A DYNAMIC PROGRAMMING BASED APPROACH TO DAY-AHEAD OPERATIONAL COST REDUCTION FOR DSOs

Bhargav Prasanna SWAMINATHAN
University of Grenoble-Alps – France
bhargav-prasanna.swaminathan@g2elab.grenoble-inp.fr

Vincent DEBUSSCHERE
University of Grenoble-Alps – France
vincent.debusschere@g2elab.grenoble-inp.fr

Raphaël CAIRE
University of Grenoble-Alps – France
raphael.caire@g2elab.grenoble-inp.fr

ABSTRACT

A massive integration of Distributed Renewable Energy Sources (DRES) has the potential to wreak havoc on electrical distribution networks, if they are operated in the current manner. Optimization of such networks is therefore required - both in operational planning, and planning - in order to ensure a smoother transition from distribution networks as they are today to the ones of tomorrow. This work proposes a day-ahead economic optimization algorithm that provides a set of Distribution System Operator (DSO) actions on the network resulting in the lowest overall cost of operation for DSOs. The results obtained when the algorithm is applied to a test network in the University of Grenoble-Alps are also presented in this paper.

INTRODUCTION

The operational planning domain for distribution networks is generally regarded as a domain that extends anywhere from 3 hours ahead to a few days ahead of actual network conditions. During this time, drastic changes in the network, such as reinforcement of lines isn't possible. The types of "flexibilities" that DSOs can use are therefore mostly limited to the ones owned by them, like On-Load Tap Changers (OLTC) and reconfiguration, or to the ones they can be provided with by third parties, like load reduction and reactive power control of DRES. The utilization of each of these flexibilities entails a cost, as do congestions and abnormal conditions in the network.

Given the technical constraints for the utilization of each of the flexibilities, and the technical constraints of the network to be optimized, this paper presents the first ideas for a day-ahead algorithm that optimizes MV distribution networks for the least overall cost of operation. The result is a set of DSO actions for each time period during the day for optimization. In order to achieve this, the algorithm also uses day-ahead forecasts of loads and DRES for each node in the network. Most of the work done in the domain concentrates on technical optimization of a "snapshot" of the network [1]. This work considers the multi-temporal nature of the optimization, and also of the flexibilities.

In this paper, the working of the algorithm is first presented, followed by a description of the test network, the models used for load and DRES forecasts, finally ending with the results obtained, and conclusions.

THE ALGORITHM

The developed algorithm operates in two modes. The first mode is the reactive mode, in which the algorithm reacts to network congestions and abnormal conditions as they occur, and only for the particular time period they occur in. The second mode is an anticipative mode that not only tries to absolve the network of congestions and abnormal conditions for the current time period, but also tries to ensure that its actions result in economic optimality for the time periods that remain. The choice between the two operating modes is made based on the nature and type of problems encountered in the network.

In this work, the choice is made based on where the problems occur. If the problems occur in only one feeder of the MV network to be optimized, the reactive mode of operation is used. The reactive mode can therefore be called *local optimization*. If problems occur across feeders, the anticipative mode is used, and it can therefore be called *global optimization*.

For radial distribution networks, among the flexibilities listed before, the only flexibility that can have a global effect on the network is reconfiguration. The other flexibilities all have only a local effect in the sense that their utilization does not change conditions in other feeders. Hence, the *local optimization* mode makes use of load reduction, OLTCs, and reactive power compensation, while the *global optimization* mode makes use of all that, plus reconfiguration. A schematic representation of the algorithm is provided in Figure 1.

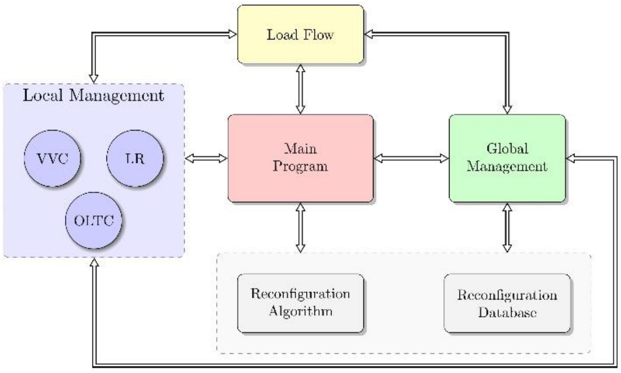

Figure 1- Schematic Representation of Algorithm Components

Reconfiguration Database

Given a forecast for loads and DRES, the reconfiguration database stores 24 optimal network configurations, one for each time period. The configurations are optimal in the sense that they guarantee technical optimality for the remainder of the time periods under the constraint that no further reconfiguration actions are to be performed. Obviously, reconfiguring networks for every time period could be the most optimal solution.

However, this is practically not possible. Hence, based on studies with different inputs (maximum load, minimum load, average load from remaining period, maximum load – minimum DRES from remaining period, and minimum load – maximum DRES from remaining period among others), the configuration closest to hourly reconfigurations in terms of technical optimality is chosen. This database is therefore a lookup table for the algorithm. The reconfiguration routine is based on a modified version of the algorithm proposed in [3].

Load Flow

The load flow routine, developed in-house, is a Newton Raphson one. It is capable of computing flows in the presence of DRES.

Local Management

The local management routine operates in reactive mode. As explained earlier, it is launched whenever the problems in the network are found only in one feeder. Each time the routine is launched, for the particular time period, it makes use of the available local flexibilities in order to have the lowest cost possible for the given time period. The total cost for one time period can be expressed as $C_{tot} = C_L + C_{vio} + C_{flex}$, where C_L is the cost of electrical losses, C_{vio} is the total cost of violated constraints, and C_{flex} is the cost of utilized flexibilities. In this case, the control is transferred to the *global management* routine. If this is not the case, the control returns to the routine that called it.

Main Program

The main program takes care of loading data, presenting results and passing control between various blocks of the algorithm. When launched, it loads all the input data. Subsequently, it calls the reconfiguration routine in order to create a database. Then it runs load flows on each of the 24 data sets with the original network in order to assess its conditions and problems. Once done, it classifies the problems according to whether they occur in one feeder or not. If for each time period with problems, only one particular feeder is affected, then the entire system can be managed with *local management* alone.

Every time *local management* is launched and is successful, the control returns to the main program. In case problems across feeders during one time period, or if *local management* fails during one of its launches, the *global management* routine (whose working is explained next) is launched. In any case, at the end of the optimization, control is returned to the main program to display the results.

Global Management

As explained earlier, this routine optimizes the network not only during the time period for which it is called, but also for subsequent time periods. It takes into account the effect network operations at one time period have on subsequent time periods. Therefore, once launched, the routine takes control of the optimization for all the remaining time periods. It works in two parallel paths. In the first parallel path, a reconfiguration is first performed. In the second parallel path, the algorithm strives to solve the problems by applying the flexibilities in *local management*, but on a global scale. Once this is done, the routine proceeds to evaluate the problems in the network for subsequent time periods. If only *local* problems are found, it launches the *local management* routine for every such instance.

However, if during any subsequent instance, problems occurring across feeders is found, the *global management* routine calls another instance of itself - a child routine – to manage the problems. It is therefore recursive in nature. And this also gives rise to the dynamic nature of the algorithm, where the optimization is split into sub-problems, each solved by a separate instance of the routine. For this routine, the ultimate goal is to minimize the cumulative cost of operation. It can be given by:

$$\text{obj} = \min\left(\sum_{p=k}^{n}\left(C_L + C_{vio} + C_{flex}\right)\right)$$

Here, k is the period when it is launched, and n is the final time period. In our case $n = 24$. The operation of the algorithm thus generates a tree of possible solutions, with each node being an instance with network problems, and chooses the shortest (lowest cost) path to a leaf.

PREDIS TEST NETWORK

The PREDIS test network [2] is a reduced-scale network at Grenoble Institute of Technology, one of the institutes in the University of Grenoble-Alps. It is an 11 kV network with 14 nodes and 17 lines. A range of DRES are connected to the network at various nodes. The network is shown in Figure 2, with some technical details presented in Table 1.

The major issues in the network, which comprises an urban, semi-urban, and rural part are either that of under-voltages or over-currents.

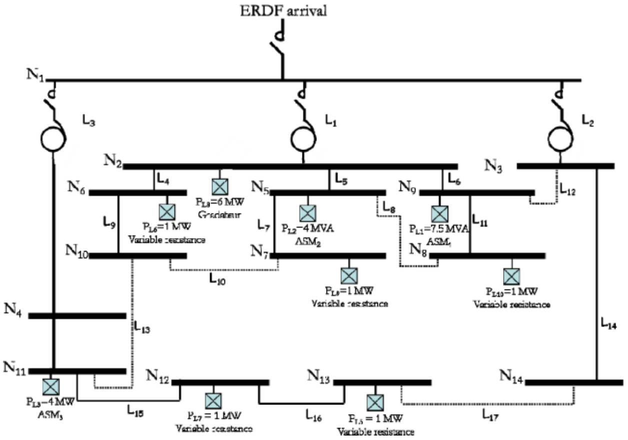

Figure 2 - The PREDIS Test Network

Parameter	Value
Voltage Level	$11\ kV$
Number of Buses	14
Number of Lines	17
Connected Load	$26.5\ MVA$
Connected DRES	$27\ MVA$

Table 1 - PREDIS Network - Technical Parameters

LOAD AND DRES FORECAST MODELS

In order to test the algorithm on the network, the connected load and DRES had to be varied throughout one day. In order to mimic a forecast, two load models and one DRES model are considered. Each model has 24 sets of values, each for one hour in the day. The models for loads were extracted from typical curves presented in [4] and [5].

In both models, four different types of loads are considered. Each node in the network has a specific share of these loads, and for each hour during the day, the loads are given weights, to be multiplied with their base value. The DRES model was constructed from typical production observations on the network. The cumulative curves for both the load models, and for the DRES model are shown in Figure 3.

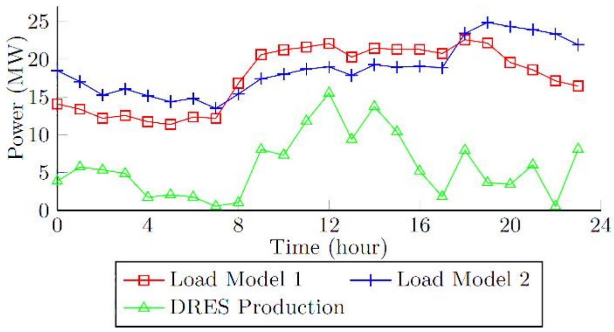

Figure 3 - Load and DRES Models

RESULTS

The results presented are divided into two subsections, depending on the load model used to obtain them. In each case, the reduction in the number of violations, a summary of network actions, and DSO expenditures are presented. In order to obtain these results, each network action, along with penalties for violations had to be defined. The parameters used for this work are summarized in Table 2.

Parameter	Value
Switching Operation	300 €
OLTC Operation	$20\ €/Tap\ Change$
Load Reduction	$6\ €/MWh$
Volt-VAr Control	$132.9\ €/MVArh$
Violations	$500\ €/Violation$

Table 2 - Valorization of Network Activities

For this work, the flexibility of load reduction at each node in the network was considered to be 10%, and each DRES is considered to be capable of, in terms of reactive power, absorbing and supplying amounts equal to 35% and 45% respectively, of its active power production. Electricity prices are based on the EPEX spot prices on October 1, 2014.

Load Model 1

The results obtained when the loads in the network vary according to load model 1 are presented here. Figure 4 shows a comparison of violations (voltage and current limits) that occur in the original and optimized networks.

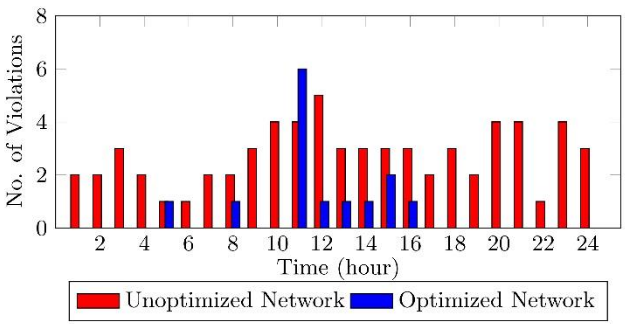

Figure 4 - Comparison of Violations - Load Model 1

The number of violations in the original network for the given DRES and load curves is 66. In the optimized network though, there are only 14 violations. Figure 5 shows the voltage profiles in the network. It can be seen that there is considerable improvement there as well. While in the original network, the minimum voltages are almost always below the permitted limit, in the optimized network, they are almost always above it. The minimum voltages in the original and optimized network are 0.862 pu and 0.908 pu respectively, with the next lowest voltages in the optimized network being 0.949 pu and 0.950 pu. A summary of network actions and other important results is presented in Table 3.

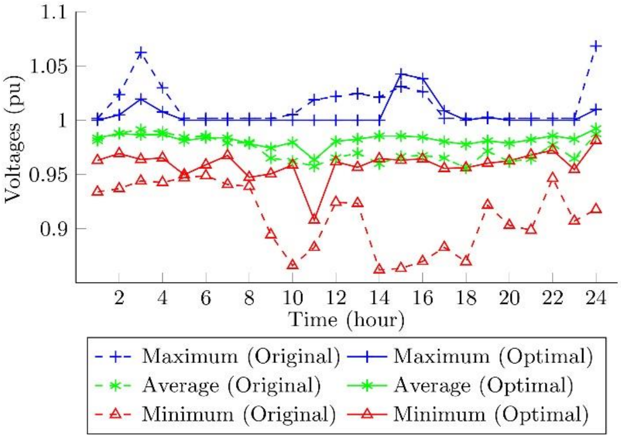

Figure 5 - Voltage Profiles in the Network - Load Model 1

Parameter	Value
DSO Expenditure (Original Network)	33 497 €
DSO Expenditure (Optimized Network)	9 349 €
Load Reduction Used	1 576.5 kWh
DRES Reactive Energy	822.7 $kVArh$
No. of Switching Operations (Reconfigurations)	12 (2)
OLTC Operations	3
Energy Losses (Original Network)	10 264 kWh
Energy Losses (Optimized Network)	3 691 kWh

Table 3 - Summary of Important Results - Load Model 1

The results show that a fair amount of all available flexibilities have been used by the algorithm to perform the optimization. The reduction in expenses for the DSO stands at 72% if the suggested network actions are adopted.

Load Model 2

The results obtained when the loads in the network vary according to load model 1 are presented here. Figure 6 shows a comparison of violations (voltage and current limits) that occur in the original and optimized networks.

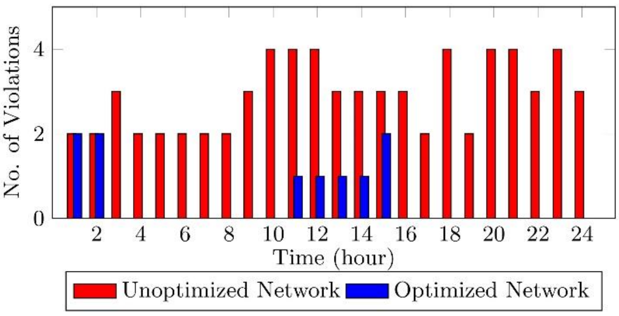

Figure 6 - Comparison of Violations - Load Model 2

The number of violations in the original network for the given DRES and load curves is 70, while the optimized network only has 10 violations. Once again, the voltage profiles improve considerably, as shown in Figure 7.

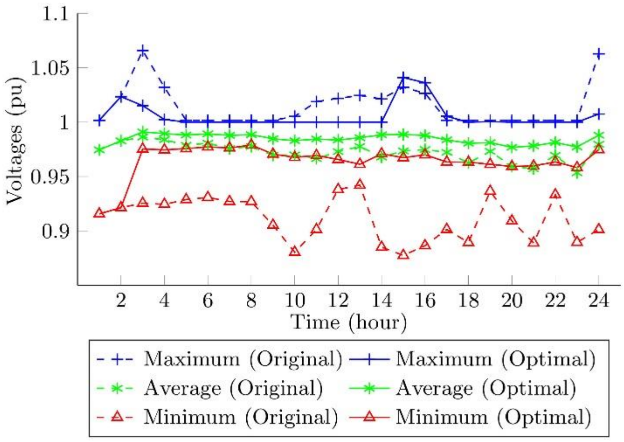

Figure 7 - Voltage Profiles in the Network - Load Model 2

In the original network, the voltages are once again almost always below the allowed limit, while in the case of the optimized network, they are almost always above the allowed limit. The minimum voltages in the original and optimized network are 0.878 pu and 0.915 pu respectively, with the next lowest voltages in the optimized network being 0.921 pu and 0.959 pu. A summary of network actions and other important results obtained with this load model is presented in Table 4.

Parameter	Value
DSO Expenditure (Original Network)	35 475 €
DSO Expenditure (Optimized Network)	6 566 €
Load Reduction Used	0 kWh
DRES Reactive Energy	0 $kVArh$
No. of Switching Operations (Reconfigurations)	10 (1)
OLTC Operations	0
Energy Losses (Original Network)	10 040 kWh
Energy Losses (Optimized Network)	3 382 kWh

Table 4 - Summary of Important Results - Load Model 2

Even though the cumulative load models look fairly similar, the results of the optimization are very different from one another. The algorithm achieves it goal in this case just with reconfiguration of the network. The reduction in expenditure for the DSO stands at 81.5%.

In both the cases, the cumulative energy losses show a similar, and very good improvement. The reduction in losses is around 62 % and 66 % respectively.

CONCLUSIONS

A day-ahead optimization algorithm based on dynamic programming logic was proposed. The goal of the algorithm is to ensure minimum operating expenditures for DSOs, based on valorisation of network actions for utilization of flexibilities and also penalties for network problems. The proposed algorithm was tested on a reduced-scale test network at the University of Grenoble-Alps under different conditions.

The results show a significant improvement in network operating conditions, and a very good overall decrease in DSO expenditures. In future, it is envisaged to develop this algorithm to handle more flexibilities, with a possible association of varied parameters for valorisation of network activities and flexibilities, and also with respect to penalties for network problems.

REFERENCES

[1] A. Zidan and E. F. El-Saadany, 2012, "Network reconfiguration in balanced distribution systems with variable load demand and variable renewable resources generation", *Proceedings IEEE PES General Meeting*, 1-8.

[2] L. Le-Thanh L et al., 2009, "Test Bench for Self-healing Functionalities applied on Distribution Network with Distributed Generators", *Proceedings of IEEE PowerTech Bucharest*, 1-6.

[3] D. Das, 2006, "A fuzzy multiobjective approach for network reconfiguration of distribution systems," *IEEE Trans. Power Del.*, vol. 21, no. 1, 202–209.

[4] E. López, H. Opazo, L. García, and P. Bastard, 2004, "Online reconfiguration considering variability demand: Applications to real networks", *IEEE Trans. Power Syst.*, vol. 19, no. 1, 549–553.

[5] S.-A. Yin and C.-N. Lu, 2009, "Distribution feeder scheduling considering variable load profile and outage costs", *IEEE Trans. Power Syst.*, vol. 24, no. 2, 652–660.

[6] The evolvdso website. [Online]. Available: http://www.evolvdso.eu