Similar Pattern Search for Time-Sectional Oscillation in Huge Plasma Waveform Database

Nakanishi H. [a,*] , Hochin T. [b] , Kojima M. [a] , and LABCOM group [a]

[a] *National Institute for Fusion Science, 322-6 Oroshi-cho, Toki 509-5292, Japan*
[b] *Osaka Prefecture University, 1-1 Gakuen-cho, Sakai 599-8531, Japan*

Abstract

Recently there has been drastic data growth in many fusion plasma experiments. Some computer-aided assistance to find similar waveforms has become indispensable for accelerating data recognition and analysis. A similarity search for slowly varying waveforms, which applies "R-tree" index with Java implementation, was already reported [1]. The next step is the search for time-sectional oscillation patterns. This requires neglecting the initial phase difference between many slices of waveforms. This new algorithm applies power spectrum density (PSD) values instead of FFT complex coefficients. To emphasize the difference between peak PSD frequency of each waveform, its Euclidean distance is multiplied by ω_j/ω_i ($\omega_j > \omega_i$). By applying "SR-tree" and fast numerical library implemented in C/C++, its computations have been accelerated. This enables the system to deal with much larger data sets. These modifications have successfully extended the application range toward them with verified accuracy.

Key words: plasma waveform, pattern matching, similarity search, PSD, Euclidean distance, SR-tree, C++

1. Introduction

In the last few years there has been drastic data growth in many fusion plasma experiments. It is due to the widespread use of 2-D cameras and 3-D multi-channel measurements to diagnose plasma profiles. As these data grow very large, human data recognition becomes more and more difficult. For instance, in the 2004-2005 campaign, LHD's data acquisition system, LABCOM, acquired up to 84.0 GB of raw data for a single long-pulse discharge. This represents a more than twenty fold increase from previous shots. Simultaneously, we also have to deal with the increasing number of plasma waveforms with longer time duration.

In such a situation, some automatic mechanism to search and retrieve similar waveforms would be quite useful. This study aims to realize the computer-aided search for waveform patterns, by which plasma scientists can find and retrieve similar ones all at once. A numerical method to quickly find similar waveforms would be able to accelerate plasma statistical analysis. It could bring about a new breakthrough in detecting the plasma behaviors or events, such as mode locking, instability growth, L-H phase transition, collapse and disruption. Furthermore, it could be extended to include plasma event prediction.

* Corresponding author. Tel.: +81-572-58-2232 fax: +81-572-58-2771

Email address: nakanisi@nifs.ac.jp (Nakanishi H.).

In previous work the similarity search was first applied for the slowly-varying trapezoidal waveforms, such as bolometer signals [1]. It uses the first $2k+1$ and largest $2m$ FFT coefficients to make the characteristic "thumb-indexed" reference in an R-tree structure [2].

An R-tree (Rectangle tree) is a simple extension for multi-dimensional index from the famous B-tree (Balance tree) [3]. By iteratively dividing one into two sub-regions, a B-tree index can reach any point within $\log_2 N$ computations, instead of N. The basic idea that R-trees can accelerate searching in two or more dimensions just as B-trees can accelerate searching one dimensional data.

This R-tree based two-step algorithm was evaluated by using 1000 LHD bolometer traces, and the values of $k = 2$ and $m = 4$ had demonstrated an optimized performance for pattern matching with such slowly-varying waveforms. This study will be oriented toward the fluctuation pattern search. For the next step, therefore, we try to extend it for the waveforms having some oscillation in a definite time period. Then, its computational speed must be improved to deal with the increased number of subdivided waveform chunks.

2. Algorithm Modifications

The basic idea of $2k+1$ and $2m$ two-step searching is also applied as previously. The waveform similarity is defined as the similarity of their characteristic value sets, i.e. major FFT coefficients [4], however, each time chunk has an arbitrary initial phase in its oscillation frequency. For instance, two chunks of the same sine wave may have different initial phases, but they should be matched in similarity. To neglect their initial phase differences, therefore, the power spectrum density (PSD) has been adopted instead of the previous real and imaginary part of FFT coefficients $X(\omega)$,

$$P(\omega) \equiv \sqrt{|\Re(X(\omega))|^2 + |\Im(X(\omega))|^2} \ . \qquad (1)$$

A set of m major PSD components, e.g.
$\vec{P} = (P(\omega_a), P(\omega_b), P(\omega_c), P(\omega_d))$, is also a characteristic vector representing the corresponding chunk of the oscillating waveform. By using m PSD components instead of $2m$ FFT coefficients, we can accurately discard the the initial phase information. This does however introduce some ambiguity to the waveform similarity.

On the other hand, the frequency dispersion of those components raises their importance relatively. When we compare two characteristic vectors $\vec{P_1} = (P_1(\omega_{1a}), P_1(\omega_{1b}), P_1(\omega_{1c}), P_1(\omega_{1d}))$ and $\vec{P_2} = (P_2(\omega_{2a}), P_2(\omega_{2b}), P_2(\omega_{2c}), P_2(\omega_{2d}))$, their amplitudes and frequency dispersion are all the information we can use for similarity check. As the degree of similarity between two chunks can be understood as the similarity of their characteristic vectors, multi-dimensional Euclidean spatial distance L between these two vectors are used to describe their differences.

$$L^2 = |\vec{P_1}(\omega_{1a}, .., \omega_{1d}) - \vec{P_2}(\omega_{2a}, .., \omega_{2d})|^2 \qquad (2)$$

L will be a simple summation of all PSD amplitudes if $\omega_{1a}, .., \omega_{1d}$ and $\omega_{2a}, .., \omega_{2d}$ are different from each other. In this case, their frequency dispersion from the reference set never affect their L values. If a query vector has a very distant frequency component from the reference, it is preferable to assign a worse L value.

To improve the effectiveness of frequency difference, we adopt an emphasizing modification for each Euclidean distance calculation. Figure 1 shows a schematic diagram of L calculation between $\vec{P_i}$ and $\vec{P_j}$. Basically, it adopts ω_j/ω_i times multiplied $P(\omega_i)$ instead of original value when $\omega_j > \omega_i$. In addition, this scheme reduces the number of comparisons to a maximum of $2m$, whereas every possible combination requires $m \times m$ times. Since we have already confirmed that the number of major frequency components is not so high in plasma oscillating waveforms, the previous $k = 2$ ($2k + 1 = 5$) and $m = 4$ (not $2m$) are also used here [1].

3. Computational Results

To improve performance the system has been recoded from Java to C/C++, employing the faster numerical libraries FFT and quick sort [5]. In addition, the index tree structure has been changed from the simple R-tree to SR-tree (Sphere/Rectangle-tree). Which provides a good improvement over R-tree accelerating the nearest neighbor search in multi-dimensional index [6].

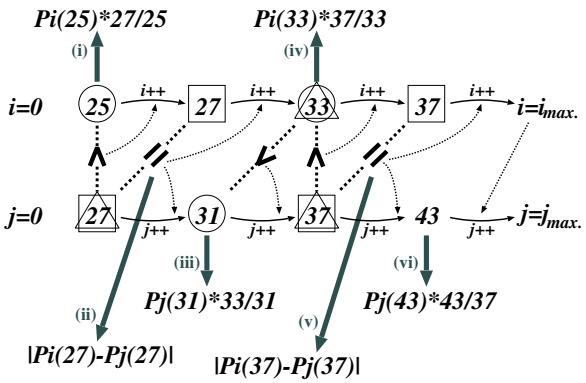


Fig. 1. Comparison scheme between two characteristic vectors $\vec{P}_i(\omega_{25}, \omega_{27}, \omega_{33}, \omega_{37})$ and $\vec{P}_j(\omega_{27}, \omega_{31}, \omega_{37}, \omega_{43})$: Once major PSD components are sorted by ascending frequencies, they are compared with one to one from the lowest. The amplitude of the smaller frequency component is multiplied by ω_j/ω_i $(\omega_j > \omega_i)$ and summed into the distance L. The remaining larger one is compared with the next component. When they have the same frequency, such as ω_{27} or ω_{37}, the amplitude difference between them will be added into L. Here, ω_j/ω_i is always equal to j/i because PSD frequency order ω_i is proportional to its index i.

As a practical test over R-tree, we have applied it to the electron temperature waveforms measured by MIT C-Mod grating polychrometer (GPC) [7] as shown in Fig. 2. Each waveform is cut into tens of shorter time chunks, each of which has 2048 data samples representing about 0.1 ms of time. For a practical evaluation using 1000 shot signals, over 24 000 time-section entries have been processed.

Under a usual computational environment of Pentium 4 3.4 GHz with 1 GB memory running on RedHat

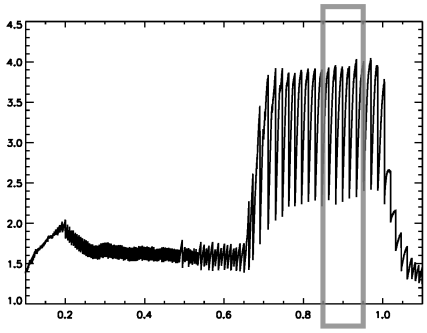


Fig. 2. Example of partly oscillating plasma waveform. The whole waveform is cut into a series of about 0.1 ms time chunks (rectangle), and their characteristic PSD vectors are independently computed to be stored into the index tree for similarity search.

Table 1
Computation time for query search and index construction: Times are user, system, and real elapsed time in second, respectively. Cpu means the percentage of cpu occupation. The chunk and times in conditions show the chunk size in samples (two bytes) and the iteration count.

conditions			index make (/s)				query search (/s)			
type	chunk	times	user	sys	real	cpu	user	sys	real	cpu
R-tree	512	1000	3.66	5.15	14.19	62%	177.9	173.0	356.6	98%
SR-tree	2048	900	12.13	0.23	12.39	99%	8.87	1.44	10.87	94%

Enterprise Linux rel. 3, the elapsed time to compute

1. FFT over 2048 samples
2. calculating PSD values from FFT coefficients
3. extracting first $2k+1$ and major m PSD indexes
4. making its vector entry in the SR-tree

on 900 chunks is 12.39 s, as shown in Table 1. Searching queries for independent 900 chunks take 10.87 s, and the generated index size is 180 224 bytes. The previous Java-based R-tree was generated in 14.19 s by using 1000 of 512 sample chunks, and 1000 times search executions took 356.6 s. Its tree size was 248 996 bytes [1]. This comparison shows a significant improvement especially in the speed of query search. It takes only 0.01 s to complete a single query search. This can be considered comfortable in practical uses of multi-user database.

Figure 3 shows typical results of two queries. Nearest candidates of the similarity search seem to be quite similar to the query pattern. Even a transient oscillating signal can be well matched by this algorithm, as shown in the latter query result. Thus, the index-based searching method for similar plasma waveforms can be concluded to be useful in massively-sized databases.

4. Discussion

As for previously mentioned anxiety about the lost of initial phase difference between multiple frequency components, we did not find cases that correspond to this fault. This is possibly because the sawtooth oscillations in plasma electron temperature have rather simple patterns even though they have some transient phases. In other words, the low-dimensional characteristic vectors seem to be enough for pattern matching of these macroscopic plasma signals.

The capability of the PSD-based searching method has been well demonstrated on the plasma waveforms

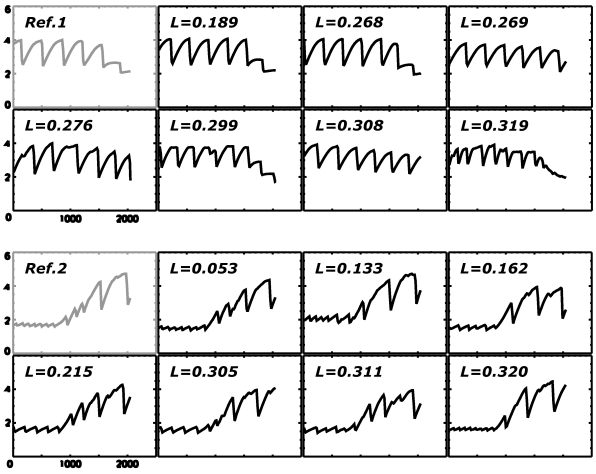


Fig. 3. Typical results of pattern matching search for oscillating wave chunks: The former are the query results for 11th section of #960115027, and the latter are for 7th of #1040212023. In both cases the first (grey) signal is the query chunk. Once 16 candidates have been picked up by the $2k+1$ index tree, and for each of them Euclidean distance from the query L is calculated by means of amplitude summation of its m major PSD components.

having some oscillating parts. However, signals with many zero-crossings still present problems to the algorithm. Even by the human recognition, it is very difficult to distinguish them. To extract their characteristics, not only the time dimensional DFT but also the spatial information would be necessary done as a mutual correlation calculation, or singular value decomposition (SVD).

The computational speed to make an index tree has not much improved even though we applied fast numerical library and C++ implementation. To deal with zero-crossing fluctuation signals, however, above-mentioned complicated analyses might require more cpu time to extract a sophisticated characteristics. Additionally, using three-dimensional data will certainly cause a large rise in the number of index entries. So, further research is needed to improve the calculation speed.

As mentioned below, this work has been carried out as an international research collaboration. The computational results show better performance on C-Mod data than LHD. It also demonstrated the good portability of this algorithm. We can expect, therefore, that such the waveform searching system will be applied to other fusion experiments in the near future.

Acknowledgement

The author would like to express his gratitude to T. Fredian, J. Stillerman, F. Kreisel, and M. Greenwald of MIT PSFC for the fruitful discussion and kindness done him. This work was partly carried out in the framework of the "Japan-US Science and Technology Cooperative Program for Fusion Research", and also supported by the National Institute for Fusion Science under NIFS01KCHH001 and NIFS05ULHH503.

References

[1] H. Nakanishi, T. Hochin, M. Kojima, LABCOM group, Search and retrieval method of similar plasma waveforms, Fusion Eng. Design 71 (1-4) (2004) 189–193.

[2] A. Guttman, R-trees: A dynamic index structure for spatial searching, in: Proc. ACM SIGMOD Int'l Conf. on Management of Data, 18-21 June, Boston, USA, 1984, pp. 47–57.

[3] R. Bayer, K. Unterauer, Prefix B-Trees, ACM Trans. Database Systems (TODS) 2 (1) (1977) 11–26.

[4] D. Radiei, A. Mendelzon, Efficient Retrieval of Similar Time Sequences Using DFT, in: Proc. 5th Int'l Conf. on Foundations of Data Organization (FODO'98), 1998, pp. 249–257.

[5] W. H. Press, B. P. Flannery, S. A. Teukolsky, W. T. Vetterling, Numerical Recipes in C, 2nd Edition, Cambridge University Press, 1992.

[6] N. Katayama, S. Satoh, The SR-tree: An Index Structure for High-Dimensional Nearest Neighbor Queries, in: Proc. ACM SIGMOD Int'l Conf. on Management of Data, 13-15 May, Tucson, USA, 1997, pp. 369–380.

[7] M. Greenwald, et al., Overview of the Alcator C-Mod Program, in: Proc. 20th IAEA Fusion Energy Conference, 1-6 Nov. 2004, Vilamoura, Portugal, 2004.

List of Tables

1 Computation time for query search and index construction: Times are user, system, and real elapsed time in second, respectively. Cpu means the percentage of cpu occupation. The chunk and times in conditions show the chunk size in samples (two bytes) and the iteration count. 3

List of Figures

1 Comparison scheme between two characteristic vectors $\vec{P}_i(\omega_{25}, \omega_{27}, \omega_{33}, \omega_{37})$ and $\vec{P}_j(\omega_{27}, \omega_{31}, \omega_{37}, \omega_{43})$: Once major PSD components are sorted by ascending frequencies, they are compared with one to one from the lowest. The amplitude of the smaller frequency component is multiplied by ω_j/ω_i ($\omega_j > \omega_i$) and summed into the distance L. The remaining larger one is compared with the next component. When they have the same frequency, such as ω_{27} or ω_{37}, the amplitude difference between them will be added into L. Here, ω_j/ω_i is always equal to j/i because PSD frequency order ω_i is proportional to its index i. 3

2 Example of partly oscillating plasma waveform. The whole waveform is cut into a series of about 0.1 ms time chunks (rectangle), and their characteristic PSD vectors are independently computed to be stored into the index tree for similarity search. 3

3 Typical results of pattern matching search for oscillating wave chunks: The former are the query results for 11th section of #960115027, and the latter are for 7th of #1040212023. In both cases the first (grey) signal is the query chunk. Once 16 candidates have been picked up by the $2k + 1$ index tree, and for each of them Euclidean distance from the query L is calculated by means of amplitude summation of its m major PSD components. 4