Fast parallel quasi-static time series simulator for active distribution grid operation with pandapower

Zhenqi Wang[12], Zheng Liu[1], Markus Kraiczy[2], Nils Bornhorst[1], Sebastian Wende-von Berg[12], Martin Braun[12]*

[1]*Department of Energy Management and Power System Operation, University of Kassel, 34121 Kassel, Germany*
[2] *Grid Planning and Grid Operation Division, Fraunhofer IEE, 34119 Kassel, Germany*
**zhenqi.wang@uni-kassel.de*

Keywords: Volt-VAR control, Parallel computing, On-load Tap Changer, Quasi-static simulation, Probabilistic grid simulation

Abstract

The increasing penetration from intermittent renewable distributed energy resources in distribution grid brings along challenges in grid operation and planning. To evaluate the impact on the grid voltage profile, grid losses, and discrete actions from assets (e.g. transformer tap changes), quasi-static simulation is an appropriate method. Quasi-static time series and Monte-Carlo simulation requires a tremendous number of power flow calculations (PFCs), which can be significantly accelerated with a parallel High-Performance Computing (HPC)-PFC solver. In this paper, we propose a HPC-PFC-solver-based grid simulation (parallel simulation) approach for a multi-core CPU platform as well as a greedy method, which can prevent the errors caused by simultaneous parallel simulation. The performance of the proposed approach and the comparison is demonstrated with two use cases.

1. Introduction

An important task in the distribution grid operation is to maintain the secure operation e.g., keep voltage limits considering the intermittency, volatility, and uncertainty from distributed energy resources (DERs) such as wind and PV in distribution grids. In the grid operation, the impact of the DER Q control mode on the grid operation can be considered in two phases:

- Planning of the operation phase: In this phase, the operational modes of DERs should be studied and simulated with the consideration of further voltage-controlled assets such as On-load tap-changers (OLTCs) so that the basic operation modes can be optimized. In this phase, long time series simulations need to be carried out.
- Real-time operation phase: In this phase the short-term forecasting of DERs and load are available for the grid operator, which, however, contains a certain degree of uncertainty. This needs to be properly considered to maintain the robustness of the operation. The uncertainty can be evaluated with Monte-Carlo probabilistic grid simulation.

As suggested in [1], the quasi-static simulation (QSS) is a suitable method for voltage stability evaluation in stationary operation (contingency-free). It is hence a suitable simulation method for the two phases mentioned above. QSS, in contrast to the RMS dynamic simulation, does not model the correlation between each time step. The DERs are simulated with external loops over multiple power flow calculations (PFCs) at each time step. The method is computational-intensive when a large number of time steps needs to be evaluated. Thus, in this paper, we proposed a method for the multi-core CPU platform to accelerate the QSS with a parallel High-Performance Computing (HPC)-PFC solver which is proposed in our previous work [2]. As already discussed in [3], performing quasi-static time series simulations (QSTSSs) in parallel for multiple time windows of a smaller number of time steps each might yield different results than a conventional serial time series simulation. The deviation from the results of the standard time series simulation will further lead to errors of discrete actions such as those of voltage controlled OLTC, which can, e.g., lead to a significant increase of tap changes. To tackle this problem, a greedy method is proposed in this paper.

To demonstrate the performance of the parallelization of QSTSS and probabilistic QSS (PQSS) methods, they are applied to case studies with open-source dataset and real MV grid data from the German DSO LEW Verteilnetz GmbH.

2. Methodology

2.1. *Serial QSS with pandapower*

QSS uses the external loop of a control logic to evaluate the voltage stability of the MV grid. In pandapower [4], the users can implement controllers for time series simulations to integrate different types of modelling and controlling effects with different priorities (e.g., fast Q compensation from DER and delayed action of OLTC).

A multi-level structure is applied with which the convergence of a controller is evaluated according to the level until all levels have converged. Convergence is defined to be achieved when the change of all variables (e.g., P, Q) stays below a predefined threshold, e.g., 1e-4 MW, and no further discrete actions are required. The serial QSS from pandapower is used as the benchmark for comparison of the performance and verification of the results.

2.2. *Modelling of DER*

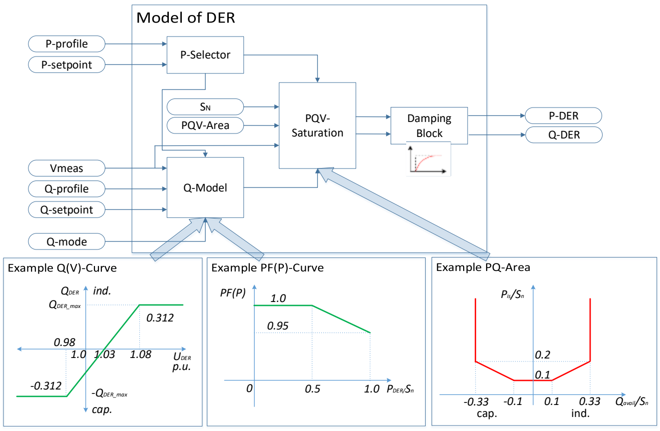

Figure 1 Modelling approach of DERs

The modelling of German grid code [5] compliant DERs comprising different local Q control strategies as well as remote setpoints from a centralized controller is shown schematically in Figure 1.

The controller takes P and Q profiles, voltage measurements at the grid connection point, and remote P and Q local control mode setpoints as inputs. The modelling is realized object-oriented in Python, which makes it flexible, universal, and easily extensible.

The P-selector block makes a DER curtailable with P setpoints, while the Q-Model block models the multiple DER Q behaviours. These include:

- Q(V)-droop control: Q of the DER is calculated based on a Q(V) curve with local V measurements which tends to stabilize the local voltage and avoid extreme voltage values.
- Power Factor of P (PF(P)): Q is calculated according to the power factor curve based on P which tries to avoid the voltage rise caused by local P feed-in.

The Q(V) curve and PF(P) curve can be user-defined. The physical and grid code limitation is modelled in the PQV-saturation block. The PQV-range limitation according to the gird code, e.g., in [5] for the MV DER, can offer Q only in the designated range, which is imposed with an additional voltage limitation. Secondly, P and Q are limited to the inverter size (nominal apparent power, S_N), which means in Q priority-mode, that the P feed-in can be curtailed to fulfill the Q requirements if exceeding the S_N.

The P and Q output from the DER needs to be damped (analog to PT1 in dynamic simulation) with a constant coefficient, e.g., 2, to avoid overcontrol, thus improving convergence.

2.3. *Parallel QSS with HPC PFC Solver*

A large number of PFCs without topological change can be efficiently solved simultaneously in parallel with a HPC-PFC solver [2]. As a performance reference, 10,000 PFCs for a typical MV grid with ~200 buses without topology change can be performed on a CPU (i7-9700k, 8 Cores) within 0.3s.

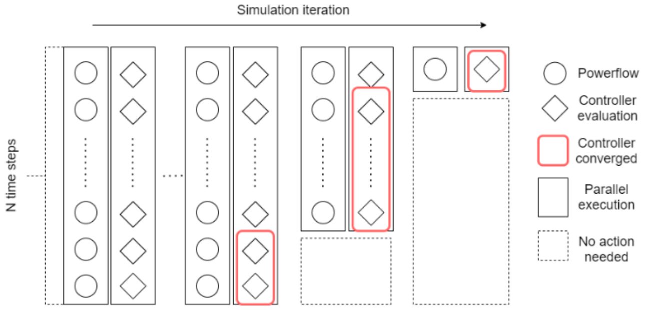

Figure 2 Principle of an example with parallel QSTSS

The principle of the proposed parallel grid simulator is shown in Figure 2. Let us consider an example of QSTSS with N time steps. In each iteration of the QSTSS, a PFC and a controller step with the result of the PFC are performed. In the serial QSTSS, the first time step is simulated until controller convergence is achieved before proceeding to the second time step and so on. In the parallel QSTSS, all time steps are simulated simultaneously in parallel. In this way, on multi-core CPU platform, a strong acceleration is achieved in combination with the HPC-PFC solver and a vectorized controller implementation with numpy [6]. After multiple iterations, when all controllers in some of the time steps have converged, the results of these time steps are exported and no more actions are required for these steps. The simulation for N time steps will stop until all controllers in the last time step (steps) have converged. PQSS can be parallelized in the same manner with the Monte-Carlo sampling.

2.4. *OLTC in QSTSS with greedy method*

In typical MV grids, an OLTC is often available at the HV-MV transformer, so that the grid operator can adjust the local grid voltage with an automatic voltage controller. It is important to coordinate the setup of the OLTC to the DER local voltage control [7], which needs to be properly simulated in the planning phase.

The parallel approach yields errors in the discrete variables such as tap positions for the following reason. In serial simulation, the state of t-1 is used as the initial state for time step t. In a parallel simulation, the states of t-1 are unknown. Thus, only a flat initialization with tap position 0 or with other predefined states can be used. A greedy method is proposed to avoid excessive tap changing of voltage-controlled OLTC caused by the parallel simulation: For every tap change from one time step to the next, an additional parallel simulation is run where the tap position is initialized with the one potentially avoiding the tap change. With this initialization, the next iteration of parallel simulations will only yield a tap change at that time step if it is unavoidable. A no-go area will be updated to mark the necessity of each tap change.

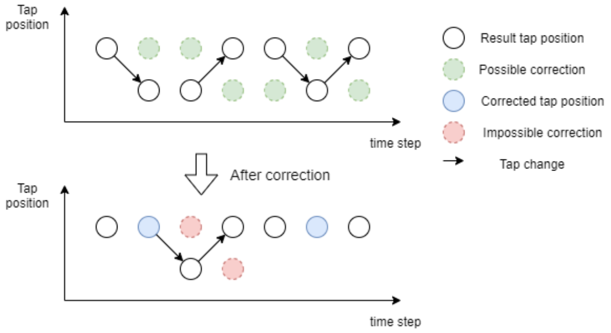

Figure 3 Example of the proposed greedy method

An example of how the abovementioned greedy method works is shown in Figure 3. With the tap position initialized at 0, after the initial parallel simulation, the resulting tap positions are depicted by white dots. In total, 4 tap changes can be observed. To verify the necessity of the tap changes, a second simulation is performed with new initial tap positions potentially avoiding unnecessary tap changes (green dots). Then, the unnecessary tap actions are identified and the tap positions corrected (blue dots), while the tap actions which are unavoidable are kept. The impossible corrections are depicted by red dots.

2.5. *Probabilistic Quasi-static simulation (PQSS)*

Since the performance bottleneck is properly handled with parallelization, Monte-Carlo probabilistic simulation can be performed in the real-time operational phase of, e.g., 8-h ahead basis. The simulation approach can be understood as running a PQSS for each time step. For each forecasted time step, the simulations are performed independently from each other. The robustness of the DER local control mode could be efficiently evaluated under uncertainties from the forecasting. Since there is no need to consider the continuity of tap positions from OLTC between the sampled profiles in the process, it is possible to consider the probability of tap changes during the simulation.

3. Case studies

Both case studies are performed on a computer with i7-9700k CPU, 64 GB DDR4 RAM and Python 3.8. In both simulations, the Q(V) curve and PF(P) characteristics are defined as in Figure 1.

3.1. *Case Study: QSTSS*

The typical MV grid data ("1-MV-urban--1-sw") with time series from the open dataset SimBench [8] is used for the result verification. The MV grid contains 133 DERs with a total installed capacity of 18.9 MW and peak P load of 21.5 MW. To simulate a future scenario, the time series of loads and DERs in 15 minutes resolution are scaled by a factor of 3. The two types of the DER control mode PF(P) and Q(V) are simulated with voltage-controlled OLTC. The MV side voltage with an allowed range of 1.02 p.u. to 1.05 p.u. is set for voltage control on OLTC.

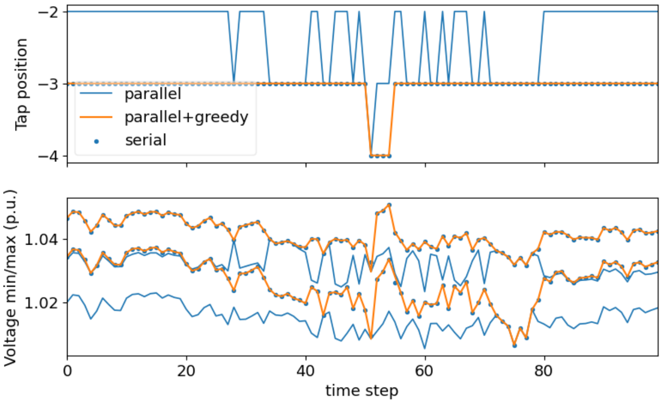

Figure 4 Example result of 100 time steps under Q(V) mode with the tap positions of trafo 1

Table 1 Result of QSTSS for 10,000 time steps

Case	Simulation methods	Computation time [s], (acceleration factor)	No. of time steps with tap position deviation	No. Of time steps with Vmin,max deviation
PF(P)	serial	3446.0 (-)	---	---
	parallel	27.9 (123x)	8,530	8,530
	parallel+greedy	108.8 (32x)	0	0
Q(V)	serial	4320.8 (-)	---	---
	parallel	35.4 (122x)	8,952	8,980
	parallel+greedy	130.9 (33x)	0	55

Figure 4 shows the results for the above-listed cases. It can be observed that the excessive tap changes caused by the parallelized simulation are significantly reduced with the proposed greedy method. The resulting tap positions of the greedy method are identical to those of the serial simulation, and only tiny voltage deviations (tolerance 1e-4 p.u., in some time steps only below 1e-2 p.u.) compared to the serial simulation can be observed.

In Table 1, the result of timing of the QSTSS is shown. The long computation time of the serial simulation can be significantly reduced using the parallel and the parallel+greedy approach. Due to the additional iterations performed by the proposed greedy method to identify unnecessary tap changes, the computation time of the parallel+greedy approach is slightly increased by a factor of 3x-4x compared to parallel only approach. The Q(V) case takes more time than the other case since the voltage-controlled controller needs more iterations to converge. The greedy method succeeds in correcting the tap position deviation between the serial approach and the parallel approaches in both, the PF(P) and the Q(V) cases. In the PF(P) case, where the Q of the DERs is relevant to the P availability, the voltage deviation between the parallel and the serial approaches can be completely avoided. In the Q(V) case, in very few time steps, the parallel+greedy approach converges to different but close and also valid results.

3.2. *Case Study: PQSS*

The real German 20 kV distribution network of the grid operator LEW Verteilnetz GmbH is used in the simulation which contains 475 buses, 112 loads (Pmax 12.19 MW), and 126 DERs (installed capacity 22.61 MW). The DERs are operated in Q(V) mode. To illustrate the concept of PQSS with the focus on the contribution to the voltage stability with the consideration of uncertainties from forecasting, the OLTC is not considered in the simulation.

Figure 5 shows the daily P-profiles of three exemplary DERs. Based on the profiles, 8h (from 07:00 to 15:00 in 15-min resolution, 28-60 in Figure 5) deterministic forecasting is used in the simulation. The probabilistic sampling with 1000 random samples at each time step is generated with the latin-hypercube method. The uncertainty is assumed as follows: P of DERs, P, Q of load forecasting, and the voltage of the external grid exhibit an equally distributed uncertainty of 5% and 1% of the deterministic values, respectively.

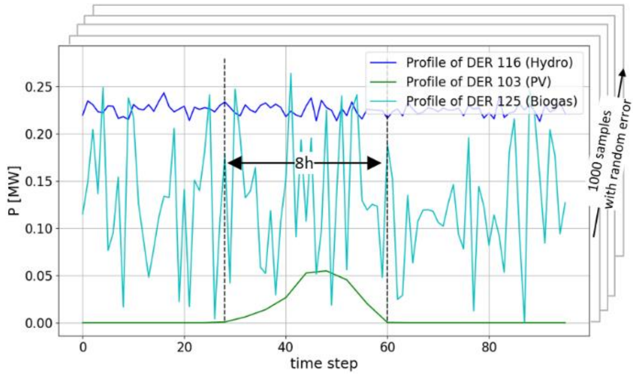

Figure 5 Example profile of DERs in the grid

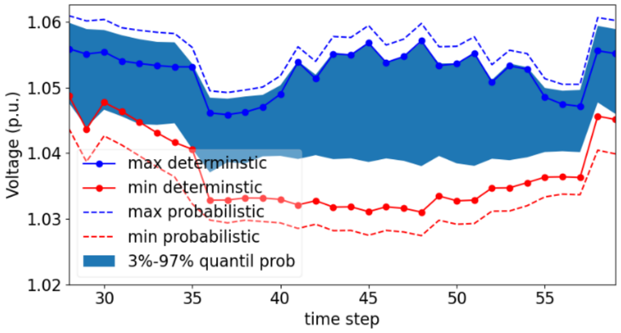

Figure 6 Voltage distribution for probabilistic and deterministic 8h time series simulation

Figure 6 shows the result of the 8h simulation: 3%-97% quantile of the voltage distribution obtained by the probabilistic simulation and the minimum and maximum voltage in both cases. Since the uncertainty from the forecasting is properly considered, the probabilistic approach can better identify the robustness via the quantile distribution and minimum and maximum voltages.

Table 2 Computation time required for different simulations and simulators

Case	Calculation steps	Simulation method	Computation time [s] (acceleration factor)
deterministic scenario for 8h	32 (4x8h)	serial	11.71 (-)
		parallel	0.71 (17x)
probabilistic scenario for 8h	32*1000	serial	8,065.5 (-)
		parallel	106.9 (75x)

Table 2 shows the computation time required for the considered simulations. The long computation time of the serial PQSS, which takes more than 2 hours, can be reduced to only 2 minutes. Comparing the resulted voltage from serial and parallel simulation, the differences can be neglected (under 1e-6 p.u.).

4. Conclusion

Using a parallelized simulation approach with our lately developed HPC-PFC solver, the time-consuming QSTSS and PQSS can be accelerated in the order of 30x-100x for the given example on a multi-core CPU platform. The proposed greedy method successfully avoids tap changes errors caused by the parallel implementation, while only slight voltage deviations in results from the parallel simulation are observed comparing to the serial simulation. Overall, our parallelization method shows a promising application in the planning phase with QSTSS and the real-time phase with PQSS.

5. Acknowledgements

The work was supported by the European Union's Horizon 2020 research and innovation programme within the project EU-SysFlex under grant agreement No 773505. Part of the work was funded within the project RPC2 (FKZ:0350003A) of the research initiative “future oriented electricity grids” funded by the Federal Ministry of Economics and Energy according to a decision of the German Federal Parliament. The authors are fully responsible for the content of this paper.

6. References

[1] IEEE Standards Coordinating Committee 21 (SCC21): ‘IEEE guide for conducting distribution impact studies for distributed resource interconnection’

[2] Wang, Z., Wende-von Berg, S., Braun, M.: ‘Fast Parallel Newton-Raphson Power Flow Solver for Large Number of System Calculations with CPU and GPU’ (17.02.2021)

[3] Azzolini, J.A., Reno, M.J., Montenegro, D.: ‘Implementation of Temporal Parallelization for Rapid Quasi-Static Time-Series (QSTS) Simulations’, pp. 2942–2949

[4] Thurner, L., Scheidler, A., et al.: ‘Pandapower—An Open-Source Python Tool for Convenient Modeling, Analysis, and Optimization of Electric Power Systems’, IEEE Trans. Power Syst., 2018, 33, (6), pp. 6510–6521

[5] VDE-AR-N 4110 (2018) Technical connection rules for the connection to and parallel operation with medium-voltage distribution networks

[6] Harris, C.R., Millman, K.J., van der Walt, S.J., et al.: ‘Array programming with NumPy’, Nature, 2020, pp. 357–362

[7] Kraiczy, M., Stetz, T., Braun, M.: ‘Parallel Operation of Transformers with on Load Tap Changer and Photovoltaic Systems with Reactive Power Control’, IEEE Trans. Smart Grid, 2018, 9, (6), pp. 6419–6428

[8] Meinecke, S., Sarajlić, D., Drauz, S.R., et al.: ‘SimBench—A Benchmark Dataset of Electric Power Systems to Compare Innovative Solutions Based on Power Flow Analysis’, Energies, 2020, 13, (12), p. 3290