HANDLING EXCEPTIONAL SITUATIONS IN A DISTRIBUTION NETWORK CONGESTION MANAGEMENT ALGORITHM

Anna KULMALA
Tampere University of Technology – Finland
anna.kulmala@tut.fi

Sami REPO
Tampere University of Technology - Finland
sami.repo@tut.fi

ABSTRACT

Congestion management is one of the key components in the transfer towards active distribution networks that enable cost-effective network interconnection of distributed generation and better utilization of network assets. Since the aim of congestion management algorithms is to guarantee acceptable network state in all loading and generation situations, they should be able to operate reasonably also in exceptional situations. This paper discusses the different exceptional situations that can be encountered during congestion management algorithm operation and presents the handling of them in one implemented algorithm.

INTRODUCTION

In weak distribution networks, the hosting capacity of the network for distributed generation (DG) is often limited by the voltage rise effect. Also thermal constraints of feeders or transformers can, in some cases, limit the hosting capacity. If the distribution network operational principles are not altered, the voltage quality and overloading problems can be solved by reinforcing the network. This can, however, be quite expensive. Another alternative is to utilize active network management (ANM) methods. They change the distribution network operational principles radically but lead, in many cases, to substantially lower distribution network total costs. The simplest ANM methods are based only on local measurements and for instance local reactive power control of DG units is already in real distribution network use. More advanced control methods require information on the state of the whole distribution network. They can increase the network hosting capacity more than the local methods and can optimize the operation of the whole network [1]. The focus of this paper is on the advanced ANM methods.

Research on ANM has been active in the past years and ANM methods of different complexity and data transfer needs have been proposed in publications. However, the practical implementation issues of the algorithms are only rarely discussed in the publications. Most publications concentrate on determining the control principles of the control algorithm and the operation of the algorithm is tested using only load flow calculations. Some methods have been further developed and tested using time domain simulations and real time simulations and a few real network demonstrations also exist. However, even these publications usually concentrate on determining the algorithm operation in normal operational conditions and describe the operation in exceptional conditions only briefly or not at all. For many studies this approach is adequate but when the methods are really implemented and applied to real distribution networks, their operation must be reliable and smart in all situations.

This paper describes the exceptional situations that need to be taken into account at the control algorithm implementation phase and presents the handling of exceptional situations in one distribution network congestion management algorithm. The paper will firstly introduce the implemented congestion management algorithm. Thereafter, exceptional situations that need to be taken into account in practical implementations are discussed and their handling in the implemented algorithm is presented in detail.

THE IMPLEMENTED CONGESTION MANAGEMENT ALGORITHM

The congestion management algorithm of this paper is based on [2] and has been further developed in IDE4L project [3]. The algorithm is implemented on a distributed automation architecture developed in the same project [4] and depicted in Figure 1. Control algorithms are executed in substation automation units (SAUs) in primary and secondary substations. The SAUs include control functions, a database for storing and exchanging information and interfaces to other SAUs and intelligent electrical devices (IEDs) such as smart meters, remote terminal units and primary controllers of controllable resources. The congestion management algorithm is one of the SAU functions. The algorithms located at the primary substation automation unit (PSAU) control the resources connected to the MV network and the algorithms located at the secondary substation automation units (SSAUs) take care of the LV network control.

The primary goal of the real time congestion management algorithm is to guarantee that the network state is always acceptable. The secondary goal is to use the network as cost-effectively as possible. The implemented algorithm utilizes sequential quadratic programming (SQP) algorithm to optimize the network state and operates through changing the set points of the primary controllers such as automatic voltage control (AVC) relays of the transformer on-load tap changers, real and reactive power controllers of DG units, reactive power controllers of reactive power compensators and real power controllers of controllable loads. The objective function is formulated to minimize network losses,

The research leading to these results has received funding from the European Union’s seventh framework program (IDE4L project FP7-SMARTCITIES-2013 under grant agreement 608860).

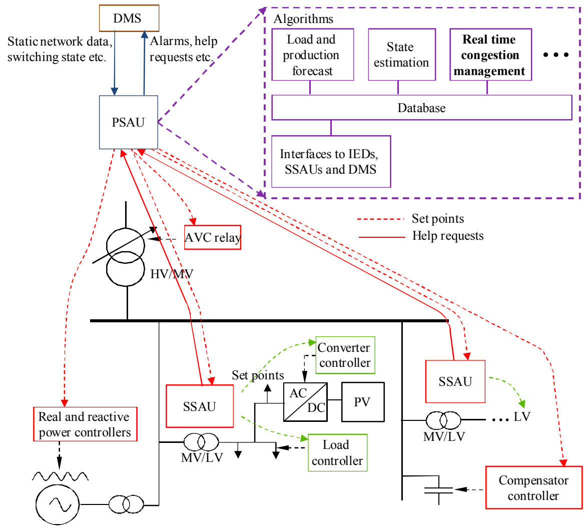

Figure 1. The automation architecture and information flows of the congestion management algorithm.

production curtailment, load control actions, the number of tap changer operations and the voltage variation at each node. Feeder voltage limits and branch current limits are taken into account in the inequality constraints of the optimization. The algorithm is implemented as an Octave [5] program and is demonstrated in IDE4L project both in laboratory and in real distribution networks. [3]

THE EXCEPTIONAL SITUATIONS AND HANDLING THEM IN THE IMPLEMENTED ALGORITHM

If the congestion management algorithm is not operating properly the network state can become unacceptable in some loading and generation situations. Therefore, the operation of the algorithm needs to be robust and the algorithm has to be able to operate reasonably also in exceptional situations. Three exceptional situations that need to be considered when implementing congestion management algorithms are identified:

1. Lack of updated state information of the control area
2. Acceptable solution not found by the congestion management algorithm
3. The network is in fault location, isolation or supply restoration state

Exception handling functionality for each of these situations has to be present in real network congestion management implementations.

The general operation of the example congestion management algorithm is depicted in Figure 2. In normal operational conditions, the algorithm operation proceeds as indicated by the green arrows in Figure 2. When the algorithm is started, there is at first an initialization phase where the static network data is read from the database and processed to a

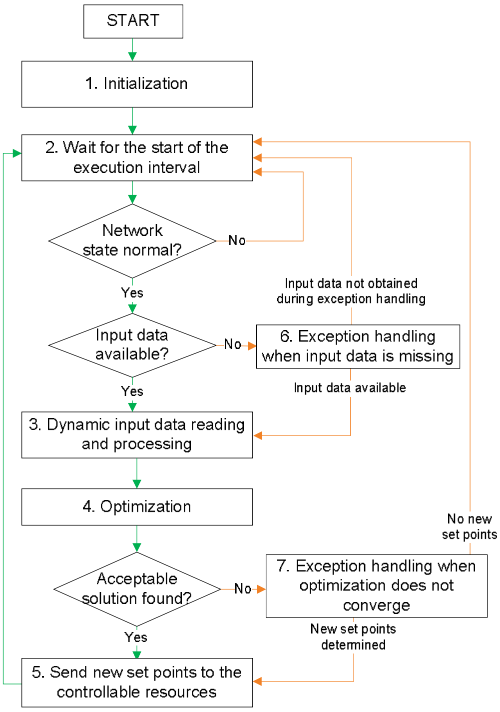

Figure 2. The operational flow chart of the implemented congestion management algorithm.

form suitable to be used by the optimization algorithm. After the initialization, the congestion management algorithm is executed with a predefined execution interval which is in IDE4L demonstrations set to be one minute but can be also any other value. When the start of the execution interval is reached or an execution request is received, the algorithm checks the network state and the availability of input data and proceeds to input data reading if the network state is not faulted and the input data is available. Database flags are used to communicate the network state and the availability of input data to the congestion management algorithm. After reading the input data, the optimization algorithm is called and new set points are calculated. These new set points are sent to the controllable resources if the improvement in the objective function value is adequate or if the network state is unacceptable prior to the optimization. The orange arrows in Figure 2 indicate the algorithm operation in exceptional situations and include all three exceptional situations listed above.

Adequate input data not available

Congestion management algorithms require information on the state of the control area to be able to determine their control decisions. The information can be obtained through direct measurements or state estimation can be utilized. If adequate input data is not available, the algorithm is unable to run reasonably and it is possible that the network state remains unacceptable for a too long time which cannot be tolerated. One possibility in these cases is to send such

predefined set points to the controllable resources that the network will remain in an acceptable state in all possible loading and generation situations.
The implemented congestion management algorithm utilizes state estimation data as its input. If the state estimation data is not available due to e.g. communication or state estimation algorithm failures, the optimization algorithm cannot be executed and in Figure 2 the algorithm proceeds to step 6. The flow chart of the operations inside block 6 of Figure 2 is depicted in Figure 3.
If state estimation results are not available, the algorithm, at first, sends an execution request to the state estimation algorithm and waits a predetermined time for the state estimator to finish its calculations. If the input data is available after the execution request, the algorithm proceeds to step 3 in Figure 2 and continues its operation normally. If state estimation data remain unavailable, the algorithm sends an alarm message to the operator and calls a function that tries to solve the problem. The function first determines where the problem is (interfaces, database, communication, measurement devices, state estimation algorithm) and after that tries to solve the problem. It can for instance reboot interface clients, measurement devices, the state estimation algorithm or even the whole SAU.
If the input data has been missing for longer than a predefined time, the congestion management algorithm sends predefined control set points to the controllable resources. The set points are determined by offline simulations of the network. The simulations need to be conducted in the worst case conditions that are maximum loading – minimum generation and minimum loading – maximum generation. The lowest network voltage can be seen in the maximum loading – minimum generation case

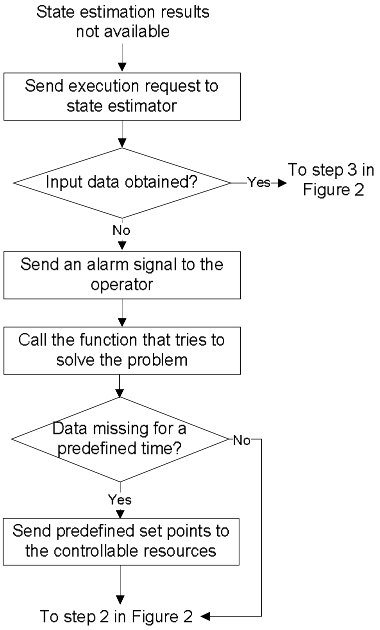

Figure 3. Flow chart of the exception handling when adequate input data is not available.

and the highest voltage in the minimum loading – maximum generation case. The largest branch current in a particular line can occur in either one of these situations but not outside these cases.
The set points need to be such that the network always remains in an acceptable state but also operation of for instance DG overvoltage protection can be taken into account. If voltage rise is the factor limiting the network's DG hosting capacity, the AVC relay set point needs to be set to the smallest value in which all network voltages are still above the lower voltage limit in maximum loading – minimum generation case. The set points of controllable reactive power resources should be set such that they are consuming the maximum amount of reactive power to mitigate the voltage rise. No limitations for DG real powers are mandatory since the DG overvoltage protection will disconnect the units in case of too high voltages. If relying on overvoltage protection is not preferred, the DG real power set points can be set to a value which leads to acceptable voltage level also in the minimum loading – maximum generation case. If overloading of lines occurs in either of the worst cases such real and reactive power set points need to be selected that overloading is prevented. Due to the presence of both load and generation it is possible that overloading occurs only somewhere down the feeder and is not visible to the protection devices at the beginning of the feeder.

Acceptable solution not found by the congestion management algorithm

There are two possible reasons for situations where the congestion management algorithm does not find an acceptable solution: either the congestion management algorithm itself does not converge or the available controllable resources are not adequate to restore the network to an acceptable state. Remaining in a network state where some technical constraint is overstepped is unacceptable and, hence, the congestion management algorithms always need to have exception handling procedures for these kinds of situations. The simplest way of tackling this would be to use the same predefined set points which are used also in case of lack of adequate input data. Also other measures can be, however, taken before resorting to this approach and the possibilities depend on the implementation of the congestion management algorithm and the automation architecture on which it is running.
The implemented congestion management algorithm is an optimizing algorithm and the convergence of optimizing algorithms cannot be guaranteed. If an acceptable solution is not found by the optimizing algorithm, the algorithm proceeds to step 7 of Figure 2. The flow chart of the exception handling in this case is depicted in Figure 4.
If the optimization does not converge, optimization is retried with recovery methods. The parameters of the optimization algorithm can be changed (initial values, tolerances, maximum number of iterations) and soft limits can be used

instead of hard limits. If an acceptable solution is found in the optimization with recovery methods the algorithm moves to step 5 in Figure 2 and continues its operation normally. If the optimization does not converge even after the recovery methods, the algorithm sends a log message to the operator and checks if the current network state is acceptable or not i.e. are all node voltages between feeder voltage limits and all branch currents below the thermal limits of the feeders. If the network state is acceptable, the failure of the optimization algorithm causes the network to operate in a non-optimal way but the situation is not critical. The algorithm returns to step 2 of Figure 2 and the optimization is retried at the next execution interval.

If the network state is, however, unacceptable something needs to be done. In this case, a rule based congestion management algorithm based on [2] is called. The rule based algorithm cannot have convergence problems but also it can be unsuccessful in restoring the network to an acceptable state if the amount of controllable resources is not adequate. If the rule based algorithm is able to restore the network to an acceptable state, the algorithm moves to step 5 in Figure 2 i.e. new set points are sent to the

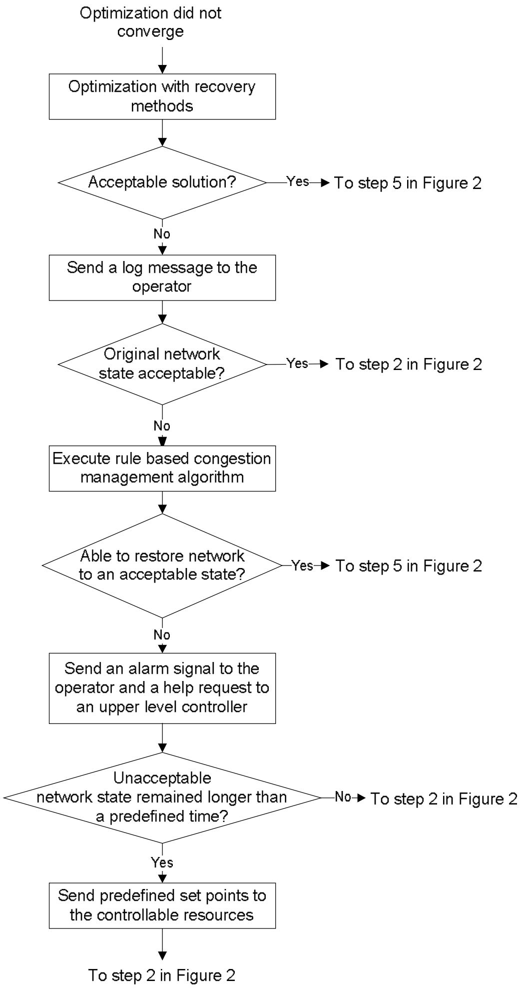

Figure 4. Flow chart of exception handling when the optimization algorithm does not converge.

controllable resources. If the rule based algorithm is not able to restore the network to an acceptable state, an alarm message is sent to the operator and a help request is sent to an upper level controller. The SSAU sends the help request to the PSAU and the PSAU to the DMS. The help request is sent only in cases where the upper level controller can possibly improve the situation. If there is a voltage problem in the LV network, the MV control can possibly help but if there is a problem with branch currents, the MV control cannot do anything. In case the unacceptable network state remains longer than a predefined time i.e. the upper level controller is not able to restore an acceptable network state, same predefined set points as in the case of lack of input data are sent to the controllable resources.

Network in fault location, isolation or supply restoration state

During network fault location, isolation and supply restoration the congestion management algorithm is suspended as the network switching state is constantly changing. When the network is in normal state or the fault has been isolated and the repair is ongoing, the algorithm is executed normally. If the network switching state is unusual (e.g. in fault repair state), the voltage limits used in the optimization can be relaxed. Automatic network data update is mandatory to be able to utilize the congestion management algorithm also in unusual switching states. In those situations the algorithm is often needed even more urgently than in normal situations.

CONCLUSIONS

In this paper, handling of exceptional situations in distribution network congestion management algorithms is discussed. The different exceptional situations are identified and handling of them in one implemented algorithm is represented in detail.

REFERENCES

[1] S. N. Liew and G. Strbac, 2002, "Maximising penetration of wind generation in existing distribution networks", *IEE Proc. , Gener. Transm. Distrib*., vol. 149, 256-262.
[2] A. Kulmala, S. Repo and P. Järventausta, 2014, "Coordinated Voltage Control in Distribution Networks Including Several Distributed Energy Resources", *IEEE Trans. Smart Grid*, Vol. 5, 2010-2020.
[3] IDE4L, deliverable 5.2/3 "Congestion Management in Distribution Networks", available: http://ide4l.eu/results/
[4] IDE4L, deliverable 3.2 "Architecture design and implementation", available: http://ide4l.eu/results/
[5] J. W. Eaton, D. Bateman, S. Hauberg, R. Wehbring, 2015, "GNU Octave version 4.0.0 manual: a high-level interactive language for numerical computations", http://www.gnu.org/software/octave/doc/interpreter/