Self-Organising Symbolic Aggregate Approximation for Real-Time Fault Detection and Diagnosis in Transient Dynamic Systems

*M. S. Gallimore, C.M. Bingham, M.J.W Riley
School of Engineering, University of Lincoln, Lincoln. UK
*Corresponding Author: mgallimore@lincoln.ac.uk

***Abstract*—The development of accurate fault detection and diagnosis (FDD) techniques are an important aspect of monitoring system health, whether it be an industrial machine or human system. In FDD systems where real-time or mobile monitoring is required there is a need to minimise computational overhead whilst maintaining detection and diagnosis accuracy. Symbolic Aggregate Approximation (SAX) is one such method, whereby reduced representations of signals are used to create symbolic representations for similarity search. Data reduction is achieved through application of the Piecewise Aggregate Approximation (PAA) algorithm. However, this can often lead to the loss of key information characteristics resulting in misclassification of signal types and a high risk of false alarms. This paper proposes a novel methodology based on SAX for generating more accurate symbolic representations, called Self-Organising Symbolic Aggregate Approximation (SOSAX). Data reduction is achieved through the application of an optimised PAA algorithm, Self-Organising Piecewise Aggregate Approximation (SOPAA). The approach is validated through the classification of electrocardiogram (ECG) signals where it is shown to outperform standard SAX in terms of inter-class separation and intra-class distance of signal types.**

1. Introduction

Similarity search is an important aspect of fault detection and diagnosis (FDD), particularly when dealing with large databases of signals. Specifically, a similarity search is a method for determining the similarity between a query object and a (typically large) database of objects. It has had important application in bioinformatics, pattern recognition and computer vision. Typical approaches involve two distinct stages. Firstly, a dimensionality reduction technique is used to create simplified signal representations. Many such algorithms have been reported e.g. such as Piecewise Aggregate Approximation (PAA) [1], Discrete Fourier Transform (DFT) [2], Discrete Wavelet Transform (DWT) [3], Singular Value Decomposition (SVD) [4] and Piecewise Linear Approximation (PLA) [5]. PAA is one such technique that has received much attention in recent years due to its relative simplicity and high performance. PAA, originally proposed by Keogh et al. [1], reduces the dimensionality of the data by taking mean values over equally spaced sized frames. For instance, consider a time series S of length n. Using the PAA approach it is possible to represent the time series in a w-dimensional vector space by a vector $\overline{s} = \overline{s}_1,\ldots,\overline{s}_w$. The i^{th} element of $\overline{s}$ is calculated from the following equation:

$$\overline{s}_i = \frac{w}{n} \sum_{j=\frac{n}{w}(i-1)+1}^{\frac{n}{w}i} s_j$$

Once the data-reduced representation is obtained, a suitable indexing structure is applied. Symbolic Aggregate Approximation (SAX) is one such technique that has received attention since being first proposed in 2003 by Lin and Keogh [6]. Considered essentially as an extension of PAA, SAX allows the time series of length n to be reduced to a symbolic string of length w ($w<n$). This is achieved by the setting of breakpoints, where mean values that fall within certain breakpoint limits are allocated a symbol or character, e.g. a, b, c, or d. These breakpoints are defined such that the normalised time series data assume a Gaussian distribution [7] as in Table 1.

β_i \ a	3	4	5	6	7	8	9	10
β_1	-0.43	-0.67	-0.84	-0.97	-1.07	-1.15	-1.22	-1.28
β_2	0.43	0	-0.25	-0.43	-0.57	-0.67	-0.76	-0.84
β_3		0.67	0.25	0	-0.18	-0.32	-0.43	-0.52
β_4			0.84	0.43	0.18	0	-0.14	-0.25
β_5				0.97	0.57	0.32	0.14	0
β_6					1.07	0.67	0.43	0.25
β_7						1.15	0.76	0.52
β_8							1.22	0.84
β_9								1.28

Table 1 – SAX Breakpoints

Table 1 shows the breakpoints for a 3-10 letter symbolic representation. For example, if a three letter representation is considered, a, b, and c, then the break points are set at -0.43 and 0.43. This means that a mean frame value falling below -0.43 would be allocated the letter a. A mean frame value falling between -0.43 and 0.43 would be allocated the letter b and a mean frame value falling above 0.43 would be allocated the letter c, as shown in Figure 1. The number of breakpoints selected should be the minimum required to retain important signal characteristics, but as this is usually unknown *a priori*, should be based on the trade-off between computational demand and information loss.

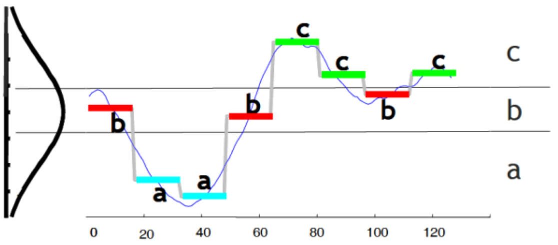


Figure 1 – SAX representation of time series data [6]

Although SAX has been shown to be very effective in providing accurate similarity search indexing, for example in the analysis of Electrocardiogram (ECG) signals [8], the fact that it employs PAA for dimensionality reduction and the allocation of symbols is based on average values falling within a certain breakpoint range, discrete changes in signal characteristics can be inadvertently missed.

Consider the two signals in Figure 2. Both are identical apart from a small discrete change (circled) added to the signal on the right.

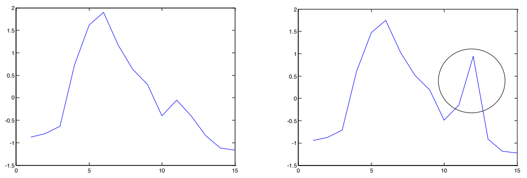


Figure 2 - Raw signals (right signal with discrete change added)

Now consider the PAA and associated three letter SAX representations of the same two signals, shown in Figure 3. It can be seen that the SAX representation (accba) is identical for both signals and hence the discrete change is lost during the reduction process.

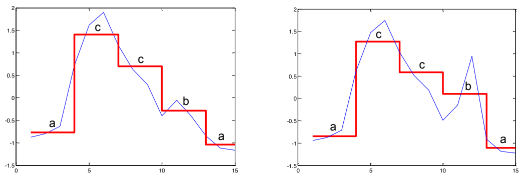


Figure 3 - SAX representations showing loss of signal information

Clearly, increasing the number of PAA frames around this area will increase the accuracy of SAX in this case. This is illustrated in Figure 4 which shows a finer PAA frame distribution around the area of interest, leading to a new SAX representation of accbca.

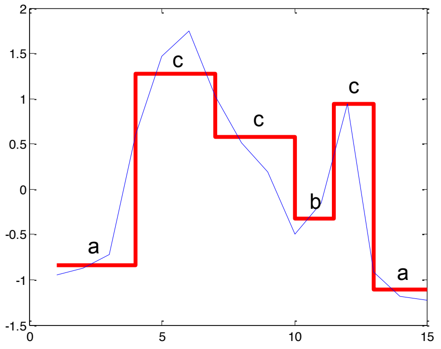


Figure 4 - Improved SAX representation without loss of key signal information

Of course, this can also be achieved by using standard PAA with more equi-distributed frames, but in so doing, leads to an increased risk of false alarms and higher computational demands. Furthermore, determining the optimum number of frames is difficult without numerous iterative reruns.

In this paper, a novel SAX-based methodology, termed Self-Organising Symbolic Aggregate Approximation (SOSAX), is proposed, for use in real-time FDD systems where fast similarity search is important but without loss of key signal information. The method utilizes an improved dimensionality reduction technique called Self-Organising Piecewise Aggregate Approximation (SOPAA) [9] that determines the optimum PAA parameters to provide maximum separation of signal types and hence leads to improved SAX representations. The approach is validated through application to a similarity search in the classification of electrocardiogram (ECG) signals, where it is shown to outperform standard SAX in terms of inter-class separation and intra-class similarity.

2. Self-Organising Symbolic Aggregate Approximation (SOSAX)

To generate SAX representations without loss of transient signal characteristics, SOPAA is applied to provide optimum dimensionality reduction. SOPAA is based on optimising the PAA parameters for individual data sets. Optimisation in this case (SOPAA) is a process of finding the optimum solution(s) for frame size, distribution and the number of classes (if unknown). The optimality of a given set of decision variables can be measured through one or more objective function(s), for instance, the number of samples correctly classified. In many practical problems, however, finding the global optima is a difficult task as the objective functions tend to be highly non-linear and there is no way of guaranteeing initial estimates that are close to the global optimum. To tackle such issues, a series of meta-heuristic optimizers have been developed. These optimizers, the Genetic Algorithm (GA), Differential Evolution (DE), Particle Swarm Optimisation (PSO) and Adaptive Simulated Annealing (ASA) being common in the literature, use a population of trial solutions and apply probabilistic rules to generate a new population which typically converge to the global optimum with high probability. Although popular, a number of problems remain with these optimizers; one being premature convergence where the population converges to a point that is a local optimum. They do have built-in functions that attempt to overcome this but frequent re-runs are good practice to give confidence that the global optimum has been reached. SOPAA utilises the DE optimiser to determine SOPAA parameters.

DE is an example of an evolutionary algorithm that uses mechanisms inspired by biological evolution; namely recombination, where two or more candidate solutions (so-called parents) are combined to give rise to one or more candidate solutions (so-called children), and mutation, where one candidate solution results in one new candidate solution. This process gives rise to a new population (so-called offspring) that competes for a place in the next generation. Considering a population of NP solutions in a D-dimensional search space, the population G for each iteration (so-called generation) is given by,

$$x_{i,G},\; i = 1,\ldots,NP$$

Two operators, mutation and crossover, are applied to each candidate solution at each generation, producing a new population. For each candidate vector solution $x_{i,G}$, a mutant vector is generated with random indexes $r_1, r_2, r_3 \in \{1,2,\ldots,NP\}$ according to,

$$v_{i,G+1} = x_{best,G} + F \cdot \left(x_{r1,G} - x_{r2,G}\right)$$

F is a real constant factor which controls the amplification of the differential variation $\left(x_{r2,G} - x_{r3,G}\right)$. Crossover is introduced in order to increase the diversity in the new population and new solution vectors generated according to,

$$u_{i,G+1} = \left(u_{1i,G+1}, u_{2i,G+1}, \ldots u_{Di,G+1}\right)$$

Each candidate solution in the new population is then compared to the corresponding candidate solution in the previous population and the best selected as a member of the next generation [10, 11]. The objective, in this case, is to maximise the classification rate of signals belonging to known classes by altering SOPAA frame number and distributions. The distribution of SOPAA frames x_i is given by,

$$D_1\left(x_i\right) = \frac{1}{2} - \left[\frac{1}{2}\left(1 - D_e\right)^{\frac{1}{B}}\right] \text{ (Left of centre)}$$

$$D_2\left(x_i\right) = \frac{1}{2} + \left[\frac{1}{2} D_e^{\frac{1}{B}}\right] \text{ (Right of centre)}$$

where, D_e is the even distribution corresponding to x_i $\{i = 1,\ldots,N\}$, N is the number of SOPAA frames and B is the form factor [0.05, 5]. To provide greater flexibility during the optimisation process, the optimiser has the freedom to define two different distributions (left distribution and right distribution) either side of the signal centre, within the same reduced representation. The distribution of SOPAA frames is governed by altering the value of the form factor, B, during the optimisation process until the global optimum is found. Once identified, the optimum SOPAA parameters are then used to generate data reduced representations of all signals within the dataset, followed by the generation of symbolic SOSAX representations.

2.1 SAX Distance Measure

Once the SOSAX representations of signals have been obtained, similarity between two symbolic strings is determined using a distance measure; the Euclidean distance measure being chosen here for convenience. This is achieved by sequentially comparing a query string $\hat{S}_q$ to each string $\hat{S}_d$ held in the database. Considering two time series, S_q and S_d, then the Euclidean distance between the two signals is calculated using,

$$D\left(S_q, S_d\right) = \sqrt{\sum_{i=1}^{n}\left(s_{qi} - s_{di}\right)^2}$$

Now, generating the data reduced PAA representations $\overline{S}_q$ and $\overline{S}_d$ and transforming them into SAX representations $\hat{S}_q$ and $\hat{S}_d$, a lower bounding approximation of the Euclidean distance measure between the signals can be calculated using,

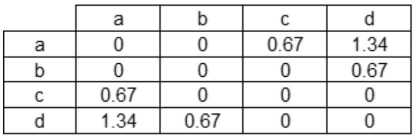
$$MINDIST\left(\hat{S}_q, \hat{S}_d\right) = \sqrt{\frac{n}{P}}\sqrt{\sum_{i=1}^{n}\left(dist\left(\hat{s}_{qi} - \hat{s}_{di}\right)\right)^2}$$

[120]

where, n is the length of the original time series, P is the number of PAA frames, and the dist() function between two symbols is given by,

$$dist = \begin{cases} 0, \text{ if } \beta_{\max} - \beta_{\min} \le 1 \\ \beta_{\max} - \beta_{\min}, \text{otherwise} \end{cases}$$

Distances can be summarised using a lookup table, as shown in Table 2 for a four-letter SAX representation.



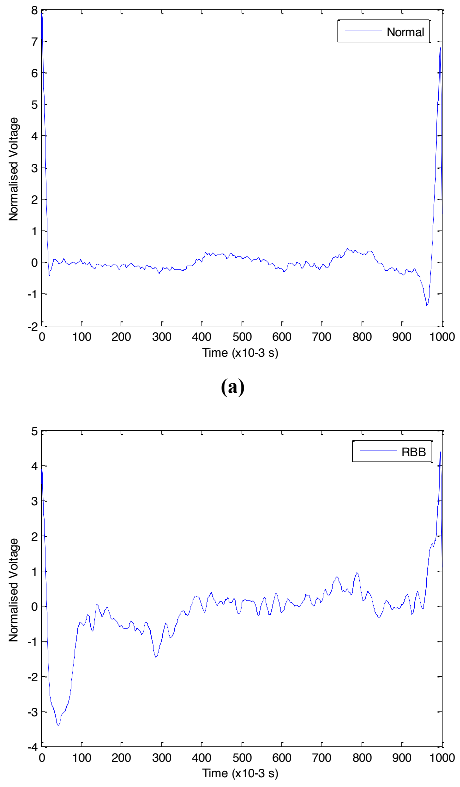
	a	b	c	d
a	0	0	0.67	1.34
b	0	0	0	0.67
c	0.67	0	0	0
d	1.34	0.67	0	0

Table 2 - 4-letter SAX lookup table

A distance of zero is considered a complete match and the aim is to minimise the distance between two strings in the same class and maximise the distance between two strings in different classes. This will lead to an increase in matching accuracy and a reduced risk of false alarms.

3. Results and Discussion

To demonstrate the improvements in search accuracy provided by the proposed SOSAX methodology, results are compared to SAX using an ECG dataset obtained from the Physionet database [12]. The data consists of 120 samples and 2 classes, Normal and Right Bundle Branch Block (RBB). Class 1 (Normal) consists of 100 individual heartbeats considered as healthy and taken from 10 patients (10 beats from each). Class 2 (RBB) consists of 20 individual heartbeats taken from two patients who have been diagnosed with a right bundle branch block. An example signal from each class is shown in Figure 5.



(a)

(b)

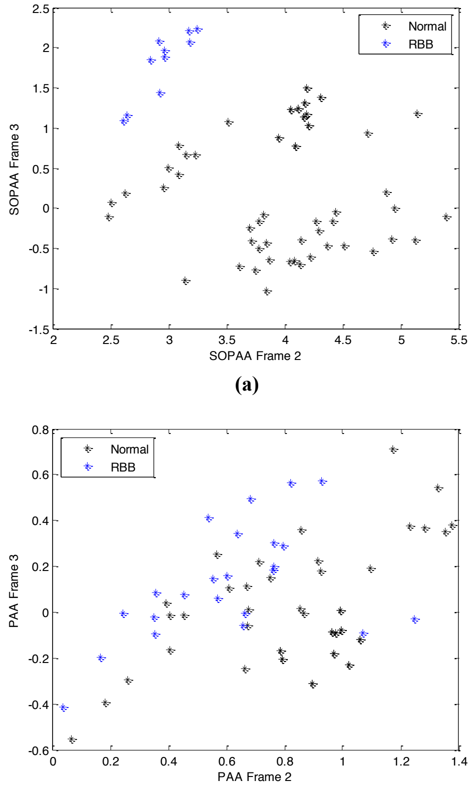
Figure 5 - ECG signals (a) Normal (b) Right Bundle Branch Block (RBB)

The dataset for each class is separated into a training and testing set, with 60 training (from six patients) and 40 testing (from 40 patients) for the Normal class and 10 training (from one patient) and 10 testing (from one patient) for the RBB set. Since the number of classes is known, single objective SOPAA is used to identify optimum PAA parameters using the training data. SOPAA identifies 13 frames as the optimum number in order to achieve a maximum classification rate. The frame distributions and classification rates for both SOPAA and PAA are shown in Table 3.

No. Frames to Left	Left Dist.	No. Frames to Right	Right Dist.	SOPAA Class. Rates	PAA Class. Rates
9	4.98	4	0.57	Total = 100% Class 1 = 100% Class 2 = 100%	Total = 56.9% Class 1 = 53.3% Class 2 = 100%

Table 3 - SOPAA parameters and classification rates

Cluster plots for frames 2 and 3 using SOPAA and PAA are shown in Figure 6 where it can be clearly seen that SOPAA achieves 100% correct classifications, as well as more compact and better separated clusters. Standard PAA with 13 frames, however, only achieves 40% Class 1 and 100% Class 2.



(a)

(b)

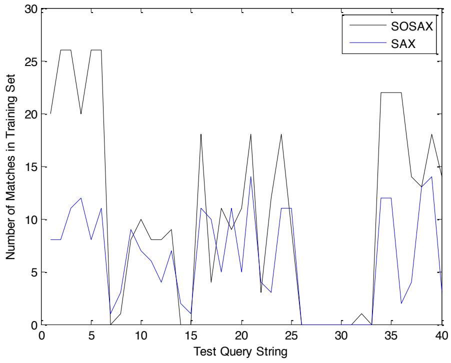
Figure 6 - Cluster plots for frames 2 and 3 (a) SOPAA (b) PAA

SAX and SOSAX representations are generated for each sample in the training and test sets using a six-letter alphabet size. Each test SAX and SOSAX query string is then compared to the respective SAX and SOSAX database of training strings and the distance measures calculated using the six-letter lookup table shown in Table 4.

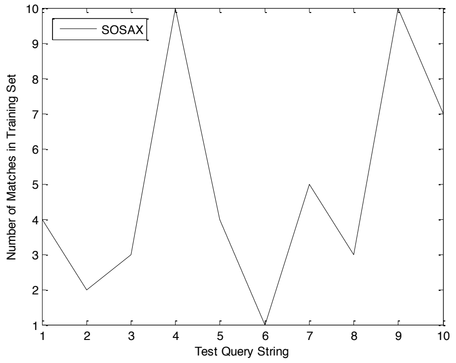
	a	b	c	d	e	f
a	0	0	0.54	0.97	1.4	1.94
b	0	0	0	0.43	0.86	1.4
c	0.54	0	0	0	0.43	0.97
d	0.97	0.43	0	0	0	0.54
e	1.4	0.86	0.43	0	0	0
f	1.94	1.4	0.97	0.54	0	0

Table 4 - 6-letter lookup table

Instances where the distance between a query string and a string in the training set is zero are considered a match. The class with the highest number of matches for a given query string is considered the 'winning class'. The objective is to maximise the classification accuracy obtained by SOSAX, as well as maximise the difference in the number of matches between each class. For example, a query string that belongs to Class 1 should match a maximum number of strings in Class 1 of the training set and a minimum number of strings in Class 2. This gives greater confidence in classification and will reduce the risk of false alarms. Classification rates are 100% Class 1 and 20% Class 2 for SOSAX and 80% Class 1 and 0% Class 2 for SAX. Although the classification rates for SOSAX and SAX are relatively close, with a 20% increase in classification rate for both classes obtained by SOSAX, the number of matches between the test data and training data in the same class increases significantly using SOSAX, particularly in Class 1. SAX fails to achieve any matches between Class 2 query strings and Class 2 training strings—as shown in Figure 7.



(a)



(b)

Figure 7 - Intra-class matches (a) Class 1 (b) Class 2

SOSAX achieves a total of 378 correct matches in Class 1 and 7 in Class 2, compared to 243 in Class 1 and 0 in Class 2 for SAX. The increase in the number of intra-class matches shows that SOSAX is able to generate much more compact and accurate symbolic representations of signals within the same class, ultimately leading to greater confidence in classification accuracy.

4. CONCLUSIONS

The paper has presented an extension to the Symbolic Aggregate Approximation (SAX) algorithm, termed Self-Organising Symbolic Aggregate Approximation (SOSAX). Optimum data reduced representations for a training dataset are achieved through the application of the single objective Self-Organising Piecewise Aggregate Approximation (SOPAA) algorithm, leading to optimum SOSAX symbolic representations. The methodology has been applied to the classification of real patient ECG data with two classes. SOSAX is shown to outperform the standard SAX algorithm in reducing the intra-class separation between SOSAX strings and improving matching accuracy and reliability. This ability to improve symbolic representations is significant and leads to more accurate and robust classification whilst maintaining computational efficiency. The methodology is applicable to the development of any FDD system where there is a requirement for fast similarity search of large databases whilst minimizing the loss of transient signal characteristics.

REFERENCES

[1] E. Keogh, K. Chakrabarti, M. Pazzani and S. Mehrotra, "Dimensionality reduction for fast similarity search in large time series databases," *Journal of Knowledge and Information Systems,* 2000.

[2] R. Agrawal, T. Imielinski and A. Swami, "Mining Association Rules between Sets of Items in Large Databases," in *Proceedings of the 1993 ACM SIGMOD Conference*, Washington DC, USA, 1993.

[3] K. Can and A. W. Fu, "Efficient Time Series Matching by Wavelets," in *15th IEEE Conference on Data Engineering*, Sydney, Australia, 1999.

[4] F. Korn, H. V. Jagadish and C. Faloutsos, "Efficiently Supporting

Ad Hoc Queries in Large Datasets of Time Sequences," in *SIGMOD Conference '97*, Tucson, Arizona, USA, 1997.

[5] Y. Morinaka, M. Yoshikawa, T. Amagasa and S. Uemura, "The L-index: An Indexing Structure for Efficient Subsequence Matching in Time Sequence Databases," in *5th Pacific Asia Conference on Knowledge Discovery and Data Mining*, Hong Kong, 2001.

[6] J. Lin, E. Keogh, S. Lonardi and B. Chiu, "A Symbolic Representation of Time Series, with Implications for Streaming Algorithms," in *DMKD '03 Conference* , San Diego, CA, USA, 2003.

[7] P. Siirtola, H. Koskimaki, V. Huikari, P. Laurinen and J. Roning, "Improving the classification accuracy of streaming data using SAX similarity features," *Pattern Recognition Letters,* vol. 32, no. 13, pp. 1659-1668, 2011.

[8] B. Kulahcioglu, S. Ozdemir and B. Kumova, "Application of Symbolic Piecewise Aggregate Approximation (PAA) analysis to ECG signals," *Journal of Applied Simulation and Modelling,* vol. 609, 2008.

[9] M. S. Gallimore, M. J. W. Riley, C. M. Bingham, "Self-Organising Piecewise Aggregate Approximation algorithm for intelligent detection and diagnosis of heart conditions," in *International Conference on Medical Health Sciences, Berlin, Germany, 2015.*

[10] R. Storn and K. V. Price, "Differential Evolution - A Simple and Efficient Adaptive Scheme for Global Optimization over Continuous Spaces, Technical Report TR-95-012," International Computer Science Institute, Berkeley, CA, 1995.

[11] R. Storn and K. A. Price, "Differential Evolution - A Simple and Efficient Heuristic for Global Optimization over Continuous Spaces," *Journal of Global Optimization,* vol. 11, pp. 341-359, 1997.

[12] “Physionet,” [Online]. Available: http://www.physionet.org/cgi-bin/atm/ATM. [Accessed 08 August 2013].