

다차원 중포 속성 색인구조의 최적 설계기법

이종학[†]

요약

본 논문에서는 객체 데이터베이스 시스템에서 중포 속성에 대한 색인구조로 다차원 색인구조를 이용하는 다차원 중포 속성 색인구조(Multidimensional Nested Attribute Index: MD-NAI)의 최적 설계기법을 제시한다. MD-NAI는 B'-tree와 같은 일차원 색인구조를 이용한 중포 속성 색인구조에서 지원할 수 없는 클래스 계층과 중포 속성이 포함된 복합 형태의 질의들에 대한 처리를 잘 지원할 수 있다. 그러나, MD-NAI는 사용자 질의 형태에 따라 색인검색의 성능이 매우 나빠질 수 있다. 본 논문에서는 질의 형태에 따른 MD-NAI의 성능 개선을 위하여, 먼저 중포 술어에 대한 질의 정보로서 색인 페이지 영역의 최적 모양을 결정하고, 이 최적 모양을 갖는 색인 페이지 영역의 모양이 되도록 하는 영역분할 전략을 적용하여 최적의 MD-NAI를 구성한다. 또한, 성능평가를 위하여 MD-NAI를 이용하여 다양한 중포 술어의 형태와 객체 분포에 대하여 실시한 실험 결과를 제시한다. 성능평가의 결과에 의하면, 주어진 질의 패턴에 따라 최적의 MD-NAI를 구성할 수 있었으며, 삼차원 MD-NAI의 경우에 질의 영역의 구간비가 1:16:256일 때 기존의 순환분할 전략에 의한 MD-NAI에 비해 성능이 5.5배 이상까지 향상되었다.

An Optimal Design Method for the Multidimensional Nested Attribute Indexes

Jong-Hak Lee[†]

ABSTRACT

This paper presents an optimal design methodology for the multidimensional nested attribute index (MD-NAI) that uses a multidimensional index structure for indexing the nested attributes in object databases. The MD-NAI efficiently supports complex queries involving both nested attributes and class hierarchies, which are not supported by the nested attribute index using one-dimensional index structure such as B'-tree. However, the performance of the MD-NAI is very degraded in some cases of user's query types. In this paper, for the performance enhancement of the MD-NAI, we first determine the optimal shape of index page region by using the query information about the nested predicates, and then construct an optimal MD-NAI by applying a region splitting strategy that makes the shape of the page regions of the MD-NAI as close as possible to the predetermined optimal one. For performance evaluation, we perform extensive experiments with the MD-NAI using various types of nested predicates and object distribution. The results indicate that our proposed method builds optimal MD-NAI regardless of the query types and object distributions. When the interval ratio of a three-dimensional query region is 1:16:256, the performance of the proposed method is enhanced by as much as 5.5 times over that of the conventional method employing the cyclic splitting strategy.

Key words: 객체 데이터베이스, 다차원 색인구조, 중포 속성, 색인 구성

본 연구는 2001학년도 대구가톨릭대학교 연구비 지원에 의한 것임.

접수일 : 2002년 11월 13일, 완료일 : 2002년 12월 3일

[†] 정회원, 대구가톨릭대학교 컴퓨터정보통신공학부 교수

1. 서 론

객체 데이터베이스 관리 시스템(Object Database Management System: ODBMS)은 객체지향 개념에 의해 순수 관계형 DBMS보다 확장된 객체지향 데이터 모델을 지원함으로써, 공학(Engineering) 데이터베이스나 멀티미디어 데이터베이스 등 새로운 응용분야를 지원하는 시스템으로 각광을 받고 있다. 그러나, 기존의 ODBMS들은 대부분 객체지향 개념은 잘 지원하지만, 데이터베이스 관리 기능의 제공은 순수 관계형 DBMS에 비하여 미흡한 단계에 있다. 최근들어 객체 데이터베이스 시스템의 질의처리 성능 최적화는 중요한 연구과제이다. 최근에 제안된 객체 데이터베이스 색인기법들은 객체지향 질의처리의 성능 향상에 크게 기여하고 있다[1]. 그러나, 이들 색인구조들은 기존의 관계형 데이터베이스의 단순 속성에 대한 색인구조에 비해 저장공간 및 개선유지 비용에 큰 부담이 있다. 또한, 색인구조의 종류에 따라 검색 성능이 다른 특성도 있다[1]. 그러므로, 객체 데이터베이스 색인기법을 통한 질의처리의 잇점이 저장공간 및 개선유지를 위한 부담으로 인해 상쇄되지 않게 하기 위해서는 색인들이 매우 신중히 구성되어야 하며, 효과적인 색인구성 방법에 관한 연구가 필수적이라 할 수 있다.

객체 데이터베이스는 클래스 집단화(aggregation) 개념에 의하여 한 클래스가 가지는 속성의 도메인이 또 다른 클래스가 되게 함으로써(이러한 속성을 복합 속성이라 함) 클래스 집단화 계층을 이룬다. 따라서, 클래스 집단화 계층상의 모든 클래스에서 정의된 어떠한 속성도 논리적으로는 루트 클래스의 속성이라고 볼 수 있다. 우리는 이런 속성을 루트 클래스의 중포 속성(nested attribute)[2,3]이라 한다. 이로 인하여 객체지향 질의어에서는 기존의 관계형 데이터베이스에서의 질의어와는 달리 중포 속성에 조건이 주어지는 중포 술어(nested predicate)[2,3]를 가진다는 특징이 있다. 중포 속성을 표현하기 위해서는 경로식(path expression)[4]이 사용되며, 경로식은 루트 클래스로부터 클래스 집단화 계층을 따라 나타나는 속성들의 나열로 표현된다.

객체 데이터베이스는 또 하나의 중요한 개념인 클래스 상속(inheritance) 개념에 의하여 클래스들 사이에 클래스 상속 계층(일반적으로 클래스 계층으로 약칭함)이라는 하나의 계층구조를 이룬다. 즉, 하나의 클

래스는 여러 개의 서브클래스를 가지며, 각 서브클래스는 또 다른 여러 서브클래스들을 가진다. 따라서, 객체 데이터베이스에서는 질의의 대상 범위를 두 가지 경우로 나타낼 수 있다. 한 경우는 질의의 대상 범위를 질의에 나타나는 클래스만으로 한정하는 것이고, 또 다른 경우는 질의의 대상 범위를 질의에 나타나는 클래스와 그의 모든 서브클래스들을 포함하는 것이다[3]. 이러한 클래스 계층에 대한 질의를 효율적으로 처리할 수 있는 색인구조는 특정 클래스에 속하는 객체들의 탐색뿐만 아니라 특정 클래스를 루트로 하는 클래스 계층의 모든 클래스들에 속하는 객체들의 탐색도 효율적으로 처리할 수 있어야 한다.

중포 속성을 표현하는 경로식에는 속성값(객체 참조자: Oid)들의 참조관계에 의한 객체와 객체 사이에 암시적 조인(implicit join)[3]의 의미를 가지고 있다. 이러한 암시적 조인은 데이터베이스 스키마에 의해 미리 예상이 가능하다. 따라서, 질의에 자주 나타나는 중포 속성에 대한 암시적 조인을 미리 계산하여 그 결과를 색인으로 구축하여 놓음으로써, 질의처리시 이를 이용하여 성능 향상을 꾀할 수 있으며 이를 중포 속성에 대한 색인기법[2,4-6]이라 한다. 그러나, 이러한 중포 속성에 대한 기존의 색인기법들은 B⁺-tree와 같은 일차원 색인구조를 이용함으로써, 클래스 상속에 의한 특징으로 질의의 대상 범위가 클래스 계층상의 임의의 클래스들로 제한되거나, 경로식에 나타나는 속성의 도메인이 클래스 계층상의 임의의 클래스들로 제한이 되는 질의들을 지원하기 어려운 문제점을 가지고 있다.

이와 같은 일차원 색인구조를 이용하는 기존의 중포 속성에 대한 색인기법들이 가지는 문제점을 해결하기 위하여, 다차원 색인구조를 중포 속성에 대한 색인구조로 이용할 수 있으며 이를 다차원 중포 속성 색인구조라 한다. 다차원 중포 속성 색인구조에서는 중포 속성의 칸값들로 구성된 키 속성 도메인과 함께, 경로식에 나타나는 각 속성마다 그 속성의 도메인 클래스 계층을 이루는 클래스들의 클래스 식별자 도메인을 할당하여 다차원 색인구조를 구성한다. 이와 같은 색인기법에서는 기존의 B⁺-tree와 같은 일차원 색인구조를 이용한 색인기법들에서 문제가 되는 질의의 대상 범위가 클래스 계층상의 임의의 클래스들로 제한되거나, 경로식에 나타나는 속성의 도메인이 클래스 계층상의 임의의 클래스들로 제한이 되는 질의들의 처리를 한번의 색인 탐색으로 가능하게 할 수 있다.

그러나, 중포 속성에 대한 색인구조로 다차원 색인구조를 단순히 이용하는 것은 키 속성 도메인의 크기와 클래스 식별자 도메인의 크기가 매우 다르고, 주어지는 질의의 형태가 다름으로 인하여 색인 검색의 성능이 매우 나빠질 수 있다. 따라서, 본 논문에서는 다차원 중포 속성 색인구조의 성능을 개선하기 위하여, 주어진 질의 패턴에 따라 색인 엔트리들의 최적의 클러스터링을 가능하게 하는 다차원 중포 속성 색인구조의 최적 설계기법을 제안한다. 제안한 설계기법에서는 질의 정보를 이용하여 키 속성 도메인과 여러 개의 클래스 식별자 도메인으로 구성된 다차원 도메인 공간상에 주어진 색인 엔트리들의 클러스터링 문제를 다룬다. 이는 경로식으로 표현된 중포 술어를 가지는 객체지향 질의는 키 속성 도메인과 여러 개의 클래스 식별자 도메인으로 구성된 다차원 도메인 공간상에 주어지는 다차원 질의 영역으로 매핑할 수 있기 때문이다.

다차원 중포 속성 색인구조의 최적 설계기법에서는 먼저, 사전에 분석한 사용자 질의 형태에 대한 정보를 이용하여 키 속성 도메인과 여러 개의 클래스 식별자 도메인 사이의 색인 엔트리들에 대한 클러스터링 정도를 구한다. 그리고, 이러한 클러스터링 정도를 유지하도록 하는 다차원 도메인 공간의 영역분할 전략을 적용하여 다차원 색인구조를 구성한다. 이러한 색인구조의 핵심 아이디어는 다차원 도메인 공간 상에서 색인 엔트리들의 클러스터링 정도를 주어진 질의 패턴에 적합하도록 조정함으로써 주어진 질의들에 의해서 액세스되는 색인 페이지의 평균 개수를 최소화하는 것이다. 사전에 분석한 질의 정보를 화일구조의 구성에 이용하는 기법은 물리적 데이터베이스 설계기법으로 지금까지 널리 사용되고 있다[7-10].

본 논문의 구성은 다음과 같다. 먼저, 제 2절에서는 관련 연구로서 객체 데이터베이스의 색인 구축에 필요한 기본 개념과 함께 기존의 색인 기법들을 살펴본다. 그리고, 제 3절에서는 객체 데이터베이스의 중포 속성에 대한 색인 기법으로 다차원 색인구조를 이용하는 다차원 중포 색인구조를 소개하고, 제 4절에서는 이 다차원 중포 색인구조의 최적 설계기법을 제안한다. 제 5절에서는 제안한 설계기법의 성능평가의 결과를 제시한다. 마지막으로, 제 6절에서는 결론을 내린다.

2. 관련 연구

객체 데이터베이스에서 사용자가 정의하는 클래스

들은 루트를 가진 이차원 방향성 그래프(rooted two-dimensional directed graph)를 형성하며, 이 그래프를 그 클래스에 대한 스키마 그래프(Schema Graph: SG)라 정의한다[3]. 그림 1은 클래스 Person에 대한 SG이다. 그림 1에서 클래스 Person은 서브 클래스 Instructor와 Student, 그리고 Instructor와 Student의 서브 클래스들을 포함하는 클래스 상속 계층구조와 Vehicle과 Company를 포함하는 클래스 집단화 계층구조의 루트이다.

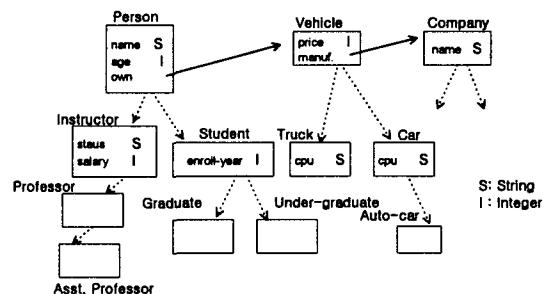


그림 1. 클래스 Person에 대한 스키마 그래프.

객체 데이터베이스에서 한 클래스가 가지는 속성들은 정수 타입이나 문자 타입과 같은 기본 타입의 값을 가지는 단순 속성과 다른 클래스에 속하는 객체의 객체 식별자(Object identifier: Oid)를 값으로 가지는 복합 속성으로 구분된다[3]. 클래스 C의 중포 속성[2]이란 클래스 C로부터 실선 링크로 연결된 클래스에서 정의된 속성들을 의미하며, 클래스 C와 속성들의 연속으로 표시한다. 그리고, 클래스 집합 C*는 클래스 C와 그의 모든 서브 클래스들을 원소로 하는 집합으로 정의한다.

2.1 객체지향 질의어의 특징

객체지향 질의도 SQL에서처럼 Select, From, Where 절로 구성할 수 있으며 각 절에서 객체지향 개념을 지원하도록 확장한다[11]. From 절에는 질의의 대상이 되는 클래스를 기술하며, Where 절에는 단순 속성에 대한 조건인 단순 술어와 더불어 중포 속성에 대한 조건인 중포 술어[2]를 사용할 수 있다.

객체지향 질의에서 중포 속성을 표현하는 방법으로 경로식[4, 11]을 사용한다. 경로식은 클래스 집단화 계층구조상에서 클래스 이름과 속성의 교차적인 나열(sequence)로서 다음과 같은 형태를 가진다. 단, A; 뒤

의 중괄호({ })는 선택적임을 나타내는 표시이다.

$$P = C_1.A_1\{[C_2]\}.A_2\{[C_3]\}...A_n\{[C_{n-1}]\} \quad (1)$$

경로식 P 에서 클래스 C_1 을 타겟 클래스, 속성 A_i 의 도메인이 되는 C_{i+1} 을 A_i 의 도메인 클래스라 정의한다. 타겟 클래스와 도메인 클래스는 클래스 대치(substitutability) 개념에 의해 경로에서 클래스 계층구조에 속하는 특정 클래스로 한정될 수 있다[3,11]. 수식(1)의 경로 P 에서 속성 A_i 의 도메인은 뒤에 대괄호([])가 생략되면 자동으로 클래스 집합 C_{i+1}^* 로 되지만, C_{i+1}^* 에 속하는 특정 클래스가 대괄호 안에 지정되면 대괄호 안의 클래스로 한정된다.

경로식에서 속성의 도메인 클래스가 특정 클래스로 한정되는 것을 도메인 대치(domain substitution)라고, 타겟 클래스가 특정 클래스로 한정되는 것을 타겟 대치(target substitution)라 한다. 이러한 클래스 대치는 질의의 범위를 특정한 클래스로 한정할 수 있도록 하여 클래스 상속의 개념을 객체지향 질의에 표현하도록 한 것이다[11].

2.2 중포 술어 질의처리를 위한 기존 색인기법

객체 데이터베이스의 중포 술어를 가지는 질의를 수행하기 위한 기법으로 순방향 운행법(forward traversal), 역방향 운행법(backward traversal)이 있다[12]. 순방향 운행법은 스키마 그래프의 집단화 계층 구조에서 상위 클래스를 먼저 탐색하고 하위 클래스를 나중에 탐색하는 방법이고, 역방향 운행법은 하위 클래스를 먼저 탐색하고 상위 클래스를 나중에 탐색하는 방법이다. 역방향 운행시에는 하위 클래스의 객체를 참조하는 상위 클래스의 객체를 탐색하기 위해서는 많은 비용을 필요로 한다.

객체 데이터베이스의 중포 속성에 대한 색인기법은 중포 속성을 표현하는 경로식에 의한 암시적 조인을 미리 계산하여 색인구조로 유지함으로써, 질의처리시 클래스 집단화 계층에 대한 운행을 생략할 수 있도록 한 것이다[1,3]. 지금까지 중포 속성에 대한 색인기법을 위해 제안된 색인구조들로, Bertino와 Kim[2]이 제안한 중포 색인(nested index), 경로 색인(path index), Kemper와 Moerkotte[4]가 제안한 ASR(Access Support Relation), 그리고 Xie와 Han[6]이 제안한 JIH(Join Index Hierarchy) 등이 있다. 임의의 경로 $p = C_1.A_1...A_n$ 에 대하여, 중포 색인은 A_n 을 키로하여 C_1 에

속하는 객체들의 Oid를 찾는 색인구조이고, 경로 색인은 A_n 을 키로하여 $C_1, …, C_n$ 에 속하는 객체들의 Oid 리스트(즉, 경로 인스턴스)를 찾는 색인구조이다. ASR은 경로식에 대한 경로 인스턴스들에서 Oid만 추출하여 릴레이션 형태로 구성한 색인구조로서 경로 색인과 유사한 색인구조이다. 그리고, JIH는 하나의 경로에서 두 개의 클래스 사이에 직접 또는 간접참조 관계가 있는 객체들의 Oid쌍들로 구성된 색인구조로서, 직접참조 관계가 있는 Oid들의 쌍들을 기본 조인 색인구조라 하였고, 간접참조 관계가 있는 Oid들의 쌍들은 기본 조인 색인구조로부터 유도하여 구축함으로써 유도된 조인 색인구조라 하였다.

그러나, 이러한 색인구조들은 클래스 상속에 의한 객체지향 데이터 모델의 특징을 반영하지 못하는 것들로서 타겟 대치 또는 도메인 대치가 있는 경로식으로 표현된 질의는 지원하지 못한다. 즉, 질의의 대상 범위가 클래스 계층상의 임의의 클래스들로 제한되거나, 경로식에 나타나는 어떠한 속성의 도메인이 클래스 계층상의 임의의 클래스들로 제한이 되는 질의들을 지원하지 못한다.

Bertino[5]는 질의의 대상 범위가 임의의 클래스들로 제한되는 타겟 클래스 대치가 있는 경로식으로 표현된 중포 속성에 대한 질의를 지원할 수 있는 NIX(Nested-Inherited Index)라고 하는 색인구조를 제안하였다. NIX는 클래스 계층에 대한 색인구조의 하나인 CH-index[13]와 중포 색인을 통합한 형태의 색인구조이다. 즉, NIX는 B-tree의 단말 노드를 색인된 중포 속성의 값을 키로하여 스키마 그래프상에서 직접 또는 간접적으로 이 중포 속성을 내포하는 모든 클래스들의 Oid들을 대상으로 클래스별 구분을 위한 클래스 딕렉토리와 함께 구성한 색인구조이다. 그러나, NIX 역시 도메인 클래스의 대치가 있는 경로식으로 표현된 질의를 지원하지 못한다. 이는 색인된 중포 속성으로부터 각 타겟 클래스까지의 경로 정보를 유지하지 않기 때문이다.

중포 속성에 대한 색인구조로 다차원 색인구조를 이용함으로써 경로상에 타겟 클래스 대치와 도메인 클래스 대치 모두가 있는 질의를 지원할 수 있다. 그러나, 중포 속성에 대한 색인구조로 다차원 색인구조를 단순히 이용하는 것은 키 속성 도메인의 크기와 클래스 식별자 도메인의 크기가 매우 다르고, 주어지는 질의의 형태가 다름으로 인하여 색인 검색의 성능이 나빠질 수 있다. 따라서, 다차원 중포 속성 색인구조의 성능을

개선하기 위하여, 주어진 질의 패턴에 따른 다차원 중포 속성 색인구조의 최적 설계기법이 필요하다. 다음 제 3절에서는 다차원 중포 속성 색인구조의 최적 설계 기법을 기술하기에 앞서 다차원 색인구조를 중포 속성에 대한 색인구조로 활용하는 다차원 중포 속성 색인구조를 간단히 소개한다.

3. 다차원 중포 속성 색인구조

객체 데이터베이스 질의어의 중포 술어에는 클래스 대치가 있을 수 있으며, 이러한 중포 술어의 처리를 지원하기 위한 색인구조로 다차원 색인구조를 이용할 수 있다. 즉, 색인할 중포 속성의 키 속성 도메인과 함께 중포 속성을 표현하는 경로상의 각 클래스 계층마다 클래스 식별자들로 구성된 한 차원씩의 클래스 식별자 도메인을 할당함으로써, (경로길이 +1) 차원의 도메인 공간을 구성하여 이를 다차원 색인구조에 적용한다. 예를 들어, 그림 1과 같은 스키마 그래프에서 경로의 길이가 2인 Person 클래스의 중포 속성 price(Vehicle 클래스의 단순 속성)에 대한 색인구조로서 X축은 중포 속성 price의 키 속성 도메인으로 하고, Y축은 Person 클래스 계층의 클래스 식별자 도메인으로 하고, 그리고 Z축은 Vehicle 클래스 계층의 클래스 식별자 도메인으로 하는 삼차원 도메인 공간을 구성할 수 있다.

다차원 중포 속성 색인구조에서 클래스 식별자 도메인은 클래스 대치가 되는 클래스 계층의 클래스 식별자들이 하나의 연속된 구간이 되도록 구성하여야 한다. 그렇지 않으면, 다차원 도메인 공간상에서 클래스 대치가 있는 중포 술어에 대한 질의 영역이 여러 개의 영역으로 분리되기 때문에 색인 탐색의 성능이 저하되거나 때문이다. 트리 구조의 클래스 계층에서 임의의 클래스 C와 그의 서브클래스들로 구성되는 클래스 집합을 C^* 로 표현할 때, 클래스 식별자들을 클래스 계층의 전위 순회(preorder traversal) 순서로 나열되게 클래스 식별자 도메인을 구성함으로써, C^* 에 포함되는 클래스 식별자들이 클래스 식별자 도메인상에서 하나의 연속된 구간으로 표현되게 할 수 있다. 본 논문에서는 클래스 계층을 트리 구조로 가정한다. 클래스 계층이 다중 상속(multiple inheritance)에 의한 DAG(directed acyclic graph) 구조인 경우에는 질의의 대상이 C^* 인 질의처리를 하나의 영역 탐색으로 가능하게 하기 위해서는 다중 상속된 클래스에 대해 클래스 식별자 도메인

의 식별자 값을 중복되게 배열하여야 한다¹⁾.

그림 2는 그림 1의 스키마 그래프에서 Person 클래스 계층에 대한 클래스 식별자 도메인의 구성 예를 나타낸다. 즉, 각 클래스 식별자들이 Person 클래스 계층의 전위 순회 순서로 매핑되는 도메인상의 값과 각 클래스 C를 루트로 하는 클래스 집합 C^* 에 대응하는 클래스 식별자 도메인상의 연속된 값들의 구간을 보여준다.

| 클래스-id (C) | 도메인 값 | C^* 의 구간 |
|-------------|-------|------------|
| Person | 0 | 0 ~ 6 |
| Instructor | 1 | 1 ~ 3 |
| Professor | 2 | 2 ~ 3 |
| Asst. Prof. | 3 | 3 ~ 3 |
| Student | 4 | 4 ~ 6 |
| Graduate | 5 | 5 ~ 5 |
| Under-grad. | 6 | 6 ~ 6 |

그림 2. 그림 1의 Person 클래스 계층의 클래스 식별자 도메인의 구성 예.

본 논문에서는 객체 데이터베이스의 중포 속성에 대한 색인구조를 다차원 색인구조의 하나인 계층 그리드 화일(mltiplelevel grid file: MLGF)[15]을 이용하여 구성하고, 이를 다차원 중포 속성 색인구조(Multi-dimensional Nested Attribute Index: MD-NAI)라 한다. MD-NAI는 디렉토리와 색인 페이지로 구성된다²⁾. 디렉토리는 다단계의 균형된 트리 구조를 가지며, 디렉토리를 구성하는 디렉토리 페이지의 구조는 MLGF[15]에서와 마찬가지이다. 색인 페이지는 색인 레코드들로 구성되며, 각 색인 레코드에는 경로상의 각 클래스 식별자 값(class-id value) 필드, 키 값(key value) 필드, 객체 또는 경로의 개수 필드, 및 이들에 대한 색인 엔트리들의 리스트 필드가 있다. 그리고, 레코드의 크기가 페이지의 크기보다 크게될 때 오버플로우 페이지를 할당하고 이를 포인트하기 위한 오버플로우 페이지를 할당하고 이를 포인트하기 위한 오버플로우

- 1) DAG 구조인 클래스 계층에 대하여 C^* 인 질의처리를 하나의 영역 탐색이 되도록 다중 상속된 클래스의 식별자 값을 중복시키지 않고, 탐색 영역의 개수를 최소화하는 클래스 계층 선형화(class hierarchy linearization) 알고리즘에 대해서는 참고문헌[14]을 참조하기 바람.
- 2) MD-NAI의 디렉토리 페이지와 색인 페이지는 B*-tree의 비단말(non-leaf) 페이지와 단말(leaf) 페이지에 해당한다.

우 페이지(overflow page) 필드가 있다.

MD-NAI는 색인 페이지의 색인 레코드에 있는 색인 엔트리의 구성방법에 따라 다차원 중포 색인구조과 다차원 경로 색인구조의 두 가지 색인구조로 분류할 수 있다. **다차원 중포 색인구조(Multidimensional Nested Index: MNI)**는 색인 엔트리를 색인된 중포 속성의 타겟 클래스 계층에 속하는 객체에 대한 객체 식별자(즉, Oid)들로 구성하며, **다차원 경로 색인구조(Multidimensional Path Index: MPI)**는 색인 엔트리를 색인된 중포 속성에 대한 경로 인스턴스(즉, Oid 리스트)들로 구성한다. MPI와 같이 색인 엔트리를 경로 인스턴스들로 구성하는 것은 색인 엔트리를 타겟 클래스 계층의 객체 식별자만으로 구성하는 경우에 발생하게 되는 데이터베이스의 변경에 따른 색인구조의 막대한 유지비용을 줄이기 위함이다[1,2]. 그림 3은 MNI의 색인 페이지 구조를 나타낸다.

중포 속성에 대한 색인기법은 중포 술어를 만족하는 객체들의 탐색에는 매우 유용하지만, 상대적으로 색인구조의 유지비용을 많이 필요로 한다[1,2,16]. 따라서, 색인을 유지하는 경로의 길이에 따라 MNI와 MPI의 선택이 달라져야 한다. 경로의 길이가 1인 경우는 MNI와 MPI가 같은 색인구조가 되며, 이는 객체 데이터베이스의 클래스 상속 계층에 대한 색인구조로 제안한 이차원 클래스 색인구조[17]와 같은 색인구조가 된다. 경로의 길이가 2인 경우는 MNI를 선택하는 것이 좋다. 이것은 MNI와 MPI 모두 색인구조의 유지 비용은 비슷한 반면에 MPI가 MNI에 비하여 많은 저장 공간의 오버헤드로 인하여 검색 성능이 떨어지기 때문이다. 경로의 길이가 3인 경우에는 MPI를 선택하는 것이 좋다. 이것은 MPI가 MNI에 비하여 검색 성능은 떨어지지만 유지비용을 적게 필요로 하기 때문이다. 그리고, 경로의 길이가 4이상일 경우에는 경로를 길이가 1, 2, 또는 3이 되는 서브경로들로 분할한 다음에, 각 서브경로에 따라 MNI 또는 MPI를 할당하여야 한다[1,2]. 따라서, 본 논문에서는 경로의 길이가 2 또는 3인 경우의 삼차원 또는 사차원 MD-NAI에 대해서

최적 설계기법을 제시하고 성능평가를 실시한다.

4. 다차원 중포 속성 색인구조의 최적 설계기법

다차원 도메인 공간상에서 한 영역의 모양은 영역을 구성하는 각 축에 대한 구간 크기의 비를 나타내는 구간비로 표현할 수 있다. 본 논문에서는 객체지향 질의에서 사용되는 중포 술어들이 다차원 도메인 공간상에 매핑되는 질의 영역들의 형태에 대한 정보를 기반으로 질의 영역들에 의해 교차하는 색인 페이지 영역들의 개수가 최소로 되는 페이지 영역의 최적 구간비를 결정하고, 가능한 이와 같은 구간비를 갖는 페이지 영역들이 되도록 하는 영역분할 전략을 사용함으로써 최적의 MD-NAI를 구성하는 설계기법을 제시한다.

4.1 MD-NAI의 최적조건

다차원 색인구조에서는 다차원 도메인 공간에 주어진 색인 페이지 영역의 구간비에 따라 질의 영역에 의해서 교차되는 페이지 영역의 평균 개수가 달라지는 특징이 있다. 참고문헌[8]에서는 이러한 특징을 이용하여 데이터의 균일 분포와 비균일 분포 각각에 대하여 주어진 질의 영역들에 대해 페이지 영역의 평균 액세스 횟수를 최소로 하는 페이지 영역의 최적 구간비를 계산하는 방법을 제안하였다. 본 절에서는 이를 소개하고, MD-NAI 색인구조에 대한 페이지 영역의 최적 구간비를 이와 같은 방법으로 계산한다.

아래 정리 1은 X, Y축의 이차원 도메인 공간상에서 데이터가 균일하게 분포할 때 페이지 영역의 최적 구간비를 계산하는 방법에 대한 정리이다. 데이터가 균일하게 분포하면 도메인을 구성하는 페이지 영역들의 크기가 일정하게 되며, 주어진 질의 영역들에 의해 교차되는 페이지 영역들의 개수를 최소로 하는 페이지 영역의 최적 구간비는 모든 질의 영역들에 대해 각 축별로 구간 크기를 더한 값의 비로서 계산할 수 있음을 나타낸다.

색인 레코드

| 레코드 길이 | 클래스 식별자1 값 | ... | 클래스 식별자n 값 | 키-값 | 오버플로우 페이지 | 객체 수 | {Oid1,Oid2,...,Oidn} | ... |
|--------|------------|-----|------------|-----|-----------|------|----------------------|-----|
|--------|------------|-----|------------|-----|-----------|------|----------------------|-----|

그림 3. MNI의 색인 페이지 구조.

정리 1 [균일분포] 크기가 $p(x) \times p(y)$ 로 일정한 페이지 영역들로 나누어져 있는 이차원 도메인 공간상에서, 임의의 위치에 주어지는 n개의 질의 영역 $q_i(x) \times q_i(y)$ ($i = 1, \dots, n$)에 대해 각 질의 영역과 만나게 되는 페이지 영역의 총 개수를 최소로하는 페이지 영역의 최적 구간비 $p(x) : p(y) = \sum_{i=1}^n q_i(x) : \sum_{i=1}^n q_i(y)$ 이다.

증명: 참고문헌[8] 참조.

이차원 도메인 공간상에서 데이터가 비균일하게 분포한다는 것은 도메인 공간의 위치에 따라 색인 엔트리의 밀집도가 다름으로 인하여 페이지 영역의 크기가 위치에 따라 달라짐을 의미한다. 즉, 밀집도가 높은 곳에서는 밀집도가 낮은 곳에 비하여 많은 페이지가 할당되므로 각 페이지 영역의 크기는 작아지게 된다. 따라서, 비균일 분포의 경우에는 질의 영역에 의해 교차되는 페이지 영역의 개수는 질의 영역의 크기뿐만 아니라 질의 영역이 주어진 위치의 데이터 밀집도에도 비례하게 되므로, 균일 분포에서와 같이 페이지 영역의 최적 구간비를 모든 질의 영역의 각 축별로 구간크기를 단순히 더한 값의 비로서 구할 수 없다. 이와 같은 경우에는 각 질의 영역의 크기에 대해 위치에 따른 데이터 밀집도를 가중치(weight)로 곱한 질의 영역의 형태를 정규화된 질의 영역(normalized query region)이라 하고, 이러한 질의 영역의 정규화를 통하여 페이지 영역의 최적 구간비를 계산할 수 있다. 아래 정리 2는 이를 정리한 것이다.

정리 2 [비균일분포] 서로 다른 크기의 페이지 영역들로 나누어져 있는 이차원 도메인 공간상에서, 임의의 위치에 주어지는 n개의 질의 영역 $q_i(x) \times q_i(y)$ ($i = 1, \dots, n$)에 대해 각 질의 영역의 객체 밀집도를 d_i 라 할 때, 각 질의 영역과 만나게 되는 페이지 영역의 총 개수를 최소로하는 페이지 영역의 최적 구간비 $p(x) : p(y) = \sum_{i=1}^n q_i(x)\sqrt{d_i} : \sum_{i=1}^n q_i(y)\sqrt{d_i}$ 이다.

증명: 참고문헌[8] 참조.

따라서, 본 논문에서는 정리 2의 결과를 다차원으로 확장하여 최적의 MD-NAI 색인구조를 구성한다. 즉, 경로의 길이가 2인 경우에 적용할 삼차원 MD-NAI

색인구조인 경우에는 X, Y, Z축으로 구성된 n개의 삼차원 질의 영역 $q_i(x) \times q_i(y) \times q_i(z)$ ($i = 1, \dots, n$)에 대해 색인 페이지 영역의 최적 구간비($p(x) : p(y) : p(z)$)를 $\sum_{i=1}^n q_i(x) d_i^{1/3} : \sum_{i=1}^n q_i(y) d_i^{1/3} : \sum_{i=1}^n q_i(z) d_i^{1/3}$ 로 계산한다. 그리고, 경로의 길이가 3인 경우에 적용할 사차원 MD-NAI 색인구조인 경우에는 W, X, Y, Z축으로 구성된 n개의 사차원 질의 영역 $q_i(w) \times q_i(x) \times q_i(y) \times q_i(z)$ ($i = 1, \dots, n$)에 대해 색인 페이지 영역의 최적 구간비($p(w) : p(x) : p(y) : p(z)$)를 $\sum_{i=1}^n q_i(w) d_i^{1/4} : \sum_{i=1}^n q_i(x) d_i^{1/4} : \sum_{i=1}^n q_i(y) d_i^{1/4} : \sum_{i=1}^n q_i(z) d_i^{1/4}$ 로 계산한다. 다음 제 4.2절에서는 이렇게 계산된 색인 페이지 영역의 최적 구간비를 갖는 MD-NAI를 구성하기 위한 색인구조의 영역분할 전략을 제시한다.

4.2 MD-NAI의 조작과 영역분할 전략

MD-NAI의 삽입, 삭제, 및 검색과 관련된 조작 연산의 구체적인 알고리즘은 참고문헌[15]에 기술된 MLGF의 조작 알고리즘과 거의 동일하나, 단지 삽입 연산의 영역분할 전략에서 차이가 있다. 따라서, 본 절에서는 페이지 영역의 구간비가 제 4.1절에서 기술한 방법에 의하여 계산되는 페이지 영역의 최적 구간비에 근접하도록 하는 영역분할 전략을 제시한다.

MD-NAI에서는 색인 엔트리가 삽입되고 삭제되는 상황에 따라 분할과 병합을 반복함으로써 동적 변화에 적응한다[15]. 새로운 색인 엔트리가 삽입되는 경우, 단계의 디렉토리를 루트로부터 최하위 디렉토리까지 탐색하여 그 색인 엔트리가 속하는 페이지 영역을 찾게되고, 그 영역에 할당된 색인 페이지에 색인 엔트리를 삽입하게 된다. 삽입 결과 색인 페이지의 용량이 초과되면(overflow), 해당 영역은 같은 크기를 갖는 두 개의 영역으로 분할(half splitting)되고 각 영역에 해당하는 새로운 두 개의 색인 페이지가 할당되며, 색인 레코드들은 두 색인 페이지에 분산된다. 지금까지, MD-NAI에서는 하나의 페이지 영역을 두 개의 영역으로 분할할 때 각 축을 번갈아 가며 분할시키는 순환 분할 전략을 사용하고 있다. 그러나, MD-NAI는 페이지 영역을 분할할 때 분할 축을 임의로 선택할 수 있으며, 분할 축을 선택하는 방법에 따라 페이지 영역의 모양을 결정할 수 있다. 따라서, 본 논문에서는 분할

축으로서 분할된 후의 페이지 영역의 구간비가 최적 구간비에 가깝게 되는 축을 선택함으로써, 객체의 지속적인 삽입으로 인한 연속된 분할시에 도메인 공간내의 모든 페이지 영역의 구간비를 최적 구간비에 근접하도록 유도할 수 있다.

아래 정리 3은 특정 모양의 질의 영역이 이차원 도메인 공간상의 임의의 위치에 주어질 때, 특정 크기의 한 페이지 영역과 만나게 되는 위치 영역의 크기는, 그 페이지 영역의 모양이 주어진 질의 영역의 모양과 같을 때 최소가 됨을 나타낸다.

정리 3 구간비가 $a:b$ 인 $a \times b$ 형태의 질의 영역 Q 가 이차원 도메인 공간상의 임의의 위치에 주어질 때, 크기가 R 인 $p(x) \times p(y)$ 형태의 한 페이지 영역 P 와 만나게 되는 위치 영역의 크기가 최소로 되는 경우는 P 의 구간비($p(x) : p(y)$)가 주어진 Q 의 구간비($a : b$)와 같을 때이다.

증명: 아래 그림 4는 $a \times b$ 형태의 질의 영역 Q 가 이차원 도메인 공간상의 임의의 위치에 주어질 때, 크기가 $R (=p(x) \times p(y))$ 인 특정 페이지 영역 P 와 만나게 되는 위치 영역을 질의 영역 Q 의 중앙점의 위치 영역(빗금 친 부분) LR 로 나타낸 것이다.

그림 4에서 LR 의 크기 $S LR(p(x), p(y))$ 는 다음 식과 같다.

$$S LR(p(x), p(y)) = (p(x) + a)(p(y) + b) \quad (2)$$

$p(x) \times p(y) = R$ 이므로, 수식(2)의 $p(y)$ 를 $\frac{R}{p(x)}$ 로 치환하면,

$$\begin{aligned} S LR(p(x), \frac{R}{p(x)}) &= (p(x) + a)\left(\frac{R}{p(x)} + b\right) \\ &= R + \frac{aR}{p(x)} + p(x)b + ab \end{aligned} \quad (3)$$

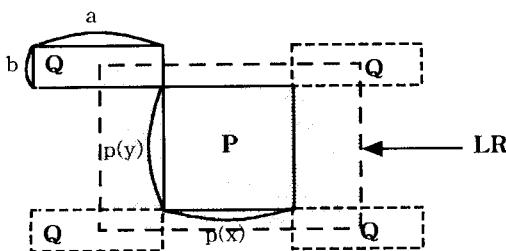


그림 4. 임의의 한 페이지 영역과 만나게 되는 질의 영역의 위치 영역.

이다. 따라서, 수식(3)의 값을 최소로 하는 $p(x)$ 를 구하면, $p(x) = \sqrt{(a/b)R}$ 이다. 또한, 이러한 $p(x)$ 에 대한 $p(y)$ 는 $p(x) \times p(y) = R$ 에 의하여 $p(y) = \sqrt{(b/a)R}$ 이다. 그러므로, $S LR(p(x), p(y))$ 를 최소로 하는 페이지 영역 P 의 구간비 $p(x) : p(y) = a : b$ 이다. □

정리 3을 이용하여 이차원 MD-NAI의 경우 페이지 영역을 분할할 때 분할된 페이지 영역의 구간비가 최적 구간비에 가깝게 되는 분할 축을 선택할 수 있다. 즉, X, Y, Z축으로 이루어진 삼차원 MD-MAI의 경우 페이지 영역의 분할 시 분할된 페이지 영역의 구간비가 미리 주어진 질의 정보들에 의해 계산된 최적 구간비에 가장 가깝게 되는 분할 축을 선택하는 방법은 다음과 같다. 먼저, 계산된 최적 구간비 $O_x : O_y : O_z$ 를 갖는 $O_x \times O_y \times O_z$ 형태의 질의 영역이 삼차원 도메인 공간상에 임의의 위치에 주어진다고 가정하고, 분할이 요구되는 $p(x) \times p(y) \times p(z)$ 형태의 페이지 영역이 각 축에 대해 분할된 후의 한 페이지 영역과 만나게 되는 질의 영역의 위치 영역 LR 의 크기를 계산한다. 그리고, 이 LR 크기의 값이 가장 작게 되는 축을 분할 축으로 선택한다. 예를 들어, 분할 축으로 X축을 선택했을 때 LR 의 크기는 $(p(x)/2 + O_x)(p(y) + O_y)(p(z) + O_z)$ 이다. 따라서, 삼차원 MD-MAI의 영역분할 전략은 다음과 같다.

삼차원 MD-MAI를 위한 영역분할 전략:

$(p(x)/2 + O_x)(p(y) + O_y)(p(z) + O_z)$ 의 값이 최소이면, X축 분할,

$(p(x) + O_x)(p(y)/2 + O_y)(p(z) + O_z)$ 의 값이 최소이면, Y축 분할,

$(p(x) + O_x)(p(y) + O_y)(p(z)/2 + O_z)$ 의 값이 최소이면, Z축 분할.

4.3 MD-NAI의 설계 알고리즘

다음 그림 5는 제 4.1절에서의 다차원 중포 속성 색인구조의 최적 조건과 제 4.2절에서의 영역분할 전략을 이용하여 경로의 길이가 2인 중포 속성에 대한 색인구조인 삼차원 MD-NAI 경우에 대한 설계 알고리즘이다. 경로의 길이가 3인 중포 속성에 대한 색인구조인 사차원 MD-NAI인 경우에도 같은 방법으로 확장하여 적용할 수 있다.

그림 5의 알고리즘에서 나타난 삼차원 MD-NAI의 전체 설계 과정은 다음과 같은 세 가지 단계로 구성된다. 첫째, 두 개의 클래스 식별자 도메인과 한 개의 키 속성 도메인으로 구성된 삼차원 도메인 공간상에 주어진 각 질의 영역에 대해서 정규화 과정을 거친다. 즉, 삼차원 도메인 공간에 주어지는 질의 영역 $q(x) \times q(y) \times q(z)$ 에 대한 정규화는 다음과 같다. 먼저, 질의 처리 결과로서 알 수 있는 질의 영역내의 객체의 개수 N 을 이용하여 각 질의 영역의 객체 밀집도 d 를 $\frac{N}{q(x) \times q(y) \times q(z)}$ 로 구하고, 질의 영역을 이루는 각 축의 구간에 가중치 $d^{1/3}$ 을 곱하여 정규화된 질의 영역 $q(x)d^{1/3} \times q(y)d^{1/3} \times q(z)d^{1/3}$ 를 얻는다.

둘째, 정규화된 모든 질의 영역에 대해서 각 축별 구간의 크기를 합산한 값의 비율로서 페이지 영역의

최적 구간비 $O_x : O_y : O_z$ 를 얻는다.

셋째, 최적 구간비에 최대한 가까운 모양의 페이지 영역으로 구성된 MD-NAI를 구축한다. 여기서는 제 4.2절의 영역분할 정책을 적용한다. 즉, 계속되는 공간 객체의 삽입으로 MD-NAI의 색인 페이지에 오버플로 우가 발생하면, 이 색인 페이지에 대응하는 페이지 영역은 구간반분 정책을 사용하여 같은 크기의 두 영역으로 분할되고, 원 색인 페이지의 객체들은 분할된 페이지 영역에 대응하는 두 개의 색인 페이지로 나뉘어 저장된다. 이때 페이지 영역의 구간반분 정책으로 제 4.2절의 영역분할 정책을 적용하는 것이다. 즉, 둘째 단계에서 결정된 최적 구간비 $(O_x : O_y : O_z)$ 와 같은 모양을 가지는 가상의 질의 영역 $(O_x \times O_y \times O_z)$ 이 임의의 위치에 주어진다고 가정하고, 분할이 요구되는 페

• 설계 정보

두 개의 클래스 식별자 도메인과 한 개의 키 속성 도메인으로 구성된 삼차원 도메인 공간(X, Y, Z축으로 구성)상에 주어진 n 개의 삼차원 질의 영역에 대하여,

- (1) 각 질의 영역의 형태: $q_i(x) \times q_i(y) \times q_i(z)$ ($i = 1, \dots, n$)
- (2) 각 질의 영역에 포함되는 객체수: N_i ($i = 1, \dots, n$)

• 설계 알고리즘

단계 1: 각 질의 영역의 정규화 ($i = 1, \dots, n$)

- (1) 각 질의 영역의 객체 밀집도 d_i 를 계산한다.

$$d_i = \frac{N_i}{q_i(x) \times q_i(y) \times q_i(z)}$$

- (2) 각 질의 영역의 정규화된 질의 영역의 형태를 구한다.

$$q'_i(x) = q_i(x) \times d_i^{1/3}$$

$$q'_i(y) = q_i(y) \times d_i^{1/3}$$

$$q'_i(z) = q_i(z) \times d_i^{1/3}$$

단계 2: 색인 페이지 영역의 최적 구간비 ($O_x : O_y : O_z$)의 결정

$$O_x : O_y : O_z = \sum_{i=1}^n q'_i(x) : \sum_{i=1}^n q'_i(y) : \sum_{i=1}^n q'_i(z)$$

단계 3: 최적의 MD-NAI 구축

- (1) 객체 삽입으로 색인 페이지에 오버플로우가 발생하면, 색인 페이지 분할

⇒ 대응하는 페이지 영역 ($p(x) \times p(y) \times p(z)$)의 분할 전략:

다음 식들의 계산에 따른 결과로서 분할 축 결정

- $(p(x)/2 + O_x)(p(y) + O_y)(p(z) + O_z)$ 의 값이 최소이면 X축 분할
- $(p(x) + O_x)(p(y)/2 + O_y)(p(z) + O_z)$ 의 값이 최소이면 Y축 분할
- $(p(x) + O_x)(p(y) + O_y)(p(z)/2 + O_z)$ 의 값이 최소이면 Z축 분할

- (2) 연속된 객체의 삽입에 따라 (1)번 항목을 반복 적용

이지 영역 ($p(x) \times p(y) \times p(z)$)이 각 축에 대해 구간반분에 의한 분할 후의 한 페이지 영역과 만나게 되는 질의 영역의 위치 영역의 크기(예를 들어, X축의 구간을 반분했을 때 그 크기는 $(p(x)/2+O_x)(p(y)+O_y)(p(z)+O_z)$ 이다.)를 각각 계산한 다음 그 값이 가장 작게 되는 축을 분할 축으로 선택한다.

5. 성능 평가

본 절에서는 제안한 다차원 중포 속성 색인구조의 설계 알고리즘에 대한 유용성을 다양한 실험을 통하여 검증한다. 실험의 목적은 MD-NAI를 구성하는 색인 페이지 영역의 모양에 대한 질의 영역별 색인 검색의 성능을 알아보고, 주어진 질의 정보로서 최적의 MD-NAI를 구성할 수 있음을 실제 실험을 통하여 검증하는 것이다. 본 성능 평가에서 사용된 비용은 질의 처리를 위하여 액세스해야 할 색인 페이지의 개수로 한다. 제 5.1절에서는 성능평가를 위하여 사용된 실험 모델에 대하여 기술하고, 제 5.2절에서는 실험 결과를 제시하고 이를 분석한다.

5.1 실험 모델

본 실험에서는 315,000개의 객체를 포함하는 두 종류의 MD-NAI를 구축한다. 하나는 키 속성 도메인인 X축과 타겟 클래스 식별자 도메인인 Y축으로 구성된 이차원 MD-NAI이고, 다른 하나는 이차원 MD-NAI에 도메인 클래스 식별자 도메인인 Z축을 하나 더 할당하여 구성한 삼차원 MD-NAI이다. 그리고, 색인구조의構성을 위하여 사용한 각 구성요소들의 값으로 색인구조들의 일반적인 구현에서 널리 사용되고 있는 다음과 같은 값을 사용한다[13,17-19]. 객체 식별자 O_id의 크기는 12바이트, 클래스 식별자의 크기는 4바이트, 색인된 키의 크기는 8바이트, 포인터는 4바이트, 각종 길이와 개수 필드의 크기는 2바이트, 2D-CHI의 리전 베타의 크기는 12바이트, 그리고 색인 페이지의 크기는 4K바이트로 한다.

그리고, 실험을 위한 데이터베이스를 다음과 같이 구성한다. 먼저, 타겟 클래스 도메인과 도메인 클래스 도메인의 구성을 위하여 사용한 클래스 계층구조로서 63개의 클래스(C_1, C_2, \dots, C_{63})로 구성된 균형된 이진 트리(balanced binary tree) 형태를 사용한다. 이런 경우 클래스 식별자 도메인의 구간 크기는 클래스 집합

C_i^* 에 속하는 원소의 개수로 1, 3, 7, 15, 31, 63인 6가지가 가능하다. 그리고 각 클래스에 대해 5000개의 키 값을 [0, 2500]의 구간내에서 표준 편차 σ 가 $2500 \times 2/5$ 인 $N(\mu, \sigma^2)$ 의 정규 분포를 취하게 하여 평균값 μ 를 임의로 조정함으로써, 클래스에 따라 색인 엔트리들이 다차원 도메인 공간상에서 집중되는 위치가 다르게 한다.

질의 패턴의 구성을 위하여 사용한 질의 영역들의 형태는 이차원 질의 영역인 경우에는 크기가 도메인 공간의 1/2000로서 일정하고, 질의 영역의 구간비가 64:1, 16:1, 4:1, 1:1, 1:4, 및 1:16인 Q_64:1, Q_16:1, Q_4:1, Q_1:1, Q_1:4, 및 Q_1:16 형태의 질의 영역 등이다. 그리고, 삼차원 질의 영역인 경우에는 크기가 도메인 공간의 1/20000로 일정하고, 질의 영역의 구간비가 1:1:1, 1:2:4, 1:4:16, 1:8:64, 및 1:16:256인 Q_1:1:1, Q_1:2:4, Q_1:4:16, Q_1:8:64, 및 Q_1:16:256 형태의 질의 영역 등이다.

5.2 실험 결과

첫 번째 실험에서는 이차원 MD-NAI 색인구조에 대해서, 다양한 타겟 페이지 영역의 구간비를 갖는 여러 개의 MD-NAI를 생성하고, 각각에 대하여 고유의 질의 영역 형태를 갖는 각 질의를 처리할 때 발생하는 평균 페이지 액세스 수를 관찰하였다. 이차원 MD-NAI의 생성을 위하여 사용한 타겟 페이지 영역의 구간비는 64:1, 32:1, 16:1, 8:1, 4:1, 2:1, 1:1, 1:2, 1:4, 1:8, 1:16, 1:32의 12가지를 선정하고, 각각에 대하여 제 4.2 절의 영역분할 전략을 사용하여 이차원 MD-NAI를 구축하였다. 그리고, 이차원 질의 영역의 형태(Q_64:1, Q_16:1, Q_4:1, Q_1:1, Q_1:4, 및 Q_1:16)별로, 질의 패턴을 구성하기 위하여 질의 영역 생성 프로그램을 개발하였으며, 이를 이용하여 1,000개의 질의 영역을 도메인 공간상에 균일하게 생성하고 이를 질의를 처리하는데 발생하는 평균 액세스 수를 측정하였다.

그림 6은 실험 결과를 그래프 형태로 나타낸 것이다. 가로축은 구성된 이차원 MD-NAI의 타겟 페이지 영역의 구간비를 나타내고, 세로축은 질의처리 시 발생되는 색인 페이지의 액세스 수를 나타낸다. 그림 6에서 제시된 바와 같이, 모든 형태의 질의 영역에 대하여 그 질의 영역의 구간비를 타겟 페이지 영역의 구간비로 가지는 MD-NAI에서 가장 좋은 성능을 보였다. 질

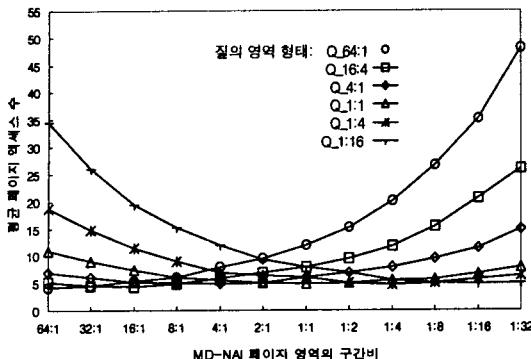


그림 6. 각각 다른 구간비의 페이지 영역을 가지는 이차원 MD-NAI들에 대한 질의 형태별 성능비교.

의 영역의 구간비가 64:1로 주어진 경우 타겟 페이지 영역의 구간비가 64:1(최적 구간비)인 MD-NAI에서 (평균 4개의 색인 페이지를 액세스) 타겟 페이지 영역의 구간비가 1:1로 구성된 MD-NAI에 (평균 12개의 색인 페이지를 액세스) 비해서는 세 배까지의 성능 향상을 보였으며, 타겟 페이지 영역의 구간비가 1:32로 구성된 MD-NAI에 (평균 47개의 색인 페이지를 액세스) 비해서는 12배까지 성능이 좋았다. 이것은 다차원 중포 속성 색인구조의 경우 반드시 주어진 질의 형태에 따라 색인구조를 달리 구성하여야 함을 보여주는 것이다.

두 번째 실험에서는 삼차원 MD-NAI 색인구조에 대해서, 첫 번째 실험에서와 같은 실험을 실시하였다. 즉, 다양한 타겟 페이지 영역의 구간비를 갖는 여러 개의 삼차원 MD-NAI를 생성하고, 각각에 대하여 고유의 질의 영역 형태를 갖는 각 질의를 처리할 때 발생하는 평균 페이지 액세스 수를 관찰하였다. 삼차원 MD-NAI의 생성을 위하여 사용한 타겟 페이지 영역의 구간비는 1:1:1, 1:2:4, 1:4:16, 1:8:64, 및 1:16:256의 다섯 가지를 선정하고, 각각에 대하여 제 4.2절의 영역 분할 전략을 사용하여 삼차원 MD-NAI를 구축하였다. 그리고, 삼차원 질의 영역의 형태(Q_1:1:1, Q_1:2:4, Q_1:4:16, Q_1:8:64, 및 Q_1:16:256)별로, 질의 패턴을 구성하기 위하여 질의 영역 생성 프로그램을 이용하여 1,000개의 질의 영역을 도메인 공간상에 균일하게 생성하고 이를 질의를 처리하는데 발생하는 평균 액세스 수를 측정하였다.

그림 7은 실험 결과를 그래프 형태로 나타낸 것이다. 그림 7에서도 알 수 있는 바와 같이, 삼차원 MD-NAI에 대해서도 모든 형태의 질의 영역에 대하여 그

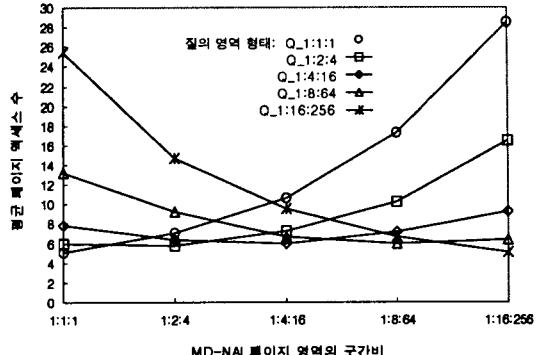


그림 7. 각각 다른 구간비의 페이지 영역을 가지는 삼차원 MD-NAI들에 대한 질의 형태별 성능비교.

질의 영역의 구간비를 타겟 페이지 영역의 구간비로 가지는 MD-NAI에서 가장 좋은 성능을 보인다. 질의 영역의 구간비가 1:16:256으로 주어진 경우 타겟 페이지 영역의 구간비가 1:16:256(최적 구간비)인 MD-NAI에서 (평균 4개의 색인 페이지를 액세스) 타겟 페이지 영역의 구간비가 1:1:1로 구성된 MD-NAI에 (평균 22개의 색인 페이지를 액세스) 비해서 5.5배까지의 성능 향상을 보인다. 이것은 삼차원 MD-NAI의 경우에도 반드시 주어진 질의 형태에 따라 색인구조를 달리 구성하여야 함을 보여주는 것이다.

세 번째 실험에서는 삼차원 MD-NAI 색인구조에 대해서, 다양한 타겟 페이지 영역의 구간비를 갖는 여러 개의 MD-NAI를 생성하고, 각각에 대하여 여러 가지 질의 영역의 형태들이 혼합되어 주어지는 하나의 혼합 질의 패턴을 처리하기 위한 평균 색인 페이지 액세스 수를 관찰하였다. 삼차원 MD-NAI의 생성을 위하여 사용한 타겟 페이지 영역의 구간비는 두 번째 실험에서와 마찬가지로 1:1:1, 1:2:4, 1:4:16, 1:8:64, 및 1:16:256의 다섯 가지를 선정하고, 각각에 대하여 제 4.2절의 영역 분할 전략을 사용하여 삼차원 MD-NAI를 구축하였다. 그리고, 혼합 질의 패턴을 구성하기 위하여, 삼차원 질의 영역의 형태(Q_1:1:1, Q_1:2:4, Q_1:4:16, Q_1:8:64, 및 Q_1:16:256)별로 200개씩 도메인 공간상에 균일하게 분포하도록 생성하여 이들을 혼합하여 사용하였다.

그림 8은 실험 결과를 그래프 형태로 나타낸 것이다. 정규화된 모든 질의 영역들에 대하여 각 축의 구간 크기를 더한 값의 비는 1:6:68으로 계산되었으며, 그림 8에서 알 수 있는 바와 같이 이 비율과 같은 구간비의 페이지 영역을 갖는 MD-NAI에서 가장 좋은 성능

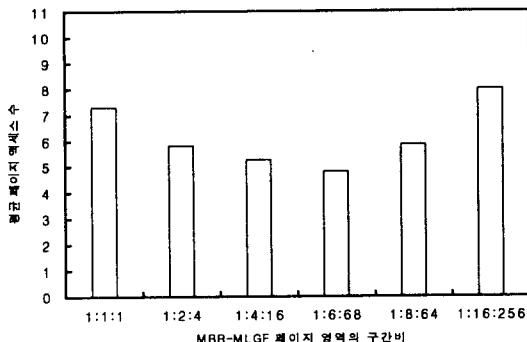


그림 8. 각각 다른 구간비의 페이지 영역을 가지는 삼차원 MD-NAI들에 대한 혼합 질의 패턴의 성능비교.

을 보인다. 이와 같은 실험 결과는 다양한 형태의 질의 영역들에 의해 교차되는 페이지 영역의 개수를 최소로 하는 페이지 영역의 최적 구간비는 정규화 과정을 통하여 주어진 질의 영역들에 대해 각 축별로 구간 크기를 더한 값의 비로서 계산할 수 있음을 보이기 위한 것이다. 또한, 이 실험의 결과는 제 4.3절에서 제시한 다차원 중포 속성 색인구조의 설계 알고리즘의 실용성을 입증하는 것이다.

끝으로, 네 번째 실험에서는 본 논문에서 제시한 다차원 중포 속성 색인구조의 설계기법을 이용하여 구성한 MD-NAI가 기존의 순환분할 전략(즉, 구간비 = 1:1:1)으로 구성한 MD-NAI과 비교하여 얼마만큼의 성능 개선 효과가 있는지를 알아본다. 먼저, 다섯 가지의 삼차원 질의 영역의 형태인 Q_1:1:1, Q_1:2:4, Q_1:4:16, Q_1:8:64, 및 Q_1:16:256에 대하여, 각 형태 별로 1000개의 질의 영역들이 도메인 공간상에 균일하게 주어지는 다섯 가지의 질의 패턴을 생성한다. 그리고, 각 질의 패턴에 대하여 최적의 구간비(질의 패턴을 구성하는 질의 영역들의 구간비와 동일)를 갖는 영역 분할 전략으로 페이지 영역들을 구성한 MD-NAI를 생성하여 그 질의 패턴을 처리할 때 발생하는 평균 색인 페이지 액세스 수를 구하고, 이 값에 대한 기존의 순환분할 전략으로 구성한 MD-NAI에서 같은 질의 패턴을 처리할 때 발생하는 평균 페이지 액세스 수의 비율을 측정한다. 그림 9는 이에 대한 실험 결과를 나타낸 것이다. 가로축은 각 질의 패턴을 구성하는 질의 영역들의 구간비를 나타내며, 세로축은 제안된 기법을 사용하는 경우의 성능 이득이 몇 배인가를 나타낸다.

그림 9에서 나타난 바와 같이 질의 영역의 구간비가 1:1:1에서 멀어질수록 제안된 기법을 사용하는 경우의

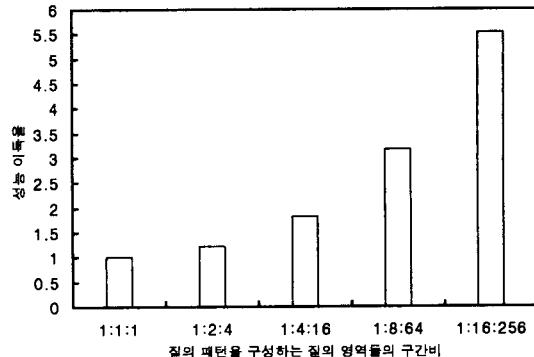


그림 9. 질의 패턴을 구성하는 질의 영역의 구간비별 다차원 중포 속성 색인구조의 설계기법에 의해서 생성된 삼 차원 MD-NAI의 성능 효율.

성능 개선 효과가 뚜렷해짐을 볼 수 있다. 즉, 질의 영역의 구간비가 1:16:256인 경우 질의처리 성능이 5.52 배까지 향상됨을 볼 수 있으며, 구간비가 더 커질수록 더욱더 향상될 수 있음을 나타낸다. 이러한 결과는 제 4.3절에서 제시한 다차원 중포 속성 색인구조 설계기법의 성능 개선 효과를 잘 나타내는 것이다.

6. 결 론

본 논문에서는 다차원 색인구조를 객체 데이터베이스의 중포 속성에 대한 색인구조로 이용하는 다차원 중포 속성 색인구조에 대하여, 질의처리 성능을 최적으로 보장할 수 있는 최적 설계기법을 제안하였다. 다차원 중포 속성 색인구조는 객체 데이터베이스의 중포 술어에 타겟 클래스 계층뿐만 아니라 도메인 클래스 계층에도 클래스 대치가 있는 경우에도 질의 처리를 잘 지원할 수 있는 색인구조이다.

다차원 중포 속성 색인구조의 최적 설계기법에서는 먼저, 객체지향 질의에서 사용되는 중포 술어들이 다차원 도메인 공간상에 매핑되는 질의 영역들의 형태에 대한 정보를 기반으로, 질의 영역들에 의해 교차하는 색인 페이지 영역들의 개수가 최소로 되는 색인 페이지 영역의 최적 구간비를 결정한다. 이는 다차원 중포 속성 색인구조의 색인 키 속성 도메인과 함께 타겟 클래스 계층으로부터 색인 키 속성이 정의된 클래스 계층까지의 여러 개의 클래스 식별자 도메인 사이의 색인 엔트리들에 대한 클러스터링 정도를 나타낸다. 그리고, 다차원 색인구조에서 이러한 최적 구간비를 갖는 페이지 영역들이 되도록 하는 영역분할 전략을 적

용함으로써 최적의 다차원 중포 속성 색인구조를 구성한다.

또한, 본 논문에서는 제안한 다차원 중포 속성 색인구조 설계기법의 성능평가를 위하여, 본 논문에서 다차원 중포 속성 색인구조로 제시한 MD-NAI를 대상으로 색인 페이지 영역의 모양이 최적 구간비에 근접하도록 하는 영역분할 전략을 제시하고, 이를 이용하여 다양한 실험을 수행하였다. 실험 결과에 의하면, 주어진 질의 패턴과 데이터 분포에 따라 최적의 MD-NAI를 구성할 수 있음을 확인하였다. 경로의 길이가 2인 경우에 주어지는 중포 술어에 대한 삼차원 질의 영역의 경우, 모양이 편향된 정도에 따라 정방형(구간비가 1:1:1인 경우) 모양의 페이지 영역으로 구성된 삼차원 MD-NAI에 비해 질의처리의 성능이 매우 크게 향상됨을 알 수 있었다. 특히, 주어진 중포 술어에 대한 질의 영역의 구간비가 1:16:256인 경우에는 질의처리를 위한 색인 성능이 5.52배까지 향상되었다. 이것은 제안된 기법이 실제적으로 매우 유용함을 보여주는 것이다.

참 고 문 헌

- [1] Bertino, E. and Ooi, B. C., "The Indispensability of Dispensable Indexes," *IEEE Trans. on Knowledge and Data Eng.*, Vol. 11, No. 1, pp. 17-27, Jan. 1999.
- [2] Bertino, E. and Kim, W., "Indexing Techniques for Queries on Nested Objects," *IEEE Trans. on Knowledge and Data Eng.*, Vol. 1, No. 2, pp. 196-214, June 1989.
- [3] Kim, W., "A Model of Queries for Object-Oriented Databases," In *Proc. Intl. Conf. on Very Large Data Bases*, pp. 423-432, Amsterdam, Aug. 1989.
- [4] Kemper, A. and Moerkotte, G., "Access Support Relations: An Indexing Method for Object Bases," *Information Systems*, Vol. 17, No. 2, pp. 117-145, 1992.
- [5] Bertino, E. and Foscoli, P., "Index Organizations for Object-Oriented Database Systems," *IEEE Trans. on Knowledge and Data Eng.*, Vol. 7, No. 2, pp. 193-209, April 1995.
- [6] Xie, Z. and Han, J., "Join Index Hierarchies for Supporting Efficient Navigations in Object-Oriented Databases," In *Proc. Intl. Conf. on Very Large Data Bases* pp. 522-533, Santiago, Chile, Sept. 1994.
- [7] Finkelstein, S. et al., "Physical Database Design for Relational Databases," *ACM Trans. on Database Systems*, Vol. 13, No. 1, pp. 91-128, Mar. 1988.
- [8] Lee, J. H. et al., "A Physical Database Design Method for Multidimensional File Organizations," *Information Sciences*, Vol. 102, No. 3, pp. 31-65, 1997.
- [9] Whang, K. Y. et al., "Separability-An Approach to Physical Database Design," *IEEE Trans. on Computers*, Vol. C-33, No. 3, pp. 209-222, Mar. 1984.
- [10] Yu, C. T. et al., "Adaptive Record Clustering," *ACM Trans. on Database Systems*, Vol. 10, No. 2, pp. 180-204, June 1985.
- [11] Kifer, M., Kim, W., and Sagiv, Y., "Querying Object-Oriented Databases," In *Proc. Intl. Conf. on Management of Data*, ACM SIGMOD, San Diego, Calif., pp. 393-402, May 1992.
- [12] Kim, K. C. et al., "Acyclic Query Processing in Object-Oriented Databases," In *Proc. Intl. Conf. on Entity-Relationship Approach*, Rome, Italy, pp. 329-346, Nov. 1989.
- [13] Kim, W. et al., "Indexing Techniques for Object-Oriented Databases," *Object-Oriented Concepts, Databases, and Applications*, (Kim, W. and Lochovsky, F.eds.), Addison-Wesley, 1989.
- [14] Mueck, T. A. and Polaschek, M. L., "A Configurable Type Hierarchy Index for OODB," *The VLDB Journal*, Vol. 6, No. 4, pp. 312-332, Nov. 1997.
- [15] Whang, K. Y. and Krishnamurthy, R., *Multilevel Grid Files*, IBM Research Report RC 11516, IBM Thomas J. Watson Research Center, Nov. 1985.
- [16] Bertino, E. and Guglielmino, C., "Path-Index: An Approach to the Efficient Execution of Object-Oriented Queries," *Data & Knowledge*

- Engineering*, Vol. 10, pp. 1-27, 1993.
- [17] Lee, J. H. et al., "A Tunable Class Hierarchy Index for Object-Oriented Databases Using a Multidimensional Index Structure," *Information and Software Technology*, Vol. 43, No. 5, pp. 309-323 , April, 2001.
- [18] Ramaswamy, S. and Kanellakis, P. C., "OODB Indexing by Class-Division," In *Proc. Int'l Conf. on Management of Data*, ACM SIGMOD, pp. 139-150, San Jose, CA, May 1995.
- [19] Sreenath, B. and Seahadri, S., "The hcC-tree: An Efficient Index Structure for Object Oriented Databases,"In *Proc. Int'l Conf. on Very Large Data Bases*, pp. 203-213, Santiago, Chile, Sept. 1994.



이 종 학

1982년 경북대학교 전자공학과
(전자계산 전공) 졸업(학사)

1984년 한국과학기술원 전산학
과 졸업(공학석사)

1997년 한국과학기술원 전산학
과 졸업(공학박사)

1991년 정보처리기술사

1984년 ~ 1987년 금성통신(주) 부설연구소 주임연구원

1987년 ~ 1998년 한국통신 연구개발본부 선임연구원

1998년 ~ 현재 대구가톨릭대학교 컴퓨터정보통신공학부
교수

관심분야 : 데이터베이스 시스템, 객체 데이터베이스, 트
랜잭션 프로세싱, 지리정보 시스템 등

E-mail: jhlee11@cataegu.ac.kr

교 신 저 자

이 종 학 712-702 경북 경산시 하양읍 금락1리 330 대구
가톨릭대학교 컴퓨터정보통신공학부