid these issues by harnessing ML to accurately predict future RB usage.

The overall framework for studying the trade-off between latency and RB usage efficiency is depicted in Fig. 3. Its top left corner shows the building blocks of our power grid model, which aims to balance local energy consumption and supply via a BC-based SC trading scheme. Thus, a robust decentralized energy market is created, allowing participants to buy and sell energy based on individual demands. The resulting SC traffic is used as input for the ML-driven communication prediction model. Finally, this predicted transmissions are used to drive the CG-based 5G network slicing scheduler.

IV. Blockchain-based Smart Contracts for Cellular Power Grids

This section details the generation of real-world Smart Contracts and the ML model for predictive 5G slice scheduling.

A. Hyperledger Fabric

A HLF network is implemented to execute an energy trading scheme between a PV plant and consumers in a cellular grid.

1) Structure of the Hyperledger Fabric Framework: The chosen HLF network design supports the goal of generating real-world data traffic. Two HLF organizations representing consumers respectively producers of electricity are implemented. As shown in Fig. 4, each organization maintains an orderer node, an anchor/leading peer for communicating with the ordering service and internally, as well as additional peers for fault-tolerance. A third orderer, run by a neutral organization, enables the consensus mechanism RAFT [20] which requires the majority of orderer nodes to be working and non-fraudulent. Cryptographic credentials are generated using HLF *cryptogen*, which replaces certificate authorities in this work. The HLF nodes are deployed as Docker containers which are connected using swarm mode.

2) Block Interval Configuration: The generation of blocks on the smart contract Blockchain significantly impacts the resulting communication load. To ensure a cellular energy grid's frequency stability, European Network of Transmission System Operators for Electricity (ENTSO-E) specifies that sufficient generating capacity (secondary reserve) has to be available within a maximum of $5\,\mathrm{min}$. Hence, fixed block transmission intervals (BIs) of $1\,\mathrm{min}$ are generated. A Block Interval (BI) consists of two phases:

- **Transactions Transmission:** Achieving a throughput of 3 transactions per second, the first $40\,\mathrm{s}$ are reserved for transmitting up to 120 previously defined energy contracts. Here, no blocks may be transmitted.
- **Block Transaction:** A block containing the ordered transactions of the BI is transmitted on the 50^{th} second of the block interval. The next $10\,\mathrm{s}$ serve to ensure successful block transmission, regardless of any communication issues (e.g. latencies), as well as to regulate energy flows.

HLF offers two parameters to control the occurrence and size of blocks. The *BatchTimeout* parameter defines how long the ordering service waits after receiving the first transaction until it creates a new block. It is hence set to $50\,\mathrm{s}$. *BatchSize* controls the maximum number of messages within a block as well as its size in Byte. To preclude premature block creation, its values are set to *MaxMessageCount = 1000* and *AbsoluteMaxBytes = PreferredMaxBytes = 1GB*.

B. Machine Learning Models for SC Traffic Prediction

In preliminary studies LSTM outperformed Seasonal Autoregressive Integrated Moving Average (SARIMA) and random forests for time-series prediction of SC traffic. Hence, its implementation is discussed below.

1) Preprocessing of the Generated Time Series Data Sets: First, the smart contract transmission dataset is split into $70\,\%$ training, $20\,\%$ validation and $10\,\%$ test data sets. The data is subsequently scaled to the interval of $[0, 1]$ using scikit-learn's MinMaxScaler [21]. Preserving causality, the scaler is solely fitted on training data which is not shuffled. Next, feature-label pairs are generated using a rolling window. The input sequence has the length of N, which is a tunable parameter,

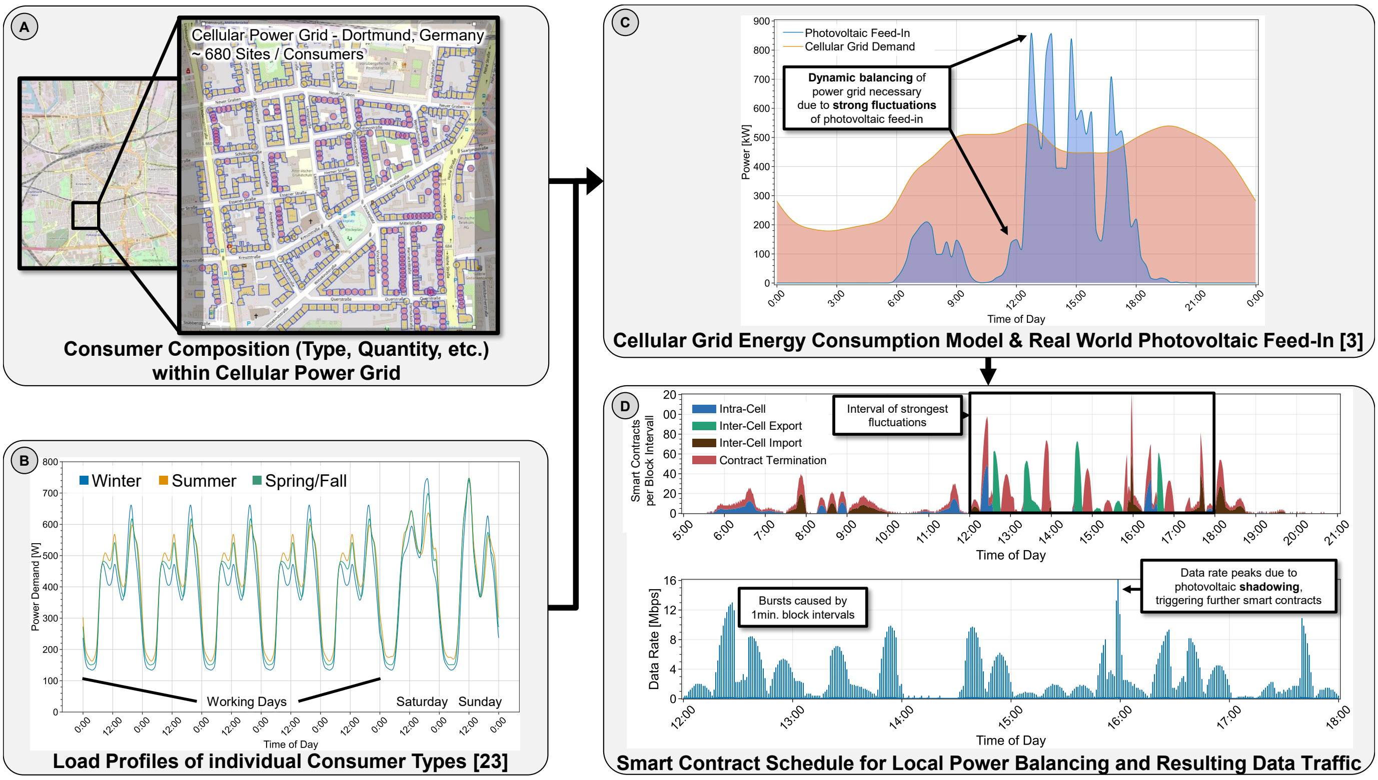

Figure 5: Cellular Power Grid Energy Model based on Real-World Data, Resulting Smart Contracts and Slice Traffic Load

and an output length of 1. After that, M input-output pairs are packed into a training batch. Hyperparameter tuning is executed using bayesian optimization.

2) LSTM Implementation: The model is implemented using Keras [22] and Tensorflow [23], utilizing the tensorflow.keras.sequential API to stack two layers. LSTM serves as its first layer, with the cell number considered as hyperparameter. On top of this a dense layer is stacked to reshape output tensors and the number of predicted features to the same dimension, namely one. Mean Squared Error (MSE) is chosen as loss function, which delivers good performance in general by penalizing large errors. The optimizer algorithm determines how errors are propagated through the network Here, the adam optimizer is considered a good choice since it finds an optimal compromise between learning the most relevant as well as less frequent features. Another hyperparameter is the learning rate.

3) Time Aggregation: Model accuracy is enhanced via time aggregation. Hence, the original sequence's temporal resolution is reduced by integrating consecutive samples into a single sample. We apply this strategy to train multiple sub-models on different aggregation levels and calculate an aggregated model as a weighted sum of the partial predictions. Longer-term forecasts broaden the overall prediction in the time domain and thus reduce the challenge of exact timing, while short-term forecasts capture the dynamics of bursts in data traffic.

V. Evaluation Scenario & Experimental Setup

To evaluate the setup, real-world Blockchain data traffic is generated using a local energy balancing scenario. For that purpose, first the Kreuzviertel area in Dortmund, Germany (Fig. 5-A) is used to model a power grid cell. Next, the number of participants, consisting of households and small businesses like bakeries and restaurants, is determined using publicly available map data [3]. For each consumer type a matching power consumption profile, e.g. of single household as depicted in Fig. 5-B, is applied via the real-world based reference data of [4]. The final power consumption model as shown by the red curve in Fig. 5-C results from summarizing the demand profiles of the energy cell's participants. Next, the power feed-in provided to the energy cell by a PV plant needs to be considered. This data is provided by the local utility DEW21 [2] and is shown in Fig. 5-C (blue). Here a fluctuating discrepancy between demand and supply can be seen, highlighting the need for dynamic and decentralized energy contracts. Thus, a minute-wise smart contract schedule is derived from the power delta, as shown in the upper half of Fig. 5-D, to ensure reliable grid operation. Four kinds of smart contracts are considered. An *intra-cell* variant is used to distribute power from the PV plant to the consumer of the local smart grid cell. In case a power surplus can be sold to an adjacent cell, *inter-cell export* contracts are used. Conversely, *inter-cell imports* are used to supply consumers from outside the local cell in cases of insufficient PV power output. As these SCs have no expiration date, they are dissolved manually whenever necessary using *contract terminations*. Finally, this strategy results in Blockchain transmissions with highly variable data rates as shown in the lower part of Fig. 5-D.

Experimental Setup for Predictive 5G Network Slicing

As discussed in the previous section, a realistic energy model of a local cellular power grid is developed and employed to drive the creation of Blockchain-based smart contracts. While electrical power flows remain purely virtual, the corresponding Blockchain network is setup on a physical testbed. The setup consists of four servers equipped with Intel Xeon-D1518 CPUs (4x $2.2\,\mathrm{GHz}$), $16\,\mathrm{GB}$ RAM, $1\,\mathrm{Gbps}$ Ethernet via Intel I350 NICs and Ubuntu 18.04.5 LTS. This enables the implementation of the aforementioned HLF-network, allowing real-world data traffic between actors to be captured for ML-analysis. The data traffic is subsequently used as input for the proposed predictive 5G network slicing scheduler.

Table I: 5G Network Slicing Configuration

Slice	Smart Contracts (SCs)	Grid Control (IEC 61850 MMS)	Best Effort (BE)
Slice Priority & 5G Profile	high mMTC	high URLLC	low eMBB
Data Rate	average: $20\,\mathrm{kbps}$ burst: $285\,\mathrm{Mbps}$	$122\,\mathrm{Byte}$ every 2 TTIs	$2370\,\mathrm{kbps}$
Latency Requirements	high	highest	low
Static Resource Allocation per UE and TTI [Byte]	1000	122	~ 1800

Overall, a setup with three 5G radio network slices is considered, as shown in Tab. I. First, the SC slice serves to support the mission critical balancing of regional energy flows by transmitting traffic of HLF orderers, anchors and peers as represented within the testbed (c.f. Fig. 4). Due to the large number of nearly 700 participants within a small geographic area, the associated data rate is highly variable as individual transmissions may occur simultaneously. Additionally, a smart grid control slice transmits International Electrotechnical Commission (IEC) 61850 Manufacturing Message Specification (MMS) data which is essential for controlling the cellular grid. Its latency requirements are even more stringent and thus serve to validate if the proposed approach is capable of handling multiple challenging traffic types, i.e. high priority slices. Finally, a Best Effort (BE) slice with comparably high data rate but low latency requirements is considered. Thereby full utilization of all available 5G air interface resources is ensured. The radio specifications used for configuration of the developed 5G scheduler framework are shown in Tab. II. To judge the effectiveness of the employed ML-based approach to predictive 5G network slicing, results are discussed in the following. A specific focus is placed on scheduling latencies as well as efficiency in terms of radio resource utilization.

Table II: 5G Radio Configuration for 5G Slice Scheduling

Transmission Time Interval (TTI) [ms]	Channel Bandwidth [MHz]	Subcarrier Spacing (SCS) [kHz]	Modulation and Coding Scheme (MCS)	Scheduling Request Occasion (SRO) Frequency [ms^{-1}]
1	20	15	15 (QAM-64)	1

VI. Evaluation Results

Depicted in Fig. 6 are the results observed in terms of average latencies caused by scheduling 5G uplink access. Values for the Smart Contract slice are represented by blue bars, with green indicating Grid Control traffic and red constituting Best Effort transmissions. Starting from the left, a static allocation of available Radio Resource Blocks (RBs) is used as a baseline reference. As dynamic changes in demand can not be accommodated, slices have to be configured for peak data rates. Accordingly, critical applications are treated preferentially compared to BE traffic, which has to wait for any remaining capacity causing the increased latencies observed. In case of dynamic reactive scheduling all three slicing achieve the same $1\,\mathrm{ms}$ latency. For SC and grid control traffic this is an increase compared to the static variant. Both may have to wait for their chance to access the radio channel as BE packets now fill in any short gaps that may arise in their transmission patterns. Hence, the delay experienced by BE users is improved slightly. Next, results of a dynamic predictive scheduling mechanism are shown, which bases on a manually configured analytic approach. It is derived by analyzing the respective traffic patterns and serves to assess possible gains unlocked by the subsequently given ML-based solution. Specifically, a decomposition of SCs transmissions overlapping sub-processes is used, namely communication by peers, anchors and orderer nodes. This strategy is also applied to the ML-based approach. Relative to static slicing, SCs experience a $21\,\%$ latency increase, while other services' delays are effectively reduced to zero. It can be seen as an indicator, that their mainly periodical traffic patterns are captured efficiently by analytic modelling, while Smart Contracts are more complex. This observation is underlined by the results of ML-driven predictive slicing. Here the latency penalty for SCs is reduced to $5\,\%$, but nevertheless falls short of the static case's performance. The highly variable flow of energy and the resulting Smart Contract data transmissions were thus not captured successfully on a ms level, as required for optimal 5G scheduling. Nevertheless, detailed analysis reveals that the points in time when data packets need to be sent is predicted with high accuracy. However, the anticipated transmission size, i.e. the amount of resources required, proves insufficient.

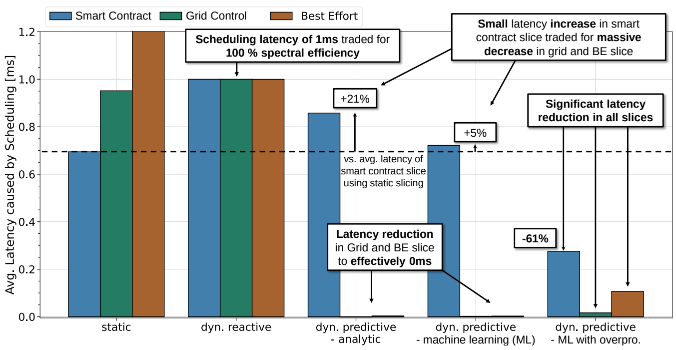

Figure 6: Reactive and Predictive 5G Network Slice Scheduling Latencies, Demonstrating the Effectiveness of ML

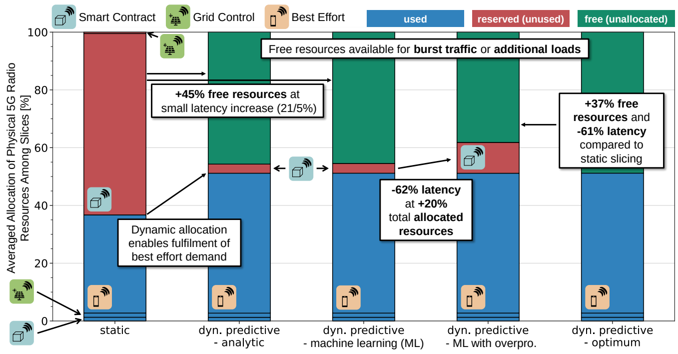

Figure 7: Efficiency of Radio Resource Usage for Traditional and ML-driven 5G Network Slicing Strategies

This is addressed by over-provisioning, i.e. scaling the size of predicted transmissions by a factor of five. As shown by the rightmost group of bars, this enables latency reductions of 61 % for SCs transmissions, which comes at the cost of slightly worse performance for grid control and BE slices. Hence, ML combined with over-provisioning yields significant gains relative to static, dynamic reactive and analytic approaches. However, it is crucial to consider if these improvements impact efficiency in terms of 5G radio resource usage. While reactive scheduling only uses RBs if required, this high efficiency comes at the cost of increased latencies as previously detailed. As shown on the left of Fig. 7, static slicing accounts for peak data rates by allocating all resources available, inevitably causing inefficiencies (red). In contrast, dynamic predictive slicing anticipates transmissions, freeing up about 45 % (green) of 5G RBs on average. This applies to both analytic and ML-based strategies, with the latter reserving slightly more unused RBs (red) but incurring a lower latency penalty versus static slicing of plus 5 % compared to 21 %. Finally, combining ML with over-provisioning requires the reservation 20 % more resource which on average remain unused (i.e. mis-predictions/allocations). Crucially, this enables latencies to be reduced by 61 %, representing an acceptable compromise for the chosen application. All dynamic mechanisms use the same amount of RBs as optimal (i.e. 100 % correct) predictive scheduling. Yet, the amount of reserved but unused spectrum varies, resulting in the discussed latency/efficiency trade-offs.

VII. Conclusion and Outlook

In this work we present a Machine Learning driven approach to predictive 5G network slicing, on the example of blockchain-based SCs for future cellular Smart Grids. First, a detailed energy model is created to drive real-world SC communications on a physical testbed. It forms the basis for creating a LSTM model to predict transmissions of the mission critical blockchain on the order of ms. These predictions drive our 5G standard-based network slicing framework to accurately simulate pro-active scheduling behaviour. The efficacy of this approach to ML-driven Smart Contract data traffic prediction is clearly demonstrated as uplink latency reductions of 61 % are achieved, while freeing up 37 % of radio resources for use by other slices and applications. Future work will focus on transferring the proposed solution onto a live 5G network.

Acknowledgement

This work is supported by the *5Gain* Project (FKZ: 03EI6018C) as well as the *6GEM* research hub (FKZ: 16KISK038), funded by the German Federal Ministry of Economic Affairs and Energy (BMWi) respectively the German Federal Ministry of Education and Research (BMBF). The authors would like to thank the consortia for their valuable contributions.

References

[1] C. Gorenflo *et al.*, 'FastFabric: Scaling Hyperledger Fabric to 20,000 Transactions per Second', in *IEEE International Conference on Blockchain and Cryptocurrency (ICBC)*, 2019, pp. 455–463. DOI: 10.1109/BLOC.2019.8751452.

[2] *5GAIN: 5G Infrastrukturen für zellulare Energiesysteme unter Nutzung künstlicher Intelligenz*, Funded by the German Federal Ministry for Economic Affairs and Energy (BMWi) (funding reference: 03EI6018C), 2019-2022. [Online]. Available: https://5gain.info/.

[3] OpenStreetMap Contributors, *Planet dump retrieved from https://planet.osm.org*, https://www.openstreetmap.org, 2021.

[4] Verband der Elektrizitätswirtschaft (VDEW), *Repräsentative VDEW-Lastprofile*, 1999. [Online]. Available: https://www.bdew.de/media/documents/1999_Repraesentative-VDEW-Lastprofile.pdf.

[5] C. Bektas, D. Overbeck and C. Wietfeld, 'SAMUS: Slice-Aware Machine Learning-based Ultra-Reliable Scheduling', in *IEEE International Conference on Communications*, Virtual Event, Jun. 2021.

[6] K. Yue *et al.*, 'A Survey of Decentralizing Applications via Blockchain: The 5G and Beyond Perspective', *IEEE Communications Surveys & Tutorials*, pp. 1–1, 2021.

[7] A. Aderibole *et al.*, 'Blockchain Technology for Smart Grids: Decentralized NIST Conceptual Model', *IEEE Access*, vol. 8, pp. 43 177–43 190, 2020.

[8] I. R. Fedorov *et al.*, 'Blockchain in 5G Networks: Perfomance Evaluation of Private Blockchain', in *Wave Electronics and its Application in Information and Telecommunication Systems*, 2021, pp. 1–4.

[9] K. Zhou *et al.*, 'Credit-Based Peer-to-Peer Electricity Trading in Energy Blockchain Environment', *IEEE Transactions on Smart Grid*, pp. 1–1, 2021.

[10] S. Saxena *et al.*, 'Design and Field Implementation of Blockchain Based Renewable Energy Trading in Residential Communities', in *Intern. Conf. on Smart Grid and Renewable Energy*, 2019, pp. 1–6.

[11] H. Malik *et al.*, 'Performance Analysis of Blockchain based Smart Grids with Ethereum and Hyperledger Implementations', in *IEEE Conference on Advanced Networks and Telecommunications Systems (ANTS)*, 2019, pp. 1–5.

[12] Y. Kim, K.-H. Kim and J.-H. Kim, 'Power Trading Blockchain using Hyperledger Fabric', in *Conference on Information Networking (ICOIN)*, 2020, pp. 821–824.

[13] F. Geyer *et al.*, 'Performance Perspective on Private Distributed Ledger Technologies for Industrial Networks', in *Conference on Networked Systems (NetSys)*, 2019, pp. 1–8.

[14] T. S. L. Nguyen *et al.*, 'Impact of Network Delays on Hyperledger Fabric', in *IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS)*, 2019, pp. 222–227.

[15] X. Shen *et al.*, 'AI-Assisted Network-Slicing Based Next-Generation Wireless Networks', *IEEE Open Journal of Vehicular Technology*, vol. 1, pp. 45–66, 2020.

[16] S. Fu, F. Yang and Y. Xiao, 'AI Inspired Intelligent Resource Management in Future Wireless Network', *IEEE Access*, vol. 8, pp. 22 425–22 433, 2020.

[17] Y. Abiko *et al.*, 'Flexible Resource Block Allocation to Multiple Slices for Radio Access Network Slicing Using Deep Reinforcement Learning', *IEEE Access*, vol. 8, pp. 68 183–68 198, 2020.

[18] D. Bega *et al.*, 'Network Slicing Meets Artificial Intelligence: An AI-Based Framework for Slice Management', *IEEE Communications Magazine*, vol. 58, no. 6, pp. 32–38, 2020.

[19] 3rd Generation Partnership Project (3GPP), *TS 23.501 V16.4.0 - Technical Specification Group Services and System Aspects; System architecture for the 5G System (5GS); Stage 2 (Release 16)*, 2020.

[20] D. Ongaro and J. Ousterhout, 'In Search of an Understandable Consensus Algorithm', in *USENIX Annual Technical Conference*.

[21] F. Pedregosa *et al.*, 'Scikit-learn: Machine Learning in Python', *Journal of Machine Learning Research*, vol. 12, pp. 2825–2830, 2011.

[22] F. Chollet *et al.*, *Keras*, https://keras.io, 2015.

[23] M. Abadi *et al.*, *TensorFlow: Large-Scale Machine Learning on Heterogeneous Systems*, 2015. [Online]. Available: tensorflow.org/.