데스크톱 그리드에서 자원 클러스터링을 이용한 작업 결과 검증에 관한 연구†

강지훈*, 송성진*, 길준민**
*고려대학교 컴퓨터교육학과
**대구가톨릭대학교 IT공학부
e-mail: {k2j23h, imagessj}@korea.ac.kr, jmgil@cu.ac

A Study on Task Result Verification using Resource Clustering in Desktop Grids

Jihun Kang*, SungJin Song*, Joon-Min Gil**
*Dept. of Computer Science Education, Korea University
**School of IT Engineering, Catholic University of Daegu

요 약

데스크톱 그리드에서는 휘발성과 이질성과 같은 동적 특성을 갖는 자원의 자율적인 수행에 의해 얻어진 작업 결과의 검증이 중요하다. 이를 위해, 본 논문에서는 자원의 동적 특성을 신뢰도와 결과반환 확률로 정의하고 k-means 클러스터링 알고리즘을 적용하여 자원들을 자원 그룹으로 분류하고, 분류된 자원 그룹에 따라 작업의 복제수를 결정하는 자원 클러스터링 기반의 결과 검증 기법을 제안한다.

1. 서론

데스크톱 그리드를 통해 수행된 작업 결과의 정확성을 보장하기 위해서는 불특정 자원에서 수행된 작업 결과의 검증이 요구된다[1]. 전형적으로 과반수투표 기반 기법[2]과 신뢰도 기반 기법[3]이 작업 결과의 검증을 위해 주로 사용되어 왔다. 그러나 이들 두 기법은 데스크톱 그리드의 동적 자원제공 환경의 대처에 미흡하여 불필요한 작업 복제와 반환시간의 증가라는 문제가 있다. 특히, 불특정 자원에 기반을 두고 작업을 수행하는 데스크톱 그리드는 다양한 오류 발생 원인을 가지고 있다. 데스크톱 그리드 환경에서 발생 가능한 오류들은 크게, CPU, OS, 메모리 등 이질적 하드웨어와 소프트웨어로 인한 정밀도(precision) 차이, CPU 내부의 잘못된 연산 등에 의한 내부적 오류와 악의적 사용자가 작업 결과를 의도적으로 변형함으로써 발생하는 외부적 오류로 분류할 수 있다.

이러한 오류들에 능동적으로 대처하기 위해, 본 논문에서는 자원의 동적 특성인 신뢰도와 결과반환 확률에 따라 자원들을 자원 그룹으로 분류하고, 분류된 자원 그룹에 따라 작업 복제수를 결정하는 자원 클러스터링 기반의 결과 검증 기법을 제안한다. 자원의 동적 특성인 신뢰도와 결과반환 확률을 고려한 본 논문의 결과 검증 기법은 작업의 마감시간까지 정확한 결과의 반환을 보장함과 동시에 자원 그룹에 따라 작업 복제수를 결정하므로 불필요한 자원 낭비를 방지하고 전체 작업의 반환시간을 줄여준다. 게다가, 시뮬레이션 결과는 본 논문의 제안 기법이 반환시간과 자원 사용량의 관점에서 기존 기법보다 성능상의 우수함을 보인다.

† 이 논문은 2014년도 정부(교육부)의 재원으로 한국연구재단의 지원을 받아 수행된 기초연구사업임(No. NRF-2014R1A1A2055463).

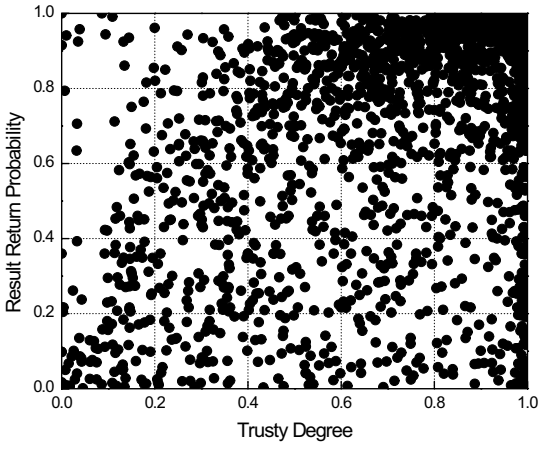
2. 자원 클러스터링

2.1 자원 분류

데스크톱 그리드 환경에서 자원의 동적 특성에 따라 자원을 분류하기 위해, 본 논문에서는 다음과 같이 자원의 신뢰도와 결과반환 확률을 정의한다[4].

정의 1: 신뢰도(Trusty Degree) - 하나의 자원에 의해 수행된 작업 결과들의 정확한 정도.

정의 2: 결과반환 확률(Result Return Probability) - 실패가 있음에도 불구하고 하나의 자원이 주어진 마감시간까지 작업을 완료할 확률.



(그림 1) 자원 분포[4]

본 논문에서는 Korea@Home 데스크톱 그리드 시스템 [5]의 실제 로그 데이터를 활용하여 정의 1과 정의 2에 따라 자원들의 분포를 추출하였다. (그림 1)은 2008년 3월 한 달을 기준으로 축적된 로그 데이터를 활용하여 얻은 자원 분포를 보여준다.

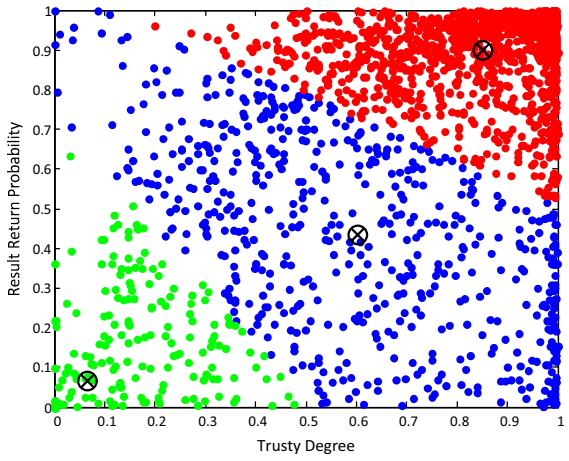
2.2 k-means 클러스터링 알고리즘

주어진 데이터 집합을 유사한 데이터들로 구성된 그룹으로 분류하기 위해 클러스터링 기법이 주로 활용되어 왔다. 본 논문에서는 다양한 클러스터링 기법 중에 k-means 클러스터링 알고리즘을 사용한다. k-means 클러스터링 알고리즘은 효율성과 단순성으로 인해 마케팅, 생물정보학, 패턴인식, 웹 마이닝, 소셜 네트워크 분석 등 다양한 분야에서 활용되고 있다. k-means 클러스터링 알고리즘은 다음과 같이 기술될 수 있다[6].

$$E=\sum_{i=1}^{k}\sum_{j\in C_i} \| x_j - c_i \| \quad (1)$$

여기서, k는 클러스터의 수, x_j는 i번째 클러스터 C_i 내의 j번째 데이터, c_i는 클러스터 C_i의 중점을 나타낸다. x_j와 c_i 사이의 거리는 유사도 측정을 위해 사용되며, 일반적으로 유클리디안 거리를 사용한다. 적용 가능한 k개의 클러스터를 얻기 위해, 수식 (1)의 E는 가능한 작은 값이 되도록 해야 한다.

(그림 1)의 자원 분류를 기반으로, 본 논문에서는 수식 (1)의 k-means 클러스터링 알고리즘을 적용하여 자원을 분류하였다. (그림 2)는 클러스터의 수를 3으로 하였을 경우(즉, k=3)에 얻은 자원 클러스터링의 결과를 보여준다.



(그림 2) k=3인 경우 자원 클러스터링 결과[4]

3. 자원 클러스터링을 이용한 결과 검증 기법

이 절에서는 본 논문에서 제안하는 자원 클러스터링 기반의 결과 검증 기법을 기술한다. (그림 3)과 (그림 4)는 본 논문의 결과 검증 기법에서 핵심이 되는 작업 복제 알고리즘과 결과 검증 알고리즘을 각각을 보여준다. (그림 3)의 작업 복제 알고리즘은 개별 작업에 대한 복제수를 자원 클러스터링에 기반하여 결정한다. 작업 풀에 속한 하나의 작업에 대해 자원 풀에서 특정 자원이 선택되면, 선택된 자원이 속한 자원 그룹의 특징에 따라 복제수를 결정한다. 이때, 자원 클러스터링에 의해 분류된 자원 그룹의 인덱스를 작업의 복제수로 사용한다. 예를 들어, 선택된 자원이 첫번째 자원 그룹에 속하면(즉, $r_j \in C_1$)이면 복제수를 1로 설정한다. 만일 세번째 자원 그룹에 속하면(즉, $r_j \in C_3$)이면, 복제수를 3으로 설정한다.

(그림 4)의 결과 검증 알고리즘은 자원들이 수행한 복제수 만큼의 작업 결과를 서버가 받으면, 과반수 투표 방식에 의해 결과의 정확도를 측정한다. 동일한 결과가 과반수 이상이면 정확한 결과로 받아들이며, 그렇지 않다면 잘못된 결과로서 해당 작업을 작업 풀에 넣고 작업을 다시 수행시키도록 한다.

```
1: TASKS = {t1, t2,..., tT}
2: RESOURCES = {r1, r2,..., rR}
3: while TASKS ≠ ∅ do
4:    waitForEvent()
5:    If (Event(ri, 'taskResult')) then
6:       select a task tk ∈ TASKS
7:       ALLOCATION = {ri}
8:       D = index of the closest cluster center to ri
9:       while (| ALLOCATION | ≤ D) do
10:          select a resource rj ∈ RESOURCES
                 with greatest contribution
11:          ALLOCATION = ALLOCATION ∪
                 {rj}
12:      end while
13:      for all ri ∈ ALLOCATION do
14:          distribute tk to ri
15:          RESOURCES = RESOURCES − {ri}
16:      end for
17:      TASKS = TASKS − {tk}
18:   end if
19: end while
```

(그림 3) 작업 복제 알고리즘

```
1: RESULTS = ∅
2: while (| RESULTS | ≤ D) do
3:    waitForEvent()
4:    if (Event(ri, 'resultReturn')) then
5:       RESULTS = RESULTS ∪ {resj of rj}
6:       RESOURCES = RESOURCES ∪ {rj}
7:    end if
8: end while
9: find a result resj ∈ RESULTS with majority
10: set S to a number of resj in RESULTS
11: if (S ≥ ⌈ D/2 ⌉ ) then
12:    send resj to a management server
13: else
14:    TASKS = TASKS ∪ {tk}
15: end if
```

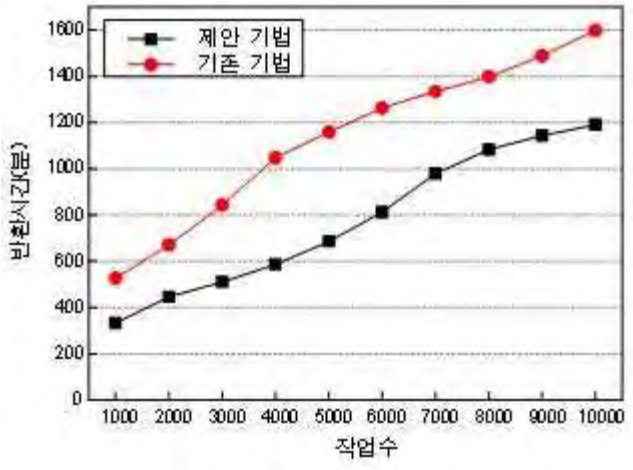
(그림 4) 결과 검증 알고리즘

4. 성능 평가

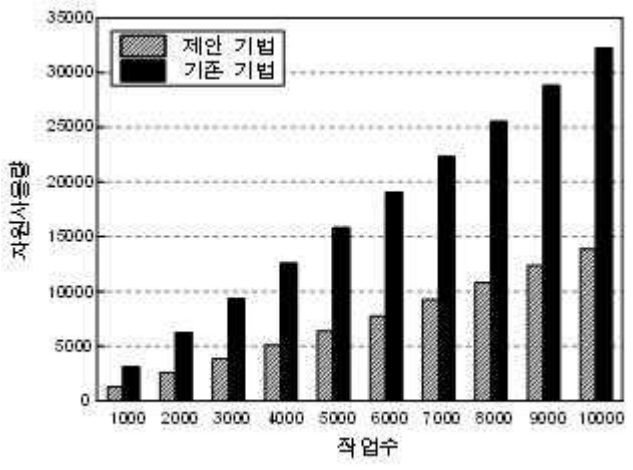
제안한 결과 검증 기법의 성능 평가를 위해 본 논문에서는 전체 작업의 반환시간과 자원 사용량의 측면에서 시뮬레이션을 수행하였다. 시뮬레이션을 위해 필요한 작업 수행과 자원 활용에 대한 데이터는 2008년 3월 한 달 동안 축적된 Korea@Home 데스크톱 그리드 시스템의 로그 데이터를 사용하였다. 성능 측정을 위해 사용된 전체 작업의 반환시간과 자원 사용량의 정의는 다음과 같다.

- 반환시간: 첫번째 작업의 제출부터 시작하여 마지막 작업의 결과가 서버에 도착할 때까지의 전체 시간
- 자원 사용량: 실패가 있음에도 불구하고 전체 작업의 수행을 위해 소비된 자원의 총 수

위의 성능 측정 항목을 사용하여 기존 기법인 과반수 투표 기반의 결과 검증 기법과 본 논문의 제안 기법을 비교하였다. 본 논문에서는 1000개부터 10000개까지의 작업 수를 1000개씩 변화해 가면서 시뮬레이션을 수행하였다. 30번의 시뮬레이션이 각 기법마다 수행되었고, 이들 시뮬레이션 수행 결과의 평균값을 비교하였다.



(그림 5) 반환시간의 비교



(그림 6) 자원사용량의 비교

(그림 5)와 (그림 6)은 반환시간과 자원사용량 관점에서 본 논문의 제안 기법과 기존 기법에 대한 성능 결과를 각각 보여준다. 기존 기법의 작업 복제수는 3개가 사용되었다. (그림 5)에서 볼 수 있듯이, 본 논문의 제안 기법은 작업수에 상관없이 기존 기법보다 빠른 반환시간을 보여준다. 또한, 자원사용량을 비교한 성능 결과인 (그림 6)은 본 논문의 제안 기법이 기존 기법에 비해 모든 작업수에 대해 적은 수의 자원이 사용되었음을 보여준다. 이러한 성능 결과는 본 논문의 제안 기법이 결과 검증 과정 중에 자원의 신뢰도와 결과반환 확률에 따라 현재 자원제공 환경에 적절한 중복수를 결정하였기 때문이다. 즉, 본 논문의 제안 기법은 자원 클러스터링에 의해 높은 신뢰도와 결과반환 확률을 가진 자원에 대해서는 적은 수의 복제수를 결정할 수 있다. 이렇게 함으로써, 불필요한 자원의 낭비를 줄이며, 더욱이 결과반환 확률에 따라 마감시간에 맞는 자원을 선택하므로 반환시간을 절감할 수 있었다. 결론적으로, 본 논문의 제안 기법은 기존 기법에 비해 적은 수의 자원을 사용함에도 불구하고 빠른 반환시간을 가진다.

5. 결론

본 논문에서는 데스크톱 그리드 환경에서 작업 결과의 정확성을 보장하기 위한 결과 검증을 지원하는 자원 클러스터링 기반의 결과 검증 기법을 제안하였다. 제안한 결과 검증 기법은 현재 자원 제공 환경에 맞게끔 작업의 복제수를 결정하기 위해 자원의 신뢰도와 결과반환 확률을 자원 분류의 파라미터로 사용하였다. 따라서 본 논문의 결과 검증 기법은 작업의 마감시간까지 정확한 작업 결과를 얻는 것을 보장해 준다. 시뮬레이션을 통한 성능 평가는 본 논문의 결과 검증 기법이 기존의 결과 검증 기법에 비해 적은 수의 자원을 사용함에도 불구하고 빠른 반환시간을 가짐을 보여 주었다.

참고문헌

[1] C. Cerin and G. Fedak, Desktop Grid Computing, Chapman and Hall/CRC, 2012.

[2] D. P. Anderson, E. Korpela, and R. Walton, “High-Performance Task Distribution for Volunteer Computing,” Proc. of the 1st Int. Conf. on e-Science and Grid Computing, Melbourne, pp. 196-203, July, 2005.

[3] S. Zhao, V. Lo, and C. G. Dickey, “Result Verification and Trust-based Scheduling in Peer-to-Peer Grids,” Proc. of the 5th IEEE Int. Conf. on Peer-to-Peer Computing, pp.31-38, Sept., 2005.

[4] J.-M. Gil, S. Kim, and J. Lee, "Task Scheduling Scheme Based on Resource Clustering in Desktop Grids," Int. J. of Communication Systems, Vol. 27, No. 6, pp. 918 - 930, June 2014.

[5] Korea@Home, http://www/koreaathome.org/eng/.

[6] R. Xu, D. Wunsch. Clustering. John Wiley & Sons: Hoboken, New Jersey, USA, 2008.