

## 효율적인 동적 작업부하 균등화 정책

저자일\*, 저자이\*, 저자삼<sup>○</sup>(발표자 표기)

\*한국컴퓨터대학교 전자계산학과

<sup>○</sup>한국디지털대학교 컴퓨터공학과

e-mail: {author1, author3}@hcu.ac.kr\*, author2@hdu.ac.kr<sup>○</sup>

## An Efficient Dynamic Workload Balancing Strategy

Ja-Il Jeo\*, Ja-E Jeo\*, Ja-Sam Jeo<sup>○</sup>

\*Dept. of Computer Science, Korea-Computer University

<sup>○</sup>Dept. of Computer Engineering, Korea-Digital University

### 요 약

본 논문에서는 고성능 컴퓨팅 시스템의 성능 향상을 위한 효율적인 동적 작업부하 균등화 정책을 제안한다. 이 정책은 시스템 자원인 CPU와 메모리를 효율적으로 사용하여 고성능 컴퓨팅 시스템의 처리량을 최대화하고, 각 작업의 수행시간을 최소화한다. 또한 이 정책은 수행중인 작업의 메모리 요구량과 각 노드의 부하 상태를 파악하여 작업을 동적으로 할당한다. 이때 작업을 할당 받은 노드가 과부하 상태가 되면 다른 노드로 작업을 이주시켜 각 노드의 작업부하를 균등하게 유지함으로써 작업의 대기시간을 줄이고, 각 작업의 수행시간을 단축한다. 본 논문에서는 시뮬레이션을 통하여 제안하는 동적 작업부하 균등화 정책이 기존의 메모리 기반의 작업부하 균등화 정책에 비해 고성능 컴퓨팅 시스템의 성능 향상 면에서 우수함을 보인다.

▶ Keyword : 작업부하(workload), 이주(migration), 부하균등화(load balancing)

### I. Introduction

한국컴퓨터정보학회 발전 방안을 위한 서론을 기술하고 있으며, 본 방안을 수행하기 위한 본 학회의 발전은 다음과 같이 기술할 수 있으며, 향후 논의를 통해 발전 방안을 제시할 수 있습니다. 추가적으로 기록 할 내용은 연구내용과 함께..

### II. Preliminaries

#### 1. Related works

##### 1.1 국내 동향

한국컴퓨터정보학회 발전방안을 위한 관련연구로서 다양한 의견들이 제시되고 있다. 먼저, 발전방안의 기술을 살펴보면 이해그룹의 장점과 단점을 살펴보고, 이를 통하여..

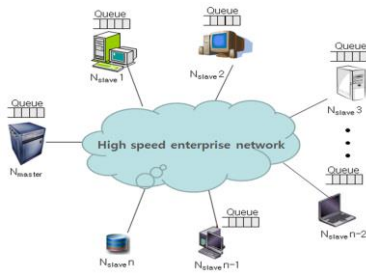


Fig. 1. System Architecture

먼저, 발전방안의 기술을 살펴보면 이해그룹의 장점과 단점을 살펴보면,

Table 1. System Environment

Item	Value
CPU Clock Speed	100 ~ 500 MIPS
Memory Size	32 ~ 256 MB
System File Size	16 MB

### III. The Proposed Scheme

한국컴퓨터정보학회 발전방안을 위한 관련연구로서 다양한 의견들이 제시되고 있다. 먼저, 발전방안의 기술을 살펴보면 이해그룹의 장점과 단점을 살펴보면, 이를 통하여..

### IV. Conclusions

한국컴퓨터정보학회 발전방안을 위한 관련연구로서 다양한 의견들이 제시되고 있다. 먼저, 발전방안의 기술을 살펴보면 이해그룹의 장점과 단점을 살펴보면, 이를 통하여.

### References (모두 영문으로 표기)

- [1] E. Smirni, and G. Ciardo, "Workload-Aware Load Balancing for Cluster Web Servers," IEEE Trans. on Parallel and Distributed Systems, Vol. 16, No. 3, pp. 219-232, March 2005.
- [2] kdhong, "An Efficient Dynamic Workload Balancing Strategy," Journal of The Korea Society of Computer and Information, Vol. 15, No. 1, pp. 1-10, Nov. 2010.

- [3] D.H.Ballard, "Computer Vision," Prentice-Hall, pp.76-79, 1991.
- [4] kdhong,, "C Programming Language" Korea-Press, pp.100-120, 1991.
- [5] SIMGRID Project, <http://simgrid.gforge.inria.fr>

