Scientific Papers of
the Institute of Electrical Power Engineering of
the Wrocław University of Technology

PRESENT PROBLEMS OF POWER SYSTEM CONTROL

4

Wrocław 2013

Guest Reviewers
Ivan DUDURYCH
Tahir LAZIMOV
Murari M. SAHA

Cover design
Piotr PIERZ

Printed in the camera ready form

Institute of Electrical Power Engineering
Wrocław University of Technology
Wybrzeże Wyspiańskiego 27, 50-370 Wrocław, Poland
phone: +48 71 320 26 55, fax: +48 71 320 26 56
www: http://www.ie.pwr.wroc.pl/; e-mail: Inst.Energ@pwr.wroc.pl

OFICYNA WYDAWNICZA POLITECHNIKI WROCŁAWSKIEJ
Wybrzeże Wyspiańskiego 27, 50-370 Wrocław
http://www.oficyna.pwr.wroc.pl
e-mail: oficwyd@pwr.wroc.pl
zamawianie.ksiazek@pwr.wroc.pl

ISSN 2084-2201

Drukarnia Oficyny Wydawniczej Politechniki Wrocławskiej. Order No. 706/2013.

model based design,
test automation, Simulink and EMTP

Piotr SAWKO*, Kevin HILL*,
George EVANS*

DESIGN AND TESTING OF POWER SYSTEM PROTECTION FUNCTIONS USING SIMULINK® AND EMTP

This paper presents how simulation of power system behaviour under different disturbances can be used to test protection algorithms. A method for co-simulation using Electromagnetic Transient Program (EMTP) and Simulink is presented along with the procedure used for testing protection algorithms against different power system waveforms. The advantages of model based design are described showing how a realistic model of protection relay can be designed and how individual components are integrated.

1. INTRODUCTION

Modern numerical protection relays are used in many different applications. It is therefore of paramount importance to verify that each of the protection functions will operate exactly as expected, even during unusual disturbances. Early detection of any problems is also beneficial as it removes the necessity of rework allowing for a shorter time to market. With advancement in computer aided design (CAD) software and wide availability of high performance computers it is possible to test power system protection functions more comprehensively than ever before. First a short description of both tools is given as well as justification for using them. Then, the principles of model based design are discussed showing what benefit it can bring for the protection designer. In the following sections the details of co-simulation between EMTP and Simulink are described. The paper concludes with test results of a simplified under frequency function as an example of the capabilities of the method. An example is shown of function design in Simulink and power system model in EMTP.

* Schneider Electric UK, Ltd.

2. SIMULATION TOOLS

2.1. SIMULINK

Simulink is a graphical design tool allowing modelling of both continuous and discrete systems. It contains a wide range of blocks suitable for modelling control and signal processing algorithms allowing the designer to use his domain knowledge and not requiring him to become a software expert. Simulink allows the user to create his own reusable libraries which promotes the idea of component reuse and encapsulation as each of the libraries are self contained. Because of the graphical interface it is much easier to explain the concept to others while still being able to demonstrate how the model behaves under different application scenarios.

Simulink was chosen as a design tool because of the already mentioned simplicity of use as well as very good integration with Matlab which facilitates scripting. This is extremely important if it is planned to automate some of the testing done on the models.

2.2. EMTP-ATP DRAW

EMTP-ATP Draw is a power system simulation tool with a graphical user interface. It contains library blocks, allowing the simulation of behaviour of different power system equipment. With the library being constantly updated, this is one of the most popular software programs for power system simulations and is widely used in both academic and industrial applications. Because of the large number of ready to use components it is easy to create a model with the right level of detail, giving a compromise between accurate results, speed of simulation and the time necessary to produce the model.

This tool has been chosen since it is widely recognized for its accurate results as well as its fast solver allowing the tests to be run in shorter time compared to other simulation programs.

3. MODEL BASED DESIGN

With the computational power of modern computers it is possible to perform extensive testing of the designed functionality without having to produce costly prototypes. Model based design (MBD) facilitates early testing of algorithms, therefore making it easier to debug them and test against different scenarios. In the early stages of development only the general requirements are known therefore the model is relatively simple and many assumptions are initially made. As the work on the model progresses it becomes closer to the final product. The confidence in the designed solution increases as more tests are conducted on the model. This can range from recording the response of the function to ideal signals (during initial stages of design) to responses to actual waveforms that can exist on the power system. After the algorithm is thoroughly tested it can then be implemented to run in the final product. This way most errors are found early in the design and little rework is required after the algorithm is implemented in the relay. The workflow of MBD is shown in Fig. 1.

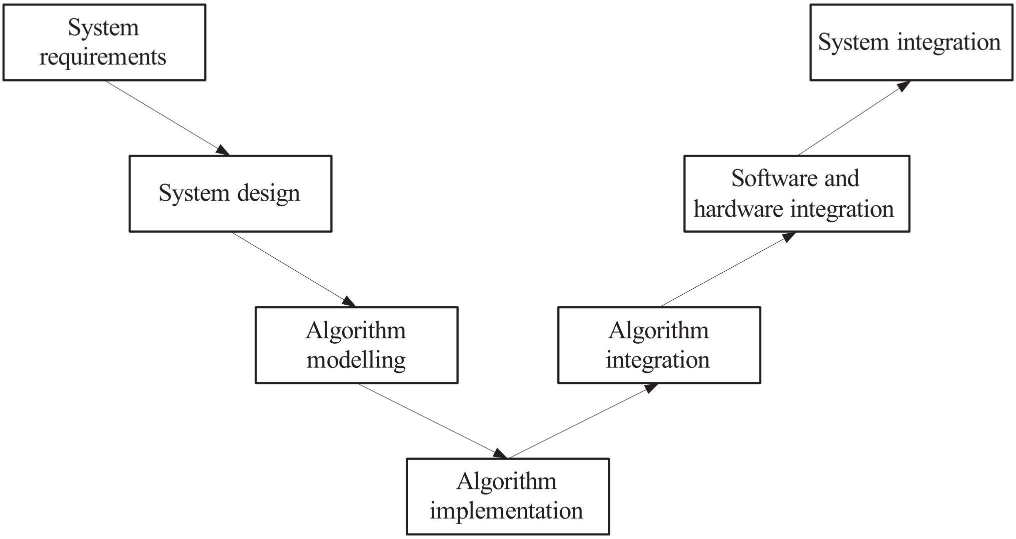


Fig. 1. Workflow in model based design development

With the initial design only the core algorithm is tested and the impact of the rest of the system is ignored. Many protection functions operate on the magnitude of a waveform and therefore in an early stage of development the testing is done using constant signals as opposed to sine waves. This simplifying assumption can be a source of many errors which will be detected in the later stages of development.

With the design tested using constant signals one can remove some of the assumptions by modelling the signal processing that will be used in the final product. This way it will be possible to confirm whether the algorithm still operates correctly even if the input signal is distorted by the signal processing. At this stage performance of the final product can be estimated with two main sources of delay being modelled. With signal processing being modelled it is possible to use both ideal sine waves as well as real power system waveforms as test vectors. This way the new algorithm can be tested using waveforms which are known to have caused problems for existing algorithms.

Since modern digital relays have advanced disturbance recording capabilities it is common to use those waveforms when testing new product or protection functions.

The level of detail of the model can be increased by modelling other parts of the final product like current transformers, anti aliasing filters, analog to digital converters, tripping contacts, etc. The more detailed the final model is the more confidence one has that performance of the model will be the same as the performance of the final product. If any problems are discovered during the design they can be addressed early on.

The important thing is that all the tests are stored, so if in the future a change in the algorithm is required regression testing can be easily done. Simulation of the algorithm used in the final product is also an easy way of error finding in case of relay mal operation.

MBD can make the design process much easier through the use of common library blocks which can be shared between designers and products. In this way common blocks of functionality have to be designed and tested only once. This reduces the risk of introducing errors in the algorithm as well as accelerating the design stage. An example of a common block can be a specific type of timer or a filter.

Simulink supports common block libraries and allows traceability between the instance of the block and the master copy of the library. In this way changes done to the library are automatically cascaded throughout the design. During initial testing sometimes it may be required to do some changes to the instance itself – this is possible by temporarily disabling the link. This allows the instance to be modified however it is still possible to restore the instance with the functionality in the library.

This paper describes how MBD can be implemented using Simulink however most of the techniques described can be realized using other tools as well.

4. CO-SIMULATION BETWEEN SIMULINK AND EMTP

The design process described in the previous section assumes that real power system waveforms are available for testing of the protection functions. This however is not always the case. Some fault scenarios are much more common than others so then it is possible that a real waveform can be used for testing. There are however many scenarios which are rare and no real waveforms are available.

This problem can be addressed by using modelling tools in order to understand the behaviour of the system during different types of disturbances.

The advantage of this approach is that many test waveforms can be generated to simulate scenarios relevant to the protection function under test. One example can be the testing of a distance relay by applying faults at specific locations on the line with different values of fault resistance. Such a comprehensive test would be almost impossible to do if one wanted to use only waveforms recorded on the power system, as this would require the fault to appear at exactly the same location but with different fault resistance each time. This however can be easily achieved by using power system simulation software.

With the increased number of tests running them all and analysing the results one by one can be a very slow process. If one continues with the distance protection example – for each fault location and fault resistance the value of parameters has to be changed manually, the test has to be run and results have to be analysed (manually as well). Figure 2 shows what task has to be done for each of the tests.

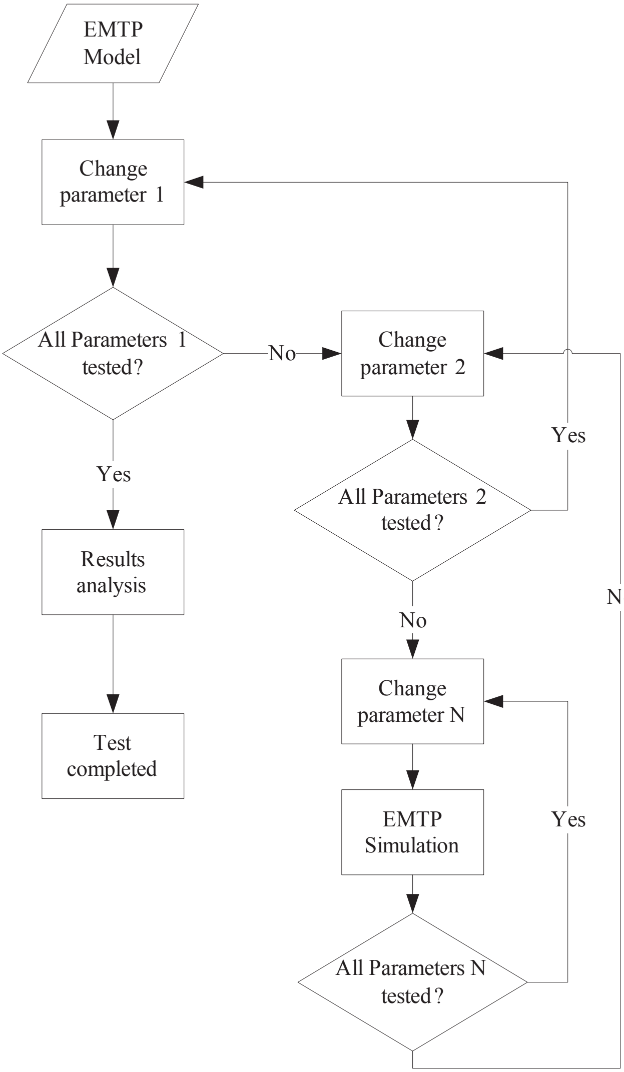


Fig. 2. Breakdown of tasks done during transient testing

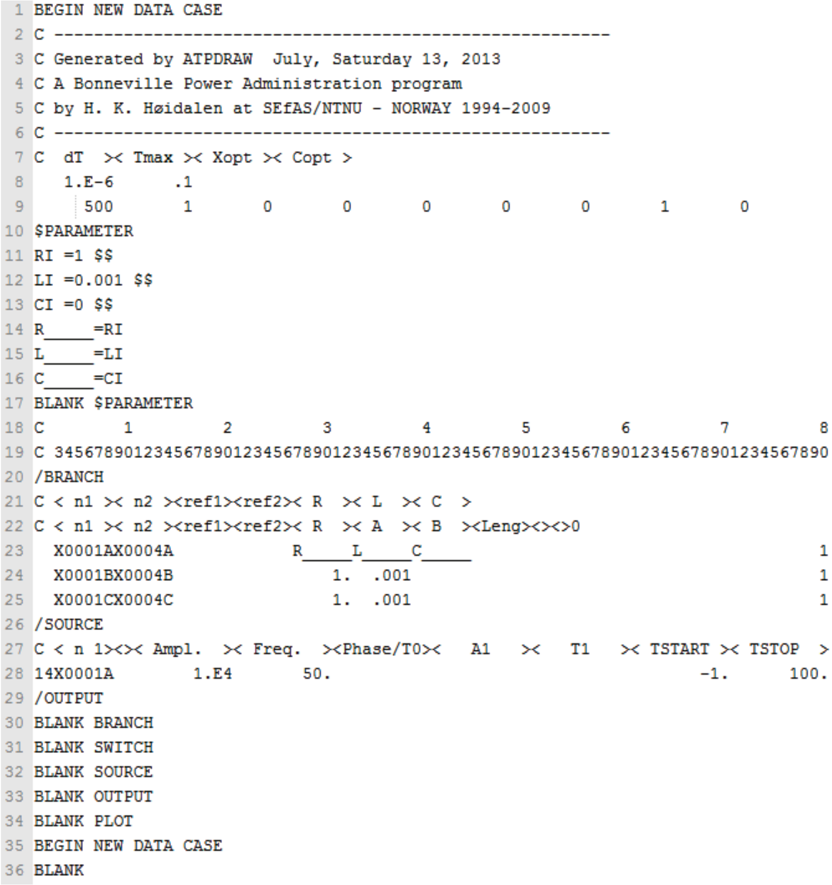
Continuing with our example if it is required to simulate faults at 10 different locations with 3 different values of fault resistance and 3 different fault inception angles this brings the total number of simulations to 90. It is easy to imagine scenarios which increase the number of simulations even further.

Conducting all those simulations manually is error prone and additionally if some changes have to be done to the system used for simulation all the scenarios have to be simulated again. All of this reduces the benefits of MBD.

Large parts of testing can be automated therefore making it easy to test against different fault scenarios. The following sections describe how EMTP simulations can be automated and how the waveforms can then be used to test Simulink based algorithms.

4.1. AUTOMATION OF ATP-EMTP TESTING

When a model of a power system is created in ATP-EMTP before it is actually simulated it gets translated into a format understandable for the EMTP solver. This



```
BEGIN NEW DATA CASE
C --------------------------------------------------------
C Generated by ATPDRAW  July, Saturday 13, 2013
C A Bonneville Power Administration program
C by H. K. Høidalen at SEfAS/NTNU - NORWAY 1994-2009
C --------------------------------------------------------
C  dT  >< Tmax >< Xopt >< Copt >
   1.E-6      .1
     500       1       0       0       0       0       0       1       0
$PARAMETER
RI =1 $$
LI =0.001 $$
CI =0 $$
R_____=RI
L_____=LI
C_____=CI
BLANK $PARAMETER
C        1         2         3         4         5         6         7         8
C 345678901234567890123456789012345678901234567890123456789012345678901234567890
/BRANCH
C < n1 >< n2 ><ref1><ref2>< R  >< L  >< C  >
C < n1 >< n2 ><ref1><ref2>< R  >< A  >< B  ><Leng><><>0
  X0001AX0004A            R_____L_____C_____                                   1
  X0001BX0004B                1.  .001                                         1
  X0001CX0004C                1.  .001                                         1
/SOURCE
C < n 1><>< Ampl.  >< Freq.  ><Phase/T0><   A1   ><   T1   >< TSTART >< TSTOP  >
14X0001A        1.E4       50.                                     -1.      100.
/OUTPUT
BLANK BRANCH
BLANK SWITCH
BLANK SOURCE
BLANK OUTPUT
BLANK PLOT
BEGIN NEW DATA CASE
BLANK
```

Fig. 3. Example .ATP file

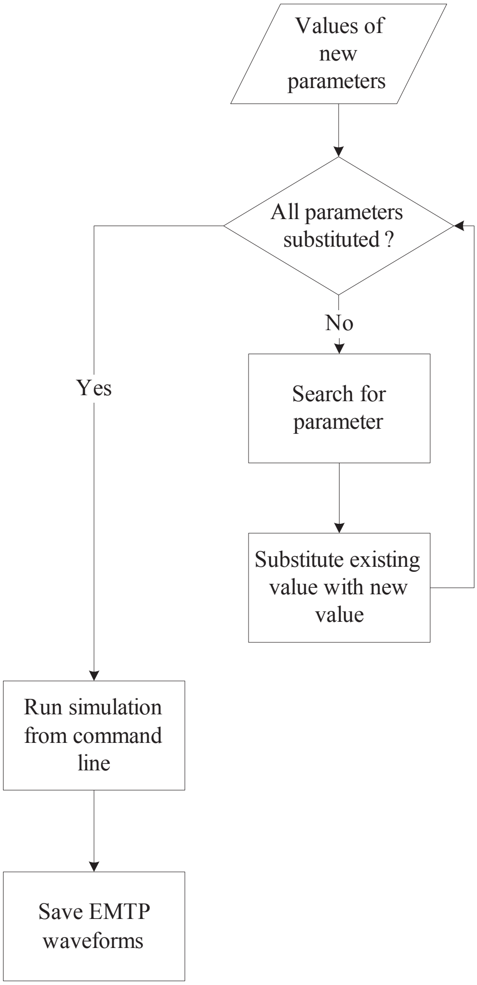

Values of new parameters
All parameters substituted ?
No
Search for parameter
Substitute existing value with new value
Yes
Run simulation from command line
Save EMTP waveforms

Fig. 4. Procedure for automation of EMTP testing

way a graphical. ACP file is converted into .ATP file which describes the circuit in a textual format. Normally when a single simulation is run the conversion is transparent to the user and there is no need to read the .ATP file unless there is a problem with the simulation.

The structure of an example .ATP file (consisting of one source and RLC element) is shown in Fig. 3. Details on how the file is structured can be found in [1].

As shown in Fig. 3 all parameters defined in the graphical tool can still be accessed in the textual file (lines 11 to 13). For each test the value of a parameter can be changed automatically by finding the name that was defined in ATP Draw. After the value is substituted the new .ATP file is used as an input to the EMTP solver.

The whole process can be easily controlled by Matlab script which would contain all the values for each of the parameters. At the end of each simulation the script would save the measurements for the particular simulation in a separate file so that waveforms can be stored and reused whenever required e.g. during regression testing.

Figure 4 shows the procedure that would have to be implemented using Matlab script.

Going back to the example of distance protection it is easy to write a script which will run a simulation for each combination of fault location, fault resistance and fault inception angle.

4.2. AUTOMATION OF SIMULINK MODEL TESTING

After an algorithm is designed in Simulink it is important to be able to thoroughly test it. One way of doing that is using waveforms generated by EMTP simulations as test vectors. For each of the tests a set of expected outputs needs to be created as well in order to be able to evaluate whether the function has operated correctly. In addition to functionally proving that the algorithm operates correctly in all simulated scenarios it is also important to asses how much of the model was actually tested.

If one takes again distance protection as an example one can have many tests which prove that the behaviour is correct for different fault locations, resistance and inception angles however we never tested whether the function can recognize that the voltage measured is not high enough to provide an accurate impedance calculation. If that feature has been implemented but one has not tested it then one needs a way of finding out about it.

This can be done using different types of model coverage. Some of the coverage types available in Simulink are decision coverage, condition coverage and modified condition-decision coverage. The detail of each of them can be found in [2] however each of them indicate how much of the implemented algorithm has actually been exercised during all the tests.

Testing automation can be achieved in Simulink using different techniques. The preferred technique is to load test vectors from a file into the Matlab workspace and run the Simulink models using them as an input. This way there is no need to create and maintain test harness as the model can be run on its own.

Depending on the type of test vector and the modelled function, there may be a need for additional pre-processing on the input signal. As described in Section 0 modelling can cover multiple domains therefore this pre-processing can include CT and ADC modelling as well as Fourier algorithm for calculating the magnitude and angle of the signal. Depending on the function this may or may not be the case as some functions can operate on raw samples.

Figure 5 shows the procedure of configuring the model to use data in Matlab workspace as input vectors.

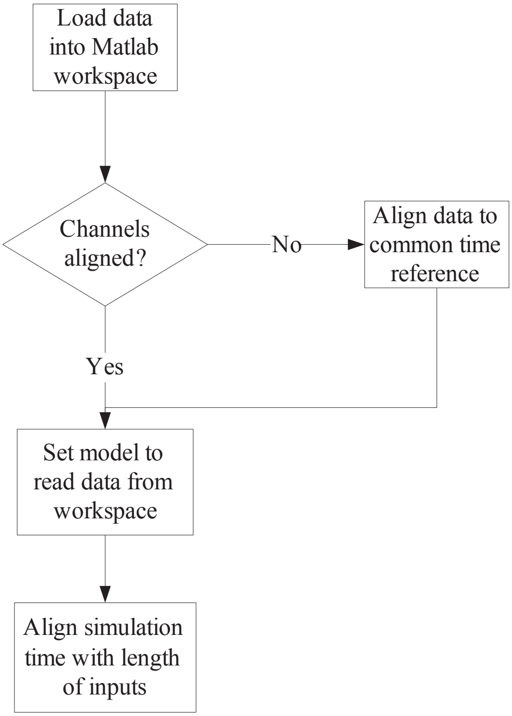

Load data into Matlab workspace
Channels aligned?
No
Align data to common time reference
Yes
Set model to read data from workspace
Align simulation time with length of inputs

Fig. 5. Procedure for using workspace variables as inputs to Simulink model

When using data from the workspace Simulink requires that all inputs are synchronised to a common time reference. If this is not the case then some pre-processing of workspace variables will need to be done. In most cases a linear interpolation is sufficient but this depends on the sampling rate of the original data.

Since the parameters of a Simulink model can be changed using a Matlab script it is possible to run a whole set of tests by simply iterating through the files containing the input data. As long as the names of the variables in those files are consistent the number of tests can be increased.

This flexible approach allows Simulink models to be tested using data collected during relay operation (i.e. disturbance records) or results of a simulation.

5. IMPLEMENTATION AND TESTING OF UNDER FREQUENCY FUNCTION

Using the technique described in the previous section allows testing of any protection function. As an example for this paper an under frequency function was chosen since large variations of frequency are rarely observed on a real power system therefore not many disturbance records of such events are available.

Under frequency protection responds to a prolonged decrease of power system frequency. This is normally caused by mismatch between power supply and demand and can occur e.g. when a main link with another major part of the power system is disconnected.

Since the reduced frequency can have an adverse effect on both the electrical load and the generator supplying the system it is important to be able to detect the decrease of frequency.

The following equation describes the relationship between the change in real power, the total inertia of the system (*H*) and the corresponding change in frequency.

$$\Delta P = 2 \cdot H \cdot \Delta f \tag{1}$$

Although this can give an estimated value of frequency over time it only takes into account the inertia of the system without considering the response of the turbine which is important for evaluating the frequency over tens of seconds.

Figure 6 shows a Simulink block diagram of an under frequency function. Detailed description of Simulink and its features can be found in [3].

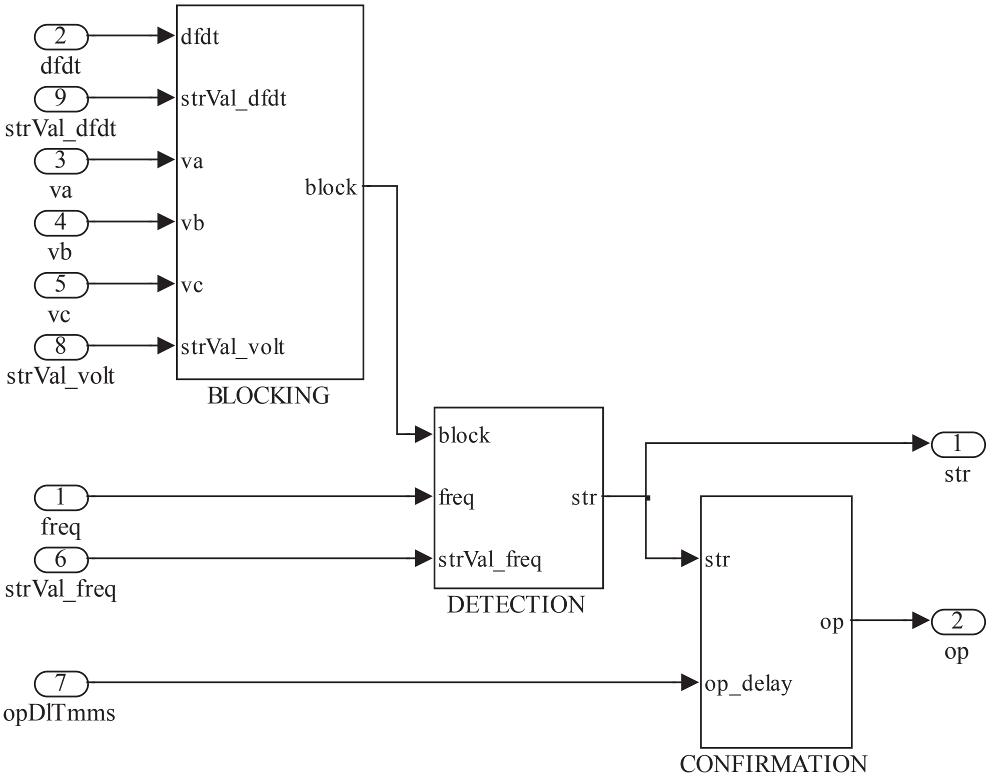


Fig. 6. Block diagram of under frequency function

The model consists of three main blocks:

- Under frequency detection where the measured value of frequency is compared to a threshold in order to determine if the function should operate. Also, any blocking signals are taken into account in this block.
- Under frequency confirmation where a certain time delay may be applied in order to make sure that the function will not operate for transient frequency excursions. The delay can normally be set by the user according to the specific application.
- Under frequency blocking where a decision is made whether protection needs to be blocked despite the frequency falling below the set threshold. One application scenario is when the magnitude of the voltage is so low that accurate measurement of frequency is impossible therefore the protection has to be blocked.

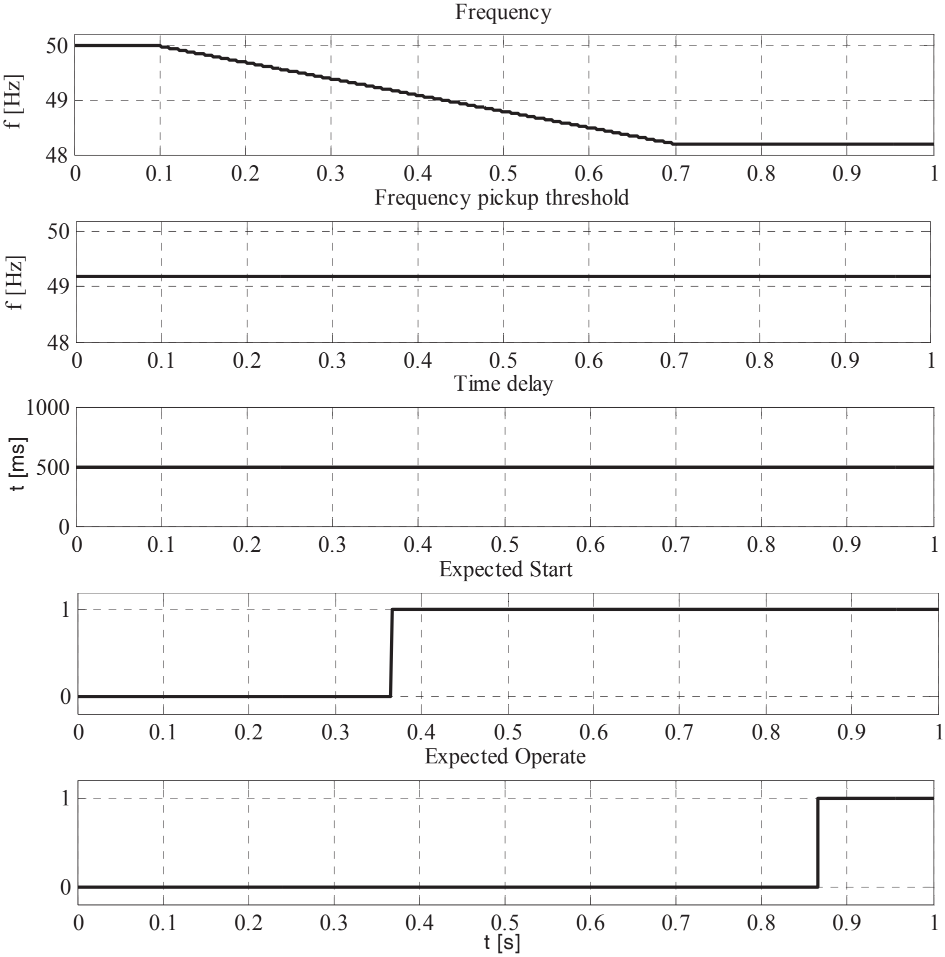


Fig. 7. Example of test vectors and expected results

Having the model designed in Simulink allows this behaviour to be tested. In this paper two types of tests are used. First the response of the function to ideal signals is tested which means that any errors due to the measurement of the frequency are ignored. Figure 7 shows example test vectors along with expected outputs. For the sake of clarity not all inputs are shown. The test vectors shown are used as inputs to the model. Expected outputs are compared with the outputs of the function after the execution of the test – the comparison can easily be implemented as a simple Matlab script. In this example a 3 Hz/sec frequency ramp is applied with the under frequency setting of 49.2 Hz. The Start output is expected to be asserted at 0.36 sec and the Operate output is expected to follow the Start with a 500 ms delay.

The aim of the test shown above is to prove that the algorithm works as required for ideal signals and not to establish whether the function will operate correctly in real world applications. The latter can be done using the second type of testing which uses power system waveforms – in this case results of EMTP simulations. This test not only mimics the behaviour of the power system but also introduces errors that exist in real products, e.g. nonlinearity of current transformer (CT) and analog-to-digital converter (ADC) as well as inaccuracies added due to the measurement technique. Depending on the required accuracy of the simulation some of the mentioned sources of errors can be ignored or some other additional errors can be added. Additional information on the subject can be found in [4].

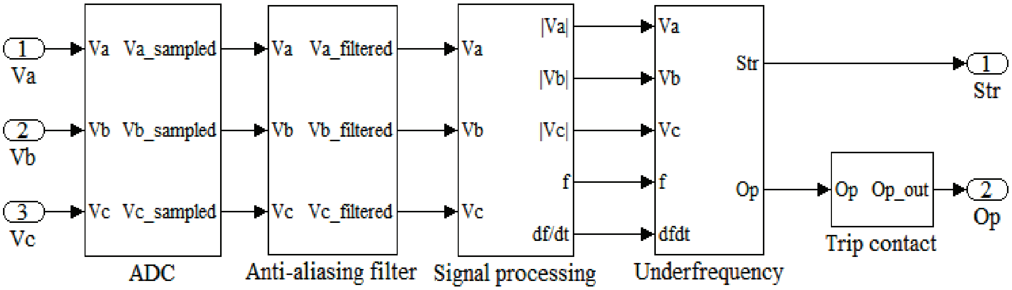


Fig. 8. Simulink model including the whole acquisition chain

Figure 8 shows the Simulink model used in these tests (for the sake of clarity all the settings were hidden). Comparing to the model from Fig. 6 additional blocks are present in order to bring the model as close to reality as possible. Starting from the left the ADC is modelled in order to take into account its dynamic range. Following the sampling of the voltage waveforms anti aliasing filtering is applied. The next block is responsible for signal processing of raw samples and producing frequency and voltage measurements. Depending on the type of filtering used this block can introduce significant delay into the total operating time of the function therefore it is important to model it accurately. The next block is the under frequency logic. The last block on the right represents the delay introduced by the relay contacts which adds to the total delay of the under frequency function.

The EMTP model used as the source of the test vectors are shown in Fig. 9 and Fig. 10. It consists of two synchronous machines connected via a transformer and a line. One of the machines represents a 10MVA small local generation which normally exports real power into the system. The generator itself is not able to provide power for all local loads therefore if the circuit breaker (CB) at the common coupling point is opened the frequency of the islanded system will decrease which should cause the under frequency function to operate.

On the other hand the under frequency function should not operate for transient frequency dips caused by faults on the system (in our case represented as a fault on the line connecting the small generator with the rest of the power system).

The third scenario to be considered is when the frequency relay is connected at a substation powering large induction motors. When a fault on the system causes the substation to become isolated from the supply for a short period of time the motors work as generators producing fast decaying voltage with decreasing frequency. It is required for the frequency protection not to operate during such a scenario.

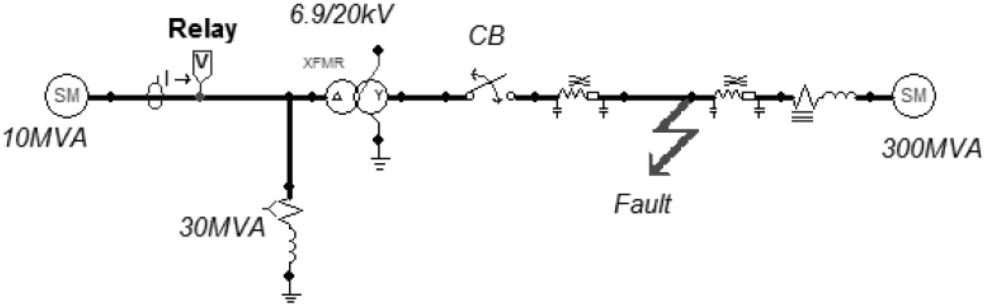


Fig. 9. Interconnection between distributed generation and the grid

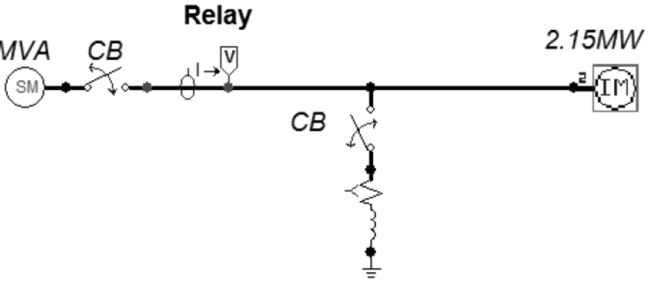


Fig. 10. Generator powering an induction motor

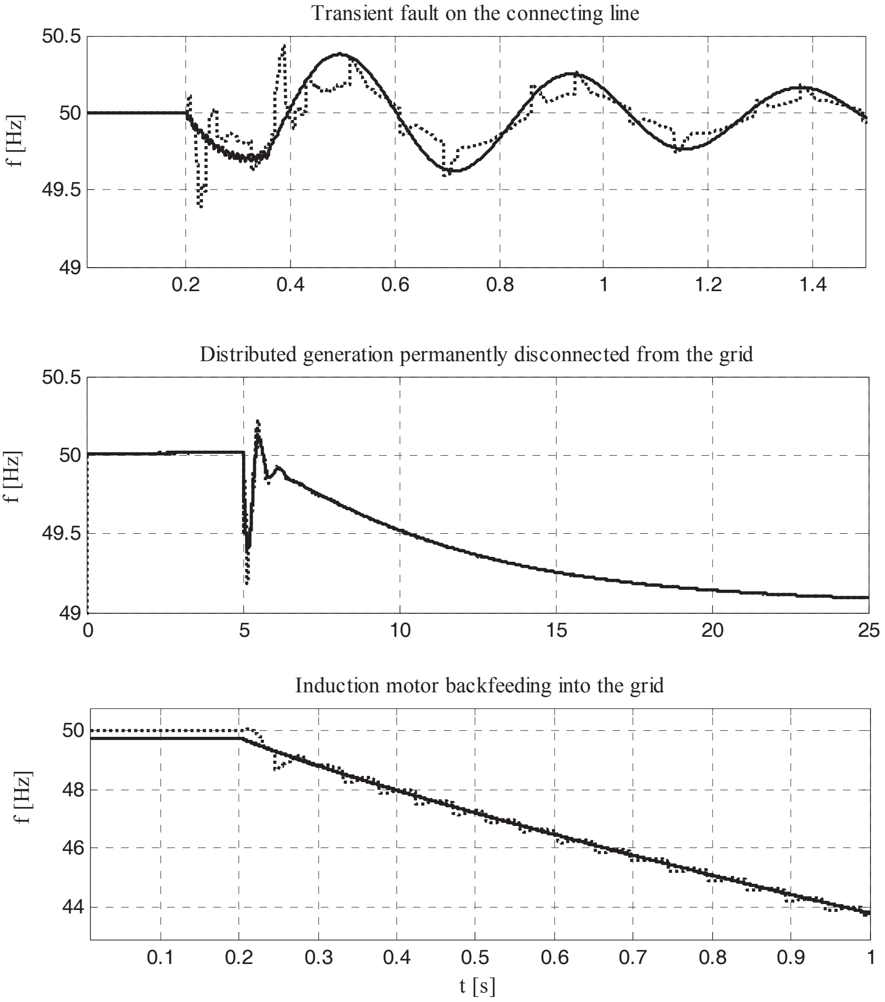


Fig. 11. Waveforms for each of the test scenarios
(solid line – actual, dotted line – measured frequency)

Figure 11 shows example results for each of the test cases described. Both the measured and actual frequencies are plotted in order to show the differences. Depending on the algorithm used for frequency measurements the error can vary. This allows selecting the best suited algorithm. When the frequency measurements are used as inputs for under frequency it also allows determining whether the relay would behave correctly in the particular scenario.

Each of the diagrams in the above figure shows both measured and actual frequencies. The measured frequency is obtained using the signal processing block and therefore is dependent on the algorithm used. The actual frequency is calculated based on the rotation speed of the rotor of synchronous generator or induction machines (depending on the test scenario). Ideally both the actual and the measured frequency should be identical however due to inaccuracies of the processing chain and delays introduced by the filter algorithms a small difference can exist. The difference varies depending on the test scenario therefore automated testing is ideal for choosing the most suitable algorithms.

Waveforms gathered during EMTP simulation are be used as input to the Simulink model as described in 4.2. This allows checking whether the settings selected for the frequency relay will work as expected for all the simulated disturbances.

6. CONCLUSIONS

This paper demonstrated how available CAD tools can be used to design and test power system protection functions. By following the described workflow it is possible to create new protection functions by using blocks of functionality provided by Simulink. A design can be first tested to prove that the function works for ideal signals. In addition Simulink allows gathering of model coverage metrics which can demonstrate that the function was tested thoroughly. The final stage of testing is by using waveforms generated using ATP-EMTP. Using the method described in the paper it is possible to simulate multiple scenarios automatically. Since all the tests can be stored and easily re-run it is very easy to conduct regression testing or identify problems with the design. By modelling the parts of hardware used to run the protection function on it is possible to make the simulation results even closer to those of a physical protective relay.

The under frequency example shows the benefits of testing the function using ATP-EMTP generated waveforms in conjunction with Simulink. By simulating multiple scenarios one can gain confidence that the protection function will operate as expected and because it is easy to extend the list of test cases one can easily check whether the function meets the new application requirements.

REFERENCES

[1] PRIKLER L., HOIDALEN H.K., *ATPDraw User's Manual version 5.6*, 2009.

[2] BARESEL A., CONRAD M., SADEGHIPOUR S., WEGENER J., *The interplay between model coverage and code coverage*, EuroSTAR Software Testing Conference 2003.

[3] MATHWORKS, *Simulink User's Guide.*

[4] GEC MEASUREMENTS, *Network protection and automation guide 3rd Edition*, 1987.