Differential Relay Models inside Real Time Simulators

F. Magrin, M.C. Tavares

***Abstract*-- This paper describes important steps when developing relays models inside a real time simulator. These relays models, as the differential element inside real time simulators, can improve the dependability of the tests. In the present paper a commercial transformer differential element model is described. In order to compare the model response to the real relay, the last was connected to the real time simulator. Also to enhance the range of the tests it was tested a built-in differential model existent in the simulator library. All kind of daily situations for a protection relay was tested including inrush and internal faults to the power transformer. The simulations were performed with RTDS.**

***Keywords*: Protection relay, Differential element, Model, Real time simulation, RTDS.**

I. Introduction

DIGITAL or micro processed relay replaced electromechanical and static relays because of their flexibility and reduced cost. Added to that the development of new algorithms became easier to be implemented what also leaded to more complex protection elements. This complexity promoted a more difficult or even not possible characteristic to be tested by the engineers using static equipment tests like test sets.

Based on these difficulties the real time simulators brought a new perspective for power system studies. Real time simulators can save engineers a lot of time and, most importantly, increasing the range of analyzed data while testing protection characteristic that are difficult to be tested with regular test sets.

However, this test structure is expensive due to the necessity of specialized workforce and the complexity to configure a test with many equipment. A test with many equipment needs a lot of wires and time what makes the test expensive and sometimes forces the engineers to test only one or two relays, depending on the equipment under test.

Considering additionally system configuration expansions, which are done frequently in the power systems nowadays, the need of relay models inside simulators were brought into existence. These models can be used in the surrounding areas of the main relay, hardware, under test. These models can give the engineers a more precise idea of the whole protection system and the impact of, for example, a new line or transformer, in the system, increasing the dependability of the whole system. Nevertheless, to accomplish this the models need to behave exactly as the real relays installed in the field. In this matter, this paper proposes the use of real relay models inside real time simulators.

The paper presents in the following section the development of a commercial differential element in real time simulator (RTDS) and compares the model with a real relay and with a built-in model. Although the relay has many protection elements, in the present paper only the differential element will be studied.

For comparison it is analyzed two main aspects, the correct operation of the main functions of the differential elements and the speed of their response.

II. Background of a Typical Relay Model

Fig. 1 shows a typical diagram of a relay processing steps. The first step still outside the relay is the connection to the external world. The conventional way is to receive regular PTs and CTs but today is also possible to receive via Ethernet using IEC 61850 and this way would remove the Signal Conditioning step of the relay box. The next step, Low-pass Filter in a conventional relay, is still an analog process but it can be also necessary to be applied, in a digital version, when using IEC 61850, depending on the project.

This paper modeled a conventional relay, i.e. the signal becomes digital just after the A/D Converter.

[1, 2, 3, 4] are some examples of the vast literature about modeling. Literatures [2, 5, 6] already present model developments in real time simulators.

Literatures [7, 8, 9] present some aspects of differential element models.

Back to Fig. 1, firstly the input signals are rearranged to levels, which fit the analog-to-digital converter. After this step, it is necessary to reduce the signal noise level, applying a low-pass filter. The following step is the analog to digital converter. At this stage, the signal is sampled at a specific rate and converted to a binary code, what makes the signal understandable by the microprocessor. Although it depends on the manufacturer and the application of the relay, it is rather common to have a digital filter to isolate the signal at a desired frequency. Following, the signal is decomposed in frequencies in order to calculate the phasors and, consequently, feed the

This work was supported by a grant from the São Paulo Research Foundation, (FAPESP) and from CNPq and CAPES, Brazil.
Fabiano Magrin and Maria C. Tavares are with the School of Electrical and Computer Engineering, University of Campinas (UNICAMP), Av. Albert Einstein, 400, 13083-250, Campinas, SP, Brazil (e-mail: magrin@dsce.fee.unicamp.br, cristina@dsce.fee.unicamp.br).

Paper submitted to the International Conference on Power Systems Transients (IPST2015) in Cavtat, Croatia June 15-18, 2015

protection algorithms with the appropriated data.

III. Differential Elements

There are many types and consequently many different ways to build an algorithm for the same purpose. For transformer differential relays there are high impedance relays and low impedance relays, being the last one the more commonly used. For this kind, there are many operations' principles as fixed pick-up, one slope, two slopes, among others. It is rather common to find two slopes relays but even for this there are many types of two slopes relays, as can be seen in Fig. 2. Added to the main protection algorithm the differential relays call for others parallel algorithm for inrush, over excitation and transformer angle compensation, see [10].

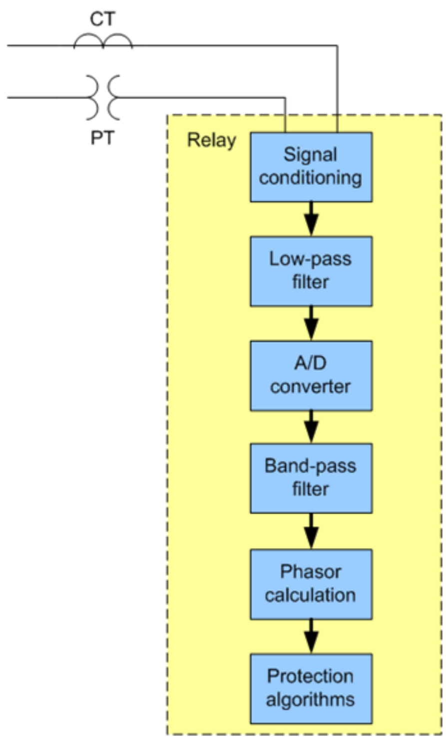

Fig. 2 – Relay Diagram

All these algorithms can be implemented in many different ways. To analyze the different responses the relay algorithm from a commercial relay and the built-in model will be compared.

The commercial relay has the following characteristics: two slope differential, instantaneous non restricted differential, harmonic blocking: cross blocking mode with second and fourth harmonics, harmonic restrained with second and forth harmonics to the restrained current, over excitation blocking based on fifth harmonic and transformer angle compensation with 13 options.

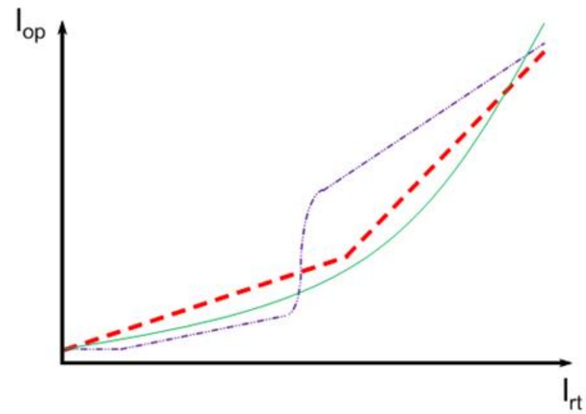

Fig. 1 – Types of Differential Elements

The built-in model has the following characteristics: two slope differential, instantaneous non-restricted differential, non-crossing harmonic blocking mode with only second harmonic, transformer angle compensation with 3 options.

It can be observed that the commercial relay has a more complex differential function compared to the built-in model.

IV. Implementation

Fig. 3 shows the signal processing block for each current phase of the differential relay. The Input comes from the CT and the Output is the phasor current in pu value.

As mentioned before the first step is the low-pass filter. The simulator in use has already some components in its library and some transients programs also can have some components in their library. However, the filter can be also implemented using difference equation as (1). For this particular relay the low-pass filter is a second order Butllerworth with cutoff frequency of 640 Hz.

$$y(k)=A_1 y(k-1)+A_2 y(k-2)+B_0 x(k)+B_1 x(k-1)+B_2 x(k-2) \quad (1)$$

It is important to notice that this stage is a model for an analog process and as mentioned in [11] the engineer must be aware of some model limitation.

The next step is the A/D converter. During the development of the model this step is divided in two. First a sampler at the specific rate as the relay and second the A/D converter itself. It is important to remember that in a software

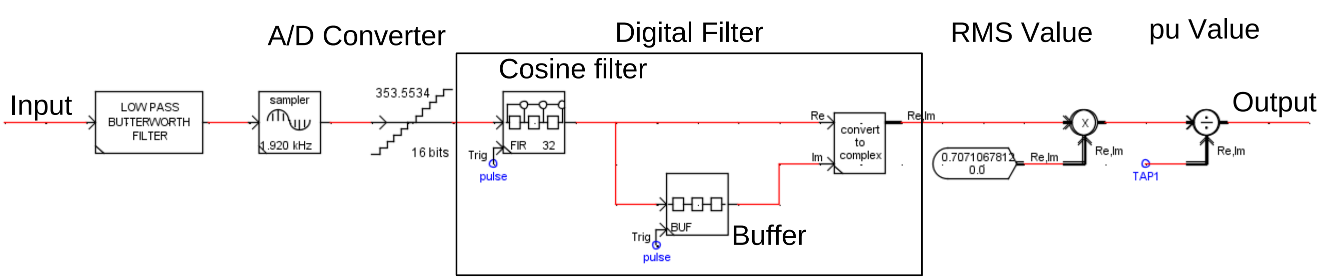

Fig. 3 – Simulator Example

environment everything is already digital but this step is crucial to emulate a real relay. The converter has two main pourposes, divide the signal in steps which means that the signal is not continuous anymore and secondly to insert bounderies to the maximum and minimum measurables values.

When reproducing a commercial relay it is important to follow each datum from the manufacturer without forgetting that sometimes some data are confidential.

To exemplify a possible error of the A/D converter Fig. 4 shows the difference between an 8 bits converter, blue, and a 16 bits converter, red. It is important to notice that the 8 bits converter does not have enough bits to represent some steps with fidelity; an 8 bits converter has 256 steps while 16 bits has 65,536 steps. A wrong representation of the steps and the maximum value measured by the relay will imply in a different sensitivity of the model.

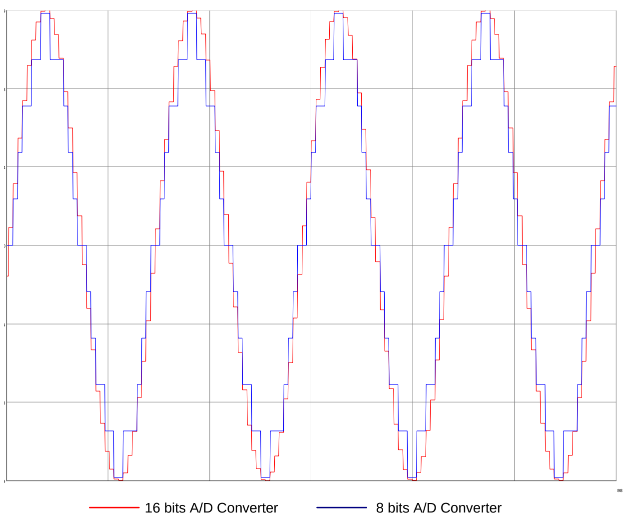

Fig. 4 – Comparison Between 8 and 16 bits A/D Converters

The A/D converter was done in the simulator using the C language. Below is the code. Notice that the code is very simple and it has to be as simpler as possible when programming in a real time simulator. Different from a regular programming, in real time, it is necessary to execute the entire code plus the other elements during the time step of the simulation, which is regularly 50 µs.

The code has three sections. The first one represents the variables, the second one the RAM code and the third one the code itself. The RAM area comprises the code that has to be executed before the simulation starts.

In the example the RAM code is the number of steps, the maximum value and the size of the step. None of these elements need to change during simulation, they only depend on the settings of the component entered by the user.

The CODE area embraces the code executed on-line and it is done every time step of the rack. Here it calculates in which step is the input and send to the output checking if it is positive or negative value.

```
/* VARIABLES */
int num_step;
int max_step;
double step;
int aux;
/* RAM */
num_step = pow(2,Bits);          /* number of steps */
max_step = num_step/2;           /* maximum step */
step = 2*Lim/(num_step-2);       /* each step value */
/* CODE */
aux = trunc(fabs(Input/step));   /* absolute value */
if (aux >= max_step)
{      aux = max_step-1;          }
if (Input < 0)
{      Output = - aux * step;     }
else
{      Output = aux * step;       }
```

After the A/D converter, the next step is extraction of the phasors for a future use in the protection algorithms. For the differential relay under analysis there are digital filters to extract not only the fundamental frequency but also the second, fourth and fifth harmonics. Many filters use two elements in quadrature to extract the real and imaginary parts of the phasor. However, as the commercial relay used has only cosine filter and this is the real part of the phasor, it is necessary to use a buffer to retain the real values and compare the output of the filter with a value that is 90° in delay, see Fig. 3. As it is necessary to extract more than the fundamental frequency, this process was performed for each harmonic of each phase of each winding. For each frequency, the buffer must be lagging 1/4 of cycle for that frequency. In Fig. 3 the signal “pulse” is paramount to give to the filter the exactly moment to take the new sample to the filter and run the buffer. The control of the signal “pulse” depends on the manufacturer and can change if there is a frequency tracking element as it can be seen in [11]. This step is another very crucial step and cannot be computed in the simulator time step to reproduce the actual relay behavior.

Still talking about the digital filter, Fig. 3 shows only the calculation of the fundamental frequency. For each phase there are 4 filters and for a two winding diferential element it is necessary 24 filters in total. This means that this step consumes a lot of processing of the simulator and it deserves special attention. In the proposed model, this step took two processors of the simulator and to accomplish the entire differential element a third processor was necessary.

It is very important when working with real time simulators to always work with imaginary numbers in rectangular format, see Fig. 3, because they demand much less effort from the processors.

To avoid the use of so many processor, one solution is to develop the element in C language and not use pre-built simulator´s blocks. Another advantage of the C language is the facility to move to other real time simulators or even to any off-line software. It is almost impossible to run the relay element model together with the system in one simulation time step therefore everything that does not need to be calculate each time step must be done in advance in RAM area. As the relay time step is much greater than the simulator time step, one solution is to divide the model tasks in parts, i.e. each simulation time step runs parts of the model task and they must be done before the relay time step.

Back to the two final blocks of Fig. 3, the output signal of the band-pass filter needs to be downgraded to rms value and corrected to pu value.

The output of all filters for all frequencies are the phasors but the data are not yet ready to be used in the protection algorithms. The relay model needs to compensate the angle differences between power transformer high side and low side. To accomplish this step it was necessary again to develop a new component. A commercial relay has all possible compensations as [10]. To reproduce this behavior a new component was built and it was assembled to have the same options of the relay and to improve the component flexibility it was also created a setting to allow phasor compensation for ABC and ACB rotation. In addition, the input and output can be used as rectangular or polar form of complex numbers, see Fig. 5, but it is important to remember that in order to run in real time, and solves the system equations in 50 μs or less, the rectangular form is always preferred.

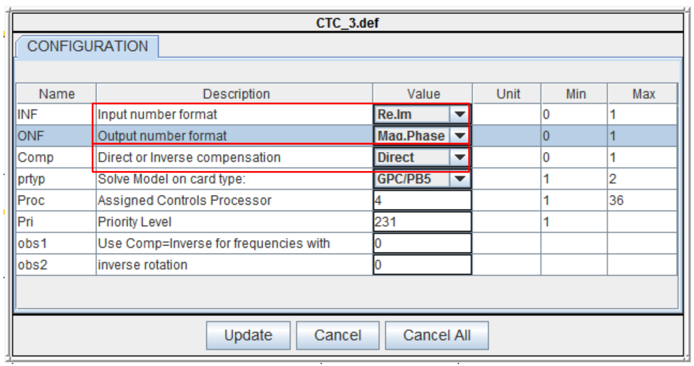

Fig. 5 – Matrix Compensation Component

After the transformer compensation, it is time to go to the differential algorithm with these compensated currents: it is possible then to calculate the operation and restrictions currents as shown in Fig. 6. The simulator usually processes the whole system at 50 μs, but this is not the relay processing time. To reproduce the same behavior of the relay processing interval an S/H (sample and hold) controlled by the "pulse240" signal was used after the operation and restriction currents calculation.

This element has the main function of putting the model at the same pace of the real relay and this is the most important step to reproduce a real relay. Without this element, the model would take decision every real time simulator time step and then it would be much faster than the real relay. This step is of paramount importance to achieve real relays responses. Literature [6] shows that the time step can change the model response and it happens because the control mentioned before was not done.

In possession of the restriction and operation currents and at the correct pace, the next step is the protection function itself. For the specific differential model, it is just necessary to develop the two slopes characteristic and the harmonics comparisons.

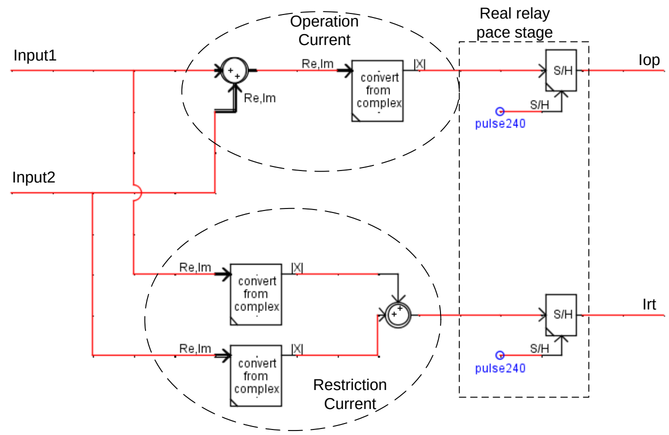

Fig. 6 – Differential Currents at the same Pace

V. System and Test Description

All tests were performed in a system as shown in Fig. 7. This system makes possible to run external faults to the differential section, internal faults to the differential section, internal faults to the transformer as turn-to-turn faults and ground faults, evolving faults (external to internal), inrush with and without fault, serial and parallel sympathetic inrush, and CTs saturation.

To accomplish all possible real situations it was necessary to test all types of faults, change the point on wave, and test the evolving faults for all kind of situations in the high side and low side. It was done 1190 tests in total and each test was applied at the same time to the real relay, model and built-in model.

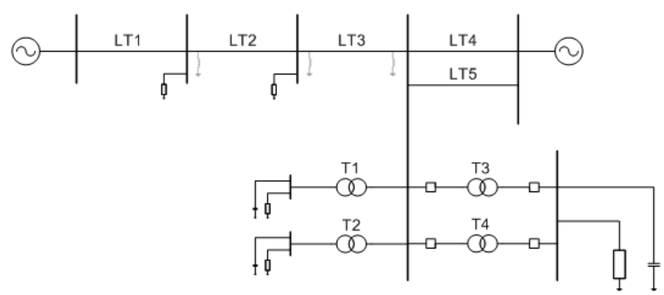

Fig. 7 – Single Line Diagram

VI. Results

This section presents the analysis of the tests and some significant results. For comparison, it was analyzed two main aspects, the correct operation of the main functions of the differential element and the speed of their responses.

Correct/true and false operation in this paper means that the two elements under comparison agreed or not in the response. It is very important to notice that the objective in the present analysis is to identify if the models reproduce the performance of the physical relay, no matter whether the operation is correct or not from the protection point of view.

The speed response was analyzed comparing the difference in cycles of 60 Hz the two elements under analysis.

Two main analysis were performed, model relay versus real relay and built-in relay versus model relay.

A. Real Relay x Model

The difference between the real relay and the model is analyzed in this section. As the model is an emulation of a commercial relay the following signals were possible to be monitored regarding pick-up and response time:

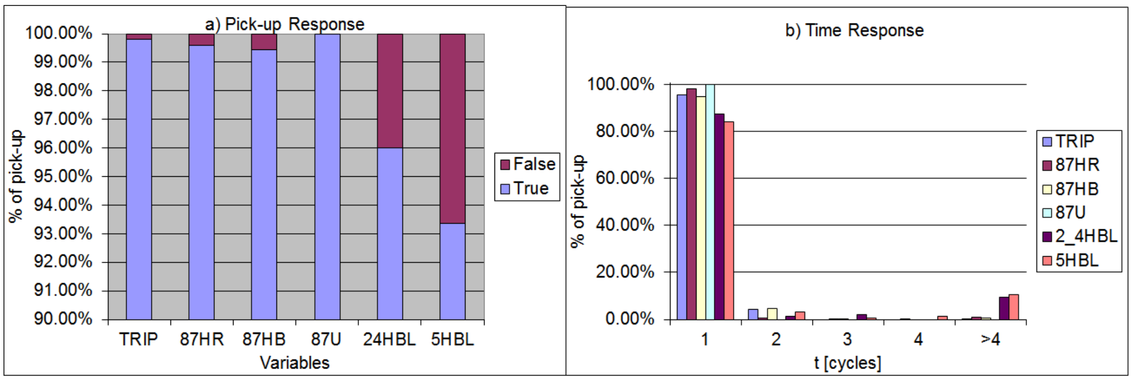

Fig. 8 –Relay x Model Responses

- TRIP - trip signal, set to an or combination of 87HR, 87HB and 87U;
- 87HR - harmonic restrained differential;
- 87HB - harmonic blocking differential;
- 87U - non restricted differential;
- 24HBL - second and fourth harmonic blocking;
- 5HBL - fifth harmonic blocking.

Looking at Fig. 8a it is possible to see the final result for all 1190 cases. Much less than 1% of the protection elements had a false trip and around 5% of the blocking elements had a false operation. These numbers are already considered satisfactory but they were analyzed below.

Looking at the time response of all functions, Fig. 8b shows that there are very few points above the second cycle of operation even for the worst case, which is the fifth harmonic blocking element. For all these cases, the pick-up occurred for just only one processing interval from the relay or from the model without any commitment of the protection results.

Analyzing the differences in the harmonic blocking elements it was notice that mainly for external faults sometimes there was a relay pick-up and sometimes a model pick-up. This variation in the pick-up shows no addiction from the model or from the relay. Although this behavior can be observed when testing two relays of the same manufacturer due to time processing differences related to the acquisition time and due to measurement errors they were analyzed more carefully. In all cases in which the model picked-up the second harmonic level, I2H2, was above the operation current, IOP2, as can be seen at Fig. 9. For these cases the second harmonic element did not achieved the minimum pick-up level. This difference can be explained because the relay has a minimum operation current, which probably varies according to each fault while the model was set to a fixed value. As mentioned before when working with models from manufacturers some differences are expected due to confidentiality of data, [6], and here there was no important impact in protection results.

Analyzing one case of false operation related to the trip elements, the 87HB bit from the model picked-up during an inrush with fault test. Although it is a different behavior, the pick-up happened almost when the circuit breaker opened and for all 1190 tests, the circuit breaker only opened after 200 ms, leaving the relay and model free to operate. Observing Fig. 10 it was concluded that this event would not happen in a regular test due to very large dead-time. Also the difference between the level of fourth harmonic, I3H4, and the operation current, IOP3, was less than 2%, which is inside any relay error.

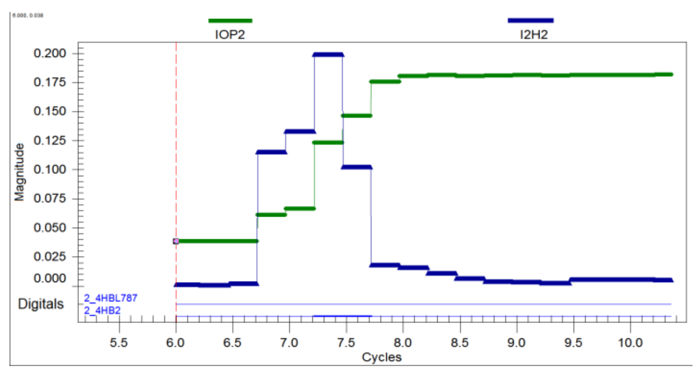

Fig. 9 – External Fault – 24HBL Example

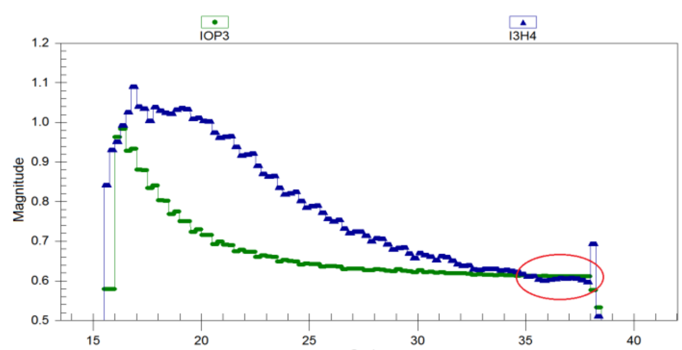

Fig. 10 – Forth Harmonic allows 87HB trip

B. Model x Built-in Model

Due to the simplicity of the built-in model, it was only possible to compare the trip and the second harmonic blocking signals.

Analyzing all tests, Fig. 11, the built-in model had a close behavior to the model, when comparing the trip signal pick-ups. However, in almost 3% of all internal faults the built-in model tripped and not the model. Looking at the second harmonic blocking element, this one presented much more false pick-ups, reaching in some test 50%. This different behavior is mainly because of the simplicity of the harmonic blocking element compared to the commercial relay.

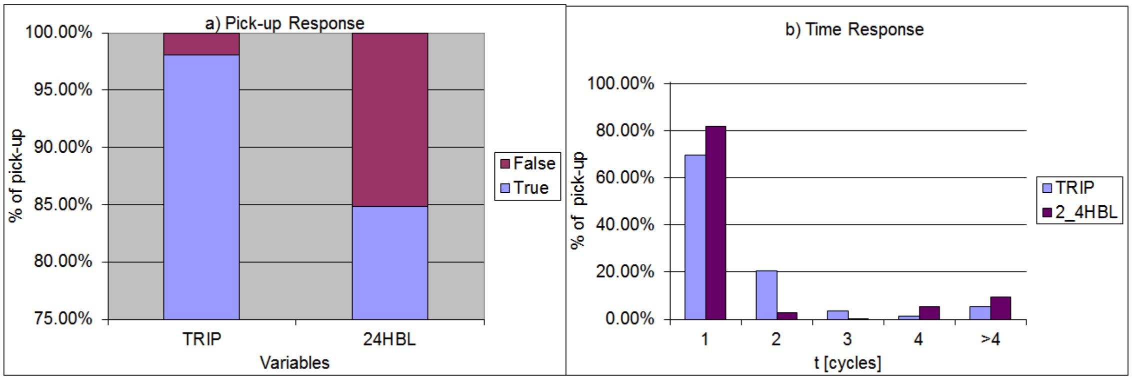

Fig. 11 –Model x Built-in Model Responses

Fig. 11 shows also the response time and the built-in model presented a much more scattered time responses. In general, the built-in model is much faster and this occurs because it does not reproduce the time processing of the actual relay. This means that for every real time simulator time step, 50 µs, the built-in model makes the whole calculation and additionally there is no output contact time.

A generic model applied in real time simulations as proposed in this paper would lead the engineers to wrong conclusions.

VII. Advantages of the Model

In a real relay, the users have information and tools, which were made available by the manufacturer. However in the proposed model every step can be followed and a complete understand of the relay and processing can be achieved. In this section some graphs are shown, which are only available in the model.

The real relay can give to the users the usual graphics as input currents or the final differential calculation. However, the model can give much more to the user. Fig. 12 shows the input signal, IAW1s, the output of the analog filter, IAW1lp, and the sampled signal, IAW1sp. Actually, a real relay does not show the input signal, but the signal after the sampling process, which is the third information, IAW1sp, of the model. From Fig. 12, for example, it is possible to see the analog filter delay and in a real relay, this part of the processing is analog and impossible to be available.

Another interesting signal is the output of the digital filter. As seen before after this filtering there are two information available: the real and imaginary part of the phasor. In a real relay, if available, the user has the magnitude of the phasor, but in the model, it is possible to plot each quantity individually.

Usually in a relay not much information is available about the harmonics, but in the model it is available the same information for harmonics as it is available for the fundamental signal. Fig. 13 shows the output of the second harmonic filter for an internal fault.

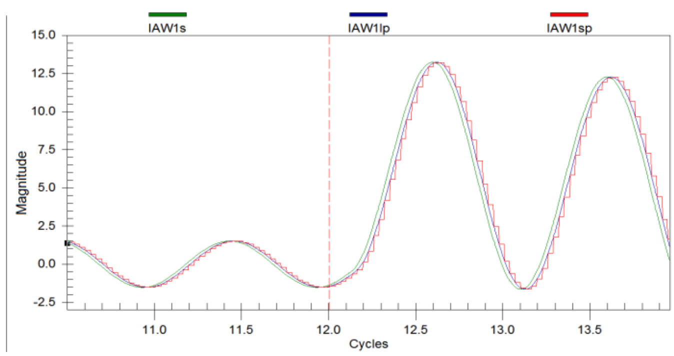

Fig. 12 – Input signal, low-pass and A/D Converter outputs

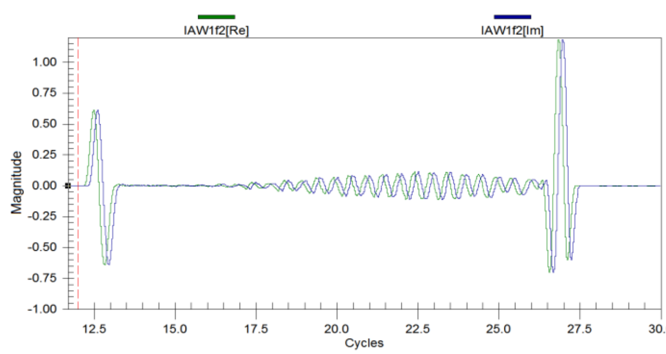

Fig. 13 – Output of the second harmonic filter during a fault

VIII. Conclusions

This paper presented a commercial differential transformer element model developed inside a real time simulator. The main aspects of the model implementation inside a real time simulator was exposed and the model performance was compared to a real relay and a built-in simulation model considering daily possible events for the differential element.

The main objective of building real relay models inside real time simulator is to allow the study not only the physical relay at the specific area under study but also the complete surrounding protection system modeled. The models can guide engineers to some specific problem that can happen with the expansion of a system and the necessity to modify the settings.

It should be emphasized that the model implementation has to be done with care to ensure the same response as the real relay regarding the time processing control and model of output contacts time.

The model, following a manufactured relay, presented a very close behavior to the real relay (hardware). There was less than 0.1% difference in the trip responses and some of them just happened due to the long circuit breaker dead-time and within measurements error. Almost 98% of all trips were at the first cycle of 60 Hz, which can be considered satisfactory as all the filter algorithms needs at least one cycle to reproduce the actual rms value. This means that the model has the same behavior compared to actual relay even during transient response of these filters.

The real relay has two types of differential elements and both algorithm in the model presented a very close similarity to the physical relay. The second harmonic blocking element had the farthest similarity but still with 92% of all results at the first cycle.

Based on the simulations it can be also concluded that built-in models can give engineers an idea about the behavior of the differential element, but it does not give a full support for the engineers to make a precise decision about a specific product. Specifically, just 60% of all trips of built-in model were at a difference of one cycle and there were some at more than four cycles. Usually these cases were at critical faults implying that it would be too risky to use the generic model to represent actual relays in simulations. There was 10% different pick-ups for the second harmonic blocking element and very different time response with almost 20% at more than four cycles. It was also noticed that there was a biased behavior of the built-in model in the pick-up differences.

Unlike the built-in model, when differences appeared in the proposed model, the pick-up sometimes was from the model and sometime was from the relay. These variation were related to few minimum value set to the sensor and to the intrinsic signal processing step. Looking at the time responses from both comparisons the difference between the two models is clear and the main reason is that the proposed model has the protection element at the same pace of the real relay.

It is therefore concluded that the presented relay model reproduces the physical relay with high fidelity and that it could be used in a real time simulation study to represent the surroundings relays, giving more reliability to the test.

It is also important to point out that the proposed model can give to the users more tools to analyze the protection algorithm behavior during a fault. In addition, these models could be used in initial protection tests while hardware is not available, and during dynamic analysis of power system and smart grid tests. An important advantage is that the model basic functions could be used as a platform to develop and test new protection algorithm, for educational and training purposes and even for relay pre-setting.

IX. Acknowledgment

The authors thank the relay manufacturer for all of their provided information.

X. References

[1] M. Sachdev, M. Nagpal, T. Adu, Interactive software for evaluating and teaching digital relaying algorithms, IEEE Trans. Power Systems. 5 (1990), 346-352.

[2] P. McLaren, E. Dirks, Using a real time digital simulator to develop an accurate model of a digital relay, Digit. Power Syst. Simulators, 1995, ICDS '95., First Int. Conf. (1995) pp. 173–178.

[3] P.G. McLaren, K. Mustaphi, G. Benmouyal, S. Chano, a. Girgis, C. Henville, et al., Software models for relays, IEEE Trans. on Power Delivery. 16 (2001) 238–245.

[4] S. Pérez, Modeling relays for use in power system protection studies, Can. Conf. Electr. Comput. Eng. (2005) 566–569.

[5] D. Ouellette, W. Geisbrecht, R. Wierckx, P. Forsyth, Modelling an impedance relay using a real time digital simulator, Dev. Power Syst. Prot. (2004) 665–668.

[6] J. C. Peqqueña-Suni, J.A. Martinez-Velasco, J. Mahseredjian, O. Saad, E. Ruppert, Real-Time Implementation of a Fault Location Algorithm for Homogeneous Systems, International Conference on Power Systems Transients - IPST. (2013) 1–6.

[7] S. Zocholl, A. Guzmán, D. Hou, Transformer modeling as applied to differential protection, 22nd Annu. West. Prot. Relay Conf. (1996) 108–114.

[8] J. Thorp, A. Phadke, A microprocessor based three-phase transformer differential relay, IEEE Trans. on Power Apparatus and Systems (1982) 426–432.

[9] K. A. Tavares, K. M. Silva, On modeling and simulating the differential protection of power transformers in ATP, 11th IET Int. Conf. Dev. Power Syst. Prot. (DPSP 2012), pp. P100–P100, 2012.

[10] C37.91 - 2008, IEEE Guide for Protecting Power Transformers, 2008.

[11] C. Apostolopoulos, G. Korres, Real-time implementation of digital relay models using MATLAB/SIMULINK and RTDS, Eur. Trans. Electr. Power. (2010) 290–305.