

# GA 기반의 성능 적응형 Job Scheduler

문용혁<sup>\*,\*\*</sup>, 서대희<sup>\*</sup>, 나재훈<sup>\*</sup>, 윤찬현<sup>\*\*</sup>

<sup>\*</sup>한국전자통신연구원, <sup>\*\*</sup>한국과학기술원

e-mail : {yhmoon, dhseo, jhnah}@etri.re.kr; chyoun@kaist.ac.kr

## A GA-based Job Scheduler for Dynamic Performance Adaptation

Yong-Hyuk Moon<sup>\*,\*\*</sup>, Dae-Hee Seo<sup>\*</sup>, Jae-Hoon Nah<sup>\*</sup>, and Chan-Hyun Youn<sup>\*\*</sup>  
<sup>\*</sup>ETRI and <sup>\*\*</sup>KAIST

### 요 약

분산 Job Scheduling 문제에서 Makespan 은 항상 타 성능지표를 대표하는 단일 목표치 (Objective) 가 되기 어려운 측면이 있다. 그러나 기존의 Job Scheduler 관련 제안들은 Makespan 만을 단일 목표치로 최적화 시킴으로써, 성능적 우수성을 입증하는 한계점이 있었다. 그러므로 본고에서는 Makespan 및 Throughput 을 동시에 최소화하여 개별 가중치로 정량화될 수 있는 다양한 성능 요구사항에 적합한 복수 대안 (Scheduling Alternatives)들을 제공할 수 있는 GA 기반 스케줄링 기법에 대해 제안한다.

### 1. 서론

최근 Grid 및 Cloud 로 대표되는 대규모 분산 컴퓨팅 환경에서 고가용도의 컴퓨팅 및 네트워크 자원을 요구하는 Job 을 분산된 자원에 효율적이고 신뢰적으로 할당할 수 있는가를 다루는 스케줄링 문제에 대한 해결하기 위한 방안의 필요성이 더욱 커지고 있다. 또한, 스케줄링 성능지표에 대한 고려가 더욱 다양화 되면서, 크게 Job (or task) 및 Resource 관점에서 관련 지표가 세분화 되고 있다.

- Job: Makespan, Throughput, Response time
- Resource: Utilization, Allocation cost

더불어 종래에는 일반적으로 Makespan (i.e., finish time of the last completed job)와 같은 대표적인 성능지표를 단일 최적화 목표(Objective)로 선정하여 스케줄링 성능의 우수성을 평가해왔다. 이에 따라, 시간과 관련된 타 지표들 (예: slowdown ratio, average response time, flow time, average throughput 등)뿐만 아니라 Resource 가용성의 대표 지표인 Utilization 의 상관관계에 대한 고려 없이 Makespan 이 우수하면 직간접적으로 타 지표들에 긍정적인 영향을 미치는 것으로 간주해버리는 지나친 단편화 문제가 있었다.

따라서, 최근 복수개의 Objectives 을 동시에 충족 (Efficient or non-dominated solution) 시켜 보다 우수한 스케줄링 성능을 보장하기 위한 제안들이 하나의 추세(MODM: Multi Objectives Decision Making)가 되고 있으며, GA, Tabu, SA 등의 Population 기반의 진화 알고리즘이나 Pareto Optimality 에 기반한 이론 등을 통해 최적화 연구가 이뤄지고 있다 [1][2].

그러므로 본고에서는 대규모 분산 컴퓨팅 환경에서 고가용도의 컴퓨팅 및 네트워크 자원을 요구하는 Job 을 분산된 자원에 효율적이고 신뢰적으로 할당할 수

있는 기법에 대해 제안한다.

### 2. Bicriteria 최적화 문제 모델링

본 장에서는 Makespan 및 Throughput 성능 모델링 및 실험을 통해 상관관계를 분석하고, 이를 기초로 최적화 모델링에 대해 논의한다.

#### 2.1. 성능지표 모델링

Makespan 은 다음과 같이 각 개별 Resource  $m$  에서의 완료시간( $ct(m)$ ) 중 최대값으로 표현된다.

$$makespan = \max \{ct(m) \mid m \in Resources\} \quad (1)$$

$$where \ ct(m) = rt(m) + \sum_{j \in Jobs, m \in Resources} etc(j)(m)$$

수식(1)에서  $rt(m)$ 은 Resource  $m$ 에서의 ready time 을 의미하며,  $etc(j)(m)$ 은 Job  $j$ 의 Resource  $m$ 에서의 개별 수행시간(execution time to complete)을 표현한다. 또한, 평균 Throughput 은 전체 Job 개수  $N$ 으로 개별 Resource  $m$ 에서의 완료시간의 합을 나눔으로써 산출된다.

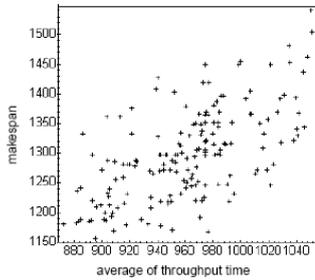
$$\overline{Throughput} = \frac{\sum_{m=1}^N ct(m)}{N} \quad (2)$$

여기서 수식(1) 및 수식(2)은 ETC matrix [2]을 기반으로 한다.

#### 2.2. 상관관계 분석

두 가지 이상의 성능지표를 동시에 고려하여 최적화하기 위해서는 먼저, 해당 성능지표 상호관계를 규명해야 한다. 이를 위해 다음 그림 1 와 같이 Makespan 및 Throughput 과의 상관관계를 Muth J.F. & Thompson G.L. 모델 [2]을 이용하여 실험적으로 확인

하였다. 이는 두 목표치가 서로 독립적(Independent Objectives)인 경우를 말하며, 동시에 Makespan 이 항상 다른 측면의 성능지표를 대표할 수 있는 단일한 목표치가 아니라는 사실을 확인시켜 준다.



(그림 1) 상관관계 결과분석: Makespan 및 Average Throughput (Random Combinational Correspondence,  $R^2 = 0.0653$ )

### 2.3. Bicriteria 최적화

단순 가중합법(A Simple Additive Weighting Method)은 각 목표치에 상대적인 가중치를 반영하여 의사 결정하는데 이용되며, Normalization 은 목표치를 최소화시키기 위한 방법론으로서 각기 다음과 같이 표현된다.

$$\max \{ \Gamma(x) \mid x \in X \} \text{ where } \Gamma(x) = \sum_{k=1}^K \omega_k z_k^T(x)$$

$$\text{and } \sum_{k=1}^K \omega_k = 1, \omega_k > 0, k = 1, 2, \dots, K. \quad (3)$$

$$n_i = \frac{z_k^{\max} - z_i}{z_k^{\max} - z_k^{\min}}$$

수식(3)에서  $\Gamma(x)$ 은 Substitute 함수를,  $\omega$  는 가중치를  $Z$  는 Objective 를 각각 의미하며,  $K$  는 최적화할 Objective 의 개수를,  $T$  는 정량화된 단일 도메인을 표현하고 있다. 단, 본 기법에서 각 Objective 는 단일한 영역으로 이미 정량화된 것으로 가정한다. 또한  $n_i$  는 개별 목표치( $Z$ )의 최대값과 최소값의 차이로 표준화시킨 값을 의미한다. 본 과정은 각 개별 목표치 간의 상호비교를 용이하게 한다.

### 3. Genetic Algorithm 을 이용한 최적화 과정

스케줄링 복수 대안들 (Alternatives) 중에서 최적 해 (Near-optimal solution)를 결정해야 하는 의사결정문제에서는 기본적으로 대안들간에 상호평가가 이뤄져야 한다. 이를 위해 유전 알고리즘 (GA: Genetic Algorithm) [2]에서는 Fitness 함수를 정의하여 주어진 초기 Population 내에서 가장 적합도가 높은 대안을 추정하는 과정을 확률적 연산자(Select, Crossover, Mutation)에 의존하여 수행한다 [2]. 전통적인 Min-Min, Min-Max, Sufferage 등의 스케줄링 알고리즘은 Local Optima 를 추적하여 Global Optima 을 추정하는 Greedy Algorithm 모델을 기반으로 하고 있는데 비해 GA 는 모집단의 다양성과 선별압력의 조화를 통해 해 수렴 속도 및 효율적 탐색을 조율할 수 있는 장점이 있다.

본 절에서는 2.1 절 및 2.3 절에서 설명한 방법론을 근거하여, 각 Individual 의 우수성을 평가하고 개별 대

안을 선별(Select)하는데 결정적인 역할을 수행하는 적합도 평가 함수를 수식 (4)와 같이 제안한다.

$$\text{fitness}(i \mid i \in I) = \omega_1 ms(i) + \omega_2 tp(i) \quad (4)$$

where  $\omega_1 \in [0, 1], \omega_2 \in [0, 1], \text{ and } \omega_1 + \omega_2 = 1$

수식 (4)에서  $ms(i)$  및  $tp(i)$ 는 각 지표의 표준화 값을 의미한다. 다음 그림 2 는 상기 제한한 각 내용을 바탕으로 GA 기반의 재구성성이 가능한 스케줄링 알고리즘을 요약한 것이다.

- |  |
|--|
| <ol style="list-style-type: none"> <li>1: Compute the ETC matrix<br/>Preference Setup::</li> <li>2: Decide <math>\omega_1</math> and <math>\omega_2</math> according to the resource site's policy<br/>GA Start::</li> <li>3: Population Initialization</li> <li>4: Evaluate the fitness values of each individuals<br/>by using equation (1), (2), and (4)</li> <li>5: Select better individuals</li> <li>6: Make offspring by crossover and mutation</li> <li>7: If the termination condition is satisfied then stop here</li> <li>8: Otherwise iterate GA procedure</li> <li>9: Make correspondence matrix between jobs and resources</li> <li>10: Scheduling according to the correspondence matrix</li> </ol> |
|--|

(그림 2) A GA-based Reconfigurable Job Scheduling Algorithm

본 제안 알고리즘에서는 두 성능 가중치  $\omega_1$  및  $\omega_2$  를 각 Resource Site 정책에 따라 해당 관리자가 지정을 하거나, Job 의 요구사항을 각 가중치 측면에서 정량화하여 표현함으로써, 서로 다른 성능지표를 동시에 최소화할 수 있는 대안 집합을 산출할 수 있다. 또한 여러 가중치를 대상으로 각기 다른 최적화 전략 (Scheduling Solution)을 획득할 수 있다.

### 4. 결론

본고에서는 성능 적응형 최적 Scheduling 조합을 찾아냄으로써, Job Scheduler 에게 재구성(Reconfiguration) 능력을 제공할 수 있는 GA 기반의 기법에 대해 제안하였다. 추가연구로서, 시뮬레이션을 통한 재구성 능력에 대한 평가가 이뤄져야 한다.

### Acknowledgement

본 연구는 지식경제부/방송통신위원회 및 한국산업기술평가관리원의 IT 신성장동력핵심기술개발사업의 일환으로 수행하였음 [2008-S-006-01, 유무선 환경의 개방형 IPTV(IPTV2.0) 기술개발].

### 참고문헌

- [1] Stanislaw Gawiejnowicz, "Time-Dependent Scheduling", Monographs in Theoretical Computer Science. An EATCS Series., published by Springer, 2008.
- [2] Fatos Xhafa and Ajith Abraham, "Metaheuristics for Scheduling in Distributed Computing Environments", Studies in Computational Intelligence., published by Springer, 2008.