MULTI-LAYERED SIMULATION PLATFORM FOR FUTURE WORLDS DISTRIBUTION SYSTEM SCENARIOS

***Han Xu*[1]*, Kyle Jennett*[1]*, Allan Downie*[1]*, Federico Coffele*[1]**

[1]*Power Networks Demonstration Centre, University of Strathclyde, Glasgow, UK*
email address: {han.xu; kyle.jennett; allan.s.downie; federico.coffele} @strath.ac.uk

Keywords: DISTRIBUTION SYSTEM OPERATOR, DISTRIBUTION FLEXIBILITY RESOURCES, FLEXIBILITY PROCUREMENT AND DISPATCH, DSO SCENARIOS

Abstract

With the current Distribution System Operator (DSO) transition, DSOs are looking for novel cost-effective solutions to manage distribution networks. To avoid operational failures these solutions must be evaluated in a realistic end to end test environment prior to deployment. To meet this requirement PNDC is presently developing a platform that integrates solutions for power system analysis, market modelling, and real-time simulation. This multi-layered simulation platform will be used to investigate the impact of different DSO operational scenarios (e.g. flexibility procurement, communication interfaces, and vendor provided solutions). To develop the case study presented in this paper aspects of the Open Networks 'Future Worlds' were utilised. The 'Future Worlds' were developed by the UK Energy Networks Association and represent potential scenarios for the UK future electricity industry structure.

This paper presents a case study using the PNDC platform. This case study reflects 'Future World' A and simulates an enforced power exchange profile at a grid supply point. In the case study a controllable demand is simulated in real-time and interfaces with power flow analysis and an optimal flexibility procurement algorithm. The case study demonstrates the capability of the multi-layered platform to manage network limitations by procuring flexibility services within a simulated distribution network.

1 Introduction

Distributed Network Operators (DNOs) in the UK are in a transition to DSOs. Within this new role they require cost-effective solutions for more active and automated network operation to compensate for changes in the energy sector e.g. decarbonisation of energy production, electrification of heat and transport, and a growing number of Distributed Energy Resources (DERs). Instead of reinforcing the network, DSOs can utilise DERs to actively manage their networks.

The ENA Open Networks project [1] investigates ways to facilitate the DSO transition. Five 'Future Worlds' [2] representing five potential scenarios for the UK electricity industry were proposed by the Open Networks project. In these 'Future Worlds', roles, responsibilities and interactions between actors (e.g. Electricity System Operator (ESO), DSO, aggregators, etc.) are defined. The role of a DSO in many of the 'Future Worlds' is to act as a neutral market facilitator and optimally manage distribution network congestion in a coordinated manner with the ESO. Therefore, DSOs need novel solutions to develop the capability to actively manage the distribution network.

There are already existing solutions developed for power system simulation [3-4], market modelling [5-6], and real time simulation [7]. However, there is not an integrated platform that is able to bring all of these capabilities together into a single unified whole for DSO scenario testing. At the Power Network Demonstration Centre (PNDC), we are developing a multi-layered simulation platform that integrates all of these components.

The multi-layered platform that is being developed at PNDC can be used for testing and demonstrating DSO required capabilities (such as communication interfaces, distributed flexibility procurement process, etc.), in order to de-risk future DSO operation. The scenarios that are being tested on this platform are based on components of the 'Future Worlds' associated with network operation. By testing these 'Future World' scenarios, the impact and operational effectiveness of flexibility resources on network operation can be assessed during testing and therefore de-risk the current DNO to DSO transition.

This paper presents the multi-layered simulation platform and demonstrates how the platform acts as an integrated whole. Section 2 explains how the different layers of the platform are configured and interconnected. Section 3 gives a case study of the platform, together with a detailed explanation of the implementation and data exchange processes between the layers. This case study illustrates how the platform operates for flexibility procurement. Case study results are presented in Section 4. The conclusions and future work are discussed in Section 5.

2. Multi-Layered Simulation Platform

The aim of building the multi-layered simulation platform is to enable component testing in an end to end system. This end to end system integrates simulation of power flow, market modelling, and interfaces to hardware and software vendor solutions. The multi-layered simulation platform is shown in Fig 1. Within this platform there are three distinct layers: software, data and real-time. The layers interact and exchange data corresponding to their functions.

The **real-time layer** simulates real-time network operation. It has been developed using the Real Time Digital Simulator (RTDS) [7]. It simulates real-time electrical power systems with a 50µs time step. At this layer, physical devices are connected via Power Hardware In the Loop (PHIL) to test the effectiveness of physical devices during real-time network operation. The physical devices can be a variety of different components, for example: network monitoring devices, communication systems, Electrical Vehicles (EVs), load controllers, Intelligent Electronic Devices (IEDs), etc.

The **data layer** is used to model and analyse power systems. It runs power flow analysis of the simulated network. At this layer control and monitoring platforms, for example Supervisory Control and Data Acquisition (SCADA) systems or Active Network Management (ANM) systems, are connected using commercial communication interfaces. The data layer sends load flow modelling results to both the **software** and **real-time layers**.

The **software layer** represents: an electricity market, actors within the market, or a real world vendor provided solution for actors within the market. This layer can demonstrate the effectiveness of market solutions or test a concept type market product before network deployment. The **software layer** sends market decision information to the **data layer**.

Equipment can be tested in isolation (e.g. an EV connected to a charger), however, this type of test does not assess how the equipment will interact as part of a larger system. The value of the multi-layered platform being developed at PNDC is that it allows equipment to be tested as part of a simulated end to end system to evaluate full system impact. All of the layers discussed in this section need to be simulated to realistically test a component connected to a single layer. This can be explained in the context of a smart EV charger testing example. An EV charger can be connected to the **real-time layer** using PHIL via the RTDS. The EV charger then becomes part of a simulated distributed network modelled in the **data layer**, via a custom built Python interface. This simulated network operation is impacted by market events modelled in the **software layer**. Interaction between the layers is bi-directional. For example, the market (**software layer**) may instruct an EV charger turn down (in the **real-time layer**). However, the EV charger state will also impact the markets to enact turn down commands. By testing the EV charger in this way the scope of the testing expands beyond simple EV charger operation to EV charger interaction with the larger distribution network and electricity market.

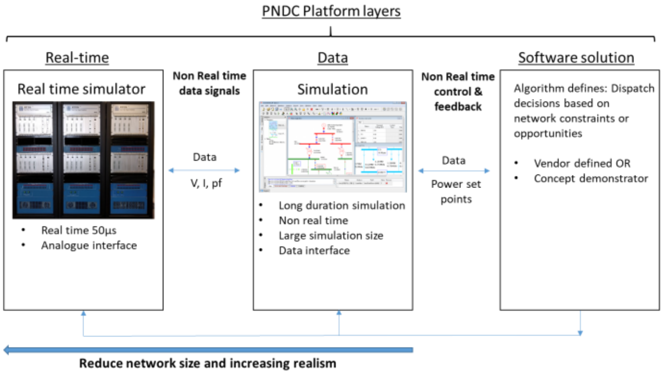


Figure 1 The multi-layered simulation platform

3. Case Study

Different DSO scenarios and configurations can be implemented in the multi-layered simulation platform. For example, finding cost-effective solutions is one of the challenges associated with the DSO transition, especially when facing choices between traditional network reinforcement and market-based procurement of distributed flexibility resources. By procuring flexibility services, network operators can benefit from deferring the high costs related to network reinforcement. The platform allows flexibility service scenarios to be simulated and evaluated.

A case study was tested on the platform (and presented in this section) to understand the stacked impact of network operation decision making. This example is designed to answer the following question, if a DNO decides to procure a flexibility service instead of upgrading the network infrastructure, what is the impact of flexibility resource utilisation on the network (e.g. voltage and capacity) over an extended period of network operation? For how long can the voltage be kept within acceptable limits? This case study reflects the Open Network ‘Future Worlds’ and procurement of distributed flexibility resource was tested on the platform.

Fig 2 shows the implementation of the platform layers and associated data exchange in the case study. The data layer uses Python pandapower [8]. The data layer is used to model a test network and runs load flow analysis every 10 seconds. The load flow results are sent to the real-time and software layers. The real-time layer (RTDS) acts as a sub-section of the test network modelled in the data layer. Controllable flexibility resources are simulated in the real-time layer so associated real-time changes in power can be implemented and monitored. The flexibility resources modelled in RTDS have their own ramping rates to simulate the real world limitations of DERs. They also have pre-defined load profiles that can be curtailed. The real-time layer interfaces with the data layer via the python interface.

If the test network is constrained, which is detected by the power flow analysis in the data layer, the data layer will inform the software layer. The software layer optimises the network operation when the network is constrained. In a constrained situation available flexibility resources may therefore be controlled to turn up/down. The optimisation is carried out by Python pandapower. The control set-points of the available flexibility resources, calculated by the optimisation algorithm, are then sent to the real-time layer. The flexibility resources in the real-time layer change their outputs based on the received set points and their inherent ramping limitations. The altered outputs of the flexibility resources are captured and sent back to the data layer. The power flow analysis in the data layer re-calculates and checks if the test network is still constrained.

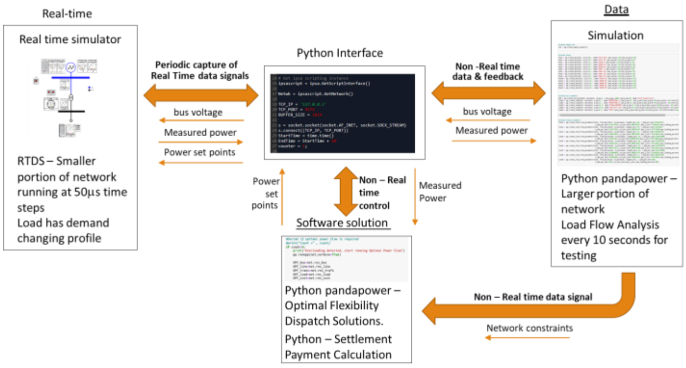


Figure 2 Implementation of the multi-layered platform

The flexibility resources instantly react to the set point, however, the rate of response is controlled due to the imposed ramp limits. This means the network will remain constrained while the flexibility resources ramp to the requested constrained setpoint. While the network is still constrained, the software layer will continue to send the setpoint command to the flexibility resources. If the constraint is relieved, the set point command will stop, and the flexibility resources will revert back to follow the pre-defined load profile. It is assumed that the constraints can always be solved with available flexibility resources. Extreme network situations, where this assumption is no longer valid, are out of the scope of this paper. However, this will be considered in the future work.

Each flexibility resource has predetermined bids and offers. Based on the procurement request and captured data from the flexibility resources, settlement payments are calculated.

4. Case Study Results

In this case study, the scenario tested on the multi-layered platform reflects aspects of the Open Networks 'Future World A' [2]. The World A has a predefined power exchange profile at the GSP, i.e. the link between transmission and distribution network. The power exchange profile limits the export and import power at the GSP level. In World A, the DSO procures flexibility resources connected on the distribution network to actively manage distribution network constraints. This case study presented in the paper reflects these aspects of World A.

The test network implemented in this case study is illustrated in Fig 3. In this test network, there is one controllable demand, which is 'Demand D2' on bus D. A predefined power exchange profile was enforced at the GSP, connected to bus A. There are no generators connected within the test network considered in this case study. Therefore, only import limitations from the transmission network are applicable (i.e. only demand within the test network is limited. Therefore, if a network limitation is breached, flexibility from the controllable demand will be procured to address the breach.

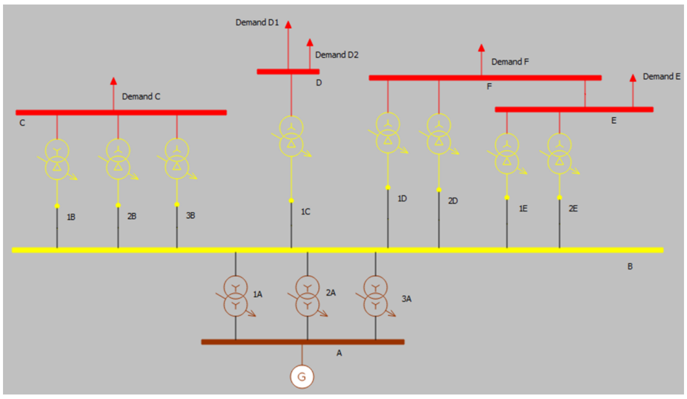


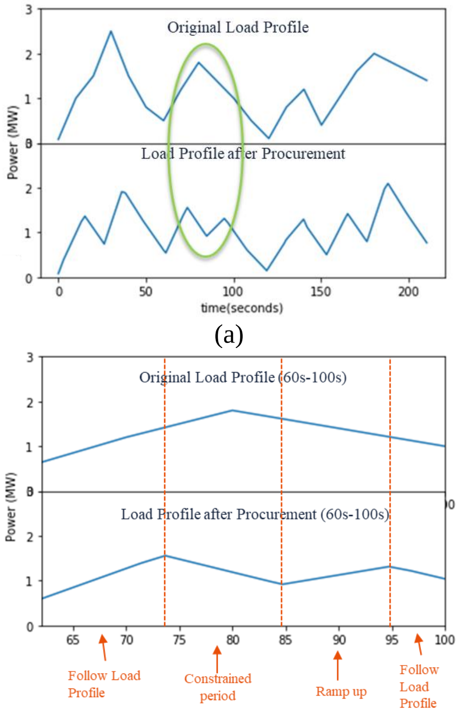
Figure 3 Case study test network

In the case study, the simulation ran for 200 seconds and the power flow analysis ran every 10 seconds. One power flow analysis will be referred to as one 'iteration' in the following explanation. As explained in section 3, if the load flow analysis detects network constraints, it invokes the optimal flexibility dispatching algorithm. The flexibility dispatching algorithm sends the control set point in iteration one and ten seconds later in iteration two the flexibility dispatching algorithm measures actual demand. The load flow analysis is then re-computed to check if the constraint still exists.

Fig 4 (a) shows the load profile of the controllable demand. The top graph shows the load profile if no flexibility procurement is implemented; the bottom graph shows the load profile if flexibility procurement is requested. Fig 4 (b) presents the load profile of the controllable demand for the period between 60 to 100 seconds. From Fig 4 it can be observed that the controllable demand was requested to decrease its consumption during specific times within the simulation (these corresponds to periods when the GSP limit is exceeded). When there was no control/procurement signal, the controllable demand followed the pre-set ramping rates and returned to the pre-set load profile.

Settlement payments were calculated after each iteration. If there was no procurement request at one iteration, there would be no settlement payment. Bids and offers of the controllable demand are given in Fig 5. Please note the bids and offers are only illustrative values and do not correspond to actual market rates.

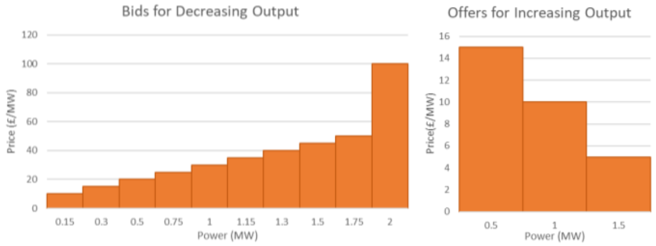
The calculated settlement payments are summarised in Table 1 for each iteration. All the settlement payments were paid for decreasing consumption of the controllable load (i.e. a turn-down service). Iterations where there is no payment correspond to instances when no constraint was required.



(a)

(b)

Figure 4 (a) Controllable demand load profile (b) Controllable demand load profile (60-100s)



(a) (b)

Figure 5 (a) Bids for decreasing output, (b) Offers for increasing output, of the controllable demand.

Table 1 Settlement Payment at each iteration

Iteration	1	2	3	4	5	6
Settlement Payment	£ -	£4.3	£ -	£8.1	£8.7	£ -
Iteration	7	8	9	10	11	12
Settlement Payment	£4.0	£ -	£8.3	£ -	£ -	£ -
Iteration	13	14	15	16	17	18
Settlement Payment	£10.0	£ -	£4.1	£ -	£5.5	£9.2

In the case study presented, the multi-layered platform has been used in a virtual environment to demonstrate procurement of flexibility to operate a network with GSP capacity limits. The capability of the platform to compute settlement payment if there is a procurement request has also been demonstrated. The settlement payment is calculated based on the real-time captured data from the flexibility resource, rather than control set points. This is reflective of how the market clears in the real world after flexibility has been procured and utilised. i.e. settlement is billed on actual response rather than procurement request.

5 Conclusions and Future Work

This paper presents a multi-layered simulation platform that is being developed at PNDC. The multi-layered platform integrates power system simulation, market modelling and real time simulation. A case study, that reflects aspects of the Open Network 'Future World' A, was tested on the platform. The case study results demonstrate the capability of the multi-layered platform to procure flexibility resources to manage network constraints. In the next stage the multi-layered platform will be extended at PNDC, to connect real world flexibility resources (e.g. EV and heat pumps etc.), i.e. PHIL. Behaviour of real world devices can be observed within real time network operation in PHIL. In addition, more DSO scenarios based on real world network data and events will be tested on the platform, in order to generate additional useful learnings for DSOs.

6 References

[1] 'Overview of the Open Networks project', https://www.energynetworks.org/electricity/futures/open-networks-project/open-networks-project-overview/, accessed 02 March 2020

[2] ENA, 'Open Networks Future Worlds', https://www.energynetworks.org/assets/files/14969_ENA_FutureWorlds_AW06_INT.pdf, accessed 03 March 2020

[3] Milano F.: 'Power system modelling and scripting' (Springer Science & Business Media; 2010)

[4] 'IPSA Power | Software for Power Systems', https://www.ipsa-power.com/, accessed 03 March 2020

[5] Geth, F., D'hulst, R. and Van Hertem, D.: 'Convex power flow models for scalable electricity market modelling', CIRED-Open Access Proceedings Journal, 2017, pp.989-993.

[6] Babic, J. and Podobnik, V. : 'A review of agent-based modelling of electricity markets in future energy eco-systems'. IEEE International Multidisciplinary Conference on Computer and Energy Science (SpliTech), July 2016.

[7] 'RTDS Technologies', https://www.rtds.com/, accessed 03 March 2020

[8] Thurner, L., Scheidler, A., Schäfer, F., et al, 'pandapower - an Open Source Python Tool for Convenient Modeling, Analysis and Optimization of Electric Power Systems', IEEE Transactions on Power Systems, 2018, 33, (6), pp. 6510-6521.