Artificial Intelligence Techniques for Managing the Massive Amounts of Data from Digital Relays

By:

Amir Makki
Softstuf
Philadelphia, PA

Maria Rothweiler
Softstuf
Philadelphia, PA

Mark Taylor
Softstuf
Ashville, AL

Tony Giuliante
ATG Consulting
Galloway, NJ

Presented before the

34th Annual
Western Protective Relay Conference
Spokane, Washington
October 16-18, 2007

Artificial Intelligence Techniques for Managing the Massive Amounts of Data from Digital Relays

Amir Makki, Maria Rothweiler, Mark Taylor, and Tony Giuliante

Abstract

Advances in technology over the past decade have changed the way relays are designed in two fundamental ways. First, the high speed embedded systems technology brought us much more powerful digital relays. These relays can and do create massive amounts of data, have plenty of storage, many analog and digital channels, lots of useful measurements and calculations, Ethernet and serial connectivity too. Next came the communications technology wave and all of a sudden we can easily collect data from a large number of relays, in real time, and from a single workstation.

Clearly, “data overload” is a visible downside. On a mid sized power system, expect a few billion bytes of data each month. It is impossible to manually examine every available byte trying to discover where the trouble spots are. The object of this paper is to show how artificial intelligence (AI) can be used to help users manage such massive amounts of data (i.e. collect, discern, sort, visualize, analyze, etc.). After WWII the British Royal Navy declared: “we can’t be everywhere all of the time, but we can be anywhere at anytime”. Same thing here, we can’t manually study and analyze all of the data, but we can use AI to help the experts locate the trouble spots.

The following is a description of the various types of AI techniques used to organize, prioritize, and classify transient and steady state conditions. The results from a 2 year case study, covering 100 circuits in 5 substations, are also presented, and the benefits of using the IEEE C37.111 and C37.232 standards for building content addressable systems are highlighted. The main intent is for the paper to serve as a tutorial on how to use artificially intelligent techniques for addressing data overload problems.

Introduction

In the realm of software there are 2 known types of AI programs. The two types are: expert systems, and adaptive learning systems. Expert systems are used to logically organize vast amounts of information and answer questions and make conclusions just like a human expert does. An expert system is basically a very smart database. The user trains the system by specifying, in some English like language, specific rules and methods to govern the analysis process.

On the other hand, adaptive learning systems such as neural networks, adaptive filters, and pattern recognizers, have nothing to do with databases. These are intelligent systems that model or simulate the adaptive learning process using mathematical equations with time varying coefficients. The user trains the system by allowing the coefficients to vary depending on the performance of the system and its ability to match the user's desired output.

For the purposes of this application, managing data from digital relays, a hybrid expert/adaptive system is needed to properly classify circuit conditions as being "good" or "bad". The expert system is needed in order to automatically collect and logically organize the vast amounts of data recorded by modern day digital relays. The organization is usually based on company name, division, station, circuit type and name, and on time of occurrence. The adaptive learning system, on the other hand, is needed to do the actual good/bad classifications (such as assigning a priority value to each event). Examples of the above are provided in the next two sections and the balance of the paper is focused on the application of these techniques.

Adaptive Example

An example bare bones diagram of an adaptive learning system is shown in Figure-1. The diagram is of a system used to study, weigh, and reconsider multiple conditions in real time (such as imbalance, overload, and inefficiency). The applied weights are adjusted using a predefined cost function that is designed to force the error signal down to zero. The cost function is removed once the system is tuned (fully trained).

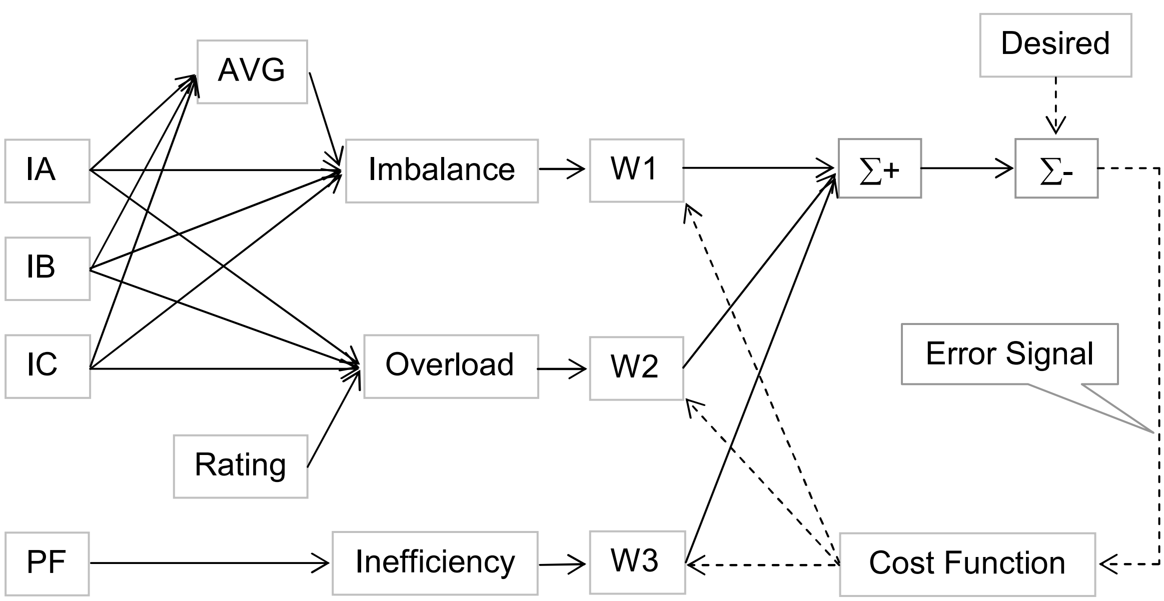

Figure-1; Adaptive example using weighted summations and cost functions

The definitions for the terms being used in the above figure are listed here below:

IA = phase-A current input (Amps RMS),
IB = phase-B current input (Amps RMS),
IC = phase-C current input (Amps RMS),
AVG = average of Phases A, B, and C inputs,
Rating = normal current rating input (Amps RMS),
PF = power factor input,
Imbalance = maximum imbalance output (max % change from average),
Overload = maximum overload output (max % change from rating),
Inefficiency = maximum inefficiency output (max % change from unity),
W1 = coefficient multiplier for the imbalance output,
W2 = coefficient multiplier for the overload output,
W3 = coefficient multiplier for the inefficiency output,
∑+ = summation node (good/bad circuit classification branch),
∑- = subtraction node (error signal generation branch),
Desired = input signal used to train the network, and
Cost function = adjusts the coefficients to force the error signal to zero.

Expert Example

An expert system is composed of three main components: a database, a set of rules, and an inference engine. The operations of the expert system are mostly controlled by the inference engine. The typical inference engine operates as follows: it reads the data from the database, extracts the key features, and then checks these key features against the list of current rules. If rules are violated, then the inference engine “fires” alarms and warning messages. If new rules are formulated (as a result of “previous experience”) then the inference engine adds them to the list of current rules. The effort needed to develop an expert system depends on the complexity of the intended application. An expert system to manage data from digital relays is a complex undertaking that requires expertise in integration, communications, collection, storage, visualization, and analysis. The following is a little flavor of what is involved at each level:

Integration**:** Digital relays have so far been integrated using a diverse array of wide area networking technologies, old and new, such as phone switches, port switches, leased lines, Ethernet switches, data concentrators and so on. In order to access digital relays and retrieve their data, an expert system must be able to work with all of these technologies. Accordingly, the expert system must be installed on a specialized computer having multiple types of communication ports capable of addressing star and multi-drop topologies.

Communications**:** Digital relay developers have created too many types of protocols using too many different types of operating nuances. These developers had to create their own protocols because of the lack of an industry wide standard that can deal with the complexities of digital relays. Accordingly, the expert system must be able to communicate using different types of protocols such as Modbus, DNP, IEC61850, C37.118, Zmodem, FTP, TCP, UDP, Telnet, VT terminal, and so on.

Collection**:** In order to collect the latest data from the digital relays, the expert system must poll them on a periodic basis. The polling period may vary from microseconds to years depending on the type of data being retrieved (such as instantaneous values, transient fault records, load values, event sequences, relay settings, and so on). Relays that form a multi-drop topology are polled in order (one at a time), and those that form a star topology are polled simultaneously (all at the same time).

Storage**:** Unfortunately, for this application, storage is the most complex part of the system because there are too many types of data storage formats and filing conventions in circulation today. It is practically impossible to house all of these different types of formats and conventions in the same database. Accordingly, C37.111 and C37.232 were developed by IEEE to address these problems. The concept behind these standards is to have a common method for formatting and filing data. The expert system must be able to discern all of the collected data, in their original proprietary form, and convert them to the noted standards in order to house this data in a common and structured way. The results are usually saved to a shared location on the company network called the "repository" where immediate, concurrent, and secure access is already available. The typical size of a repository will vary from as low as a few mega bytes to as high as a few terra bytes depending on the selected poll period and on the total number of digital relays being polled.

Visualization**:** This is the most important component of the system. It is imperative that the user be able to verify, validate, and study the accuracy of any conclusions and/or deductions made by the expert system. The problem is that we can't manually study each byte used by the system in making its decisions because there is too much data involved (that's why we needed an expert system in the first place). Accordingly, a "good" expert system must have an array of creative displays for presenting tremendous amounts of data in a single screen. For example, Figure-2 shows all of the data from one circuit for the 2003 spring and summer seasons (15 minute readings starting at the top of the circular charts and continuing in a clockwise direction). The left chart shows an overlaid plot of Mega Watts (brown) and Mega Vars (green), and the right chart shows an overlaid plot of the current channels IA, IB, IC, Rating and AVG (blue, brown, green, purple, and red respectively). Clearly, a good efficiency condition, as visualized from the left chart, is a small green dot in the center with a big brown ring around it. A good balanced condition is when the right chart has only one thick red ring in it. Also, a good current condition (meaning a no overload condition) is when the purple circle in the right chart (the Rating) completely encircles all of the current traces.

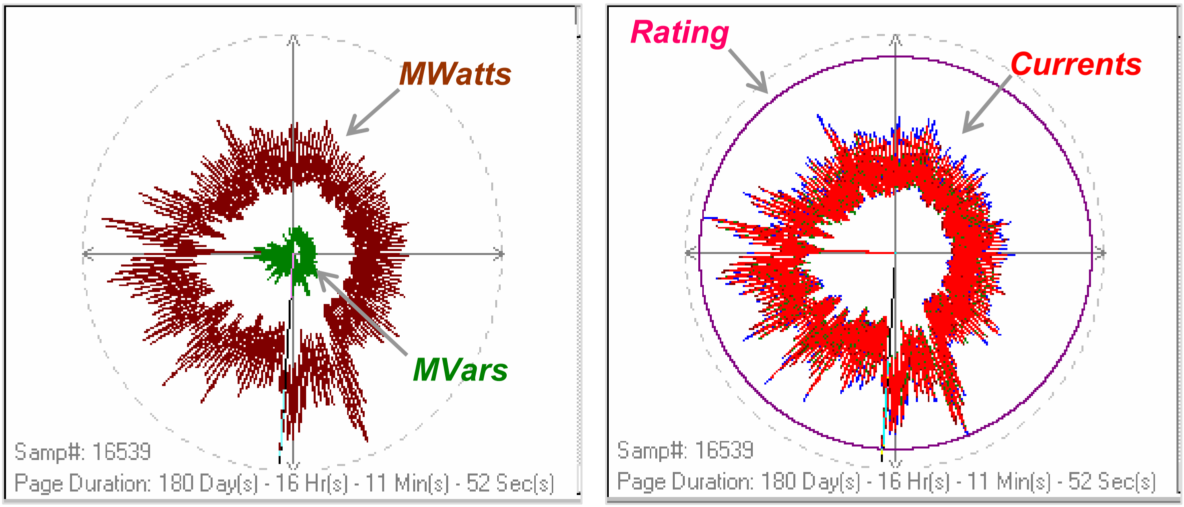

Figure-2; Visualizing 6 months of load information (spring and summer)

Another example is shown in Figure-3. It provides a visualization of a breaker's performance during a trip and close operation. The duration is for 10 seconds at 1920 Hz per channel (80,000 measurements displayed). As seen in this figure, once the trip coil was energized, the air compressor kicked in and bounced for a few cycles. Clearly, it is important to have visual confirmation before issuing a maintenance order to go visit the substation.

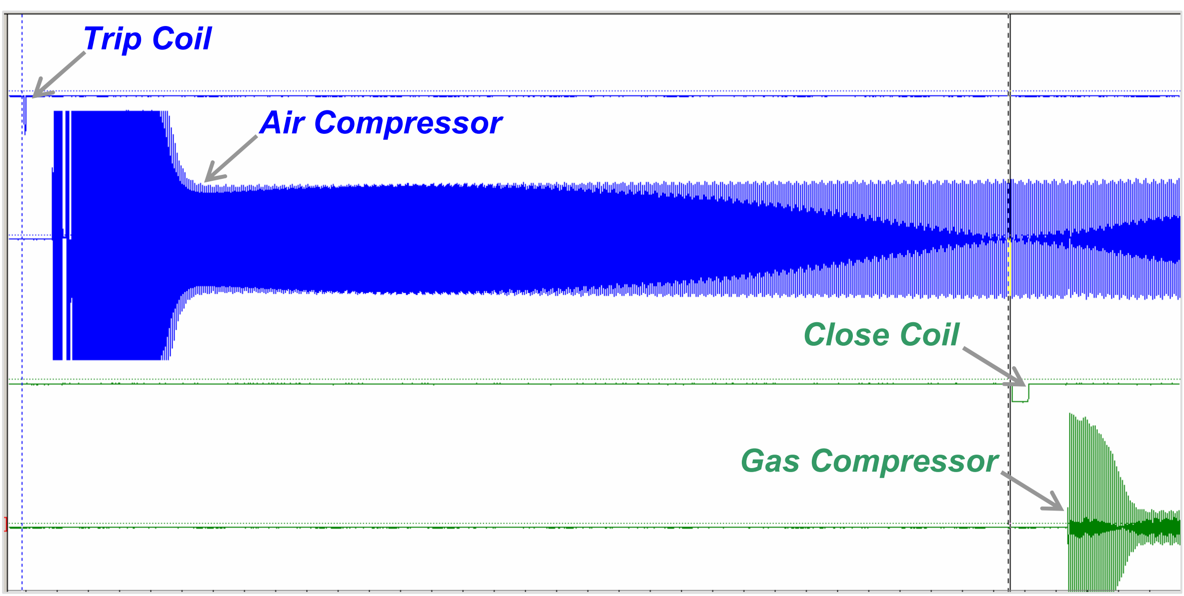

Figure-3; Visualizing 10 seconds (1920 Hz) of breaker trip/close operations

***Analysis*: **Sometimes a simple visual inspection may not be good enough to validate the claims, alarms, and warning messages being generated by the expert system. For example, the expert system, in making its decision, may have had to calculate a missing phase, or a harmonic, or an envelope, or a sequence component. Depending on the preference of the developers, such calculations may or may not appear in the "default" displays. So it is important for the expert system to provide the user with manual access to all of the available analysis tools along with the capability of individually visualizing each result.

The Experiment

The following is a brief description of an AI experiment that was conducted over a period of 4 years, from 2002 to 2005. In 2002 an expert system was installed and commissioned in 5 substations which included about 100 circuits. By the year 2005 the system had collected more than a billion bytes of periodic load measurements (at 15 minute intervals). When the system planners sat down to chart this data in order to find the worst performing substations, they were quickly overwhelmed even though they were only considering the five substations, and even though the expert system had already organized the measurements in compliance with the IEEE C37.111 and C37.232 standards. The reality is such that the mere thought of having to open and chart 100 different files is simply overwhelming in itself. Now imagine if we had a system with hundreds of substations and thousands of circuits. The task would be monumental. Hence the decision was made to add an adaptive engine on top of the expert system to overcome this level of data overload.

A small adaptive network composed of filters, rules, and methods collectively aspiring to classify circuit behavior was developed and trained in accordance with Figure-1 as shown above. The over one billion bytes of collected data were then automatically processed by the adaptive system. The processing duration was about a minute for each circuit. The adaptive system successfully spotted, ranked and classified all of the abnormal conditions in each circuit (abnormal meaning either imbalanced, or overloaded, or inefficient, or a combination thereof). The performance was repeatable, and the time needed to visually verify the results was amazingly short. The system used the highest rank to pick out the worst performing circuit, and used the highest averaged rank per substation to pick out the worst performing substation. Needless to say, the planners were very happy with these results.

The results for the 5 substations are shown in Figure-4. The figure shows the weighted sum of the calculated imbalance, overload, and inefficiency conditions for each circuit (based on the percentage, level, and duration of each condition). The weighted sums are shown under the priority column and are used to rank the performance of each circuit. The maximum possible priority is 999 and the minimum priority is 0. Of the 5 substations studied, the worst performing one was substation number 5 (with a rank of 222), and the worst performing circuit in that substation was circuit number 4009 (with a rank of 459).

-	Imbalance			Overload			Inefficiency			Priority
-	(Trigger	MaxAmps	Dur)	(Trigger	Rating	Dur)	(Trigger	AvgAmps	Dur)	(Rank)
* CUMULATIVES										
STN5 #54 :	029%	0591A	148D	007%	1098A	001D	038%	0538A	070D	0222
STN3 #21 :	020%	0603A	073D	012%	1060A	003D	014%	0555A	046D	0171
STN2 #52 :	017%	1192A	098D	001%	0893A	000D	023%	0542A	062D	0129
STN4 #39 :	009%	0572A	097D	005%	1145A	002D	013%	0533A	058D	0110
STN1 #49 :	013%	1388A	079D	002%	1097A	000D	013%	0841A	043D	0077
System Averages:	017%	0869A	099D	005%	1058A	001D	020%	0601A	055D	0141
* STN5 #54										
4009 #13 :	063%	0295A	255D	000%	0442A	000M	083%	0232A	249D	0459
4008 #12 :	068%	0304A	077D	000%	0378A	000M	089%	0209A	244D	0424
4001 #5 :	030%	0297A	247D	000%	0351A	000M	089%	0275A	151D	0417
4006 #10 :	053%	0290A	075D	000%	0294A	000M	086%	0238A	196D	0397
4005 #9 :	015%	0226A	022H	000%	0294A	000M	089%	0217A	197D	0371
4013 #17 :	026%	0521A	284D	022%	0464A	001D	089%	0443A	003D	0319
4010 #14 :	041%	0399A	167D	000%	0464A	000M	058%	0348A	165D	0314
4011 #15 :	030%	0552A	275D	029%	0464A	011D	000%	0473A	000M	0292
4012 #16 :	017%	0578A	005D	016%	0545A	001D	068%	0543A	001D	0217
4002 #6 :	039%	0464A	315D	015%	0442A	001D	000%	0406A	000M	0137
4003 #7 :	034%	0498A	316D	017%	0464A	004D	000%	0427A	000M	0136
4004 #8 :	032%	0519A	262D	017%	0485A	021H	000%	0458A	000M	0131
4014 #18 :	016%	0502A	102D	013%	0485A	002D	000%	0472A	000M	0086
4007 #11 :	041%	0249A	149D	000%	0290A	000M	000%	0190A	000M	0081
XFMR-3 #3 :	000%	1449A	000M	000%	4272A	000M	000%	1409A	000M	0000
XFMR-2 #2 :	000%	1462A	000M	000%	4272A	000M	000%	1406A	000M	0000
XFMR-1 #1 :	000%	1448A	000M	000%	4272A	000M	000%	1406A	000M	0000
STN5 Averages :	029%	0591A	148D	007%	1098A	001D	038%	0538A	070D	0222

Figure-4; Average ranks and the worst performing substation (and its circuits)

On the other hand, the worst performing circuit, from the entire batch of 100 circuits, was circuit number 4010 in substation number 3 (rank = 646) as shown in Figure-5.

-	Imbalance			Overload			Inefficiency			Priority
-	(Trigger	MaxAmps	Dur)	(Trigger	Rating	Dur)	(Trigger	AvgAmps	Dur)	(Rank)
* STN3 #21										
4010 #16 :	020%	0492A	006D	064%	0319A	011D	028%	0416A	170D	0646
4006 #13 :	030%	0600A	271D	034%	0480A	013D	017%	0514A	005D	0370
4008 #9 :	045%	0299A	155D	020%	0272A	003D	034%	0275A	172D	0305
4003 #11 :	060%	0410A	195D	004%	0415A	014H	041%	0398A	147D	0300
4005 #21 :	000%	0481A	000M	037%	0379A	017D	000%	0448A	000M	0295
4012 #19 :	000%	0561A	000M	029%	0471A	006D	021%	0549A	024D	0275
4004 #10 :	062%	0461A	186D	019%	0425A	002D	000%	0357A	000M	0198
4007 #12 :	029%	0278A	163D	000%	0272A	000M	028%	0245A	231D	0170
4015 #15 :	000%	0478A	000M	021%	0404A	017D	000%	0448A	000M	0167
4011 #20 :	023%	0332A	052D	000%	0415A	000M	025%	0318A	130D	0123
4014 #7 :	000%	0368A	000M	000%	0415A	000M	055%	0351A	001D	0110
4001 #22 :	045%	0307A	266D	000%	0372A	000M	000%	0271A	000M	0091
4002 #17 :	029%	0329A	007D	000%	0440A	000M	017%	0322A	005D	0063
4013 #18 :	032%	0297A	061D	000%	0337A	000M	014%	0261A	001D	0060
4016 #8 :	019%	0371A	043D	009%	0376A	020H	000%	0363A	000M	0054
4009 #14 :	000%	0349A	000M	006%	0351A	019H	000%	0339A	000M	0026
XFMR-3 #26 :	000%	1619A	000M	000%	4668A	000M	000%	1524A	000M	0000
XFMR-2 #25 :	000%	1755A	000M	000%	4668A	000M	000%	1585A	000M	0000
XFMR-1 #24 :	000%	1674A	000M	000%	4668A	000M	000%	1565A	000M	0000
STN3 Averages :	020%	0603A	073D	012%	1060A	003D	014%	0555A	046D	0171

Figure-5; Assigned ranks and the worst performing circuit

As seen in the above two figures, each circuit has 3 sets of calculated values (imbalance, overload, and inefficiency). The sets form a 3-D space providing a vector depiction of the circuit conditions. This type of representation provides an alternative way for assigning priority values based on distance to origin and angle of inclination. The angle of inclination is needed in order to provide a distinction between single and multiple conditions.

Visual Confirmation

The following provides the needed visual confirmations, along with some comments, for each of the “worst” circuits mentioned in the previous section. First to be considered is circuit 4009 (substation 5) because it was ranked the worst circuit of the worst substation. Figure-6 shows a chart of the circuit’s data for spring and summer of 2003. The outputs from the adaptive system were:

Imbalance:	63% for 255 days
Overload:	0% for 0 hours
Inefficiency:	84% for 249 days
Summer rating:	442 amps
Maximum average:	232 amps
Maximum current:	295 amps
Priority value:	459 out of 999

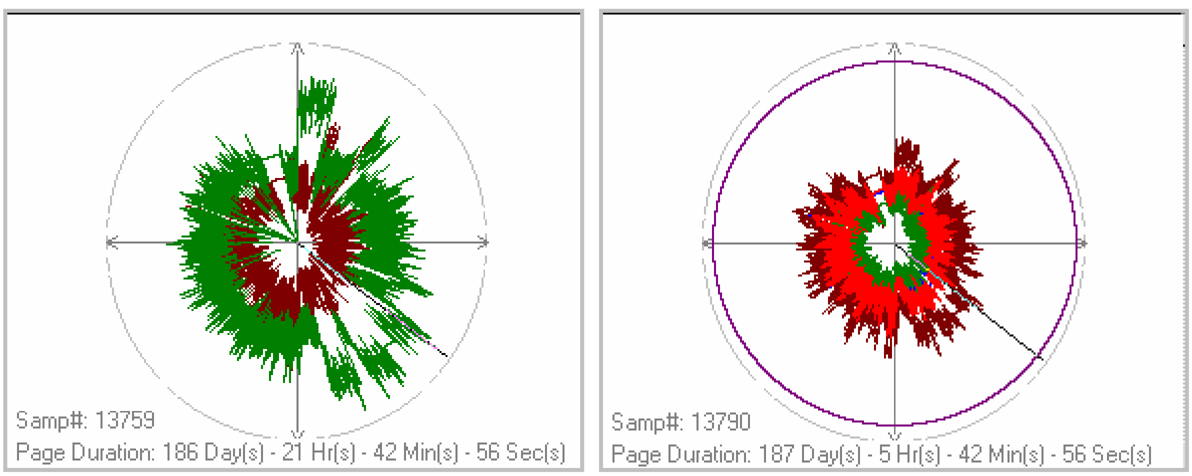

Figure-6; 2003 spring and summer chart for circuit 4009 (substation 5)

A quick inspection of the charts shows that the brown ring in the left chart is totally enclosed by the green ring (which is a severe inefficiency condition); and, the red ring in the right chart is now sandwiched by 2 other rings (which is a severe imbalance between the green and brown phases); but, the purple circle in the right chart is positioned at the boundary far away from the current traces (which means there is no overload condition and that the circuit has room for additional load). Clearly, the circuit was operating at about half capacity but still got a high priority! The imbalance and inefficiency discoveries are high but are only at 50% of maximum load. The priority should have been less. There must be some other reason that is causing the adaptive system to magnify the priority. To study further, the circuit's data for the next year (spring and summer 2004) was then charted and reviewed (see Figure-7).

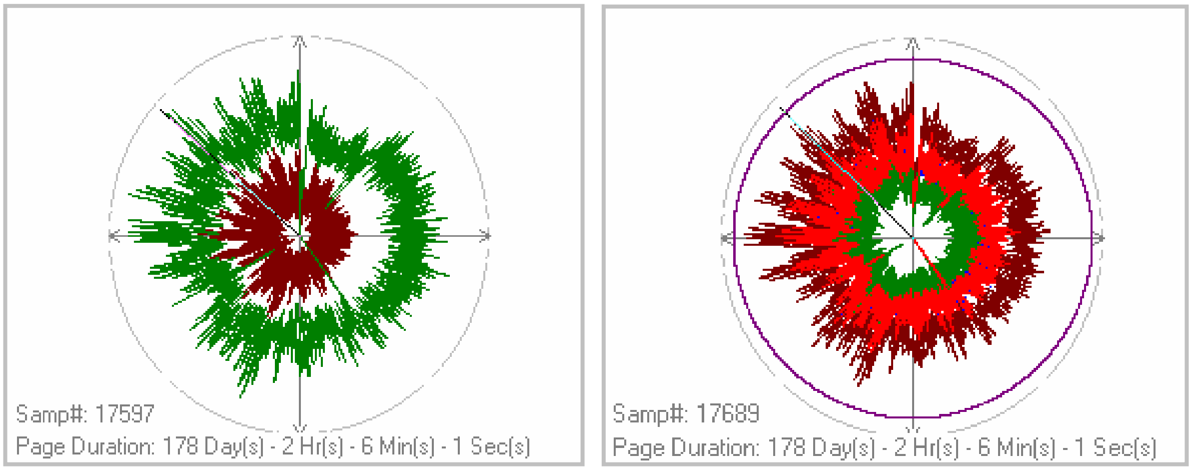

Figure-7; 2004 spring and summer chart for circuit 4009 (substation 5)

Unlike 2003, the circuit is now operating very close to capacity. Therefore, we have a growing or evolving load condition which explains why the priority assignment was so magnified. At this point the system planners felt comfortable issuing a "work order" because the conditions of this circuit were bad in 2003 but they got much worse in 2004. Clearly, this type of verification process is very important when dealing with AI systems.

As for the "worst" circuit (out of the 100 circuits from the 5 substations considered in this experiment) it was circuit 4010 in substation 3. Figure-8 shows a chart of the circuit's data from spring and summer of 2004. The outputs from the adaptive system were:

Imbalance:	20% for 6 days
Overload:	64% for 11 days
Inefficiency:	28% for 170 days
Summer rating:	319 amps
Maximum average:	416 amps
Maximum current:	492 amps
Priority value:	646 out of 999

A quick inspection of the right chart shows that the purple circle has now moved almost half way into the middle of the chart (which indicates a severe overload condition). The chart also shows that the red ring is dominant indicating no significant imbalance conditions. The left chart shows that the inefficiency condition improves as the load increases. Cleary, the circuit was operating far above capacity and therefore produced the highest priority value issued by the system. This outcome is understandable because overload conditions should naturally be assigned to the highest weights. A "work order" to shift the load was immediately issued.

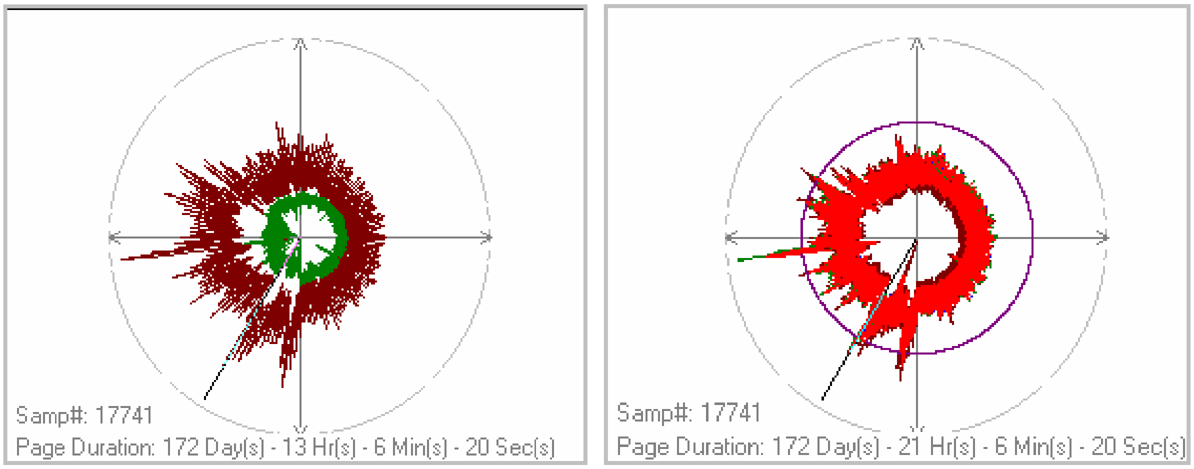

Figure-8; 2004 spring and summer chart for circuit 4010 (substation 3)

Remarks and Conclusions

This experiment is solid proof that AI can be used to analyze power system data and automatically identify the potential trouble spots. It is not feasible to manually study and analyze all of the data, but we can use AI techniques to help us locate trouble spots and verify conditions. For the record, this entire experiment was designed around the IEEE C37.111 standard (Comtrade) and was based on the IEEE C37.232 standard (Naming Convention for Time Sequenced Data Files). This level of simplicity and commonality, as shown in this paper, would not have been possible without these standards. The C37.232 standard was especially helpful in simplifying the database addressing schemes by making them "content addressable" thus enabling random access at very high speeds.

It is important at this point to mention that AI systems are here to assist the "experts" during the decision making process. AI systems are not designed for use by the non-experts because the user must be able to understand and verify the outputs of an AI system before acting on them. AI is a byproduct of human knowledge and will only do what it was trained to do. When faced with "out of the norm" exceptions (something that only a human mind can comprehend), AI systems will fail and will produce the wrong type of diagnostics which could lead to, or directly cause, a major injury such as "cutting off the wrong leg". Accordingly, we should never become comfortable with being dependant on and trusting of such technologies, we must always verify.

References

"Naming Convention for Time Sequence Data File (ComNames Convention)," IEEE, PES, PSRC Standard C37.232, 2007

"Synchrophasors for Power Systems (Synchrophasors)," IEEE, PES, PSRC Standard C37.118, 2006

“Database Principles, Programming, and Performance,” Patrick O’Neil and Elizabeth O’Neil, Academic Press, New York, 2001

"Common Format for Transient Data Exchange for Power Systems (Comtrade Format)," IEEE, PES, PSRC Standard C37.111, 1999

"Survey of Event File Saving Schemes," Makki, Taylor, & Semati, Proceeding of the Fault and Disturbance Conference, Atlanta, Georgia, May 1999

"Survey of Event File Naming Schemes," Makki and Taylor, Proceeding of the Fault and Disturbance Analysis Conference, Atlanta, Georgia, May 1998

“A Guide to Commercial Artificial Intelligence,” Rauch-Hindin, Prentice Hall Publishing Company, 1988

“Pattern Recognition Principles,” Tou and Gonzalez, Addison-Wesley Publishing Company, 1974

Biographies

Amir Makki is chairman and cofounder of Softstuf, Inc., Philadelphia, PA (founded in1991). His professional contributions include over 40 publications, 5 copyrights, 3 patents, and 2 trademarks. He holds BS and MS degrees in Electrical Engineering from Tennessee Tech University, and from 1986 to 1991, he was a presidential fellow at Temple University helping develop their first Ph.D. program in Software Engineering. Amir is a member of the HKN honor society, the IEEE Standards Association, and the Protection Systems Relay Committee of the Power Engineering Society.

Maria Rothweiler is president and cofounder of Softstuf, Inc., Philadelphia, PA (founded in1991). Her professional contributions to date include over 25 publications, copyrights, and trademarks. From 1984 to 1986, Maria attended Bucks County Community College where she prepared for her associates degree in mathematics, and from 1986 to 1990, Maria attended Temple University where she earned her BA degree in computer science. Maria is a member of the IEEE and a member of the IEEE Standards Association.

Mark Taylor is the chief engineer at Softstuf, Inc. and is based in Asheville, Alabama. Mark has over 25 years of experience in fault analysis and data collection. From 1982 to 2001, Mark worked at Alabama Power Company as a principle engineer supervising their fault data group. He is a frequent contributor to the Fault and Disturbance Analysis conference and has provided many presentations on communications, analysis and automation. Mark holds a BS degree in Electrical Engineering from the University of Alabama (in 1982), and he is an active member of the IEEE, the IEEE Standards Association, and the Protection Systems Relay Committee of the Power Engineering Society.

Tony Giuliante is president and founder of ATG Consulting. Prior to forming his company in 1995, Tony was Executive Vice President of GEC ALSTHOM T&D Protection and Control Division, which he started in 1983. From 1967 to 1983, he was employed by General Electric and ASEA. In 1994, Tony was elected a Fellow of IEEE for “contributions to protective relaying education and their analysis in operational environments.” He has authored over 50 technical papers and is a frequent lecturer on all aspects of protective relaying, including electromechanical, solid state and digital based equipment. Tony is a past Chairman of the IEEE Power System Relaying Committee 1993-1994, and past Chairman of the Relay Practices Subcommittee. He has degrees of BSEE and MSEE from Drexel University 1967 and 1969.