§29. Improvement of Similarity Retrieval in Fusion Experiment Multimedia Data Archives

Hochin, T., Nomiya, H. (Kyoto Inst. of Tech.),
Nakanishi, H., Kojima, M., Nagayama, Y.,
Ohdachi, S., Emoto, M., Ohsuna, M.

Experiments of the fusion phenomena produce a lot of sequences of time-varying values which form waveforms. If the waveforms similar to a desired one can be obtained by using computer system, the burden of researchers in searching similar waveforms will extremely be decreased. We have proposed an indexing method of the severely and quickly changing plasma waveforms for accelerating search and retrieval of their subsequences [1]. It is, however, not clear that the insertion of feature values as well as data themselves could be accomplished within the time between the measurements of two successive shots. It is also considered to be hard to contain all of feature values in an index. This paper studies on the indexing methods for addressing to these issues.

In the fusion plasma experiments, a shot is recorded every about three minutes. The major sampling frequency is 1 MHz. The typical time of a shot is 10 sec. Approximately 10 million points of time series data are obtained every shot. We separate time series data into segments, each of which contains 4096 points [1]. About 2500 segments are obtained from a shot. The Fast Fourier Transform (FFT) is applied to a segment. As the fifteen major FFT coefficients are used for a segment, it is represented with a 15-dimentional point. Therefore, 2500 15-dimentional points must be stored within three minutes. Moreover, we have 14400 shots a year because we have about 180 shots a day, about 80 days a year. Therefore, 14400 times 2500 points must be stored into an index in a year.

Additionally, it is impossible to handle files larger than 2GB on the 32bit file system under the control of the 32bit Operating System. The number of 15-dimensional point data stored in a 2GB file is about 10 million. Approximately 36 million, the number of data obtained in a year, exceeds the limitation of the file size.

Two methods are proposed in order to resolve the issues described above. In the first method, an index is created one by one. When the size of an index reaches the limitation, a new index is created. This method is called the one_by_one method. In the second method, several indexes are created in advance. Data are inserted into the indexes according to the Round-robin scheme. This method is called the Round-robin method.

The proposed methods are compared with the method without any split. Random data ranging from 1 to 6,000,000 of each dimension are used in this experiment. In the one_by_one method, the limitation of the number of data is set to 700,000. In the Round-robin method, the number of indexes is 8. A computer (Intel Xeon E5620 2.40GHz, 15.6GB memory, RAID5) is used in the experiment.

In the experiment of insertion, the times of inserting 2,500 data are measured. The total numbers of the data stored in the indexes are 1, 2, 3, 4, and 5 million. In this experiment, the data on the main memory are compulsorily released before every measurement. The result is shown in Fig. 1. In the cases of the no-split method and the Round-robin method, insertion time becomes longer according to the number of data.

In the experiment of retrieval, the times of retrieving 500,000 nearest neighbor data are measured from the index storing 5,000,000 data. The indexes are accessed through eight threads in parallel. The data on the main memory are not released before the retrieval. The retrieval times are shown in Fig. 2. Both of the proposed methods show better performance than the no-split method according to the number of divisions. The performance of the proposed methods is more than twice of the no-split method when the numbers of divisions are four and eight.

The evaluation experiments of insertion and retrieval showed that the one-by-one method provides the best insertion performance and that the proposed methods provide better retrieval performance than the no-split method. Parallel processing to the indexes divided by the proposed methods could accelerate the retrieval.

1) Hochin, T., Yamauchi, Y., Nakanishi, H., et al.: Indexing of plasma waveforms for accelerating search and retrieval of their subsequences, Fus. Eng. Des., **85**(5) (2010) 649-654.

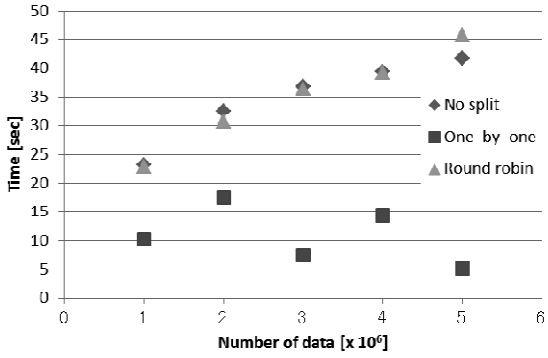

Fig. 1. Insertion time.

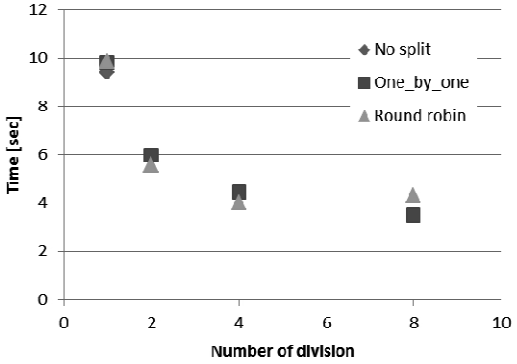

Fig. 2. Retrieval time.