Journal of Korea Multimedia Society Vol. 7, No. 11, November 2004(pp. 1505-1514)

시계열 데이터베이스에서 DFT-기반 다차원 인덱스를 위한 물리적 데이터베이스 설계

김상욱[†], 김진호[††], 한병일[†††]

요 약

시퀀스 매칭은 시계열 데이터베이스로부터 질의 시퀀스와 변화의 추세가 유사한 데이터 시퀀스들을 검색하는 연산이다. 기존의 대부분의 연구에서는 효과적인 시퀀스 매칭을 위하여 다차원 인덱스를 사용하며, 데이터 시퀀스를 이산 푸리에 변환(Discrete Fourier Transform: DFT)한 후, 단순히 앞의 두 개 내지 세 개의 DFT 계수만을 구성 속성(organizing attributes)으로 사용함으로써 고차원의 경우 발생하는 차원 저주(dimensionality curse) 문제를 해결한다. 본 논문에서는 기존의 단순한 기법이 가지는 성능 상의 문제점들을 지적하고, 이러한 문제점들을 해결하는 최적의 다차원 인덱스 구성 기법을 제안한다. 제안된 기법은 대상이 되는 시계열 데이터베이스의 특성을 사전에 분석함으로써 변별력이 뛰어난 요소들을 다차원 인덱스의 구성 속성으로 선정하며, 비용 모델(cost model)을 기반으로 한 시퀀스 매칭 비용의 추정을 통하여 다차원 인덱스에 참여하는 최적의 구성 속성의 수를 결정한다. 제안된 기법의 우수성을 규명하기 위하여 실험을 통한 기존 기법과의 성능 비교를 수행하였다. 실험 결과에 의하면, 제안된 기법은 기존의 기법에 비교하여 매우 큰 성능 개선 효과를 가지는 것으로 나타났다.

Physical Database Design for DFT-Based Multidimensional Indexes in Time-Series Databases

Sang-Wook Kim[†], Jin-Ho Kim[††], Byung-Il Han[†††]

ABSTRACT

Sequence matching in time-series databases is an operation that finds the data sequences whose changing patterns are similar to that of a query sequence. Typically, sequence matching hires a multi-dimensional index for its efficient processing. In order to alleviate the *dimensionality curse* problem of the multi-dimensional index in high-dimensional cases, the previous methods for sequence matching apply the Discrete Fourier Transform(DFT) to data sequences, and take only the first two or three DFT coefficients as organizing attributes of the multi-dimensional index. This paper first points out the problems in such simple methods taking the first two or three coefficients, and proposes a novel solution to construct the optimal multi-dimensional index. The proposed method analyzes the characteristics of a target database, and identifies the organizing attributes having the best discrimination power based on the analysis. It also determines the optimal number of organizing attributes for efficient sequence matching by using a cost model. To show the effectiveness of the proposed method, we perform a series of experiments. The results show that the proposed method outperforms the previous ones significantly.

Key words: Time-Series Databases(시계열 데이터베이스), Multidimensional Indexes(다차원 색인), Sequence Matching(시퀀스 매칭)

※ 교신저자(Corresponding Author) : 김상욱, 주소 : 서울시 성동구 행당 1동(133-791), 전화 : 02)2290-1736, FAX : 02)2290-1886, E-mail : wook@hanyang.ac.kr
접수일 : 2004년 1월 7일, 완료일 : 2004년 6월 10일
[†] 한양대학교 정보통신대학 정보통신학부 부교수
[††] 강원대학교 컴퓨터과학과 교수
(E-mail : jhkim@kangwon.ac.kr)
[†††] (주) 참좋은 인터넷 개발팀장
(E-mail : cshbi@sogood.co.kr)

1. 서 론

데이터 시퀀스(data sequence: 이후부터 간략히 시퀀스라 칭함)는 한 객체와 관련된 속성 값을 시간별로 측정하여 저장한 실수 값의 연속이다. 시계열 데이터베이스(time-series database)란 이러한 시퀀스들의 집합이며, 대표적인 예로는 주가 데이터, 환율 데이터, 기온 데이터, 제품 판매량 데이터, 기업 성장률 데이터 등을 들 수 있다[1,9]. 시퀀스 매칭(sequence matching)이란 주어진 질의 시퀀스(query sequence)와 변화의 패턴이 유사한 시퀀스들을 시계열 데이터베이스로부터 찾아내는 연산이다[1]. 시퀀스 매칭은 시계열 데이터베이스를 기반으로 하는 데이터 마이닝(data mining) 및 데이터 웨어하우징(data warehousing) 분야에서 중요한 연산으로 사용된다[17].

시퀀스 매칭에 관한 많은 연구에서는 길이 n의 시퀀스를 n 차원 공간상의 한 점으로 모델링 한다. 또한, 길이가 동일한 서로 다른 두 시퀀스 $X(=[x_0, x_1, \ldots, x_{n-1}])$와 $Y(=[y_0, y_1, \ldots, y_{n-1}])$간의 유사성을 측정하는 척도로서 아래와 같이 정의되는 유클리드 거리(Euclidean distance) D(X, Y)를 사용한다[1,7-9][Gol95s][17,18][1]. 주어진 질의 시퀀스 X로부터 사용자에 의하여 지정된 유사 허용치 ε 이하의 유클리드 거리 내에 있는 시퀀스 Y는 X와 유사하다고 간주된다.

$$D(X,Y) = \sqrt{\sum_{i=0}^{n-1} (x_i - y_i)^2}$$

시퀀스 매칭은 전체 매칭(whole matching)과 부분 매칭(subsequence matching)으로 구분된다[1]. 전체 매칭은 시퀀스들과 질의 시퀀스의 길이가 동일하다는 조건하에 수행되며, 질의 시퀀스와 유사한 시퀀스를 검색한다. 반면, 부분 매칭은 이러한 조건이 불필요하며, 질의 시퀀스와 유사한 서브시퀀스를 포함하는 시퀀스를 검색한다.

유클리드 거리만을 이용한 시퀀스 매칭을 통해서는 사용자가 원하는 시퀀스들을 검색하지 못하는 경우가 빈번하게 발생한다. 따라서 응용 분야의 특성에 따라 유사한 정도를 유연하게 정의할 수 있도록 전처리 변환(pre-processing)을 지원하기도 한다. 참고문헌 [1,9]에서는 전처리 변환이 상용되지 않았으나, 최근에는 정규화(normalization)[2,8,11,14,17], 이동평균(moving average)[17,18,13], 타임 워핑(time warping)[5,16,17,21] 등 다양한 변환을 지원하는 기법 등이 제안되었다.

대부분의 시퀀스 매칭에서는 빠른 검색을 위하여 R-트리 계열(R-tree family)[3]의 다차원 인덱스(multidimensional index)[Gra98]를 사용한다. 다차원 인덱스에서는 인덱스 액세스 성능이 차원 수의 지수 함수로 저하되는 고차원 문제(dimensionality curse)가 발생한다[6,20]. 특히, 시퀀스 매칭에서 발생하는 높은 차원 수는 고려할 때, 시퀀스 매칭에서의 이러한 고차원 문제는 더욱 심각하다[1,9]. 이를 해결하기 위하여 기존의 연구에서는 고차원 공간상의 점들을 저차원 공간상의 점들로 변환하는 방법을 사용한다. 사용 가능한 변환 함수로는 DFT(Discrete Fourier Transform), DCT(Discrete Cosine Transform), 웨이블릿 변환(Wavelet Transform) 등이 있으나, 기존의 연구에서는 DFT를 가장 널리 사용하고 있다[1,8,9,11,17,18,13,14].

본 논문에서는 시퀀스 매칭의 성능 극대화를 위한 최적의 다차원 인덱스 구성 방안에 관하여 다루고자 한다. 기존의 기법들에서는 각 시퀀스를 DFT한 후, 앞의 몇 개(<<n)의 DFT 계수만을 다차원 인덱스를 위한 구성 속성(organizing attributes)으로서 사용한다. 여기서, 구성 속성이란 다차원 인덱스를 구성하는데 참여하는 속성을 의미한다[12]. 그러나 최적의 성능을 얻기 위하여 다차원 인덱스의 구성 속성으로서 어떤 계수를 몇 개 선택해야 하는가에 대한 체계적인 지침에 대해서는 아직 논의된 바 없다.

본 논문에서는 먼저 앞에 위치한 몇개의 DFT 계수만을 단순히 선택하는 기존 기법들이 가지는 약점들을 지적하고, 이러한 문제점들을 해결하는 새로운 기법을 제안한다. 본 논문에서는 이 문제를 물리적 데이터베이스 설계(physical database design) 관점에서 접근한다. 제안된 기법은 먼저 응용의 대상이 되는 시계열 데이터베이스의 특성을 사전에 분석한다. 이는 시퀀스로부터 추출된 요소들 중 변별력이 뛰어난 것들을 파악하기 위한 것이다. 이 결과, 변별력이 뛰어난 요소들을 다차원 인덱스의 구성 속성으로 선정한다. 요소들의 변별력이 뛰어날수록 다차원

1) 유사성을 측정하기 위한 척도로서 유클리드 거리 이외에 맨하탄 거리(Manhattan distance) 등 다른 척도도 사용될 수 있다[2].

인덱스 검색을 통하여 반환되는 후보 수가 줄어듦으로 빠른 시퀀스 매칭이 가능하다. 또한, 본 연구에서는 다차원 인덱스 검색 비용과 후처리 비용을 함께 최소화할 수 있는 최적의 다차원 인덱스의 구성 속성의 수를 결정하는 기법을 제시한다. 제안된 기법의 우수성을 검증하기 위하여 실험에 의한 성능 평가를 수행한다. 실험 결과에 의하면, 제안된 기법은 기존의 기법에 비교하여 매우 큰 성능 개선 효과를 가지는 것으로 나타났다.

본 논문의 구성은 다음과 같다. 먼저, 제 2장에서는 본 논문에서 해결하고자 하는 문제를 정의하고, DFT를 기반으로 하는 기존 기법들의 공통적인 문제점들을 지적한다. 제 3장에서는 이러한 문제점들을 해결하기 위하여 최적의 다차원 인덱스 구성을 위한 새로운 기법을 제안한다. 제 4장에서는 실험을 통한 성능 분석을 통하여 제안된 기법의 우수성을 검증한다. 제 5장에서는 본 논문을 요약하고, 결론을 내린다.

2. 연구 동기

본 장에서는 본 연구에서 해결하고자 하는 문제를 정의하고, 본 논문의 연구 동기로서 DFT를 기반으로 하는 기존 기법의 약점들을 지적한다.

2.1 DFT 기반의 기존 기법

시퀀스 매칭에서는 빠른 검색을 위하여 다차원 인덱스(multidimensional index)[Gra98]가 널리 사용된다. 시계열 데이터베이스에서 시퀀스는 매우 긴 것이 일반적이므로, 다차원 인덱스의 고차원 문제(dimensionality curse)[6,20]가 발생한다[1,9]. 기존의 많은 시퀀스 매칭 기법에서는 이를 해결하기 위하여 DFT를 사용한다[1,8,9,11,17,18,13,14]. 다음의 식에 의하여 시간 도메인(time domain)상의 n개의 실수 값으로 구성된 시퀀스 x(=[x_i], i=0,..,n-1)는 주파수 도메인(frequency domain)상의 n개의 복소수 값으로 구성되는 X(=[X_F], F=0,..,n-1)로 DFT된다.

$$X_F = 1/\sqrt{n} \sum_{i=0}^{n-1} x_i \exp(-j 2\pi Fi/n),$$
$$F = 0, 1, \ldots, n-1$$

시퀀스 매칭과 관련된 DFT의 중요한 두 가지 특성은 다음과 같다.

- 특성 1: DFT 이전의 x의 에너지와 DFT 이후의 X의 에너지는 같다. 이 결과, 임의의 두 시퀀스 간의 유클리드 거리는 DFT 이후에도 그대로 보존된다(Parseval 정리[15]).
- 특성 2: 브라운 잡음(brown noise)의 특성을 갖는 시퀀스들의 DFT 변환 후의 대부분의 에너지는 가장 앞쪽의 몇 개의 DFT 계수에 집중된다[15]. 또한, 주가 데이터, 환율 데이터 등은 브라운 잡음의 특성을 갖는다.

기존의 기법에서는 이러한 두 가지 특성을 이용하여 각 시퀀스 x를 DFT한 새로운 시퀀스 X에 대하여 앞쪽에 위치하는 k(<<n)개의 DFT 계수만을 이용하여 2*k개의 구성 속성을 가지는 다차원 인덱스를 구성한다. 여기서, 2*k 차원이 되는 이유는 주파수 도메인에서 DFT 계수가 실수부와 허수부로 구성되는 복소수의 형태를 갖기 때문이다. 이와 같이, n과 비교하여 훨씬 작은 2*k 차원의 인덱스가 생성되므로 인덱스를 위한 저장 공간 및 액세스 시간 비용을 크게 줄일 수 있다. DFT의 특성 1에 의하여 이 기법에서는 질의 결과에 포함되어야 할 시퀀스를 찾아내지 못하는 착오 기각(false dismissal)[1]이 발생하지 않는다.

구성 속성으로 채택되지 못하는 k개 이외의 DFT 계수들로 인하여 에너지 손실이 발생한다. 따라서 다차원 인덱스를 통하여 찾아낸 시퀀스들 중에서는 질의 시퀀스와의 실제 유클리드 거리가 ε를 초과하는 내에 있지 않는 것들이 존재할 수 있는데, 이를 착오 채택(false alarm)[1]이라 한다. 따라서 다차원 인덱스를 통하여 검색한 후보 시퀀스들을 실제로 액세스함으로써 올바른 결과만을 선별하는 작업이 추가된다.

2.2 문제 정의

시퀀스 매칭을 처리하기 위한 전체 비용은 크게 후보 필터링을 위한 인덱스 액세스 비용과 착오 채택 해결을 위한 시퀀스 액세스 비용으로 구성된다. 순차 스캔을 이용하는 경우, 후보 필터링 과정이 없으므로 인덱스 액세스 비용은 0이지만, 시퀀스들을 모두 액세스해야 하므로 시퀀스 액세스 비용이 매우 크다. 반면, 시퀀스 전체 정보를 이용하여 다차원 인덱스를 구성하는 경우에는 착오 채택이 발생하지 않으므로 객체 액세스 비용은 0이지만, 고차원 문제로 인하여

인덱스 액세스 비용이 매우 커진다.

DFT 등과 같은 저차원 변환을 이용하는 기법들은 위의 양극단 사이에 위치하며, 인덱스 액세스 비용과 객체 액세스 비용을 모두 허용하되 극단적인 비용의 크기를 줄이려는 것을 목표로 한다. DFT 기반의 기존 기법에서 큰 k 값(=구성 속성으로 사용하는 DFT 계수의 수)을 사용하는 경우, 에너지의 손실이 작으므로 착오 채택 해결을 위한 객체 액세스 비용이 작아진다. 반면, 구성 요소의 수가 커지므로 고차원 문제로 인하여 인덱스 액세스 비용이 커진다. 반대로 작은 k값을 사용하는 경우, 구성 속성의 수가 작아지므로 인덱스 액세스 비용이 작아진다. 반면, 에너지 손실이 커지므로 착오 채택 해결을 위한 객체 액세스 비용이 커진다.

본 논문에서는 시퀀스 매칭의 성능 극대화를 위한 최적의 다차원 인덱스 구성 방안에 관하여 다루고자 한다. 본 논문에서는 이 문제를 다음과 같은 정의를 기반으로 한 물리적 데이터베이스 설계(physical database design) 관점에서 접근하고자 한다.

- 정의 1: 본 논문에서는 *시계열 데이터베이스에서 다차원 인덱스를 위한 물리적 데이터베이스 설계*를 시계열 데이터베이스와 여기서 사용될 시퀀스 매칭 질의들이 주어질 때, 이러한 질의들의 처리하기 위한 전체 비용을 최소화하는 최적의 다차원 인덱스의 특성을 결정하는 문제라 정의한다.

여기서, 전체 비용은 인덱스 액세스 비용과 시퀀스 액세스 비용을 합한 것을 의미하며, 다차원 인덱스의 특성이란 다차원 인덱스를 위한 구성 속성의 수와 구성 속성의 종류를 의미한다[2].

2.3 DFT 기반 기존 기법의 약점

DFT를 기반으로 하는 기존의 기법에서는 각 시퀀스를 DFT한 새로운 시퀀스 X에 대하여 앞쪽에 위치하는 k(<<n)개의 DFT 계수만을 이용하여 2*k개의 구성 속성을 가지는 다차원 인덱스를 구성한다. k 값으로는 2 혹은 3을 널리 사용한다[1,8,9,11,17,18,13,14]. 이러한 기존의 기법은 최적의 다차원 인덱스를 구성한다는 목표를 달성하는데 있어 다음과 같은 약점들을 갖는다.

첫째, 구성 속성을 선정하는 기준의 문제이다. 일반적으로, 인덱스의 구성 속성을 선정하는 요건은 해당 구성 속성을 이용하여 데이터베이스내의 객체들을 얼마만큼 효과적으로 변별할 수 있는가 하는 것이다. 기존 기법에서 DFT 계수들 중 앞쪽에 위치하는 것들을 취한다는 것은 단순히 DFT 계수가 가지는 평균 에너지 크기를 구성 속성을 선정하는 기준으로서 사용한다는 것이다. 그러나 평균 에너지의 크기가 큰 DFT 계수가 반드시 뛰어난 변별력을 가지는 것은 아니다. 예를 들어, 평균 에너지의 크기가 가장 큰 DFT 계수라 할지라도 데이터베이스 내의 모든 시퀀스들이 이 DFT 계수에 대하여 동일한 값을 갖는다면, 이 DFT 계수는 구성 속성으로서의 가치가 없다. 따라서 구성 속성을 선정하기 위한 새로운 기준이 필요하다.

둘째, 다차원 인덱스를 위한 구성 속성의 수에 대한 명확한 지침이 없다는 것이다. 전술한 바와 같이 기존의 기법에서는 응용의 대상이 되는 데이터베이스의 특성에 관계없이 k값으로서 2 혹은 3을 추천하고 있다. 그러나 시퀀스 매칭을 최적으로 처리하기 위한 다차원 인덱스의 구성 속성의 수는 응용 대상이 되는 데이터베이스의 특징과 사용될 시퀀스 매칭 질의의 특징에 따라 상이하다. 따라서 이러한 특징들을 반영하여 구성 속성의 수를 결정할 수 있는 체계적인 방법이 필요하다.

셋째, 구성 속성을 선정하는 단위의 문제이다. 기존 기법에서는 DFT 계수를 구성 속성을 선정하는 단위로서 사용한다. 즉, DFT 계수는 실수 부분과 허수 부분으로 구성되므로 하나의 DFT 계수가 선정되면, 이 계수의 실수 부분과 허수 부분이 모두 구성 속성으로서 참여하게 된다. 그러나 같은 DFT 계수에 대해서도 실수 부분과 허수 부분의 변별력은 차이가 있을 수 있다. 따라서 구성 속성을 선정하는 단위를 보다 세분화하는 것이 필요하다.

3. 제안하는 기법

본 장에서는 제 2.3절에서 기술한 기존 기법의 문

2) 참고 문헌 [12]에서와 같이 영역 분할 전략(region splitting strategy)을 물리적 데이터베이스 설계 요소로 고려할 수 있으나, 시퀀스 매칭 질의에서는 다차원 인덱스의 특정 구성 속성에 대한 선호도가 별도로 존재하지 않으므로 본 연구에서는 이를 특성에서 배제하였다.

제점을 해결함으로써 최적의 다차원 인덱스를 구성하는 새로운 기법을 제안한다.

3.1 기본 아이디어

제안된 기법에서는 기존 기법이 가지는 약점들을 다음과 같은 방법으로 해결한다.

3.1.1 구성 속성의 선정 기준

데이터베이스 내의 시퀀스들이 어떤 속성의 도메인 상의 특정 구간 내에만 집중적으로 분포된다면, 그 속성은 인덱스를 위한 구성 속성으로서의 가치가 떨어진다. 어떤 속성의 변별력이 뛰어나다는 것은 시퀀스들이 해당 속성의 전체 도메인 내에 산발적으로 분포한다는 것을 의미한다. 본 연구에서는 구성 속성을 선정하는 기준으로서 계수의 에너지 자체가 아닌 에너지의 표준 편차를 이용한다. 구성 속성으로 선정된 계수의 에너지 표준 편차가 클 경우 시퀀스들이 그 계수와 대응되는 도메인 상에서 산발적으로 분포함을 나타내며, 이것은 그 계수가 시퀀스들을 변별하는데 유용함을 의미한다.

3.1.2 구성 속성의 선정 단위

본 연구에서는 기존 기법과는 달리 각 DFT 계수의 실수 부분과 허수 부분을 독립적인 요소로서 간주한다. 이를 위하여 DFT 계수의 실수 부분과 허수 부분을 요소 계수(element coefficients)라 정의한다. 이와 같이, 실수 부분과 허수 부분을 분리하여 선정하는 이유는 변별력이 떨어지는 요소 계수가 구성 속성으로서 참여하는 것을 방지하기 위한 것이다.

3.1.3 구성 속성의 수

구성 속성의 수 k가 결정된 경우, 위에서 논의한 구성 속성의 선정 기준과 단위를 이용하여 좋은 성능을 가지는 다차원 인덱스를 구성할 수 있다. 그러나 또 하나의 중요한 문제는 과연 어떤 k 값을 사용하는가 하는 것이다. 일반적으로 최적의 다차원 인덱스 구성을 위한 k 값은 데이터베이스의 특성과 질의 특성에 큰 영향을 받는다. 따라서 본 연구에서는 비용 공식(cost formula)을 이용하여 최적의 다차원 인덱스 구성을 위한 k 값을 결정한다. 즉, 응용의 대상이 되는 데이터베이스의 특성과 질의 특성을 사전에 분석하고, 비용 공식을 기반으로 시퀀스 매칭 질의의 처리비용을 추정함으로써 이 비용을 최소화하는 k 값을 선정한다.

3.2 비용 공식

시퀀스 매칭을 위한 비용은 다차원 인덱스를 액세스하는 비용과 실제 시퀀스를 액세스하는 비용으로 구성된다. 데이터베이스 환경에서는 디스크 액세스가 처리비용의 대부분을 차지하므로 본 연구에서는 디스크 액세스 횟수를 고려하는 비용 공식을 사용한다. 시퀀스 매칭에서 사용되는 다차원 인덱스로서 현재 가장 널리 사용되는 R^*-트리를 사용한다고 가정한다. 또한, 시퀀스들은 별도의 클러스터링(clustering) 특성을 고려하지 않고 독립적으로 저장된다고 가정한다. 본 연구에서는 시퀀스 매칭을 위한 비용의 추정을 위하여 프랙탈 차원(Fractal dimension) D_0와 상관 프랙탈 차원(correlation Fractal dimension) D_2를 이용한 참고 문헌 [10,4]의 비용 공식을 확장하여 사용한다.

3.2.1 프랙탈 차원 및 상관 프랙탈 차원

다차원 공간상에 분포하는 점들의 집합이 모든 스케일에 대하여 자기 유사성(self-similarity)을 보일 때, 이를 프랙탈이라 한다[19]. 여기서, 점은 본 논문의 응용에서 나타나는 각 시퀀스와 일대일 대응된다. 아래 공식은 프랙탈 특성을 갖는 점들의 집합에 대한 프랙탈 차원(q=0)과 상관 프랙탈 차원(q=2)을 수학적으로 정의한 것이다.

$$D_q \equiv \frac{1}{q-1} \times \frac{\partial \log \sum_i p_i^q}{\partial \log r} = \text{constant} \quad (r \in [r_1, r_2]) \qquad \text{(공식 3.1)}$$

여기서 r은 다차원의 공간을 정방형 그리드 영역(grid cell)들로 분할하는 경우 각 그리드 영역의 한 변의 길이이며, p_i는 적어도 하나의 점을 포함하는 그리드 영역의 수를 의미한다. 실질적으로 프랙탈 차원과 상관 프랙탈 차원은 데이터베이스의 분석을 기반으로 하는 참고 문헌 [4]의 알고리즘을 수행함으로써 구할 수 있다.

3.2.2 인덱스 액세스 비용

본 연구에서는 인덱스 액세스를 위한 비용 추정을 위하여 참고 문헌 [10]에서 제안한 다음의 공식을 그

대로 이용한다. 프랙탈 차원 D_0를 이용하는 이 공식은 R*-트리를 이용하여 범위 질의(range query)를 처리할 때 발생하는 인덱스 액세스 비용을 추정하기 위한 것이다.

$$\mathrm{IndexAccessCost} = \sum_{j=0}^{h-1} \frac{N}{C_{eff}^{\,h-j}} \prod_{i=0}^{k} (\sigma_j + \varepsilon)$$

(공식 3.2)

여기서, N은 전체 시퀀스들의 수, k는 다차원 인덱스의 구성 속성의 수, C_{eff}는 R*-트리의 각 노드가 가지는 평균 엔트리의 수, ε는 유사 허용치, σ_j (j=0, 1,...,h-1)는 R*-트리의 j 번째 단계 내의 각 엔트리가 표현하는 영역의 각 차원에서의 변의 길이, h는 R*-트리의 높이 (= $\log_{C_{eff}} N$)를 의미한다. 이때 σ_j는 점들의 집합에 대한 프랙탈 차원 D_0를 기반으로 다음 식을 통하여 추정된다.

$$\sigma_j = \left(\frac{C_{eff}^{\,h-j}}{N} \right)^{\frac{1}{D_0}}$$

3.2.2 시퀀스 액세스 비용

참고 문헌 [4]에서는 n 차원 공간상에서 N개의 점들이 분포할 때, 지름이 ε인 구 형태의 범위 질의에서 검색되는 점들의 수를 추정하기 위한 다음과 같은 다음의 공식을 제안한 바 있다. 여기서 D_2는 상관 프랙탈 차원, $\mathrm{Vol}(\varepsilon, \boxdot)$ 는 한 변의 길이가 ε인 정방형 영역의 체적, $\mathrm{Vol}(\varepsilon, \odot)$는 반지름이 ε인 구형 영역의 체적을 의미한다.

$$\mathrm{NumPointsRetrieved} = (\mathrm{Vol}(\varepsilon, \odot) / \mathrm{Vol}(\varepsilon, \boxdot))^{D_2/n} \times (N-1) \times 2^{D_2} \times (\varepsilon)^{D_2}$$

(공식 3.3)

본 연구에서는 후보 시퀀스 액세스를 위한 비용 추정은 이 (공식 3.3)을 기반으로 한다. 후보 시퀀스들이란 시퀀스 매칭의 최종 결과 시퀀스가 아니라 인덱스 액세스의 결과로 반환되는 시퀀스를 의미한다. 따라서 (공식 3.3)은 그대로 사용할 수 없고, 다음과 같은 두 가지 사항을 추가로 고려해야 한다. 첫째, n은 원래의 차원 수가 아니라 다차원 인덱스를 위한 구성 속성의 수 k로 바뀌어야 한다. 둘째, 다수의 요소 계수들이 구성 속성 선정 과정에서 제외됨으로써 발생하는 에너지 손실로 인하여 구성 속성의 수가 적어질수록 반지름이 ε인 구형 질의 영역 내에 포함되는 최종 결과 시퀀스이외의 다른 시퀀스들의 수가 증가한다. 따라서 이러한 경향을 반영할 수 있도록 (공식 3.3)을 수정하여야 한다.

구성 속성의 대상에서 한 요소 계수를 제외하게 되면, 주어진 질의 시퀀스와의 거리가 ε이상 ($\varepsilon + \alpha$)이하인 시퀀스들이 추가로 후보 시퀀스로 들어가게 된다. 여기서 α는 제외된 요소 계수로 인하여 감소된 질의 시퀀스와 시퀀스의 유클리드 거리이다. 이렇게 추가로 후보 시퀀스가 되는 객체 수를 예측하기 위한 방법은 질의 영역 ε을 ($\varepsilon + \alpha$)로 확대시키는 것이다.

따라서 본 논문에서는 (1) 구성 속성들과 대응되는 k 차원 공간상에서 분포하는 N개의 시퀀스들을 대상으로 (2) 지름이 ε인 구 형태의 범위 질의를 (3) k 차원 인덱스를 이용하여 검색할 때, 결과로 나타나는 후보 시퀀스들의 수를 추정하기 위한 공식을 아래와 같이 제안한다.

$$\mathrm{ObjectAccessCost} = (\mathrm{Vol}(\varepsilon+\alpha, \odot)/\mathrm{Vol}(\varepsilon+\alpha, \boxdot))^{D_2/k} \times (N-1) \times 2^{D_2} \times (\varepsilon+\alpha)^{D_2}$$

(공식 3.4)

여기서 D_2는 구성 속성들과 대응되는 k 차원 내에 존재하는 시퀀스들의 상관 프랙탈 차원, α는 다차원 인덱스에 참여하지 않는 요소 계수의 표준 편차들의 합, $\mathrm{Vol}(\varepsilon, \boxdot)$ 는 한 변의 길이가 ε인 정방형 영역의 체적, $\mathrm{Vol}(\varepsilon, \odot)$는 반지름이 ε인 구형 영역의 체적을 의미한다.

3.3 알고리즘

다차원 인덱스의 최적 구성을 위한 구체적인 알고리즘은 그림 3.1과 같다. 단계 1에서는 데이터베이스를 액세스함으로써 프랙탈 차원과 상관 프랙탈 차원 측면에서의 시계열 데이터 특성을 분석하고, 단계 2와 3에서는 DFT를 수행함으로써 전체 시퀀스들에 대한 요소 계수들의 표준 편차들을 구한다. 단계 4에서는 요소 계수들의 가능한 각 조합에 대하여 다차원 인덱스를 구성하였을 경우의 향후 시퀀스 매칭의 비용을 3.2절에서 제시한 비용 모델을 통하여 예측한다. 단계 5에서는 이중 최소 값을 갖는 경우를 최적의 다차원 인덱스 구성으로 간주하고, 해당 요소 계수들을 반환한다.

입력: 길이 n인 N개의 시퀀스들을 포함하는 시계열 데이터베이스와 빈번하게 사용될 q개의 시퀀스 매칭 질의

출력: 다차원 인덱스의 구성 속성으로 사용되는 요소 계수 집합

1. 데이터베이스 내의 시퀀스들을 차례로 액세스하여 프랙탈 차원 D_0와 상관 프랙탈 차원 D_2를 구한다.
2. 데이터베이스 내의 모든 시퀀스들을 대상으로 DFT를 수행한다.
3. DFT에 의한 요소 계수들을 표준 편차를 기준으로 내림차순으로 정렬한 후, 요소 계수의 식별자를 배열 EC(i)에 넣는다 (1<=i<=2*n).
4. 각 요소 계수 EC(i)에 대하여 (1<=i<=2*n)
   4.1. EC(1), EC(2), ..., EC(i)내의 존재하는 i개의 요소 계수들을 구성 속성로 사용하는 다차원 인덱스가 존재한다고 가정한다.
   4.2. 주어진 시퀀스 매칭 질의 Q(j)에 대하여 (1<=j<=q) 제 3.2절에서 소개한 추정 공식을 이용하여 단계 4.1에서 가정한 다차원 인덱스 액세스 비용 IndexAccessCost(i,j)와 객체 액세스 비용 ObjectAccessCost(i,j)를 구한다.
   4.3. q개의 시퀀스 매칭 질의를 단계 4.1에서 가정한 다차원 인덱스를 이용하여 모두 처리하는데 필요한 비용 TotalCost(i)를 IndexAccessCost(i,j)와 ObjectAccessCost(i,j)를 이용하여 구한다.
5. TotalCost(i) 중 최소 값을 갖는 TotalCost(w)를 찾는다.
6. EC(1), EC(2), ..., EC(w)내에 나타난 요소 계수를 최적의 다차원 인덱스를 위한 구성 속성으로서 선정한다.

그림 3.1. 시계열 데이터베이스를 위한 다차원 인덱스의 최적 구성 알고리즘.

4. 성능 평가

본 장에서는 실험에 의한 성능 분석을 통하여 제안하는 기법의 우수성을 규명한다. 먼저, 제 5.1절에서는 성능 평가를 위한 실험 환경을 설명하고, 제 5.2절에서는 기존 기법과의 비교 실험을 통한 제안된 기법의 성능 개선 효과를 제시한다.

4.1 실험 환경

본 연구에서는 성능 평가를 위하여 참고 문헌 [1]과 동일한 방법을 사용하여 100,000개의 시퀀스들로 구성된 합성 데이터를 생성하였다. 합성 데이터 내의 각 시퀀스 $S = \langle s_1, s_2, ..., s_n \rangle$ 는 다음과 같은 랜덤 워크(random walk) 형태를 가진다.

$$s_i = s_{i-1} + z_i$$

여기서 z_i는 구간 [-500, 500] 사이에서 균일한 분포를 취하는 랜덤 변수이며, 시퀀스의 첫 요소 값 s_1은 구간 [5,000, 10,000] 사이의 임의의 값을 취하도록 하였다. 또한, 질의 시퀀스 $Q = \langle q_1, q_2, ..., q_n \rangle$ 는 데이터베이스로부터 하나의 시퀀스를 임의로 선택하여 각 요소 값에 적절한 범위 내의 임의의 값을 더하는 방식으로 변형하여 생성하였다. 선택된 시퀀스 내에 속하는 요소 값들의 표준 편차를 std라 할 때, 이 범위는 [-std/10, std/10]이다. ε의 값으로는 $\sqrt{n \times 1,000}$ (n=8)을 기본 값 ε_1로 간주하여 2배인 ε_2, 4배인 ε_3, 6배인 ε_4, 8배인 ε_5를 사용하였다. 초기에 주어지는 질의 시퀀스들의 수는 1,000개로 선택하였으며, 이들을 성능 평가 실험에서 그대로 사용하였다.

인덱싱을 위한 자료 구조로서 512-Byte 페이지 크기를 갖는 R*-트리를 사용하였다. 또한, 각각의 데이터 시퀀스를 별도의 페이지 내에 저장시킴으로써 데이터 클러스터링(data clustering) 효과에 의한 영향을 제거하였다.

시퀀스 매칭의 처리에서 사용된 다차원 인덱스의 구성 속성으로서 기존 기법에 대한 실험을 위해서는 DFT 변환 후 나타나는 앞쪽의 2개와 3개의 DFT 계수를 사용하였으며, 제안된 기법에 대한 실험으로는 다차원 인덱스의 구성 속성으로서 그림 3.1에 나타난 알고리즘에 의하여 선정된 요소 계수들을 사용하였다. 각 경우에 대해서 1,000개의 서로 다른 서브시퀀스 매칭에 대해서 실험한 후 평균을 취한 값을 실험 결과로 하였다. 실험 결과를 평가하기 위한 성능 지수로서 인덱스 액세스 단계 및 시퀀스 액세스 단계에서 액세스되는 디스크 페이지의 수를 측정하였다.

성능 평가를 위한 하드웨어 플랫폼은 700MHz Pentium III와 512MB의 주기억장치가 장착된 PC이며, 소프트웨어 플랫폼은 MS Windows 2000 및 Visual C++ 6.0이다. 실험 중 다른 프로세스들과의

상호 간섭을 방지하기 위하여 모든 사용자 프로세스들을 제거한 상황에서 실험하였다.

4.2 실험 결과

그림 4.1은 실험 결과를 나타낸 것이다. 가로축은 사용된 시퀀스 매칭 질의에서 사용된 유사 허용치 ε_1, ε_2, ε_3, ε_4, ε_5를 의미하며, 세로축은 1,000개의 시퀀스 매칭의 처리에서 발생된 디스크 액세스 수의 평균값을 나타낸다. 기존 기법 중 하나는 DFT 변환 후 나타나는 앞쪽의 2개의 DFT 계수들만을 이용하여 구성된 다차원 인덱스를 이용한 시퀀스 매칭의 처리 결과이며, 기존 기법의 또 다른 하나는 앞쪽의 3개의 DFT 계수들만을 이용하여 구성된 다차원 인덱스를 이용한 시퀀스 매칭의 처리 결과이다. 마지막 하나는 그림 3.1에 나타난 알고리즘에 의하여 선정된 요소 계수들을 이용하여 구성된 다차원 인덱스를 이용하여 수행한 시퀀스 매칭의 처리 결과를 보여준다.

그림 4.1. 제안된 기법과 기존 기법의 성능 비교.

모든 기법에서 유사 허용치가 증가함에 따라 디스크 액세스 횟수가 커지는 것으로 나타났다. 이것은 유사 허용치가 증가하면, 인덱스 페이지 액세스 횟수와 인덱스 액세스 결과로 나타나는 후보 시퀀스 액세스 횟수가 모두 증가하기 때문이다. 기존의 기법들 중에서는 두 개의 DFT 계수만을 이용하여 구성된 다차원 인덱스를 이용하는 시퀀스 매칭의 성능이 세 개를 이용한 경우보다 모든 경우에서 나은 성능을 보였다. 이것은 본 실험에서 높은 차원의 인덱스를 구성함으로써 얻어지는 필터링 효과에 비하여 인덱스 액세스 수가 증가하는 영향이 더 크게 나타났기 때문이다.

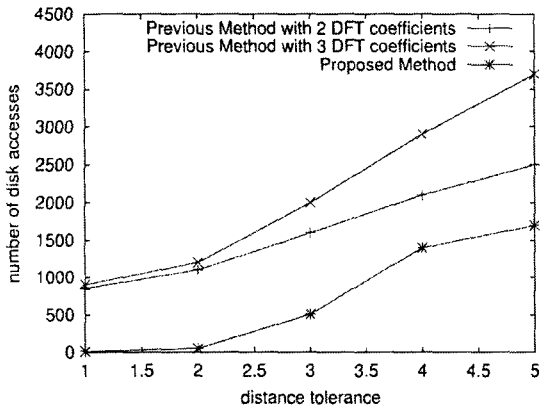

그림 4.1 제안된 기법과 기존 기법의 성능 비교

반면, 제안된 기법은 모든 경우에서 기존의 기법들에 비하여 큰 성능 개선 효과를 보였다. 이것은 물리적 데이터베이스 설계를 기반으로 하는 제안된 기법을 통하여 변별력이 큰 요소 계수들이 다차원 인덱스 구성에 참여하였고, 그 수도 적절하였음을 보여주는 것이다. 또한, 성능 개선 효과는 약 3배에서 100배에 이르는 것으로 나타났으며, 유사 허용치가 작아질수록 성능 개선 효과는 커지는 것으로 나타났다. 이와 같이, 유사 허용치가 작아질수록 성능 개선 효과가 커지는 것은 매우 바람직한 현상이다. 그 이유는 실제 응용에서 사용자는 많지 않은 수의 최종 결과를 검색하는 시퀀스 매칭을 수행하기 때문이다. 따라서 제안된 기법은 실제 응용 환경에서 매우 유용하게 사용될 수 있다.

5. 결 론

시퀀스 매칭은 시계열 데이터베이스로부터 질의 시퀀스와 변화의 추세가 유사한 데이터 시퀀스들을 검색하는 연산이다. 효과적인 시퀀스 매칭을 위해서는 다차원 인덱스의 사용이 필수적이다. 시계열 데이터베이스의 특성에서 비롯된 고차원 문제로 인하여 기존의 대부분의 연구에서는 데이터 시퀀스를 DFT 등의 변환 기법을 이용하여 저차원 공간상의 점으로 변환하여 사용한다. 그러나 다차원 인덱스를 위한 구성 속성으로서 어떤 DFT 계수를 몇 개를 이용해야 하는가에 관해서는 논의된 바 없다.

본 논문에서는 이러한 문제에 연구의 초점을 맞추어 시퀀스 매칭을 효과적으로 지원하기 위한 최적의 다차원 인덱스를 구성하는 방안에 관하여 논의하였다. 첫째, DFT 계수가 가지는 실수부와 허수부를 독립적인 요소 계수로 간주함으로써 변별력이 떨어지는 요소 계수가 구성 속성으로서 참여하는 것을 방지하였다. 둘째, 구성 속성을 선정하는 기준으로서 에너지의 표준 편차를 이용함으로써 대상이 되는 데이터베이스 특성의 사전 분석을 통하여 변별력이 뛰어난 요소 계수를 올바르게 선택할 수 있도록 하였다. 셋째, 비용 공식을 이용한 시퀀스 매칭 비용의 추정을 통하여 다차원 인덱스에 참여하는 구성 속성의 수를 올바르게 결정할 수 있도록 하였다. 제안된 기

법의 우수성을 규명하기 위하여 실험을 통한 성능 평가를 수행하였다. 실험 결과에 의하면, 유사 허용치에 따라 제안된 기법은 기존 기법에 비하여 매우 좋은 성능 개선 효과를 보이는 것으로 나타났다.

감사의 글

이 논문은 2003년도 한국학술진흥재단의 선도과학자 연구비 지원(KRF-2003-041-D00486), 첨단정보기술연구센터(AITrc)를 통한 한국과학재단(KOSEF)의 지원, 그리고 강원대학교 IT 연구 센터를 통한 연구비 지원을 받았습니다.

참 고 문 헌

[1] R. Agrawal, C. Faloutsos, A. Swami, "Efficient Similarity Search in Sequence Databases", Proc. of the 4th Int'l Conference on Foundations of Data Organization and Algorithms, Chicago, Oct. 1993.

[2] R. Agrawal, K. Lin, H. S. Sawhney, K. Shim, "Fast Similarity Search in the Presence of Noise, Scaling, and Translation in Time-Series Databases", Proc. of the 21st Int'l Conference on Very Large Databases, Zurich, Switzerland, Sept. 1995.

[3] N. Beckmann, H.-P. Kriegel, R. Schneidel, and B. Seeger, "The R*-tree: an Efficient and Robust Access Method for Points and Rectangles", In Proc. Intl. Conf. on Management of Data, ACM SIGMOD, pp. 322-331, May 1990.

[4] A. Belussi, and C. Faloutsos, "Estimating the Selectivity of Spatial Queries Using the Correlation Fractal Dimension", In Proc. Intl. Conf. on Very Large Data Bases, VLDB, pp. 299-310, Sept. 1995.

[5] D. J. Berndt and J. Clifford, "Finding Patterns in Time Series: A Dynamic Programming Approach", Advances in Knowledge Discovery and Data Mining, pp. 229-248, 1996.

[6] S. Berchtold, D. A. Keim, and H.-P. Kriegel, "The X-tree: An Index Structure for High-Dimensional Data", In Proc Intl. Conf. on Very Large Data Bases, VLDB, pp. 28-39, 1996.

[7] K.-P. Chan and W.-C. Fu, "Efficient Time-Series Matching by Wavelets", In Proc. Intl. Conf. on Data Engineering, IEEE, pp. 126-133, 1999

[8] K. K. W. Chu, and M. H. Wong, "Fast Time-Series Searching with Scaling and Shifting", In Proc. Intl. Symp. on Principles of Database Systems, pp. 237-248, May 1999..

[9] C. Faloutsos, M. Ranganathan, Y. Manolopoulos, "Fast Subsequence Matching in Time-series Databases", In Proc. Intl. Conf. on Management of Data, ACM SIGMOD, pp. 419-429, May, 1994.

[10] C. Faloutsos, and I. Kamel, "Beyond Uniformity and Independence: Analysis of R-tree Using the Concept of Fractal Dimension", In Proc. Intl. Conf. on Management of Data, ACM SIGMOD, pp. 4-13, May 1994.

[11] D. Q. Goldin and P. C. Kanellakis, "On Similarity Queries for Time-Series Data: Constraint Specification and Implementation", In Proc. Intl. Conf. on Principles and Practice of Constraint Programming, Sept. 1995.

[12] Lee, J. H. et al., "A Region Splitting Strategy for Physical Database Design of Multidimensional File Organizations", In Proc. of the 23th Int'l Conf. on Very Large Data Bases, pp. 416-425, Aug. 1997.

[13] Loh, W. K., Kim, S. W., and Whang, K. Y., "Index Interpolation: An Approach for Subsequence Matching Supporting Normalization Transform in Time-Series Databases", In Proc. ACM Int'l. Conf. on Information and Knowledge Management, pp. 480-487, 2000.

[14] Loh, W. K., Kim, S. W., and Whang, K. Y., "Index Interpolation: A Subsequence Matching Algorithm Supporting Moving Average Transform of Arbitrary Order in Time-Series Databases", IEICE Trans. on Information and Systems, Vol. E84-D, No. 1, pp. 76-86, Jan. 2000.

[15] A. V. Oppenheim and R. W. Schafer, Digital Signal Processing, Prentice-Hall, 1975.

[16] S. H. Park, W. W. Chu, J. H. Yoon, and C.

Hsu, "Efficient Searches for Similar Subsequences of Difference Lengths in Sequence Databases", In Proc. IEEE Int'l. Conf. on Data Engineering, pp. 23-32, 2000.

[17] D. Rafiei and A. Mendelzon, "Similarity-Based Queries for Time-Series Data", In Proc. Intl. Conf. on Management of Data, ACM SIGMOD, pp. 13-24, 1997.

[18] D. Rafiei, "On Similarity-Based Queries for Time Series Data", In Proc. IEEE Intl. Conf. on Data Engineering, pp. 410-417, 1999.

[19] M. Schroeder, Fractals, Chaos, Power Laws: Minutes from an Infinite Paradise, W. H. Freeman and Company, 1991.

[20] R. Weber, H.-J. Schek, and S. Blott, "A Quantitative Analysis and Performance Study for Similarity-Search Methods in High-Dimensional Spaces", In Proc. Intl. Conf. on Very Large Data Bases, VLDB, pp. 194-205, 1998.

[21] B. K. Yi, H. V. Jagadish, and C. Faloutsos, "Efficient Retrieval of Similar Time Sequences Under Time Warping", In Proc. IEEE Intl. Conf. on Data Engineering, pp. 201-208, 1998.

김 상 욱

1989년 2월 서울대학교 컴퓨터공학과 졸업(학사)
1991년 2월 한국과학기술원 전산학과 졸업(석사)
1994년 2월 한국과학기술원 전산학과 졸업(박사)
1991년 7월~8월 미국 Stanford University, Computer Science Department 방문 연구원
1994년 2월~1995년 2월 KAIST 정보전자연구소 전문연구원
1999년 8월~2000년 8월 미국 IBM T.J. Watson Research Center Post-Doc.
1995년 3월~2000년 8월 강원대학교 컴퓨터정보통신공학부 부교수
2003년 3월~현재 한양대학교 정보통신대학 정보통신학부 부교수
관심분야 : 데이타베이스 시스템, 저장 시스템, 데이타 마이닝, 멀티미디어 정보 검색, 공간 데이터베이스/GIS, 주기억장치 데이타베이스, 트랜잭션 관리

김 진 호

1982년 2월 경북대학교 전자공학과 졸업(학사)
1985년 2월 한국과학기술원 전산학과 졸업(석사)
1990년 2월 한국과학기술원 전산학과 졸업(박사)
1995 8월~1996년 7월 미국 미시간 대학교 방문 교수
2003년 3월~2004년 2월 미국 Drexel 대학교 방문 교수
1999년 9월~현재 KAIST 첨단정보기술연구센터 연구원
1990년 8월~현재 강원대학교 컴퓨터과학과 교수
관심분야 : 데이타 웨어하우징 및 OLAP, 데이타 마이닝, 데이타베이스 저장 시스템, 주기억장치 데이타베이스 시스템, XML 및 웹 데이타베이스

한 병 일

1997년 2월 삼척산업대학교 전자계산학과 졸업(학사)
1999년 2월 강원대학교 컴퓨터과학과 졸업(석사)
1999년 3월~2001년 8월 삼척산업대학교 시간강사
1999년 3월~2000년 5월 (주) 파이널테크 연구원
2000년 6월~2002년 2월 (주) 극동케이디아이 책임연구원
2002년 8월~현재 (주) 참좋은 인터넷 개발팀장
관심분야 : 데이타 마이닝, 웹 솔루션 개발, 웹 검색, 모바일 응용, WIPI