Constraint-based Modeling of Smart Grid Services in ICT-Reliant Power Systems

Frauke Oest
Department of Computer Science
University of Oldenburg, Germany
Email: frauke.oest@uni-oldenburg.de

Sebastian Lehnhoff
Department of Computing Science
University of Oldenburg, Germany
Email: sebastian.lehnhoff@uni-oldenburg.de

***Abstract*—The future Information and Communication Technology (ICT) infrastructure in distribution grids requires a significant network and computational resources for potentially running all so-called Smart Grid Services (SGS). The insufficient infrastructure may create network and computational congestions and resource shortages, which can lead to e.g., delayed critical messages in the power system and thus affect the power system stability. This paper presents a model for the configuration of SGSs with consideration of the underlying ICT infrastructure as a constraint satisfaction problem. This model is studied in a nominal and overvoltage scenario. The resulting over-constrained problem in the second scenario is relaxed by SGS data-rate reduction, SGS migration or SGS distribution. We show that an over-constrained problem can be relaxed with our proposed strategies.**

Keywords*—*smart grids*; *information and communication technology*; *quality of service*; *constraint satisfaction problem

I. Introduction

The operation of modern energy systems is based on a number of measurement, control and automation tasks for monitoring and operating extensively distributed resources, e.g., voltage stability monitoring or state estimation, for which different Quality of Service (QoS) requirements must be guaranteed. They serve as the basis for higher optimization functions that realize reliable, efficient and forward-looking overall system operation. In transmission systems, a dedicated and high-performance real-time communication infrastructure guarantees timing behavior and allows parallel execution of these communication-intensive functions and services with very heterogeneous and homogeneous latency requirements and communication demands. Such requirements range from, e.g., 10 ms up to several minutes maximum latency [1]. Due to the expansion of renewable energies at lower voltage level distribution networks and the shift of system responsibility to (operators of) these systems, similar functions and services must also be implemented in distribution systems – so-called *Smart Grid Services* (SGSs) with similar QoS requirements and guarantees. In addition, the future smart distribution system will be supplemented by additional SGSs comprising market and user-based applications, which may use the same communication infrastructure to enable a synergetic use [2]. This would require, similarly to transmission systems, a dedicated, over-provisioned communication infrastructure to run planned and future SGSs. Due to economical reasons, such infrastructure is not likely to be available soon (if ever) for smart distribution grids. Other solutions like wireless-based dedicated infrastructures or the usage of public networks do not provide sufficient and reliable resources to ensure QoS requirements of all potentially running SGSs. Additionally, the behavior and criticality of SGSs may change in different states of the power grid and thus QoS requirements may vary. For instance, in case of sudden changes of fluctuating renewable feed-in, the power grid may enter a critical state in which SGSs (e.g., feed-in management, voltage control, or congestion control) need to stabilize the system. In such critical situations, a reconfiguration might be necessary to meet new QoS requirements, e.g., due to higher sampling rates for increased measurement precision [3]. A solution to this problem could be the reconfiguration of SGS in the ICT-system with regards to the power system. Such reconfiguration can include a controlled reduction of data rates, migration of SGS to another server or a change in the overlay topology. This implies, that besides network QoS, also computational demands of SGSs must be considered to avoid congestion on servers. These SGS requirements and the limitations of the physical ICT infrastructure (i.e., computational resources on server, maximum line bitrates in the communication infrastructure) can be denoted as constraints and hence, this may be modeled as constraint satisfaction problem.

A. Related Work

Several aspects of flexibilization of communication for smart grids can be found in the literature. Virtualization concepts like Software-Defined Networking (SDN) may build the foundation to enable such flexibilization. In SDN, the network control (running on the SDN controller) is separated from the data flow (running on SDN devices). The infrastructure components (SDN devices) will be more quickly and adaptively configurable through abstract SDN applications in the SDN controller. One of the central features of SDN is the rapid adaptability of flows and packet routes and the implementation of QoS mechanisms [4]. In smart grids, the use of SDN has enhanced the reliability of field device communication through fast migration of functionality from a failed device towards a redundant device [5] and enabled a power system-dependent prioritization of SGS communication triggered by one SGS [6]. Another approach to virtualization in the communication system is called Network Function Virtualization (NFV), in which the network functions (e.g., routing, firewall, load balancer) are virtualized as an entire function and can thus be flexibly moved and multiplied [7]. This has been applied in smart grid communication where it demonstrates how this flexibility in communication may increase the dependability of

 ISBN: 978-1-61208-967-6

metering communication [8] or functionality may be moved towards the edge of the network near field devices in order to reduce traffic and communication delays [9]. Similar to NFV, the Grid Function Virtualization (GFV) concept enables the migration of SGSs. This concept has been studied in a simulation in case of server failures with a running SGS for voltage control. The affected SGS could be migrated after the failure, improving the voltage quality much faster compared to non-GFV simulations [10]. In another study, GFV was used to decrease the data rates of a non-prioritized SGS to favor QoS requirements of the prioritized SGS [11].

B. Contribution

This paper presents a constrained-based model for SGS configurations considering QoS and computational requirements for the underlying ICT infrastructure, which can be solved by using constraint satisfaction programming. This model is studied in a use case with two power system-based scenarios in which one scenario leads to non-satisfiability of constraints. With this, the usage of SGS flexibilities, such as data-rate reduction, migration and distribution, is motivated as a better controllable form of problem relaxation.

The remainder of the paper is structured as follows: The constraint-based model is defined in Section II. In Section III this model is integrated into an experimentation environment and scenarios with exemplary SGSs are defined. Furthermore, flexibilities of SGSs are introduced to resolve the constraint satisfaction problem if the scenario is over-constrained (e.g., due to changes in QoS of one SGS). Section IV presents the results for the nominal scenario and different over-constrained scenarios with and without using SGSs flexibilities showing the selected solutions and the computational performance of the process. Finally, Section V summarizes the paper, draws a conclusion, and presents ideas for future research.

II. Constraint-based Model

A Smart Grid Service (SGS) is an application serving the operation of the power grid. It contains field devices, such as sensors and actors in the power grid, and may contain a central server for processing field device data. SGSs impose requirements on the ICT infrastructure, such as maximum latency or minimum bitrate requirement, and minimum computational resources. The allocation of SGSs require a feasible path configuration of the single SGS in a given communication infrastructure and with regards to QoS requirements of SGSs. Therefore, this can be categorized as a combinatorial problem (such as other resource allocation problems) and may be formalized as a Constraint Satisfaction Problem (CSP). Our problem formalization is based on the definition of CSPs as described in Bartak et al. [12]. For this, the i-th SGS from a set of n SGSs will be associated with a decision variable x_i. In the following description, we continue to denote i for the SGS i specifics. The domain of each decision variable comprises a set of tuples $d_{i,j} \in D_i$ and represents a set of feasible solution candidates for SGS i, in which a single solution candidates $d_{i,j}$ can be selected by x_i. The solution candidate $d_{i,j}$ can be denoted as follows:

$$d_{i,j} = (p_{h_s}, m_{h_s}, s_{h_s}, a_{\max}, B) \text{ with } d_{i,j} \in D_i. \qquad (1)$$

The constraints $C = \{c_1, c_2, \ldots, c_p\}$ may be defined by n-ary functions, which limit the values from the domain that can be assigned to the decision variable. The domain is determined by the parameterization of the SGS i and therefore, regards the field devices H_{f_i} and server h_{s_i} belonging to this SGS, as well as network (latency α_i, bitrate β_i) and computational QoS requirements (CPU p_i, memory m_i, storage s_i) and the configured overlay topology (centralized or distributed). In a centralized topology, field devices are connected to the central server in a star topology, whereas in the distributed topology, field devices are directly connected, creating a fully-meshed topology. The computational properties are defined as CPU demand p_{h_s}, memory demand m_{h_s} and storage demand s_{h_s}, which are applied to the server node $h_s \in H_{s_i}$. The network properties are determined by maximum path latency $a_{\max}$ and by the bitrate demands for all edges B which are determined by the paths representing the routing or data flow in the underlying ICT infrastructure. The ICT infrastructure is represented by the physical graph $G(V, E)$ defined by vertices V and edges E. A weighted edge is defined as $e = (v', v'', a, b)$, with v' as a source node connected to v'' with the edge weight properties a as line propagation latency and b as maximum line bitrates. The set of vertices V includes infrastructure nodes (e.g., router) R and hosts, such as server nodes H_s and field nodes H_f. To construct the network properties of a solution candidate, subgraphs of the physical graphs are constructed, which contain one feasible simple path for each end-to-end connection. This end-to-end connection is defined by the overlay topology and the host nodes of the SGS i. To create such subgraphs, the paths have to be determined first (e.g., by depth-first search). A path from the source h_1 to the target h_2 is determined by

$$p(h_1, h_2) = (h_1, r_1, \ldots, r_q, h_2) \qquad (2)$$

where $r \in R$ and $h_1, h_2 \in H_s \cup H_f$. The path $p(h_1, h_2)$ is part of the set $P(h_1, h_2)$ comprising all potential paths from h_1 to h_2, which occur if there are cycles in the physical graph. In a preliminary step, a set of path combinations Z_i is constructed comprising one feasible path per end-to-end connection, i.e., one path combination z_i contains the topological information to construct subgraphs connecting all host nodes $h \in H_i$ of SGS i. For this, all end-to-end connections per SGS i are defined by the set of tuples $Q_i = \{(h_1, h'_1), \ldots, (h_n, h'_n)\}$ with $h, h' \in H_i$. The set of path combinations can then be constructed as a cartesian product by

$$Z_i = \prod_{q \in Q} P(q). \qquad (3)$$

Based on this, the subgraph $f_{i,j}$ can be built by adopting the edge latency of the physical graph $G(V, E)$ if the vertices $(v', v'') \in z_{i,j} \in Z_i$ are also $v', v'' \in V(G)$. The edge bitrates $e(b)$ are determined for the edge $(v', v'') \in z_{i,j}$ by the bitrate

 ISBN: 978-1-61208-967-6

QoS requirement β_i of the SGS i for the solution candidate j, such that

$$f_{i,j}(e(b)) = e(b) + \beta_i. \quad (4)$$

This implies, that an edge in this subgraph is used by multiple simple paths, the bitrate weight is adjusted by summing the bitrate weight with the number of paths containing this edge. The bitrate resource B of the solution candidate $d_{i,j}$ can be derived from the subgraph f_i as an adjacency matrix B_{adj}. The end-to-end latency a_{max} of the solution candidates $d_{i,j}$ is determined by

$$a_{\text{max}} = \max_{h_1,h_2 \in H_i} \left(\sum_{e \in E(P_{h_1,h_2})} e(a) \right). \quad (5)$$

The amount of SGSs running on a server $h \in H_s$ is limited by its server resources for cpu h_p, memory h_m and storage h_s. As x_i chooses a solution candidate $d_{i,j}$, the information can be accessed by the function $p_{h_s}(x_i)$ for using the CPU usage on server h_s of this solution candidate. This also applies to $m_{h_s}(x_i)$ and $s_{h_s}(x_i)$. Therefore, the server resource constraints are defined by the constraints c_p, c_m, and c_s

$$c_p := \sum_{x_i \in X} p_{h_s}(x_i) \leq h_p \quad (6)$$

$$c_m := \sum_{x_i \in X} m_{h_s}(x_i) \leq h_m \quad (7)$$

$$c_s := \sum_{x_i \in X} s_{h_s}(x_i) \leq h_s \quad (8)$$

The network constraints are defined by the latency constraint c_a and the bitrate constraint c_b, which should not exceed the maximum physical bitrates of the weighted adjacency matrics E_{adj} of G with bitrates as weights.

$$c_a := a_{\text{max}}(x_i) \leq \alpha_i \quad (9)$$

$$c_b := \sum_{x_i \in X} B_{\text{adj}}(x_i) \leq E_{\text{adj}} \quad (10)$$

In this constraint model, constraints can be relaxed by manually adjusting the SGS parameterization, such as the overlay topology, network or computational QoS requirements.

III. METHODOLOGY

The aforementioned constrained-based model is integrated into a constraint satisfaction program and then exemplary scenarios are defined based on a physical graph and five SGSs with network and computational requirements. These requirements considered in the scenarios depend on the state of the power system (e.g., nominal or overvoltage). The overvoltage scenario leads to a non-solvability of the problem. Therefore, relaxation strategies are presented at the end of this section.

A. Experimentation Setup

The constrained-based model is implemented using NetworkX [13] for solution candidate generation and Minizinc [14] for constraint satisfaction programming and solving. For this, the physical graph is modeled in NetworkX comprising all nodes and edges with latency and bitrate properties. SGSs can be defined by the QoS requirement parameters (latency, bitrate, CPU, memory, storage), the assigned field devices and server, and the overlay topology (centralized or distributed). NetworkX is used to determine the simple paths in order to create subgraphs containing all end-to-end connections to all host nodes (i.e., server and field devices) of an SGS. These subgraphs can be transformed to adjacency matrices for easier bitrate calculation in Minizinc. Thus, NetworkX is used to determine the adjacency matrix for bitrate calculations and the corresponding end-to-end path latency as part of the solution candidate. The SGSs and physical constraints are implemented as a Minizinc model, the solution candidates are integrated via a python interface to Minzinc as a model instance. Gecode [15] is used as a solver in Minizinc. The output of Minizinc consists of a feasible solution candidate for each SGS, which can be visualized in NetworkX.

The model of the physical communication network in Figure 1 is based on the modeling of field devices in the CigreMV power grid [11]. The inner communication infrastructure is adjusted with regards to bitrates and latency. Routers are connected in three subnetworks and one core network linking the subnetworks, which creates a hierarchical infrastructure [16] [17]. The core network is defined by the routers R10, R20 and R30; the subnetworks comprise the routers R11 - R12, R21 - R23 and R31 - 32. The edge of the network is modeled by the 17 field devices (F11 - F16, F21 - F26, F31 - F35) and the two servers S1 and S2. The modeled infrastructure is the most powerful in its core network, followed by the subnetworks and the edge network. Therefore, the core network is modeled with 200 kByte / s bitrate resource and a propagation delay of 20 ms on each edge. The subnetwork is defined with 100 kByte / s bitrate and a latency of 25 ms and the edge network is denoted with a bitrate and latency property of 25 kByte / s and 30 ms. The server S1 contains 6 CPU cores, 16 GB memory and 50 GB storage. The server S2 is modeled with 4 CPU cores, 8 GB memory and 20 GB storage capacity.

B. Scenarios

We test our model from Section II with two scenarios representing two states of the power system. In both scenarios, we consider the SGSs Adaptive Relaying (AR), Coordinated Voltage Control (CVC), Line Monitoring (LM), State Estimation (SE) and Virtual Power Plant management (VPP). The first scenario describes the nominal state of the power system, in which the power system properties (voltage, active and reactive power, temperature of the operating equipment) are within the ideal range, so that grid critical services only need the minimum computational and network resources. The bitrate and latency requirements for SGS AR and LM are based on [18] and the requirements for the SGSs CVC and

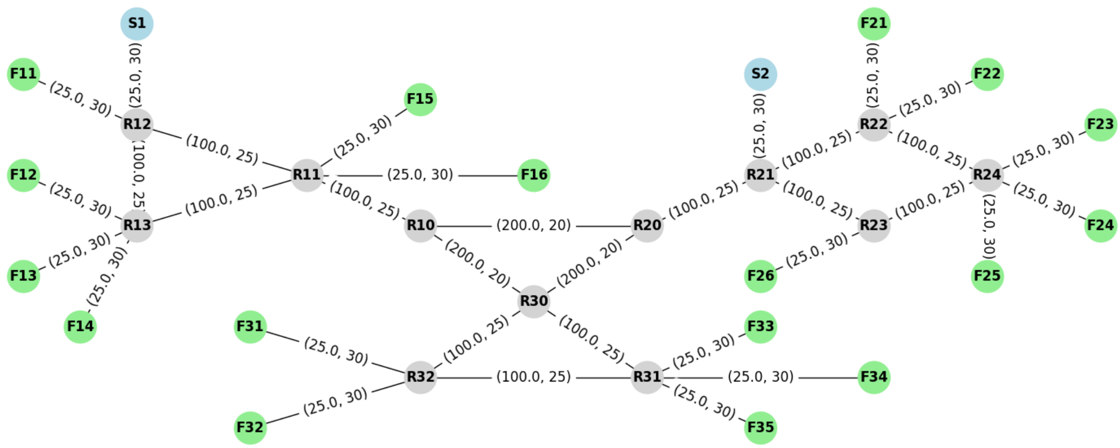


Figure 1. Physical graph with bitrates and latency on the edges.

SE are based on [19]. The bitrate requirements are derived from the sampling rates. For this, we estimate the size of a TCP/IP packet including payload to 100 Byte per sampling value. The QoS requirements for the SGS VPP are derived from the assumptions and results of [20].

The computational demands of SGSs are very dependent on the implementation of the service. Therefore, we try to categorize the demands based on a likely implementation. For this, we determine the computational demands into the categories low, medium and high for each computational demand characteristic. In Table I, these categories are translated into concrete requirements. The CPU resources are modeled as proportions (as millicpu) of one hyper thread on a bare-metal Intel processor. These computational demands are idealized abstractions but may work as a rough estimate.

TABLE I. COMPUTATIONAL RESOURCE DEMANDS FOR SGSS.

Resource demand	*low*	*medium*	*high*
CPU	500m	1.0	4.0
Memory	200MB	2GB	8GB
Storage	500MB	4 GB	16GB

The SGS demands for computational resources are estimated based on some implementation characteristics, e.g., whether historical data is needed, if the SGS can be parallelized (by, e.g., a multi-agent system or some machine learning methods) or the amount of measurement data. Therefore, SGS demands are determined based on the following assumptions: The purpose of the SGS LM is the evaluation of (a limited amount of) measurement data against a threshold [21]. The computational resource demands should be low in each category. The SGS SE for distribution grids is often based on a weighted least-squares approach and thus, needs pseudo-measurements to approximate missing measurements. Those can be achieved by a trained machine learning model [22]. Hence, we estimate the CPU demand as high, the memory demand as medium, and the storage demand as low. For the SGS AR, knowledge about topology, load flow, and characteristics of DGs are needed as an input for, e.g., a machine learning or linear programming-based approach [23]. Therefore, we assume high CPU demand, and medium memory and storage demands. The SGS CVC is threshold-based but needs short-term load and generation forecasts to avoid unnecessary control and switching signals, which may be caused by weather-dependent generators [24] [25]. For this, we assume that weather forecasts are included in the voltage calculation. Therefore, the resource demands are medium in each category. For the SGS VPP, we assume it to be implemented as a multi-agent system that can be run either centralized on a server or distributed by each agent representing a distributed generator. To find an optimal combined operational schedule, each agent may choose a schedule from a pre-defined set of feasible schedules for each generator participating in this VPP. [20]. The resources demands are assumed to be high for CPU, memory and storage. The resulting parameterization for SGSs can be taken from the following Table II.

TABLE II. PARAMETERIZATION OF SGSS FOR THE NOMINAL SCENARIO.

SGS	β *Byte/s*	α *ms*	*c* *cores*	*m* *GB*	*s* *GB*	*field devices*	*server*
AR	2.5	100	1	2	4	F36	S2
CVC	3	500	1	2	4	F16, F22, F35	S1
LM	1	1	0.5	0.2	0.5	F12, F24, F32	S2
SE	0.5	1000	1	2	0.5	F11, F15, F21, F23, F33, F34	S1
VPP	2	800	4	8	4	F13, F14, F25, F31	S1

The second scenario is describing an overvoltage situation in the power grid. Therefore, the QoS and computational requirements of CVC increase leading to a bitrate requirement of at least 6 kByte/s, a maximum end-to-end latency of 200 ms and a CPU demand of 4 cores. The CSP cannot be solved anymore, as this introduces a bitrate shortage from R12 to S1, all paths $P(F35, S1)$ exceed the latency requirement and server S1 cannot supply enough computational resources. We propose the following SGS flexibilities to relax this overconstrained problem:

Copyright (c) IARIA, 2022. ISBN: 978-1-61208-967-6

Reduction: Reduction of QoS requirements of a non-prioritized SGS. In this scenario, the bitrate requirement of SE is reduced to β=1 kBit / s

Virtualization: Migration of an SGS to another server. In this scenario, the SGS CVC is migrated from S1 to S2.

Distribution: Change of overlay topology of an SGS. In this scenario, the SGS VPP switches from centralized to distributed operation, which does not need communication to any server.

We have conducted one experiment for the nominal scenario and six for the overvoltage scenario testing each constraint violation with and without relaxation.

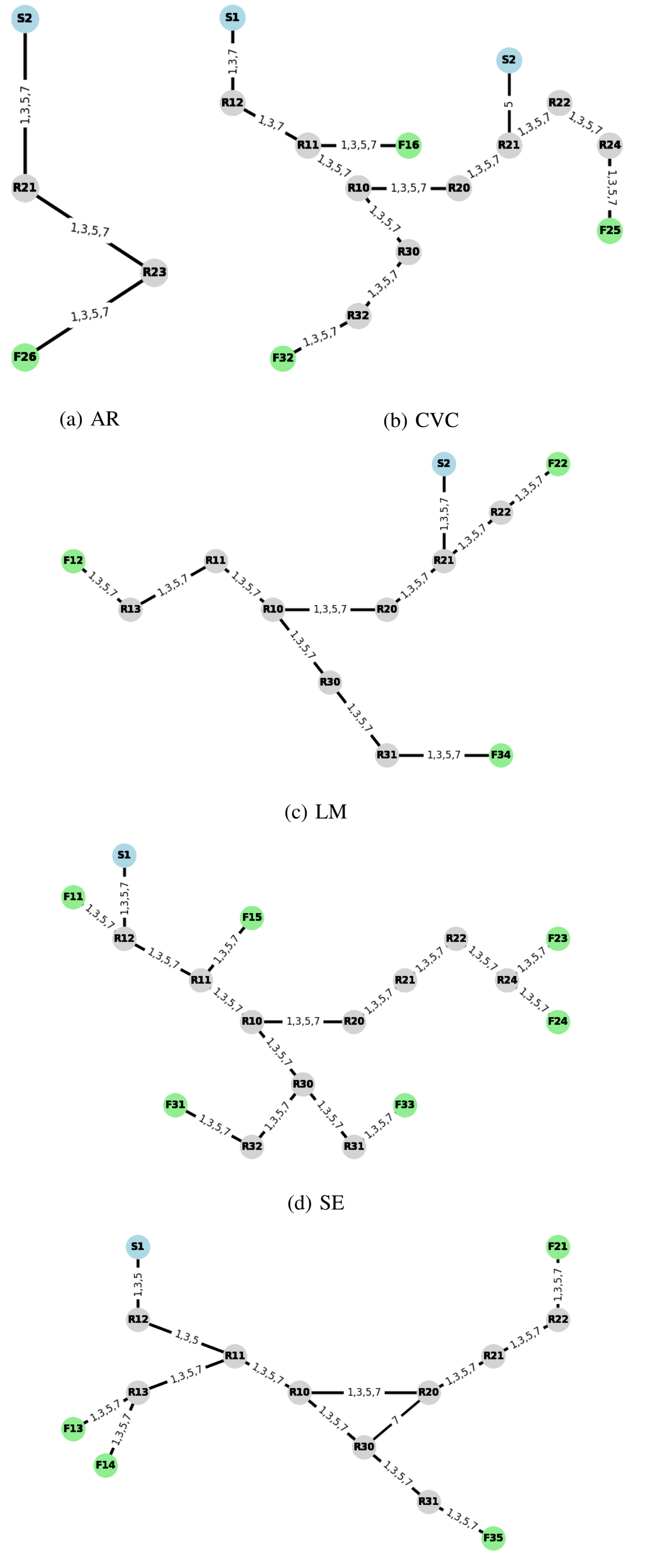
IV. Results

We have conducted seven experiments and present the resulting solutions (each as a possible SGS configuration) and the accompanied solution process data, such as the number of solution candidates per SGS and the calculation time of each step. The experiments are enumerated as follows:

1) Nominal case.
2) Overvoltage case, bitrate demand of CVC is increased, no problem relaxation.
3) Overvoltage case, bitrate demand of CVC is increased, SE communicates with reduced bitrate requirements.
4) Overvoltage case, latency demand of CVC is increased, no problem relaxation.
5) Overvoltage case, latency demand of CVC is increased, CVC is migrated to another server.
6) Overvoltage case, CPU demand of CVC is increased, no problem relaxation.
7) Overvoltage case, CPU demand of CVC is increased, VPP operates in the distributed mode.

A resulting feasible configuration is displayed in Figure 2 for each SGS. The chosen topologies for the SGSs AR (2a, CVC (2b), LM (2c, SE (2d), and VPP(2e) are displayd by the experiment number on the edge of the graphs. Following the idea of a CSP, the first valid solution for the experiment is used. Therefore, often the same first valid solution candidate is chosen. Experiment 1 shows a feasible solution for the nominal scenario in which all constraints can be fulfilled and hence, no relaxation is needed. This experiment is used to illustrate topological changes in the following experiments. In case of overvoltage without a problem relaxation with SGS flexibilization, no solutions can be found. For this reason, experiments 2, 4, 6 cannot be visualized in Figure 2. The successful problem relaxations are visible in experiments 3, 5 and 7. The bitrate reduction of SGS SE conducted in experiment 3 lead to the same topology as in the nominal scenario. In experiment 5, SGS CVC shifted the end-to-end connectivity from the field devices from server S1 to server S2, visible by the edge from R21 to S2. In experiment 7, SGS VPP Management does not have a connection from R12 to S1.

Table III shows the number of generated solution candidates per SGS, the accompanied time to generate solution candidates



(a) AR (b) CVC (c) LM (d) SE (e) VPP

Figure 2. The resulting SGSs configurations as solutions of the CSP.

TABLE III. SIZE OF THE SET OF SOLUTION CANDIDATES AND TIME MEASURES OF THE SOLUTION FINDING PROCESS.

exp.	*number of solution candidates*					*generation time of solution candidates (s)*					*solving duration (s)*	
ID	*AR*	*CVC*	*LM*	*SE*	*VPP*	*AR*	*CVC*	*LM*	*SE*	*VPP*	*mean*	*std.*
1	2	128	32	8192	256	0.0011	0.0355	0.0101	3.3635	0.0857	18.2850	0.4118
2	2	128	32	8192	256	0.0011	0.0348	0.0100	3.5954	0.0884	18.1686	0.3248
3	2	128	32	8192	256	0.0011	0.0346	0.0099	3.5270	0.1037	18.0044	0.2305
4	2	128	32	8192	256	0.0011	0.0355	0.0101	3.5528	0.0775	10.4416	0.2333
5	2	16	32	8192	256	0.0011	0.0061	0.0223	3.5493	0.0821	18.0519	0.3075
6	2	128	32	8192	256	0.0011	0.0347	0.0101	3.5908	0.0740	16.7446	0.4263
7	2	128	32	8192	32768	0.0011	0.0355	0.0099	3.4880	14.5952	84.9177	1.0081

and the mean time and standard deviation to find a solution for the problem created in the experiment. It shows the change of generated solution candidates for SGS CVC and VPP in experiments 5 and 7, where a topology change was conducted by migration or the switch to a distributed topology. In experiment 5, only 16 CVC solution candidates were generated, whereas in experiment 7, 32768 VPP candidates were created, which is 128 times the size of the centralized VPP topology. These differences are caused by the placement of field devices and the server in this example physical graph. In experiment 5, the size decreases due to the decrease of hops per end-to-end connection. The fully-connected topology in experiment 7 leads to six end-to-end connections, which may include 3 cycles in the physical graph. As each traversed cycle creates 2 possible paths, this leads to 8 paths for 5 of 6 end-2-end connections (there is only one possible path for $P(F13, F14)$. This results in a total of 32768 solution candidates.

The time measurements for the generation of solution candidates and the solving are conducted on an Intel Xeon CPU E5-2680 with 4×2.40 GHz and 24 GB RAM. The solving process was performed 25 times and all time measures are in seconds. The time to generate solution candidates is similar in each experiment apart from experiment 5 and 7, where the duration for CVC decreases and the duration for VPP increases. Also, the time to solve the CSP is similar in experiments 1, 2, 3, 5 and 6. The decrease of solving time in experiment 4 is caused by the definition of the latency constraint, as this violation is easy to detect. The increase in solving time in experiment 7 is determined by the size of potential solution candidates.

V. CONCLUSION AND FUTURE WORK

In this paper, we have presented how to flexibilize SGSs and we have shown the potential of this approach by creating a sporadically occurring high ICT-demand situation on a small communication network with five SGSs. We have formalized ICT demands as a constraint satisfaction problem considering communication network requirements (latency and bitrates), and computational requirements (CPU, memory and storage usage). For this, we have studied 7 experiments based on two scenarios: nominal operation and overvoltage. The first scenario serves as a baseline experiment. The latter is defined by an increase of several requirements of the service CVC, leading to an over-constrained problem. We have conducted experiments with each individual adjusted requirement and a corresponding problem relaxation strategy using the flexibilities of SGSs, such as reduction of requirements, migration to other servers or switching to a distributed communication topology. By using these flexibilities, the new requirements of CVC could be fulfilled.

These findings show the potential of such SGS flexibilities to operate the power system resiliently under QoS considerations without the need for a strong over-provisioned dedicated communication infrastructure. SGS flexibilities may improve the maximization of the number of SGS requirements fulfilled by migration or distribution of some SGSs and a controlled degradation of requirements of low prioritized SGSs. The proposed approach needs further extensions in future work. So far, we only distinguish between the two categories prioritized and non-prioritized SGS. Furthermore, there is no optimized selection of the most suitable relaxation strategy. The approach can be enhanced by integrating a fine-grained order of SGSs based on pre-defined power system states and refining our SGS model by defining an order of SGS flexibilities with regard to performance degradations. For larger scale experiments based on larger physical networks and an increased amount of SGSs, we consider refining our constraint-based model and search space modeling (e.g., by pre-filtering the solution candidates) to improve the solving time. We will integrate those ideas to an adaptive process, which will be evaluated in a simulation.

ACKNOWLEDGMENT

This work was funded by the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation).

REFERENCES

[1] D. Griffith, M. Souryal, and N. Golmie, "Wireless networks for smart grid applications," *Smart Grid Communications and Networking*, vol. 9781107014, pp. 234–262, 2010.

[2] B. Sörries, S. Lucidi, L. Nett, and M. Wissner, "Gutachten: Digitalisierung der Energiewende. Topthema 3: TK-Netzinfrastruktur und TK-Regulierung [Report: Digitization of the energy transition. Topic 3: Telecommunications network infrastructure and telecommunications regulation]," Wissenschaftliches Institut für Infrastruktur und Kommunikationsdienste, Tech. Rep., 2019.

[3] P. Das, A. Narayan, D. Babazadeh, P. Baboli, and S. Lehnhoff, "Real-time context-aware operation of digitalized power systems by reporting rate control of pmus," in *2021 IEEE Madrid PowerTech*, 2021, pp. 1–6.

[4] M. H. Rehmani, A. Davy, B. Jennings, and C. Assi, "Software defined networks-based smart grid communication: A comprehensive survey," *IEEE Communications Surveys Tutorials*, vol. 21, no. 3, pp. 2637–2670, 2019.

[5] F. von Tüllenburg, P. Dorfinger, A. Veichtlbauer, U. Pache, O. Langthaler, H. Kapoun, C. Bischof, and F. Kupzog, "Virtualising redundancy of power equipment controllers using software-defined networking," *Energy Informatics*, vol. 2, pp. 1–20, 2019.

ISBN: 978-1-61208-967-6

[6] N. Dorsch, F. Kurtz, S. Dalhues, L. Robitzky, U. Häger, and C. Wietfeld, “Intertwined: Software-defined communication networks for multi-agent system-based smart grid control,” in *2016 IEEE International Conference on Smart Grid Communications (SmartGridComm)*, 2016, pp. 254–259.

[7] R. Mijumbi, J. Serrat, J. L. Gorricho, N. Bouten, F. De Turck, and R. Boutaba, “Network function virtualization: State-of-the-art and research challenges,” *IEEE Communications Surveys and Tutorials*, pp. 236–262, 2016.

[8] M. Niedermeier and H. de Meer, “Constructing Dependable Smart Grid Networks using Network Functions Virtualization,” *Journal of Network and Systems Management*, p. 449–469, 2016.

[9] F. Kurtz, N. Dorsch, and C. Wietfeld, “Empirical comparison of virtualized and bare-metal switching for sdn-based 5g communication in critical infrastructures,” *IEEE NETSOFT 2016 - 2016 IEEE NetSoft Conference and Workshops: Software-Defined Infrastructure for Networks, Clouds, IoT and Services*, pp. 453–458, 2016.

[10] C. Krüger, A. Narayan, F. Castro, B. Hage Hassan, S. Attarha, D. Babazadeh, and S. Lehnhoff, “Real-time test platform for enabling grid service virtualisation in cyber physical energy system,” in *2020 25th IEEE International Conference on Emerging Technologies and Factory Automation (ETFA)*. IEEE, 2020, pp. 109–116.

[11] S. Attarha, A. Narayan, B. Hage Hassan, C. Krüger, F. Castro, D. Babazadeh, and S. Lehnhoff, “Virtualization management concept for flexible and fault-tolerant smart grid service provision,” *Energies*, vol. 13, no. 9, 2020.

[12] R. Barták, M. A. Salido, and F. Rossi, “Constraint satisfaction techniques in planning and scheduling,” *Journal of Intelligent Manufacturing*, p. 5–15, 2010.

[13] “NetworkX - network analysis in python,” https://networkx.org/, accessed: 2022.04.06.

[14] “Minizinc,” https://www.minizinc.org/, accessed: 2022.04.06.

[15] “Gecode,” https://www.gecode.org, accessed: 2022.04.06.

[16] J. F. Kurose, *Computer networking : a top-down approach*. Addison-Wesley, 2017, ch. Network Layer, p. 335.

[17] G. Bedürftig, J. Kapanen, E. M. Metsälä, and J. T. Salmelin ha, *LTE backhaul: planning and optimization*. John Wiley and Sons, 2016, ch. LTE Backhaul, pp. 3–45.

[18] C. Samitier, *Field Device to Central Platform Applications*. Cham: Springer International Publishing, 2017, pp. 29–36.

[19] L. E. Miller, A. Silverstein, D. Anand, A. Goldstein, Y. Makarov, F. Tuffner, and K. Jones, “PMU Data Quality: A Framework for the Attributes of PMU Data Quality and a Methodology for Examining Data Quality Impacts to Synchrophasor Applications,” U.S. Department of Energy, Tech. Rep. 3, 2017.

[20] F. Oest, M. Radtke, M. Blank-Babazadeh, S. Holly, and S. Lehnhoff, “Evaluation of communication infrastructures for distributed optimization of virtual power plant schedules,” *Energies*, vol. 14, no. 5, 2021.

[21] D. Douglass and et al., “Ieee standard for calculating the current-temperature relationship of bare overhead conductors (ieee std 738™-2012),” IEEE Power and Energy Society, Tech. Rep., 2013.

[22] K. Dehghanpour, Z. Wang, J. Wang, Y. Yuan, and F. Bu, “A survey on state estimation techniques and challenges in smart distribution systems,” *IEEE Transactions on Smart Grid*, vol. 10, no. 2, pp. 2312–2322, 2019.

[23] C. Chandraratne, T. Logenthiran, R. T. Naayagi, and W. L. Woo, “Overview of adaptive protection system for modern power systems,” in *2018 IEEE Innovative Smart Grid Technologies - Asia (ISGT Asia)*, 2018, pp. 1239–1244.

[24] K. E. Antoniadou-Plytaria, I. N. Kouveliotis-Lysikatos, P. S. Georgilakis, and N. D. Hatziargyriou, “Distributed and decentralized voltage control of smart distribution networks: Models, methods, and future research,” *IEEE Transactions on Smart Grid*, vol. 8, no. 6, pp. 2999–3008, 2017.

[25] G. C. Kryonidis, E. O. Kontis, A. I. Chrysochos, C. S. Demoulias, and G. K. Papagiannis, “A coordinated droop control strategy for overvoltage mitigation in active distribution networks,” *IEEE Transactions on Smart Grid*, vol. 9, no. 5, pp. 5260–5270, 2018.

 ISBN: 978-1-61208-967-6