PERIODICITY DETECTION ALGORITHMS IN TIME SERIES DATABASES-A SURVEY

JishaKrishnan
Dept. of ComputerScience And Engineering
SCT College of Engineering
Trivandrum,India
jishkrish@gmail.com

Chitharanjan K
Dept. of ComputerScience And Engineering
SCT College of Engineering
Trivandrum,India
Chitharanjan_k@yahoo.com

***Abstract*—Periodicity mining is used for predicting different applications such as prediction,forcasting etc.It has several application in Timeseries databases.Several algorithms are present for detecting the periodicity.But most of the algorithm do not take into account the presence of noise or partial periodicity.Here we compare four different types of algorithm.Based on timewraping ,the first algorithm wraps the time axis to optimally remove the noise at various locations.The second algorithm can be viewed as a variation of the approximate string matching algorithm. The third algorithm is used for partial periodicity detection and in the fourth one periodic detection using suffix tree is done .This algorithms detects periodicity in noise and also detects partial periodicity .Here acomparison of three algorithms are done.**

***Index Terms*—Periodicity detection,Time series,Segment periodicity,Symbol periodicity, Partial periodicity, Time wraping,Convolusion,Suffix Tree**

I. INTRODUCTION

A time series play a vital role in real life. A time series can be defined as a collection of datavalues gathered at uniform interval of time. Several examples of time series in real life is weather condition of a particular location, transaction in superstore ,gene expression data analysis[5] etc.One of the main features of time series is repeating cycles..Examples are seawaves,revolution of earth around sun etc There are many examples of this cycle in our daily life. For instance, there is a traffic jam twice a day when the school is open, no of transaction in a superstore is high at certain period. So finding periodic pattern in the time series database is an important task.

A time series can be represented in discrete form[1].It is discretized by considering an distinct ranges such that all values in a range is represented by one symbol .For example, consider the transaction in a superstore.Let [0] is denoted by a,[1-200]transaction is denoted by b,[201-400] transactions are denoted by c and [401-600] transactions are denoted by d.Then the time series T= 243,267,355,511,120,0,0,197 can be discretized into T = cccdbaab[1].

The Periodicity are of three types 1) Symbol periodicity 2) Partial periodicity 3) Segment periodicity[1]. Symbol periodicity means only one symbol is periodic.Ex T= acdabcadb.Here only a is periodic with p = 5. If more than one symbol is periodic and occur partially it is called partial periodicity .Eg: T= bbaaabbdabcaabbcabcd .Here sequence ab is periodic with periodicity p=4 and perodicity starts from the position 4 .Segment periodicity means the whole timeseries is represented as a periodic pattern.

In time series periodicity can be full or partial[4]. Also several noise can be present in the time series. Methods used for finding full periodicity cannot be used for detecting partial periodicity .Partial periodicity is a looser kind of periodicity than full periodicity and it exist ubiquitously in the real world[4] An example partial periodic pattern may state that Jim reads the Vancouver Sun newspaper from 7:00 to 7:30 every weekday morning but his activities at other times do not have much regularity.So partial periodic detection is a expensive mining process because it is the mixture of periodic and nonperiodic events. Another problem occur in periodic detection is the presence of noise .Most of the algorithm have poor resilience to noise.

Another problem in periodicity detection is perfect periodicity[1]. All the periodicity in time series database is not perfect. The degree of perfection of time series can be represented in terms of confidence .Confidence is defined as the ratio of actual frequency in the series over its expected perfect frequency in the time series. Consider an imperfect periodicity in the example T = abefd abcde acbfe abedf, here the confidence is 4/5.

The periodicity mining algorithm requires user to specify a periodic length that determines the rate at which the time series is periodic.This cannot be done in trial and error method.The solution of this problem is to devise a technique for discovering the potential periods in the time series datafollowed by the application of any existing pattern mining technique to determine the interesting pattern.[4]

To sum up,time series exist frequently in our daily life and their analysis could lead to valuable discoveries.So there is need for noise resilence algorithm that can tackle the problem of i) Identifying three different type of periodic pattern ii) handling asynchronous periodicity.Here a comparative study of four algorithm are done such as Time wraping for Periodicity Detection usually called WARP,Periodicity Detection Algorithm using Convulusion usually called CONV ,Partial Periodicity Detection usually called Parper and Periodicity Detection using suffix tree usually called STNR.

II. PROBLEM DEFINITION

Let T $=e_0,e_1 \ldots e_{n-1}$ be a time series of length n where e_i denotes the event at time i and let T contains symbols taken from an alphabet set ∑ with enough symbols. Let T= bbcdbababababcdaccab could represent one time series.Then it can be represented in 5 tuples(s,p,stpos,endpos,conf) where s denotes the periodic pattern, p denotes the periodic length,stpos denote the start position and endpos denote the end position and conf denotes the confidence . So here T can be represented as(ab,2,5,12,1).Here we mainly concentrate on symbol periodicity,segment periodicity and sequence periodicity.The definitions of the terms are explained below.

Definition 1: (Perfect Periodicity). Consider a time series T, a pattern X is said to satisfy perfect periodicity in T with period p if starting from the first occurrence of X until the end of T every next occurrence of X exists p positions away from the current occurrence of X. It is possible to have some of the expected occurrences ofX missing and this leads to imperfect periodicity.[1]

Definition 2: (Confidence).The confidence of a periodic pattern X occurring in time seriesT is the ratio of its actual periodicity to its expected perfect periodicity. Formally, the confidence of pattern X with periodicity p starting at position stP os is defined as:

$$conf(p, stP\,os, X) = \frac{Actual\ Periodicity(p, stP\,os, X)}{Perfect\ Periodicity(p, stP\,os, X)} \qquad (1)$$

Where Perfect periodicity(p,stPos,X)=T-stpos+1/p and Actual periodicity is computed by counting the number of occurance of X in T.For example, in T =abbcaabcdbaccdbabbca, the pattern ab is periodic with stPos = 0, p = 5 , and conf(5, 0, ab)=3/4[1]

Definition 3: (Symbol Periodicity). A time series T is said to have symbol periodicity for a given symbol s with period p and starting position stPos if the periodicity of s in T is either perfect or imperfect with high confidence, i.e., s occurs in T at most of the positions specified by stpos+i*p, where p is the period and integer i >0 takes consecutive values starting at 0.[1]

For ex in T =bacdbabcdaccdabc, the symbol a is periodic with stP os = 1 and p = 4, i.e., a occurs in T at positions 1, 5, 9, and 13.

Definition 4 (Partial Periodicity). A time series T is said to have partial periodicity for a partial periodic pattern X starting at position stP os, if., X occurs in T at most of the positions specified by stP os + i * p, where p is the period and integer i >=0 takes consecutive values starting at 0.[1]

For example, in T =abbcaabcdbabcdbabbca, the pattern ab is partially periodic starting from stP os = 0 and period value of 5.

III PERIODICITY DETECTION ALGORITHMS

A.Time Wraping Periodic Detection(WARP)

In WARP segment periodicity is detected. Two segments are similar if there are enough symbols that matchwise. Here a dynamic computation of timewraping [9] is done and hence it is called Dynamic Time Wraping .DTW can be described as follows[7]. Let $X = [x_0, x_1, \ldots, x_{n-1}]$ and $Y = [y_0, y_1, \ldots, y_{n-1}]$ be two finite length sequences of symbols, each of length n. Let X be the sequence X after removing the first element x, that is $X= [x_1, \ldots, x_{n-1}]$[3]

The standard definition of time wraping [3] is

$$DTW(X,Y) = d(x_0,y_0) + \min\begin{cases} DTW(X,Y') \\ DTW(X',Y) \\ DTW(X',Y,) \end{cases}$$

Here $d(x_0,y_0)$ is the distance between two symbols. The value of this is either 0 or 1 depending on whether there is match or not .ie[7]

$$d(x_i,y_j) = \begin{cases} 0 & x_i = y_j \\ 1 & x_i \neq y_j \end{cases}$$

Here an nxn matrix is constructed where the cell(i,j) contains the value d(i,j). A wraping path is a contiguous path $M=m_0,m_1. .m_{k-1}$ where m_k corresponds to the cell (i_k, j_k).Many wraping paths are present in matrix.Cost of each wraping path is computed .Cost of the path is the total distance to walk through this path. The path having minimum wraping cost is taken.

The main idea in periodic detection is shift the timeseries p positions and compare the original timeseries to the shifted version. The algorithm is based on the DTW matrix described above .Let a time series be denoted by $T=e_0,e_1,e_2\ldots\ldots e_{n-1}$.A DTW matrix can be constructed as in figure

	e_0	e_1	e_2	e_3	....	e_{n-1}
e_0	0	$d(e_0,e_1)$	$d(e_0,e_2)$	$d(e_0,e_3)$	....	$d(e_0,e_{n-1})$
e_1	$d(e_1,e_0)$	0	$d(e_1,e_2)$	$d(e_1,e_3)$	....	$d(e_1,e_{n-1})$
e_2	$d(e_2,e_0)$	$d(e_2,e_1)$	0	$d(e_2,e_3)$	....	$d(e_2,e_{n-1})$
e_3	$d(e_3,e_0)$	$d(e_3,e_1)$	$d(e_3,e_2)$	0	....	$d(e_3,e_{n-1})$
.						
.						
e_{n-1}	$d(e_{n-1},e_0)$	$d(e_{n-1},e_1)$	$d(e_{n-1},e_2)$	$d(e_{n-1},e_3)$	....	$d(e_{n-1},e_{n-1})$

Fig 1: DTW matrix of time series T

Here the matrix is constructed with cell values$(ei_, e_J)$ having the values $d(ei_, e_J)$ where $d(ei_, e_J) = 0$ if $ei_= e_J$ otherwise is 1.Here all the comparisons between the symbols is considered .The comparison of original time series to the shifted version is done by each sub diagonal. So each diagonal denote a particular period. The subdiagonal having minimum wrapping cost is taken. The period corresponding to that subdiagonal is the original period of the timeseries.

B.Parial Periodic Detection Algorithm(Parper)

Here time series considered as a set of features as $S= D_1,D_2,D_3\ldots,D$ .In this algorithm L set of underlying features are considered. So a pattern $s=s_1,s_2,\ldots sp$ be a non empty set of pattern over $(2^L$-{Ø})U{*}. |S| is the period of pattern d. A pattern s' is a subpattern of s if s and s' have same length and s'c s. Ex a*{a,c} is a pattern of length 5 and a*{a.c}** and **cde are two subpattern of s.[4]

In this algorithm frequent count and confidence of a pattern are taken and it is defined as

Frequency_count(s) ={i|0≤ i<m and the string s is true in S}

Confidence(s) = frequency count(s)/m where m is the maximum no of length|*s*|.

For example, a*b is a pattern of period 3,its frequency count in the feature series a{b,c}bacbacd is 2 and its confidence is 2/3 where 3 is the maximum number of period of length 3.The frequency count of a{b,c}* in a{}b,c,d}ca{b,c}aabc is 2/3 .

Here the segment periodicity is determined by mining association rule[7]. ie a pattern is a frequent partial pattern in the time series if its confidence is larger than or equal to a threshold min_conf.. So finding the frequent partial periodic pattern the input given is the time series S, a specified period or a range of period and the length of the period.

This algorithm uses single period apriorimethod[4]. The key idea behind this is a apriori property[7] is if one subset of an itemset is not frequent, then the itemset itself cannot be frequent.Suppose s' is a subpattern of a frequent pattern s.Then s' is obtained by changing some letters from s.Hence s is more restrictive than s' and frequency count of s' is greater than or equal to s .Thus s' is frequent as well..

The method used here is find all frequent pattern of period p based on the idea of Apriori ie all pattern that has frequency count no less than min_conf which is the threshold confidence value

C. Periodic Detection using convolusion method(CONV)

This algorithm resembles the string matching algorithm. Here the approach used is computing the score vector for a string of length n. The algorithm has a drawback that it is very difficult to convert to an external algorithm to handle large datasets. This algorithm computes the score vector by a convolution[2] which uses FastFourier Transform(FFT)[17].This has two advantages such as the time complexity is reduced and rhe algorithm is scalable since an external FFT algorithm is adapted for large datasets

Here mainly symbol periodicity and segment periodicity detection is done. In segment periodicity detection idea used is a variation of string matching problem[10] which consider finding small variations of pattern string of length m in a text string T of length n. Here the matching is done by using a score vector [2]of match between T and P .The scorevector $Ci^{T,P}$ is the ith component of $C^{T,P.}$ where $Ci^{T,P}$ is the no of matches between the text and the pattern where the first letter of the pattern is positioned in front of the ith letter of the string where i= 0,1…n-1.Consider the test string T= abcdaccd and a pattern string p = bcd the score vector can be computed as $C^{T,P}$=[0,3,0,0,1,2,0,0,0] ie $C_1^{T,P}$ = 3 and $C_5^{T,P}$ =2.So according to the score vector there is a perfect match between the pattern and the text at the position 1 of the text and there is a approximate match at the position 5.This method can be used for segment periodicity detection. In segment periodicity detection T = P .For ex let T = abcdaccd .The $C^{T,T}$ =[8,1,0,1,3,0,0,0] ie $C_4^{T,T}$ =3 is an approximate match at the position 4..There are some segment periodicity properties according to score vector.[2]

- The value 1,2,n/2 are the only possible values for candidate period for T
- The maximum value for any C_i^T is n-i.
- If p is a perfect period for T this C_p^T=n-p

In symbol periodicity detection algorithm only one or two symbol is considered rather than full segment.So here only one symbol is considered at a time and all other symbols are replaced by ‘*’.Ex T= abcdaccd. So here T is converted to T= **c**cc* by considering the symbol c only. When only one symbol is used then the symbol can be replaced by 1 and all other by 0.So there is a match when we meet 1 and otherwise a mismatch. The disadvantage of this symbol periodicity detection algorithm is that it requires large no of scans over the time series one per symbol which is very costly.

D.Periodic Detection using SuffixTree(STNR)

This algorithm is done in two phases.In the first phase suffixtree [9]is constructed and in the second phase this suffix tree is used to calculate the periodicity in time series databases.A suffix tree for the string T=abcabbabb is constructed as Fig:2[1].

Suffix tree can be used to find a substring in an original sting and also find frequent substring.All suffix in the string can be represented in a suffixtree.The suffix represent the path from root to leaf.The string of length n has n suffix and therefore has n leaves. Edges are represented by the string it represents. Each leaf node represents the starting position of the suffix when traversing from root to leaf.The number in the intermediate node represent the length of the substring read when traversing from root to leaf.

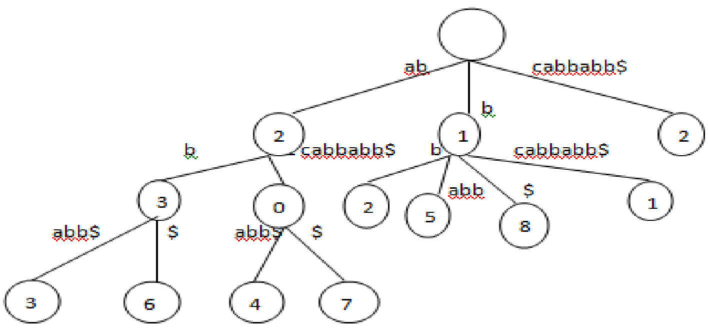

Fig 2:Suffix tree for the string abcabbabb$

Suffix tree can be used to find a substring in an original sting and also find frequent substring.All suffix in the string can be represented in a suffixtree.The suffix represent the path from root to leaf. The string of length n has n suffices and therefore has n leaves. Edges are represented by the string it represents. Each leaf node represents the starting position of the suffix when traversing from root to leaf. The number in the intermediate node represent the length of the substring read when traversing from root to leaf.

After the tree is created second phase starts. The first step during the start of second phase is the construction of occurance vector[1] for each edge connecting an internal node to its parent. The occurance vector of each edge contains the index positions at which the substring from the root to edge exist in the original string.The occurance vector of each node is represented in the figure 3[1]

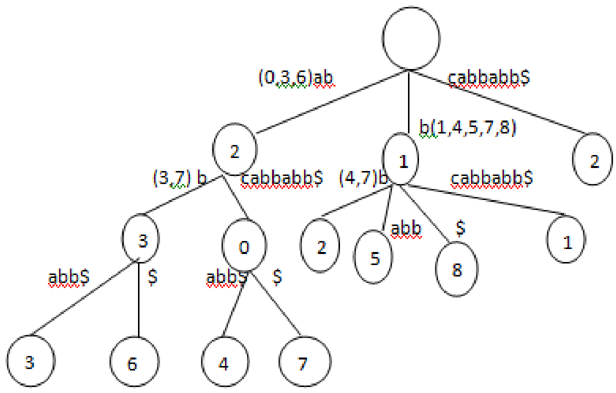

Fig 3:Occurance vector of the string abcabbabb$

The periodicity detection algorithm uses the occurance vector to check whether the string represented by the edge is periodic or not. In the second phase at each intermediate edge the periodic detection algorithm is applied[1].Here a difference vector(diff vect) is calculated as the difference between the elements in the occurance vector .In the algorithm for each candidate period (p=diffvect[j]),the algorithm scans the occurance vector starting from its corresponding value(stpos=occurvect[j]) and increase the frequency count of the period freq(p) if and only if the occurance vector value is periodic with regard to starting position and p. If the frequency of a particular period is more than a threshold value .then p is taken as the candidate period..

The main attraction of the algorithm is it is used where noise is present. It is a noise resilent algorithm[6].The noise present here are insertion, replacement and deletion noise .For making the algorithm more efficient in noise, a new parameter called time tolerance is used.The idea here is that periodic occurance can be drifted within a specified limit called time tolerance.So drifting the time period within the time tolerance value should not be problem.

Partial periodicity detection can be done using the suffix tree.For detecting the partial periodicity dmax and minlength are utilized.The first parameter dmax denote the maximum distance between any two occurance of a pattern to be part of the same periodic pattern.If the distance is more than dmaxthen it is marked as the end of periodic section.The second parameter minlength is used to specify the minimum length required to periodic section

III COMPARISON OF ALGORITHMS

For comparing the different types of algorithms such as WRAP,CONV ,Parper ,STNR, different features such as type of periodicity detected, noise ratio,time performance and algorithmic complexies are take into account

By using STNR all type of periodicity such as symbol ,segment and partial periodicity and be detected .By using CONV only symbol and segment periodicity is detected. By using WRAP only segment periodicity is detected and Parper detects only partial periodicity

Next feature considered is the time performance.By comparing STNR against Parper,Parper performs well since it specify only partial periodicitywhile STNR is general and finds the periodicity for all patterns which are periodic for any periodic value starting and ending anywhere in the timeseries.So time performance of Parper is better than Stnr but it only detects partial periodicity. By comparing CONV with STNR the time performance of CONV is better than STNR.WRAP has the worse time performance among the four type of algorithms.

By comparing the complexity of the four algorithm CONV perform best running time complexity of O(n logn).The time complexity of STNR is $O(n^2)$ but it perform better because Stnr applies various optimization strategies like redundant period pruning technique.The complexity of WRAP is worse than remaining algorithm since its running time is $O(n^2)$ .Parper has best running algorithm of O(n) but it is not best because it only detect partial periodicity

When comparing the time performance of three algorithms in different noise ratio, the STNR takes more time if the noise ratio is small. When the noise ratio increases the efficiency of STNR increases. The efficiency of WRAP in various noise ratio is worst. When the noise increases then the efficiency of WRAP decreases. CONV and Parper does not have any effect on noise ratio.Also WRAP and STNR have good effect on noise resilience.

Features	STNR	CONV	WARP	PARPER
Periodicity detection	All type	Segment periodicity	Segment periodicit y	Partial periodicity
Complexity	$O(n^2)$	O(nlog n)	$O(n^2)$	O(n)
Time Performance	Average	Best	Worst	Best
Noise resilience	Good	Worst	Good	Worst
Time performance in various noise ratio	Take more time	Does not effect	Does not effect	Does not effect

IV CONCLUSION

The periodicity detection in time series play an important role in many application.Periodic detection algorithm should detect all type of periodicity and also partial periodicity. STNR is a suffix-tree-based algorithm for periodicity detection in time series data. Our algorithm is noise-resilient and run in $O(n^2)$ in the worstcase. The single algorithm can find symbol, sequence (partial periodic), and segment (full cycle) periodicity in the time series. It can also find the periodicity within a subsection of the time series. We performed several experiments to show the time behavior, accuracy, and noise resilience characteristics of the data. We run the algorithm on both real and synthetic data. The reported results demonstrated the power of the employed pruning strategies.WRAP detects segment periodicity and A DTW matrix is used. In CONV authors used fast Fourier transforms to reduce the complexity from $O(n^2)$ to O(nlogn); and used wavelet transform to solve the problem in linear time.Parper uses a apriori method for periodicity detection. This algorithms are compared based on their time performance ,noise resilence and complexities.From the comparison of four algorithms it is infer that STNR is the most efficient algorithm.

V REFERENCES

[1] "Faraz Rasheed, Mohammed Alshalalfa, and Reda Alhajj, Associate Member, IEEE", "Efficient Periodicity Mining in Time Series Databases Using Suffix Trees" , IEEE Transactions On Knowledge And Data Engineering, Vol. 23, No. 1, January 2011

[2] M.G. Elfeky, W.G. Aref, and A.K. Elmagarmid, "Periodicity Detection in Time Series Databases," IEEE Trans. Knowledge and Data Eng., vol. 17, no. 7, pp. 875-887, July 2005.

[3] M.G. Elfeky, W.G. Aref, and A.K. Elmagarmid, "WARP: Time Warping for Periodicity Detection," Proc. Fifth IEEE Int'l Conf. Data Mining, Nov. 2005.

[4] J. Han, Y. Yin, and G. Dong, "Efficient Mining of Partial Periodic Patterns in Time Series Database," Proc. 15th IEEE Int'l Conf. Data Eng., p. 106, 1999.

[5] E.F. Glynn, J. Chen, and A.R. Mushegian, "Detecting Periodic Patterns in Unevenly Spaced Gene Expression TimSeries Using Lomb-Scargle Periodograms," Bioinformatics, vol. 22, no. 3 pp. 310-316, Feb. 2006.

[6] F. Rasheed and R. Alhajj, "STNR: A Suffix Tree Based Noise Resilient Algorithm for Periodicity Detection in Time Series Databases," Applied Intelligence, vol. 32, no. 3, pp. 267-278, 2010.

[7] R. Agrawal and R. Srikant. Fast algorithms for mining association rules. In Proc. 1994 Int. Conf. Very Large DataBases, pages 487–499, Santiago, Chile, September 1994.

[8] S. Ma and J. Hellerstein, "Mining Partially Periodic Event Patterns with Unknown Periods," Proc. 17th IEEE Int'l Conf. Data Eng., Apr. 2001.

[9] D. Brendt and J. Clifford. Using dynamic time warping to find pattern in time series. In AAAI Workshop on Knowledge Discovery in Databases, 1994.

[10] M. Atallah, F. Chyzak, and P. Dumas. *k* RaiJ.doinized Algorithm for Approximate String Matching. *Algorithmic.a,* Vol. 29, No.3, pages 468-486, 200l.
[11] H. Brown et al., "Sequence Variation in S-Antigen Genes of Plasmodium falciparum," Molecular Biology and Medicine, vol. 4, no. 6, pp. 365-376, Dec. 1987.
[12] C.-F. Cheung, J.X. Yu, and H. Lu, "Constructing Suffix Tree for Gigabyte Sequences with Megabyte Memory," IEEE Trans. Knowledge and Data Eng., vol. 17, no. 1, pp. 90-105, Jan. 2005.
[13] R. Grossi and G.F. Italiano, "Suffix Trees and Their Applications in String Algorithms," Proc. South Am. Workshop String Processing, pp. 57-76, Sept. 1993.
[14] J. Han, W. Gong, and Y. Yin, "Mining Segment-Wise Periodic Patterns in Time Related Databases," Proc. ACM Int'l Conf. Knowledge Discovery and Data Mining, pp. 214-218, 1998.
[15] E. Hunt, R.W. Irving, and M.P. Atkinson, "Persistent Suffix Trees and Suffix Binary Search Trees as DNA Sequence Indexes," Technical Report TR2000-63, Univ. of Glasgow, Dept. of ComputingScience, 2000.
[16] P. Indyk, N. Koudas, and S. Muthukrishnan, "Identifying Representative Trends in Massive Time Series Data Sets Using Sketches," Proc. Int'l Conf. Very Large Data Bases, Sept. 2000.
[17] D. Knuth. *The Art of Computer Programming,* Vol. 2,Second Edition, Series in Computer Science and Information Processing, Addison-Wesley, Reading,