

# 분산 데이터베이스 시스템에서의 최적 파일 이동 정책

## Optimal File Migration Policies in Distributed Database Systems

이 기 태\*  
 Lee, Ki-Tae  
 김 재 련\*\*  
 Kim, Jae-Yearn

### Abstract

The allocation of files is essential to the efficiency and effectiveness of a distributed system that must meet geographically dispersed data processing demands. In this paper, we address an optimization model that generates optimal file migration policies in distributed database systems. The proposed model is a more generalized model that includes system's capacity constraints - computing sites' storage capacity and communication networks' capacity - which have not taken into consideration in previous researches. Using this model, we can establish initial file allocation, file reallocation and file migration policies that minimize a system operating cost under system's capacity constraints at an initial system design or reorganization point. The proposed model not only can be adopted by small-sized systems but also provides a foundation for effective and simple heuristics for adaptive file migration in large systems.

### 1.1 서론

분산 데이터베이스 시스템은 자원 분할과 부하 밸런싱을 통해 시스템 신뢰도와 가용도의 향상, 시스템 운영비용과 통신비용 절감, 반응시간 단축을 가져온다. 분산 정보 시스템을 설계하는 데 있어서 가장 중요한 문제는 만족스러운 시스템 수행도를 달성하기 위한 적당한 파일 중복 수준과 할당을 결정하는 것이다. 파일 복사본이 할당되어 있는 컴퓨팅 사이트는 빠르고 저렴한 정보검색을 할 수 있지만 파일 저장과 유지 보수의 부담을 감수해야만 한다. 원거리에서 파일을 참조할 경우에는 나머지 사이트에서 통신비용이 발생되고 이동 지연이 발생하게 된다. 그러므로, 시스템 운영비용을 최소화하는 파일 할당을 결정하는 것은 중요한 문제가 된다.

통신 트래픽이 빈번히 원거리에서 발생하는 경우, 전체적인 시스템 운영비용은 증가하고 통신 서비스의 질이 떨어진다. 이러한 상황에서 하나의 대안은 참조 파일의 복사본을 요청 사이트로 이동시켜서 파일 접근이 지역적으로 처리될 수 있도록 하는 것이다. 그러나, 이 경우 전체 시스템의 인터럽션이 발생하는 문제점을 가지고 있다. 시스템 전반에 걸친 파일 재할당과는 달리, 파일 이동은 요청된 한 종류의 파일에 대해서만 재할당을 수행하므로 시스템 인터럽션을 거의 일으키지 않는다. 파일 이동을 통해 짧은 기간 동안 파일 할당을 수정함으로써 시스템 수행력을 높이고 시스템 운영비용을 절감할 수 있다.

동적인 파일 접근 패턴과 빈도를 나타내는 분산 데이터베이스 시스템에 대해서 정적 할당으로 설계하는 것은 비용이나 수행력의 비효율성을 가져온다. 과거의 연구에서는, 대부분 초기 설계나 적용된 관리정책이 시간에 따라 변함이 없다는 가정하에서 다루어져 왔다.

본 연구에서 제시하는 모형은 파일 접근 패턴과 빈도가 시간에 따라 변하는 분산 시스템 환경에서 컴퓨팅 사이트의 저장 용량과 네트워크의 통신 용량에 대한 시스템 제약을 만족하면서 파일 보관 비용과 질의/갱신 통신비용을 최소화하는 동적 파일 이동 정책을 수립하는 모형이다. 분산 데이터베이스 시스템의 초기 설계단계와 재설계 단계에서 보유해야 하는 최적의 파일 복사본의 수와 그 파일 복사본들의 저장 위치를 결정해 주고, 동시에 각 시간 구간 내에서의 최적 파일 이동 정책을 제시해 준다.

\* 한양대학교 산업공학과 대학원

\*\* 한양대학교 산업공학과 교수

## 1.2 파일 할당 문제와 파일 이동 문제

정적 파일 할당 문제(static file allocation)[1,2,7,11,12]는 시스템 설계 시에 계획한 정책에 의한 파일의 배치가 계획 기간 전반에 걸쳐서 변하지 않고 고정되는 문제이다. 그러므로, 파일 접근 패턴과 빈도가 계획 기간 내에서 일정한 안정적인 시스템에 적합하다. 그러나, 현실적으로 파일 접근 패턴이나 빈도가 시간의 변화에 상관없이 일정한 시스템이 거의 존재하지 않으므로 적용할 수 있는 분야가 매우 적다.

동적 파일 할당(dynamic file allocation)[12,13,14,15]은 파일 접근 패턴과 빈도의 변화에 따라 파일 배치를 재할당하는 것이다. 파일을 재배포하기 위해서는 엄청난 비용이 들고 시스템 전반적인 지연 현상이 발생되며 시스템 활동의 변화에 대한 과거의 지식이 요구되므로, 시스템 운영 정책으로 빈번하게 사용하는 것이 어렵다.

파일 이동 문제(file migration)[4,9,10]는 파일 접근 패턴과 빈도에 따라 파일의 배치를 수정한다는 점에서는 동적 파일 할당과 마찬가지로이다. 그러나, 동적 파일 할당에서는 전체 계획 기간으로 봐서는 파일 배치가 동적이지만 일정 시간 구간 내에서는 정적인 반면, 파일 이동 문제에서는 전 계획 구간에서 파일 배치가 동적인 파일 배치 문제이다. 즉, 파일 접근 요청이 발생할 때 마다 요청 사이트에서 작업을 수행할 것인지, 요청 파일의 복사본을 다른 지역에서 가져와서 수행할 것인지, 아니면 원거리에서 요청을 수행하고 그 결과만을 요청 사이트로 가져올 것인지를 결정해 준다. 이렇듯 파일 이동은 접근 요청을 받은 단일 파일의 일시적인 배치를 결정하는 것이므로 시스템 내의 전체 파일의 재배포를 다루는 동적 파일 할당에 비해 거의 시스템 인터럽션을 일으키지 않으므로 빈번하게 사용될 수 있다. 그러므로, 예측하기 힘든 사용 패턴의 일시적인 변화에 대해서 동적 할당에 비해 민감하게 대응할 수 있으며, 파일 접근 패턴이 짧은 기간 동안 빈번하게 변하는 시스템에서도 응용 가능하다.

본 논문에서는 Liu Sheng[4,9,10]의 연구를 확장하여 시스템 제약하에서의 최적 파일 이동 정책을 결정하는 모형을 제시한다. 시스템 제약으로 컴퓨팅 사이트의 저장 용량과 네트워크의 통신 용량이 고려된 보다 확장된 모형이다.

## 2. 수리적 모형의 설정

본 연구에서 사용하는 가정 및 용어를 설명하고, 확장된 모형을 2.4절 및 2.5절에서 제시한다.

### 2.1 모형 가정

- 1) 각각의 파일 접근 요청(file access request)은 단일 파일을 참조하며, 파일 접근 요청의 빈도가 시스템에 알려져 있다.
  - 2) 다른 파일에 대한 요청은 독립적으로 발생하며 모두 포아송 프로세스를 따른다.
  - 3) 파일 요청 빈도는 계획 기간 중 유동적이다. 계획 구간을 복수 시간 구간으로 나누는 데, 각각의 시간 구간에서는 파일 접근 빈도가 똑같은 수준을 유지한다.
  - 4) 파일 이동에 대한 의사결정 과정은 Markov renewal process를 따른다. 여기에서 renewal point 즉, 파일 접근 요청이 도착하는 시점은 파일 이동에 대한 의사결정 시점이다.
- 단일 파일 접근과 다른 종류의 파일간의 접근 독립성 가정에 의해, 복수 파일 시스템에서 파일 이동을 포함하는 파일 배치를 설계하기 위해서는 각각의 최적 파일 이동 정책을 독립적으로 이용함으로써 해결할 수 있다. 그러므로, 단일 파일 이동 모형을 개발하는 것으로 충분하다.

### 2.2 파일 이동 옵션의 종류

시스템에서 파일 접근 요청을 처리하기 위해서 사용되는 파일 이동 옵션은 다음과 같다.

- 1) 파일 생성  
참조 파일의 새로운 복사본이 현존하는 복사본으로부터 생성되고 파일이 저장되지 않은 어떤 사이트로 이동된다.
- 2) 파일 이동(transfer)  
참조 파일의 현존하는 복사본이 현재의 위치에서 제거되고 파일이 저장되지 않은 어떤 다른 위치로

이동(transfer)된다.

3) 파일 소거

파일 복사본이 이 옵션에 의해 소거된다. 파일 소거는 파일 접근 요청이 완료될 때 고려된다. 파일 소거 작업에 의해 시간이 소모되며, 그것의 비용은 무시하는 것으로 가정한다.

4) 원거리 처리

이것은 비이동 옵션이다. 파일 접근 요청이 현존하는 원거리 복사본에서 처리된다.

2.3 기호 및 용어 설명

1) 시스템 파라미터

$I$  : 모든 컴퓨팅 사이트들의 인덱스 집합

$n$  :  $n = |I|$ , 사이트의 수

$i, g, j, k, z$  : 컴퓨팅 사이트의 인덱스

$Y$  : 계획 기간에서의 시간 구간의 수

$y$  : 시간 구간의 인덱스

$T_y$  : 구간  $y$ 의 지속 시간

$\overline{T}_y$  : 상태 전이 간의 기대 시간 간격

$B$  : 이동 파일의 크기

$F_i$  : 의사결정 시점에서 사이트  $i$ 에 저장되어 있는 모든 파일의 크기 합

2) 비용 파라미터

$C_{ij}(\cdot)$  : 사이트  $i$ 와  $j$ 간의 단위 통신비용 함수

$S_i(\cdot)$  : 사이트  $i$ 에서의 단위 파일 저장(유지 보수)비용 함수

$C^s$  : 상태  $\vec{s}$ 에서 옵션  $\vec{a}$ 수행에 따른 할당의 기대 파일 저장 비용

$C^m$  : 옵션  $\vec{a}$ 에 의한 파일 이동 통신비용

$C^p$  : 파일 복사본 접근 통신비용

$C_{\vec{s}}^{\vec{a}}$  : 시스템 상태  $\vec{s}$ 에서 현재의 파일 접근 요청을 옵션  $\vec{a}$ 로 처리하는 작업 비용

3) 파일 접근 요청 파라미터

$r$  : 파일 접근 요청의 트랜잭션 유형(질의 또는 갱신)

$R = \{q, u\}$ , 트랜잭션 유형의 인덱스 집합

$d$  : 파일 접근 요청의 통신량 등급

$D_{ri}$  : 사이트  $i$ 에서 트랜잭션 유형  $r$ 인 요청에 대한 통신량의 등급 인덱스 집합

$\lambda_{rd}(y)$  : 사이트  $i$ 에서 트랜잭션 유형이  $r$ 이고 통신량의 등급이  $d$ 인 파일 접근 요청이 구간  $y$ 에서 발생하는 빈도

$f_{rd}, b_{rd}$  : 트랜잭션 유형이  $r$ 이고 통신량의 등급이  $d$ 인 파일 접근 요청에 의해 발생된 전/후방 통신 트래픽

$l_{rd} : f_{rd} + b_{rd}$

$M_y$  : 구간  $y$ 에서의 파일 접근 요청의 기대 횟수

#### 4 이기태

##### 4) 파일 할당 파라미터

$\vec{X}$  : 이진 파일 할당 벡터;  $\vec{X} = (x_i)_{i \in I}$ , 만일 어떤 파일 복사본이 사이트  $i$ 에 할당되어 있으면  $x_i = 1$ , 그렇지 않으면  $x_i = 0$

$G$  : 가능한 모든 파일 할당의 집합

$I(\vec{X})$  :  $\vec{X}$ 에 의해 파일 복사본이 할당된 컴퓨팅 사이트의 인덱스 집합

$\bar{I}(\vec{X})$  :  $\vec{X}$ 에 의해 파일 복사본이 할당되지 않은 컴퓨팅 사이트의 인덱스 집합

$\vec{X}_{a_1}$  : 옵션의  $a_1$ 모드를 처리한 결과에 따른 파일 할당

##### 5) 시스템 용량 제약 파라미터

$SC_i$  : 사이트  $i$ 의 최대 저장 용량

$NC_{ij}$  : 사이트  $i$ 와  $j$ 간의 통신 네트워크의 최대 통신 용량

##### 6) 모형 파라미터

$\vec{s}$  :  $(\vec{X}, g, r, d)$ . 시스템 상태 벡터. 여기에서  $\vec{X}$ 는 현재의 파일 접근 요청 발생 시점에서 관측된 파일 할당이며,  $g$ 는 파일 접근 요청이 발생한 사이트의 인덱스

$W$  : 상태 공간. 즉, 시스템에서 가능한 모든 상태들의 집합

$\vec{a}$  :  $(a_1 a_2, j, k, z)$ . 파일 이동 옵션 벡터. 여기에서  $j, k, z$ 는 차례로 파일 송신 사이트, 수신 사이트, 소거 사이트를 나타내며  $a_1$ 은 파일 접근 요청의 처리 모드이고  $a_2$ 는 소거 모드를 나타냄.

$A(\vec{s})$  : 옵션 공간. 즉, 상태  $\vec{s}$ 에서 가능한 모든 옵션

$v_{\vec{s}}^{\vec{a}}(y, m)$  : 시스템 상태  $\vec{s}$ 에서 옵션  $\vec{a}$ 를 현재의 의사결정 시점에서 수행하였을 때, 구간  $y$ 에서  $m$ 번째 파일 접근 요청을 처리하는 기대 시스템 운영비용

$v_{\vec{s}}(y, m)$  : 시스템 상태  $\vec{s}$ 에서 구간  $y$ 의  $m$ 번째 파일 접근 요청을 처리하는 최소 기대 시스템 운영비용

$V(y, \vec{X})$  : 초기 파일 할당  $\vec{X}$ 로 시스템을 운영했을 때, 구간  $y$ 에서의 최적 기대 시스템 운영비용

$P_{\vec{s}_1}^{\vec{a}}$  : 구간  $y$ 에서, 현재의 의사결정 시점에서의 상태  $\vec{s}$ 에서 다음 번 의사결정 시점에서의 상태  $\vec{s}_1$ 으로의 전이 확률(transition probability)

$P_{\vec{X} \vec{X}_1}^{\vec{a}}$  : 옵션  $\vec{a}$ 가 수행되었을 때, 파일 할당이  $\vec{X}$ 에서  $\vec{X}_1$ 으로 전이될 조건부 확률

$P_{rd}(y)$  : 구간  $y$ 에서 파라미터  $r, g, d$ 를 가진 파일 접근 요청이 발생할 확률

$e_j, e_k, e_z$  : 차례로  $j$ 번째,  $k$ 번째,  $z$ 번째  $n$ 차 단위 벡터

## 2.4 수리 모형

각 컴퓨팅 사이트의 저장 용량과 통신 네트워크의 용량을 시스템 제약으로 고려하여 Liu Sheng[4,9,10]의 모형을 확장한 동적 파일 이동 정책 수립을 위한 수리모형은 다음과 같다.

### 2.4.1 시스템 상태

어떤 파일 접근 요청이 발생했을 때의 시스템 상태는 벡터  $\vec{s} = (\vec{X}, g, r, d)$ 로 나타낼 수 있다. 파일 접근 요청이 발생했을 때 가능한 시스템 상태의 집합은 다음과 같이 정의된다.

$$W = \{ \vec{s} = (\vec{X}, g, r, d), \text{ where } \vec{X} \in G, g \in I, r \in R \text{ and } d \in D_{rg} \}$$

여기에서, 본 연구에서 제안한 확장모형에서 각 컴퓨팅 사이트의 저장 용량의 제약에 의해 다음의 두 식이 만족된다.

$$x_i = 1, \exists i \text{ for } SC_i \geq F_i + B$$

$$x_i = 0, \forall i \text{ for } SC_i < F_i + B$$

첫 번째 식은 사이트에 저장되어 있는 파일 크기의 합에 이동될 파일의 크기를 합한 것이 사이트의 저장 용량을 초과하지 않는 사이트가 반드시 적어도 하나는 존재해야 한다는 것을 나타낸다. 두 번째 식은 사이트에 저장되어 있는 파일 크기의 합에 이동될 파일의 크기를 합한 것이 사이트의 용량을 초과하는 모든 사이트에는 파일이 할당되지 않는다는 것을 나타낸다.

### 2.4.2 파일 접근 처리 옵션

$$\vec{a} = (a_1 a_2, j, k, z)$$

$$a_1 = \begin{cases} 0, & \text{local processing} \\ 1, & \text{file creation} \\ 2, & \text{file transfer} \\ 3, & \text{standard remote processing} \end{cases}$$

$$a_2 = \begin{cases} 0, & \text{no file deletion} \\ 1, & \text{file deletion} \end{cases}$$

파일 이동 결정을 위한 옵션 공간  $A(\vec{s})$ 는 다음과 같이 정의된다.

$$\vec{a} = (a_1 a_2, j, k, z) \in A(\vec{s}) \forall \vec{s} = (\vec{X}, g, r, d) \in W, \text{ iff}$$

$$a_1 = \begin{cases} = 0 & \text{if } x_g = 1 \\ \in \{1, 2, 3\} & \text{otherwise} \end{cases}$$

$$a_2 \in \{0, 1\}$$

$$j = \begin{cases} = 0 & \text{if } a_1 = 0 \\ \in I(\vec{X}) & \text{otherwise} \end{cases}$$

$$k = \begin{cases} = 0 & \text{if } a_1 = 0 \text{ or } 3 \\ \in \bar{I}(\vec{X}) & \text{otherwise} \end{cases}$$

$$z = \begin{cases} = 0 & \text{if } a_2 = 0 \\ \in I(\vec{X}_{a_1}) & \text{otherwise} \end{cases}$$

### 2.4.3 재귀 관계

m번째 파일 접근 요청에 대한 최적 기대 운영비용은 다음과 같다.

$$v_s^{\vec{a}}(y, m) = \min_{\vec{s} \in A(\vec{s})} v_{\vec{s}}^{\vec{a}}(y, m) \quad (1)$$

최적성의 원리에 따라  $v_s^{\vec{a}}(y, m)$ 는 다음 결정 시점까지 구간을 분할함으로써 유도된다. 연속적인 최적 파일 이동 결정간에는 다음과 같은 재귀 관계가 성립된다.

$$v_s^{\vec{a}}(y, m) = c_s^{\vec{a}} + \sum_{s_1 \in W} P_{s s_1}^{\vec{a}}(y) v_{s_1}^{\vec{a}}(y, m+1), \quad \forall \vec{s} \in W \quad (2)$$

식(2)를 식(1)에 대입하면 다음과 같다.

$$v_s^{\vec{a}}(y, m) = \min_{\vec{s} \in A(\vec{s})} \{c_s^{\vec{a}} + \sum_{s_1 \in W} P_{s s_1}^{\vec{a}}(y) v_{s_1}^{\vec{a}}(y, m+1)\}$$

초기 파일 할당  $\vec{X}_0$ 로 시스템을 운영했을 때, 기간  $y$ 에서의 최적 기대 시스템 운영비용(optimal expected operating cost)은 다음과 같다.

$$V(y, \vec{X}) = \sum_{g \in I} \sum_{r \in R} \sum_{d \in D_n} P_{rgd}(y) v_s^{\vec{a}}(y, 1) \quad \forall \vec{X} \in G$$

결론적으로, 모형은 다음과 같은 식으로 정리된다.

$$\begin{aligned} & \therefore \forall \vec{s} = (\vec{X}, g, r, d) \in W \text{ and } 1 \leq y \leq Y \\ v_s^{\vec{a}}(y, m) &= \begin{cases} \min_{\vec{a} \in A(\vec{s})} \{c_s^{\vec{a}} + \sum_{s_1 \in W} P_{s s_1}^{\vec{a}}(y) v_{s_1}^{\vec{a}}(y, m+1)\} & \text{if } 0 < m < M_y + 1 \\ \min_{\vec{X}_1 \in G} \sum_{j \in I(\vec{X}_1)} [ \min_{i \in I(\vec{X})} c_{ij}(B) ] + V(y+1, \vec{X}_1) & \text{if } m = M_y + 1 \text{ and } y < Y \end{cases} \end{aligned}$$

$$M_y = \sum_{i \in I} \sum_{r \in R} \sum_{d \in D_n} \lambda_{rid}(y) \times T_y$$

- 최적 초기 파일 할당

$$\min_{\vec{X} \in G} V(1, \vec{X})$$

## 2.5 상태 전이 확률과 작업 비용

### 2.5.1 상태 전이 확률

$$P_{s s_1}^{\vec{a}}(y) = P_{\vec{X} \vec{X}_1}^{\vec{a}} P_{r, g, d_1}(y)$$

$$P_{r, g, d_1}(y) = \frac{\lambda_{r, g, d_1}(y)}{\sum_{i \in I} \sum_{r \in R} \sum_{d \in D_n} \lambda_{rid}(y)}$$

$$P_{\vec{X} \vec{X}_1}^{\vec{a}} = \begin{cases} 1 & \text{if } \vec{X}_1 = \vec{X}^{\vec{a}} \text{ and } x_i = 0, \forall i \text{ for } SC_i < F_i - B \\ 0 & \text{otherwise} \end{cases}$$

$$\vec{X}^{\vec{a}} = \begin{cases} \vec{X}_{a_1}^{\vec{a}} & \text{if } a_2 = 0 \\ \vec{X}_{a_1}^{\vec{a}} - e_2 & \text{if } a_2 = 1 \end{cases} \text{ and } \vec{X}_{a_1}^{\vec{a}} = \begin{cases} \vec{X} & \text{if } a_1 = 0 \text{ or } 3 \\ \vec{X} - e_k & \text{if } a_1 = 1 \\ \vec{X} - e_k & \text{if } a_1 = 2 \end{cases}$$

2.5.2 작업 비용

작업 비용은 다음 식과 같이 저장비용, 파일 이동 통신비용, 파일 접근 통신비용, 갱신 통신비용의 합으로 이루어진다.

$$c_s^{\vec{a}} = c^s + c^m + c^p \quad \forall \vec{s} \in W, \vec{a} \in A(\vec{s})$$

작업 비용의 각 요소는 다음과 같다.

저장비용

$$c^s = \sum_{i \in I(X_j)} S_i(B) \cdot \overline{T}_y \quad \forall \vec{s} = (\overline{X}, g, r, d) \in W \text{ and } \vec{a} = (a_1 a_2, j, k, z) \in A(\vec{s})$$

여기에서, 본 연구에서 수리모형에서 고려한 컴퓨팅 사이트의 저장 용량 제약으로 인하여 다음식이 성립한다.

$$S_i(B) = \infty, \forall i \text{ for } SC_i < F_i + B$$

통신비용은 파일 이동 옵션을 사용하여 파일을 이동시키는 데 드는 파일 이동 통신 비용과 파일 이동 작업이 수행되든 안되는 간에 현재의 파일 접근 요청을 완료하기 위해 파일 복사본에 접근하는 통신 비용으로 나눌 수 있다. 각 통신비용은 본 모형에서 고려한 통신 네트워크의 통신 용량의 제약으로 인해 다음과 같이 나타낼 수 있다.

- 파일 이동 통신비용

$$\forall \vec{s} = (\overline{X}, g, r, d) \in W, \text{ and } \vec{a} = (a_1 a_2, j, k, z) \in A(\vec{s})$$

$$c^m = \begin{cases} c_{jk}(B) & \text{if } a_1 = 1 \text{ or } 2 \\ \infty & \text{if } B > NC_{jk} \\ 0 & \text{if } a_1 = 0 \text{ or } 3 \end{cases}$$

- 파일 접근 통신비용

$$\forall \vec{s} = (\overline{X}, g, r, d) \in W, \text{ and } \vec{a} = (a_1 a_2, j, k, z) \in A(\vec{s}) \text{ and } r = q$$

$$c^p = \begin{cases} 0 & \text{if } a_1 = 0 \\ c_{gk}(f_{jd}) + c_{kg}(b_{rd}) & \text{if } a_1 = 1 \text{ or } 2 \text{ and } \max(f_{rd}, b_{rd}) \leq NC_{gk} \\ \infty & \text{if } a_1 = 1 \text{ or } 2 \text{ and } \max(f_{rd}, b_{rd}) > NC_{gk} \\ c_{gi}(f_{jd}) + c_{ig}(b_{rd}) & \text{if } a_1 = 3 \text{ and } \max(f_{rd}, b_{rd}) \leq NC_{gk} \\ \infty & \text{if } a_1 = 3 \text{ and } \max(f_{rd}, b_{rd}) > NC_{gk} \end{cases}$$

- 갱신 통신비용

$$\forall \vec{s} = (\overline{X}, g, r, d) \in W, \text{ and } \vec{a} = (a_1 a_2, j, k, z) \in A(\vec{s}) \text{ and } r = u$$

$$c^p = \begin{cases} \sum_{i \in I(X_j)} [c_{gi}(f_{jd}) - c_{ig}(b_{rd})] & \\ \text{if } a_1 = 0 \text{ and } \max(f_{rd}, b_{rd}) \leq NC_{gi} & \\ \infty & \text{if } a_1 = 0 \text{ and } \max(f_{rd}, b_{rd}) > NC_{gi} \\ c_{gk}(f_{jd}) - c_{kg}(b_{rd}) - \sum_{i \in I(X_j)} [c_{gi}(f_{jd}) - c_{ig}(b_{rd})] & \\ \text{if } (a_1 = 1 \text{ or } 2) \text{ and } \max(f_{rd}, b_{rd}) \leq NC_{gk} \text{ and } \max(f_{rd}, b_{rd}) \leq NC_{gi} & \\ \infty & \text{if } (a_1 = 1 \text{ or } 2) \text{ and } \max(f_{rd}, b_{rd}) > NC_{gk} \text{ and } \max(f_{rd}, b_{rd}) > NC_{gi} \\ c_{gi}(f_{jd}) - c_{ig}(b_{rd}) - \sum_{i \in I(X_j)} [c_{ii}(f_{jd}) - c_{ii}(b_{rd})] & \\ \text{if } a_1 = 3 \text{ and } \max(f_{rd}, b_{rd}) \leq NC_{gi} \text{ and } \max(f_{rd}, b_{rd}) \leq NC_{ij} & \\ \infty & \text{if } a_1 = 3 \text{ and } \max(f_{rd}, b_{rd}) > NC_{gi} \text{ or } \max(f_{rd}, b_{rd}) > NC_{ij} \end{cases}$$

3. 수치 예제

본 논문에서는 3개의 컴퓨팅 사이트가 서로 완전하게 연결되어 있는 분산 데이터베이스 시스템을 예로 든다. 다음의 도표는 차례로 컴퓨팅 사이트의 저장 용량, 통신 네트워크의 통신 용량과 통신비용컴퓨팅 사이트의 저장비용을 나타낸다.

<Table 1> The storage capacities of computing sites(Mb)

$SC_1$	$SC_2$	$SC_3$
20	15	10

<Table 2> The capacities of communication networks (Mb)

$NC_{12}$	$NC_{23}$	$NC_{31}$
18	15	12

<Table 3> Communication costs (unit cost/Mb)

$C_{ij}$	1	2	3
1	0	6	12
2	6	0	6
3	12	6	0

<Table 4> Storage cost (unit cost/Mb)

$S_1$	$S_2$	$S_3$
2	4	1

컴퓨팅 사이트에 도착하는 파일 접근 요청의 패턴과 빈도, 파일 접근 요청의 평균 통신 트래픽은 다음과 같다.

<Table 5> The intensities of file access request

	Query						Update		
Period(t)	1	2	3	1	2	3	1	2	3
Class(d)	1			2			3		
Site(i)	Intensities of file access request								
1	1	1	0	1	2	0	1	1	1
2	0	1	2	0	1	1	1	2	1
3	2	0	1	1	1	1	1	1	1

<Table 6> The duration of time periods

$T_1$	$T_2$	$T_3$
0.375	0.5	0.25

<Table 7> Average communication traffics

	Transaction type(r)		
	Query		Update
Class(d)	1	2	1
$f_{rd}, b_{rd}$	14	5	1

이동 계획을 세우려는 파일의 크기는 12(Mb)이며, 계산상의 편의를 위하여 모든 의사결정 시점에서 각 사이트에 이미 저장되어 있는 파일의 크기 합은 0이라고 가정한다. 각 시간 구간에서의 파일 접근 요청의 기대 횟수와 상태 전이 간의 기대 시간 간격은 다음과 같이 계산된다.

<Table 8> The expected number of file access request

$M_1$	$M_2$	$M_3$
3	5	2

<Table 9> The expected time interval

$\bar{T}_1$	$\bar{T}_2$	$\bar{T}_3$
0.125	0.1	0.125

본 논문에서는 해법 중 수치 반복법(value iteration)을 사용하였다. 이 방법은 수치 공간(value space)에서 계속적인 평가를 수행하여 해를 구하는 방법이다.  $v_s^m(y, M_y + 1)$ 이 주어졌을 때,  $m = M_y$  부터  $m = 1$ 까지 재귀식을 풀어서 해를 구할 수 있다.  $v_s^1(y, M_y + 1) = 0, \forall s \in W$ 이라고 두고 문제를 풀어 나간다. 그 결과의 일부는 다음과 같다.



<Table 10> The expected operating costs over period 1 with an initial file allocation  $\vec{X}$

V(1,1,0,0)	V(1,0,1,0)	V(1,1,1,0)
876.0854	910.8010	184.875*

위의 결과는 최적 초기 할당을 계산 한 결과이다. 사이트 1과 2에 파일을 할당하는 것이 최적의 초기 할당이라는 것을 나타낸다. 다음은 파일 이동 정책을 나타낸 표이다.

<Table 11> The expected operating costs in period 1

y=1, m=3

( $v_{\vec{s}}^{\vec{a}}(y, m)$ , \* is optimal)

$\vec{s} \backslash \vec{a}$	(1,0,0, 1,q,1)	(1,0,0, 1,q,2)	(1,0,0, 1,u,1)	(1,0,0, 2,q,1)	(1,0,0, 2,q,2)	(1,0,0, 2,u,1)	(1,0,0, 3,q,1)	(1,0,0, 3,q,2)	(1,0,0, 3,u,1)
(00,0,0,0)	3*	3*	3*						
(01,0,0,1)									
(01,0,0,2)									
(10,1,2,0)				81	81	93	249	141	105
(10,2,1,0)									
(11,2,1,0)				78	78	78	246	138	90
(11,2,1,1)				75*	75	87	243*	135	99
(11,2,1,2)									
(20,1,2,0)									
(20,2,1,0)				78	78	78	246	138	90
(30,1,0,0)									
(30,2,0,0)				171	63*	15*	∞	123*	27*
(31,1,0,1)									
(31,1,0,1)									
(31,1,0,2)									
(31,2,0,1)									
(31,2,0,2)									

앞의 표는 시간 구간 1에서 발생한 3번째 파일 접근 요청에 대한 처리 비용을 나타낸 것이다. 표에서 보듯이 시스템 상태가  $\vec{s}=(1, 0, 0, 3, q, 1)$ 일 때의 최적해는  $\vec{a}=(11, 1, 2, 2)$ 이다. 이것은 사이트 1에만 파일이 저장되어 있을 때, 사이트 3에서 통신량이 1등급인 질의가 시간 구간1에서 3번째로 발생한다면, 사이트1에 저장되어 있는 파일을 사이트2로 생성하여 질의를 처리한 후 사이트 2에 저장된 파일을 소거하는 것이 최적의 파일 접근 처리 옵션이라는 것을 나타낸다. 다음 표는 재할당 정책을 나타낸 표이다. 시간 구간이 2에서 3으로 바뀔 시점에서 파일이 사이트 1과 2에 할당되어 있다면, 사이트 1에 있는 파일을 소거하고 사이트 2에만 파일을 저장하는 재할당 정책을 수행하는 것이 최적이라는 것을 나타낸다.

<Table 12> The file reallocation policies (y=2 y=3)

$v_{\vec{s}}(2, 6)$

$\vec{X} \backslash \vec{X}_1$	(1,0,0)	(0,1,0)	(1,1,0)
(1,0,0)	135.0937*	146.25	147.9375
(0,1,0)	207.0937	74.25*	147.9375
(1,1,0)	135.0937	74.25*	75.9375

#### 4. 결 론

분산 데이터베이스 시스템을 정적 파일 할당 정책으로 운영하는 것보다 동적 파일 할당 정책으로 운영하는 것이 경제적이다[9]. 분산 시스템에서 파일 이동은 주기적인 파일 재할당에 비해 데이터 처리 능력 향상과 시스템 운영비용 절감을 가져온다. 연속적인 파일 이동 정책들간의 재귀 관계로부터 Liu Sheng[4,9,10]의 Markov 결정 모형을 확장하였다. 제시된 모형은 시스템 제약하에서 동적 파일 이동 정책을 제시하는 보다 확장된 모형이다. 시스템 제약으로 기존 연구에서 고려하지 않았던 각 컴퓨팅 사이트의 저장 용량과 통신 네트워크의 용량이 고려된 보다 현실에 가까운 모형이다. 이 모형을 사용하여 시스템 설계시 또는 재설계시에 시스템 제약을 만족시키면서 시스템 운영비용을 최소화하는 초기 파일 할당 정책, 파일 재할당 정책, 파일 이동 정책을 수립할 수 있다. 수립된 정책을 분산 정보 시스템에 반영하여 시스템의 운영을 온라인 동적 제어한다면, 효율적이고 경제적으로 분산 시스템을 운영할 수 있을 것이다. 파일 접근 요청 간의 종속성이 존재할 경우나 복수 파일을 참조하는 경우의 파일 이동 정책에 관한 연구와 파일의 가용도나 시스템의 신뢰도를 도입한 확장 모형의 제시, 그리고 효율적인 휴리스틱 해법의 개발이 주요 추후 연구과제이다.

#### 참 고 문 헌

1. Casey, R. G., "Allocation of copies of a file in an information network", In proceedings of the Spring Joint Computer Conference IFIPS, pp. 617-625, 1972.
2. Chu, W. W., "Optimal file allocation in a multiple computer system", IEEE Trans. Comput., Vol. C-18, pp. 885-889, 1969.
3. Gavish, B. and Pirkul, H., "Computer and database location in distributed computer systems", IEEE Trans. Comput., Vol. C-35, pp. 583-590, 1986.
4. Gavish, B. and Liu Sheng, O. R., "Dynamic file migration in distributed computer systems", Commun. ACM, Vol. 33, pp. 177-189, 1990.
5. Ghosh, D. and Murthy, I., "A solution procedure for the file allocation problem with file availability and response time", Computers Ops Res., Vol. 18, pp. 557-568, 1991.
6. Irani, K. B. and Khabbaz, N. G., "A methodology for the design of communication networks and the distribution of data in distributed super computer systems", IEEE Trans. Comput., Vol. C-31, pp. 419-434, 1982.
7. Lanning, L. J. and Leonard, M. S., "File allocation in a distributed computer communication network", IEEE Trans. Comput., Vol. C-32, pp. 232-244, 1983.
8. Lee, H. and Liu Sheng, O. R., "A multiple criteria model for the allocation of data files in a distributed information system", Computers Ops Res., Vol. 19, pp. 21-33, 1992.
9. Liu Sheng, O. R., "Optimization of file migration policies in distributed computer systems", Computers Ops Res., Vol. 19, pp. 335-351, 1992.
10. Liu Sheng, O. R., "Analysis of optimal file migration policies in distributed computer systems", Management Science, Vol. 38, pp. 459-482, 1992.
11. Mahmoud, S., and Riordan, J. S., "Optimal allocation of resources in distributed information networks", ACM Trans. Database System, Vol. 1, pp. 67-78, 1976.
12. Morgan, H. L., and Levin, K. D., "Optimal program and data locations in computer networks", Commun. ACM, Vol. 20-5, pp. 315-322, 1977.
13. Segall, A., "Dynamic file assignment in a computer network", IEEE Trans. Automat. Contr., Vol. AC-21, pp. 161-173, 1976.
14. Segall, A. and Sandell, N. R., "Dynamic file assignment in a computer network-part II: Decentralized control", IEEE Trans. Automat. Contr., Vol. AC-24, pp. 709-715, 1979.
15. Wha, B. W., "File placement on distributed computer systems", IEEE Comput., Vol. 17, pp. 23-30, 1984.