A HOLISTIC APPROACH TO ACCELERATOR RELIABILITY MODELING

M. Reščič *, Rebecca Seviour †, University of Huddersfield, Huddersfield, UK
W. Blokland ‡ SNS, Oak Ridge National Laboratory, Oakridge, USA

Abstract

Reliability has been identified as a key factor limiting the development of certain particle accelerator applications, for example Accelerator-Driven Systems (ADS) for energy production and waste-transmutation. Previous studies of particle accelerator reliability have been undertaken using conventional techniques, such as Reliability Block Diagrams (RBD), Fault Tree Analysis (FTA), etc. Although limited data surrounding components and their failure modes limits the applicability of conventional techniques for analysing the reliability of particle accelerators. In addition industrial applications of particle accelerators, i.e. energy production, require a real time response to failure. In this paper we examine a holistic approach to accelerator reliability modelling using Electric Network Frequency (ENF) criterion to look for emergent behaviour of the particle accelerator, from complex datasets, such as beam current/charge, created by the diagnostics systems during the machines operation. To look for predictive characteristics just prior to a machine trip.

INTRODUCTION

The Electric Network Frequency (ENF) criterion is a procedure from audio forensics [1] where an audio recording's autheticity is validating by extracting low frequency mains hum and trying match it to a reference database. If a match is found the sample is determined as authentic (not tampered with, taken at the presented date and time), if matching fails the sample can be considered tampered with and not authentic. The success of the matching algorithm can vary depending on the sample size and analysis method, as described in [2]. Simplified, the procedures can be reduced to the following steps:

1. continuously record and save a reference database of mains electrical hum
2. from a given audio sample extract the low frequency hum
3. try to match the sample to the database using different techniques (e.g. root-means squared)
4. if match is successful the date and time of sample are confirmed

In this paper we propose an analogous approach to determine the *type* of accelerator pulses. Instead of determining the time the pulse occured we are interested if we can establish and determine if a given pulse is preceding a machine trip. Focus of the paper is to describe and present result on:

- the type of data used to create a reference database, as well as the associated procedures
- matching method parameter estimation
- the matching algorithm with two different matching methods
- conclusions from the results of the matching

* miha.rescic@hud.ac.uk
† rebecca.seviour@hud.ac.uk
‡ blokland@ornl.gov

DATA AND DATABASES

Mains hum is the sound associated with alternating current at the frequency of the mains electricity. Depending on the country and it's power-line frequency, the fundamental frequency of this sound is usually 50 Hz or 60 Hz. An ENF database then consists of a continuous recording of this signal for a given country, see [1] for example. One suitable implementation is to store the sampled signal as a series of waveforms with associated timestamps. Figure 1 shows a sample signal from such a database.

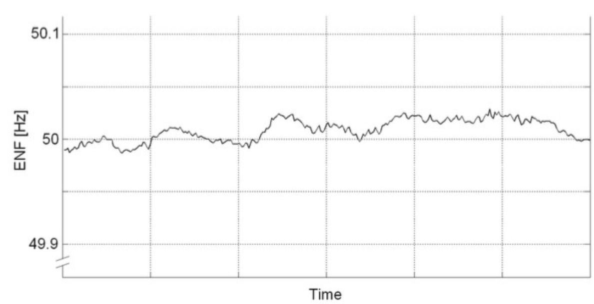

Figure 1: ENF Signal

Therefore, the question is if similar signal can be observed with pulsed particle accelerators. Our proposed approach is to use the measured beam current from the SNS accelerator, as described by [3].

The characteristics of an acquired single pulse waveform are: 25.000 datapoints recorded at the speed of 100 MHz, each waveform representing 25 μs machine time, see Fig 2.

Acquired waveforms represent three unique types of signals - the last normal pulse before a machine trip (refered to as **Previous** pulse), the off-normal, tripped pulse and the first successful pulse after the accelerator recovered from the off-normal fault (refered to as the **Next** pulse). For our purposes, we decided to create two different databases:

ISBN 978-3-95450-147-2

- Database composed of last *normal* pulses (the last pulses considered normal before the pulse that was registered as off-normal (i.e. caused machine trip) was generated)
- Database composed of first pulses considered normal again after machine recovered successfuly from fault

Signal extraction

One can observe in Fig 2 that the trailing part of the measurement is zero, or very close to zero. This region of the waveform is common to all measurements and will be for our purposes omitted, since it does not differ from one measurement to another. The initial part of the measurement on the other hand has very distinct particle bunch-by-bunch measurements of beam current, see Fig 4 for zoomed region of the measurement.

The first step in creating the database is to extract the non-zero region of the signal. The region of interest is determined by a simple algorithm:

if more than 10 *consequtive samples are above the threshold of* 0.25 mA *mark the start of the region of interest. If more than* 50 *consequtive samples are below the threshold mark the end of the region of interest.*

An extracted datasample can be seen in Fig 3. Described algorithm extracts on average about 14 500 to 15 000 datapoints.

Database creation

The databases are created by concatenating respective extracted signal vectors into a larger, continous signal vector. The threshold of 0.25 mA in the extraction algorithm was chosen after experimentation to remove the trailing zeros and the initial ramp-up of the signal creating strong discontinuty in the concatenated signal.

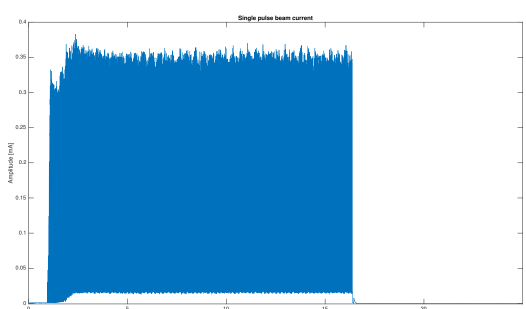

Figure 2: Single pulse signal

MEASURING VECTOR LIKELINESS AND VECTOR MATCHING

If we are to determine if a certain signal matches a database or not, we need to define a measure of how different

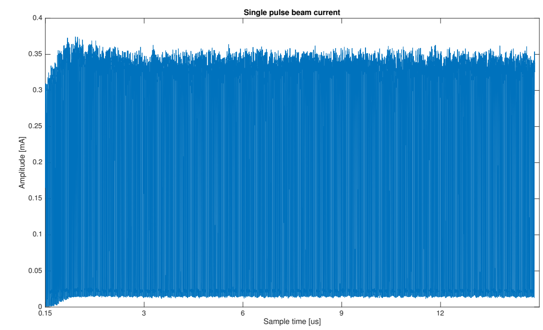

Figure 3: Extracted single pulse signal

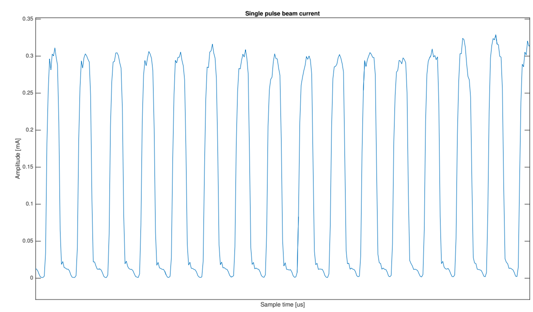

Figure 4: Single pulse signal, zoomed

(or similar) two vectors are. For ENF criterion [1] proposes two methods:

- Square root of summed squared difference between two vectors, i.e. root mean squared (RMS) between two vectors x and y of length L, defined as

$$RMS(x,y) = E_{rms} = \sqrt{\frac{\sum_{i=1}^{L}(x[i]-y[i])^2}{L}} \tag{1}$$

  The smaller the value, the more two vectors are alike (E_{rms} for identical vectors equals0).

- Correlation coefficient (CC) between two vectors x and y of length L, defined as

$$CC(x,y) = \frac{\sum_{i=i}^{L}(x[i]-\bar{x})(y[i]-\bar{y})}{(L-1)\sigma_x\sigma_y} \tag{2}$$

  where $\bar{x}, \bar{y}$ represent vector averages and σ_x, σ_y standard deviations respectively. The coefficient has values in the range $\rho \in [-1, 1]$, the closer to 1, the more the two vectors are similar (i.e. stronger correlation).

Matching process

To determine if a certain vector is the same type as the database we compare it to, we need to define thresholds that would decide this. Ideally using the RMS method, two vectors are alike if RMS between the two equals 0 But is it acceptable if error is 0.1 ? To get an estimation of such

[0] It is to be explored what is the appropriate amount of signal that should be removed in order to create smoother transitions in the database vector or if some other method should be applied

errors we've created two database from roughly 200 pulse samples, each database only consisting of either *previous* or *next* pulses. Then we took 100 000 random, non-overlapping samples of from each of the database and calculated RMS and CC values for them respectively. We repeated measurements for vector lengths from 100 to 15 000

Results are summarized in Table 1 and Table 2. Observed values give us an estimation of what could be considered 'random noise' in the database (approach is analogous to threshold estimation in [1]). Our assumption is that any vector of given length with RMS value lower than in respective RMS column entry or CC value higher than in the CC column is considered **matched** to that database. Figures 5 and 6 illustrate an example of a RMS mathed signal to database and a CC matched signal respectively.

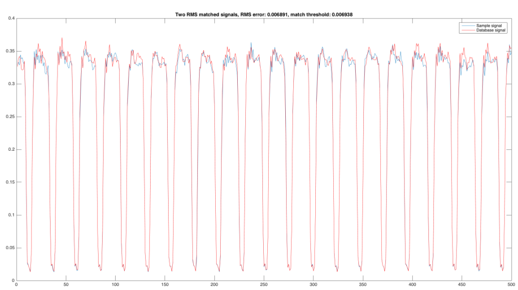

Figure 5: RMS Matched Signal, Vector Length 500

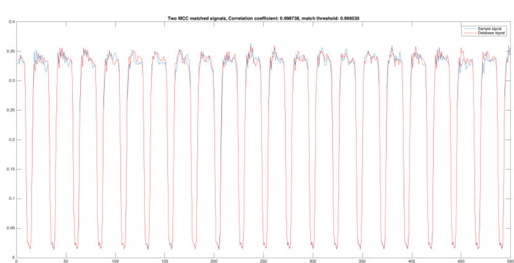

Figure 6: CC Matched Signal, Vector Length 500

Table 1: Minimum RMS and maximum CC values for *Previous* pulses database

Vector length	Minimum RMS	Maximum CC
100	0.00529395	0.99907746
500	0.00744177	0.99852720
1000	0.00741592	0.99828814
2500	0.00789533	0.99810360
5000	0.00833440	0.99785245
7500	0.00851705	0.99783553
10000	0.00863439	0.99770650
12500	0.00871000	0.99767169
15000	0.00883290	0.99774651

Table 2: Minimum RMS and maximum CC values for *Next* pulses database

Vector length	Minimum RMS	Maximum CC
100	0.00620465	0.99891061
500	0.00770854	0.99824145
1000	0.00802875	0.99816530
2500	0.00820027	0.99797494
5000	0.00834594	0.99798076
7500	0.00844571	0.99782432
10000	0.00849537	0.99778060
12500	0.00933407	0.99766753
15000	0.00926642	0.99753439

MATCHING SETUP, PROCEDURE AND RESULTS

In this section we present the matching setup and the observed results.

Database and sample setup

For the experiment two databases were constructed from two training sets:

- *Previous* database was constructed from 1201 *Previous* extracted pulse waveforms, total database waveform length: 18299571.
- *Next* database was constructed from 1177 *Next* extracted pulse waveforms, total database waveform length: 17791999.

For the matching in the experiment, 2 sets of independent samples were used

- 100 extracted *Previous* pulses
- 99 extracted *Next* pulses

The thresholds applied were: 0.00863439 for RMS and 0.99783553 for CC values.

Matching procedure

The matching procedure has the following steps:

1. Define thresholds for RMS and CC values
2. From set of samples, elect sample s of length L
3. From databases, select database D of length N
4. For every index i from $[0..N-L]$ take a section of the database of length L, $d = D[i, \ldots, i+L]$
5. Calculate $RMS(s, d)$ and $CC(s, d)$
6. Check if thresholds are reached, if yes, match is found, increase i by L (step over matched section)

In simplified terms: for a given sample, we walk along the database vector and calculate RMS and CC values. If they reach the threshold, we record as matched and step over the matched section.

ISBN 978-3-95450-147-2

Matching results

The matching algorithm would take a a *Previous* or *Next* sample and try to match it to both *Previous* and *Next* database. For each sample there are 4 possible outcomes:

- If a sample of type *A* has at least one match to database of type *A*, we record that as correctly identified, or, **Correct**
- If a sample of type *A* has at least one match to database of type *B*, we record that as incorrectly identified, or, **False**
- If a sample of type *A* has at least one match matched to both database of type *A* AND database of type *B*, we record that as identified **As both**
- If a sample of type *A* has NO matches to database of type *A* OR type *B*, we record that as **Not identified**

Results are summarized in Table 3 and Table 4.

Table 3: *Previous* Sample Matching Results, Procentage of All Samples

Method	Correct	False	As both	Not identified
RMS	6%	20%	23%	51%
CC	9%	15%	29%	47%

Table 4: *Next* Sample Matching Results, Procentage of All Samples

Method	Correct	False	As both	Not identified
RMS	30%	0%	34%	35%
CC	32%	2%	42%	23%

CONCLUSION

Although some interesting results have been observed we believe the procedures described above needs further refinement. Further focus on data filtering and extraction (and classification) to create better databases is required. Matching algorithm also needs improving since the choice of threshold levels might not be suitable for our analogous but modified approach.

We have already seen finer data (4*x* higher acquisition speed) available which means more detailed waveforms and databases which could also further improve out methods and yield more precise results.

ACKNOWLEDGEMENT

We would like to express our gratitude to Willem Blokland from SNS for sharing with us his research and data thus making our research possible.

REFERENCES

[1] M. Huijbregtse, Z. Geradts, “Using the ENF criterion for determining the time of recording of short digital audio recordings”, in *Proceedings of Computational Forensics*, The Hague, The Netherlands, 2009, pp 116–124.

[2] C. Grigoras, “Digital audio recording analysis–the electric network frequency criterion”, in *International Journal of Speech Language and the Law 12*, June 2005, pp 63–76.

[3] W. Blokland, C and Peters, “A New differential and Errant Beam Current Monitor for the SNS Accelerator”, in *Proceedings of IBIC2013*, Oxford, UK, 2013, paper THAL2, pp 921–924.