

웹 서비스 합성 구성을 위한 QoS고려 서비스 선택 기법

윤 경 아[†] · 김 윤 희^{††}

요 약

웹 서비스 합성은 기존 단일 서비스들을 합성해서 제공하는 방식으로, 사용자의 다양한 요구를 만족시키기 위해서 새로운 웹 서비스를 만들 기보다는 재사용과 효율성 측면에서 고려되고 있다. 그러나 유사 기능을 제공하지만 다른 서비스 품질을 제공하는 서비스의 수가 증가함에 따라, 서비스 합성 문제는 사용자의 전역 제약 조건과 합성서비스의 QoS를 만족시키기 위해서 어떤 구성 서비스들을 선택해야 하는지에 대한 선택 문제가 되었다. 합성 서비스의 수행은 구성 서비스의 수행에 의해 결정되므로, 합성 서비스에 포함될 구성 서비스 선택을 위해서는 가격, 지속성, 응답시간과 같은 QoS에 대한 고려는 필수적이다. 본 논문에서는 합성 서비스 선택 시 QoS를 고려한 기법으로 합성 유전자 알고리즘을 적용하였다. 유전자 알고리즘에 지역 탐색 방법을 결합하여 빠른 시간 안에 합성 서비스의 전반적인 QoS뿐만 아니라 구성 서비스의 품질을 향상시킬 수 있는 서비스 선택 기법을 제시한다. 본 연구는 실제 웹상에 존재하는 실제 데이터를 이용하여 서비스 증가에 따른 시간 측정 및 최적화 정도를 비교 분석을 통해 선택 알고리즘의 유효성을 검증하였다.

키워드 : 웹서비스 합성, QoS, 웹서비스 선택, 혼합형 유전자 알고리즘

A QoS-aware Service Selection Method for Configuring Web Service Composition

Kyoungh-a Yoon[†] · Yoonhee Kim^{††}

ABSTRACT

To fulfill the complex user requirement, composition web service comprised of existing services is considered from the efficient and reusable point of view instead of making entirely new web service. However, with the growing the number of web services which provide the same functionality but differ in quality value, the service composition becomes a decision problem on which component services should be selected such that end-to-end QoS constraints by the client and overall QoS of the composition service are satisfied. QoS of service aspects is a determinant factor for selecting the services, since the performance of the composed service is determined by the performance of the involved component web service. In this paper, hybrid genetic algorithm is presented to select component services to take part in the QoS-aware composition. The local search method is used to be combined with the genetic algorithm to improve the individuals (component service) in population as well as composed service. The paper also presents a set of experiments conducted to evaluate the efficiency of selection algorithm using the real web service data.

Keywords : Web Service Composition, QoS-aware, Web Service Selection, Hybrid Genetic Algorithm

1. 서 론

최근 웹 기술의 많은 발전과 더불어 다양한 기능을 제공하는 서비스들이 웹상에 존재하지만, 사용자의 요구를 만족하기에는 저마다 제한된 기능을 제공한다. 대부분의 단일 서비스는 사용자의 복잡한 요청에 적절히 응답하기에는 한

계가 있기 때문에, 사용자가 제시한 요청을 달성하기 위해 서비스의 재사용과 효율성 측면에서 기존 다른 서비스들을 합성하여 제공할 필요가 있다.

웹 서비스 합성을 위해서는 우선, 사용자의 요구사항이 들어오면 그림 1과 같이 작은 단위의 기능으로 나누게 되는 데, 단위 서비스의 기능(task)을 제공하는 것을 추상 서비스(Abstract Service)라고 한다[1]. 이러한 추상 서비스들을 위크플로우 형태로 연결하여 합성 계획(Composition plan)이 구성되고, 계획을 바탕으로 각 추상 서비스마다 제시된 기능을 제공하는 후보 서비스(Candidate Web Service)들이 검색되고 그룹화 된다. 그룹화 된 서비스(Alternative Web

※ 본 연구는 숙명여자대학교 2011학년도 교내연구비 지원에 의해 수행되었음.

† 준 회원 : 숙명여자대학교 컴퓨터과학과 석사

†† 종신회원 : 숙명여자대학교 컴퓨터과학부 교수

논문접수 : 2012년 1월 25일

심사완료 : 2012년 5월 30일

* Corresponding Author : Yoonhee Kim(yulan@sookmyung.ac.kr)

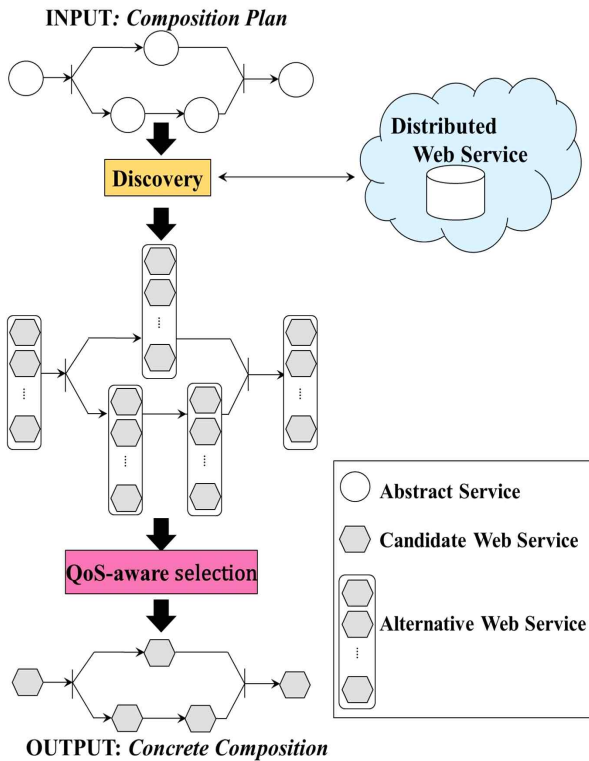


그림 1. QoS 고려 합성 서비스 선택과정
Fig. 1. A QoS-aware selection process for composition service

Service)에 포함되는 후보 서비스들은 유사 기능을 제공할 수 있지만 다른 서비스 제공자에 의해 제공된 것들이기 때문에 저마다 다른 품질을 가지고 있다. 이러한 이유로 유사 기능을 제공하는 서비스들을 구별 시 서비스의 QoS(Quality of Service)비교는 서비스 선택의 결정 요소가 된다. QoS를 고려한 서비스 선택의 목적은 각 리스트로부터 합성 서비스에 포함되는 QoS의 값이 사용자의 합성 서비스에 대한 QoS 요구조건(Constraints)을 만족시키면서, 합성 서비스의 QoS값을 최적화될 수 있도록 구성 서비스의 리스트를 찾는 것이라고 할 수 있다. 이것은 합성 서비스를 위한 후보 서비스들의 조합 최적화 문제로 볼 수 있으며, 본 논문에서는 이러한 문제를 풀기 위하여 유전자 알고리즘을 기반으로 하여 지역 최적화와 전역 최적화의 혼합 선택 방법을 적용하였다. 이 방법은 기존 지역 최적화와 전역 최적화의 장단점을 상호 보완하는 방법으로, 기존 유전자 알고리즘에 합성 서비스의 전반적인 QoS를 최대화하는 조합을 찾을 뿐만 아니라 합성 서비스에 포함되는 단일 서비스들의 품질을 보장할 수 있도록 한다. 본 논문의 구성은 다음과 같다. 2장에서는 웹 서비스 합성에서 선택방법과 관련된 연구를 소개하고, 3장에서는 본 연구가 제안하는 합성 서비스 선택 알고리즘과 전체 시스템 시나리오 및 시스템 모델, 서비스 평가 모델에 대해 소개한다. 4장에서는 연산시간 및 적합성을 분석하여 제안한 방법의 성능을 검증하고 마지막으로 5장에서는 결론을 정리하였다.

2. 관련 연구

합성 서비스의 최적화 선택 문제를 해결하기 위해 다양한 방법론들이 제시되고 있으며[2, 3, 4] 여기에서는 관련된 최적화 선택 방법에 대해서 다룬다. [4]에서는 이러한 문제를 풀기 위하여 다차원 다중 선택 배낭(multidimensional multi-choice knapsack) 문제를 사용하였고, [2, 3]에서는 웹 서비스 선택 문제에 대한 최적화 결과를 찾기 위해 선형(linear), 비선형(nonlinear) 프로그래밍 방법을 사용하였다. Ardagna와 Pernish[2]는 서비스 선택 문제를 혼합 정수 선형 계획법(mixed integer linear programming) 문제로 풀었고, [4]에서는 정수 계획법(integer programming)을 사용하였다. 그러나 위와 같은 최적화 방법들은 문제의 크기가 작을 때만 효과적이고, 문제의 크기가 커질수록 시간 복잡도가 기하급수적으로 커지기 때문에 확장성(scalability)이 좋지 않다[5, 6].

이러한 확장성 문제를 해결하기 위하여, 유전자 알고리즘(genetic algorithm, GA)과 개미집단 최적화 알고리즘(ant colony optimization algorithm, ACO)이 사용된다. [6]에서 저자는 유전자 알고리즘이 비선형 합성 함수(nonlinear aggregation functions)를 갖는 QoS속성을 다루는 데 장점이 있을 뿐만 아니라 후보 서비스들의 개수가 많아지면 다른 방법에 비해 확장성이 좋다는 것을 실험을 통해 증명하였다. 그러나 위 논문에서 사용한 순수(기본) 유전자 알고리즘(GA)은 공간 탐색능력이 뛰어나지만, 초기 점에서 최적점까지 도달하는데 많은 시간이 소요된다. 이는 유전 연산들이 확률적으로 일어나기 때문에 최적점 근처에서의 미세조정(fine tuning) 기능이 떨어지기 때문이다. 이러한 유전자 알고리즘의 결함을 극복하기 위해 향상된 유전자 알고리즘에 대한 연구(Hybrid GA, Parallel GA, Coevolution, Eco 등)가 진행되었으며, 이 중 혼합 유전알고리즘(Hybrid GA)은 진화연산을 거친 해집단에 지역 최적화 알고리즘을 적용하는 방식으로, 미미틱 유전자 알고리즘(memetic GA)이라고도 한다[7]. 본 논문에서는 합성 서비스 선택 시 선택 기법으로 혼합 유전자 알고리즘 형태로 적용하였다.

3. QoS를 고려한 서비스 선택 알고리즘

3.1 수행 시나리오

본 논문에서 제시한 선택 기법에서 합성에 포함될 수 있는 서비스들을 하나의 염색체(Chromosome)로 표현 하고, 각각의 추상 서비스를 유전자(Gene)로 표현한다[6]. 합성 서비스를 위한 최적화 수행 시나리오는 그림 2와 같으며, 여기서는 5개의 작업 수행을 위한 5개의 추상 서비스가 있다고 가정하고 설명한다. 합성을 위한 초기 계획에 따라 합성에 포함되는 추상 서비스(A)의 개수만큼 배열이 생성이 되고(A₁ ~ A₅), 각 추상 서비스의 값은 후보 서비스(Candidate Service)의 인덱스에 의해 결정된다. 그리고 서비스 합성 문제를 염색체로 표현하고 코드화(Encode)하여 초기 집단(population)을 생성한다.

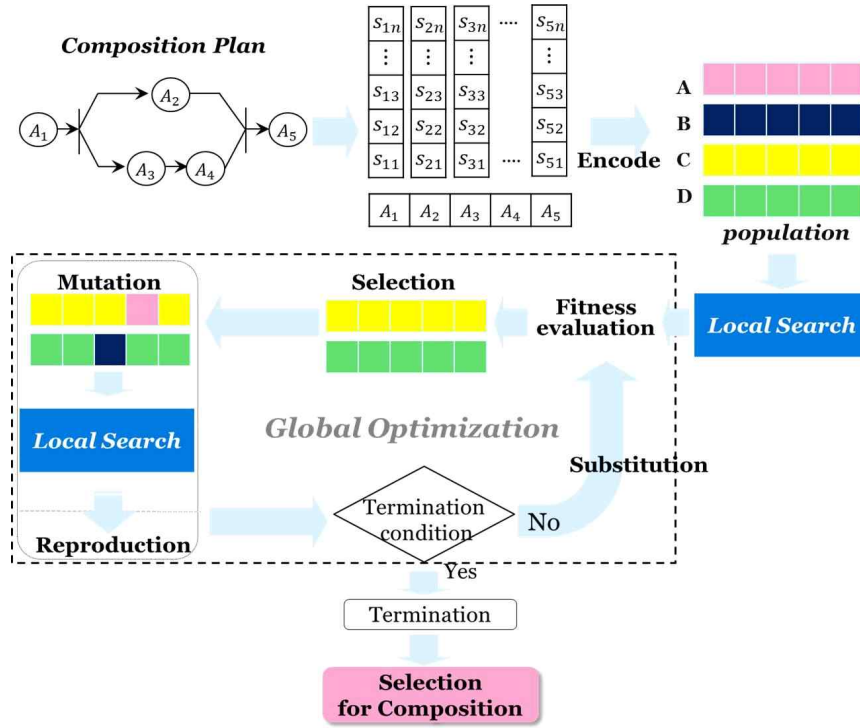


그림 2. 합성 서비스 선택 최적화를 위한 수행
Fig. 2. Optimization selection method based on genetic algorithms

초기집단에는 다양한 후보 서비스들의 조합으로 구성된 합성 서비스 계획이 포함된다. 그 다음 단계에서는 초기 집단 향상을 위해 지역 탐색(Local Search) 즉, 합성에 포함될 단위 서비스에 대한 최적화를 적용한다. 지역 탐색 수행으로 향상된 모집단을 구성하고, 종료조건이 만족될 때까지 다음 진화 과정을 반복적으로 수행 한다.

적합성 함수를 이용하여 각각의 서비스 즉 유전자 마다 적합성(Fitness)을 계산 하게 되고, 다음 교차에 참여할 염색체를 선택한다. 그리고 선택된 염색체들의 유전자를 임의로 변경(Mutation)한 후 새로운 개체군을 만들고, 생성된 개체군에 대해 전역 근사 해 근처에서 미세 조정을 위한 지역 탐색을 적용 시켜 새로운 개체(Reproduction)를 만든다. 여기서의 지역 탐색은 유전자 알고리즘 자체가 서비스 합성 레벨의 최적화(Global Optimization)는 보장하지만 서비스 레벨의 최적화는 보장하지 않는 한계를 극복하기 위해 추가된 단계로, 이를 통해 서비스 레벨에서 최적화된 리스트를 제공할 수 있다. 그리고 마지막으로 종료조건을 만족하여 최적화 진화 과정 수행이 모두 끝난 후에는 합성 계획의 적합성을 토대로 합성서비스에 포함되는 구성 서비스를 선택한다.

3.2 서비스 평가 모델

3.2.1 QoS 모델

합성 서비스의 QoS 값은 합성 서비스에 포함되는 구성 서비스 QoS 값들의 공평한 연산(fair computation)에 의해

서 결정 된다. 연산은 QoS 속성에 따라 달라질 수 있으며, QoS 속성은 부정적(negative) 속성과 긍정적(positive) 속성으로 나누어 볼 수 있다. 부정적(negative) 속성은 값이 높아질수록 품질이 낮아져 QoS에 부정적인 영향을 주게 되는 것으로, 응답시간(response time), 비용(cost) 등이 포함되며 최소화 될 필요가 있는 속성이다. 반대로, 두 번째 속성 클래스는 값이 높아질수록 품질이 좋아지는 긍정적(positive) 속성으로 신뢰도(reliability), 평판(reputation)과 같은 요소들이 포함되며 최대화 될 필요가 있는 속성이다. 본 논문에서는 4가지의 QoS 속성에 대해 다루며 부정적 속성으로 응답 시간과(Responsible time), 처리 시간(Latency) 그리고 긍정적 속성으로 가용성(Availability)과 신뢰도(Reliability)를 서비스에 대한 QoS 모델로 적용 한다. 표 1은 순차적인(Sequential) 합성에 포함되는 구성서비스 s_i 의 각 QoS에 대한 합성 함수를 나타낸다.

표 1. 구성 서비스들의 합성 함수
Table 1. Aggregation function for configuration service

QoS Attribute	Aggregation Function
Response Time	$RT = \sum_{i=1}^n RT(s_i)$
Latency	$LA = \sum_{i=1}^n LA(s_i)$
Availability	$AV = \prod_{i=1}^n AV(s_i)$
Reliability	$RE = \prod_{i=1}^n RE(s_i)$

3.2.2 지역 탐색 평가 모델

지역 탐색(Local Search)이란 서비스 수준에서 QoS를 보장하며 구성 서비스들을 찾는 것으로, 서비스 선택을 위한 평가 방법으로 단순 가중치법(Simple Additive Weighting, SAW)을 사용하였다[4, 8]. 이 방법은 웹 서비스 합성에 포함될 수 있는 각각의 구성 서비스들에 대하여 일정한 척도로 계산된 속성 값에 중요도 가중치를 곱하여 가장 큰 값을 선택하는 것으로, 이 방법을 사용하여 구성 서비스들의 우선순위를 정하도록 하였다. 우선 스케일링 과정을 통해 상반된 QoS속성을 갖는 값에 대해 표준화(normalize)가 이루어지고, 수식(1), (2)를 통해 [0, 1]사이의 값으로 표현한다. 부정적 속성의 경우 수식 (1)을, 그리고 긍정적 속성의 경우 수식 (2)를 따른다.

$$Scaling_{i,j} = \begin{cases} \frac{Q_j^{max} - Q_{i,j}}{Q_j^{max} - Q_j^{min}}, & \text{if } Q_i^{max} - Q_i^{min} \neq 0 \\ 1 & , \text{if } Q_i^{max} - Q_i^{min} = 0 \end{cases} \quad (1)$$

$$Scaling_{i,j} = \begin{cases} 1 & , \text{if } Q_i^{max} - Q_i^{min} = 0 \\ \frac{Q_{i,j} - Q_j^{min}}{Q_j^{max} - Q_j^{min}}, & \text{if } Q_i^{max} - Q_i^{min} \neq 0 \end{cases} \quad (2)$$

위 수식에서 $Scaling_{i,j}$ 은 후보 서비스 s_i 의 QoS 속성 j 번째 QoS속성을 표준화된 값으로 나타낸 것으로, 이 값은 현재 값 $Q_{i,j}$ 후보 서비스 리스트 s_i 의 j 번째 속성 최댓값 Q_j^{max} 과 최솟값 Q_j^{min} 에 의해서 결정된다. 후보 서비스 각 속성에 대한 표준화된 값을 구한 후, 수식 (3)을 통해 후보 서비스의 QoS속성들 값은 하나의 값으로 표현될 수 있다.

$$Weight(s_i) = \sum_{j=1}^4 (Scaling_{i,j} \times w_j) \quad (3)$$

여기서 $w_j \in [0,1]$, $\sum_{j=1}^4 w_j = 1$ 이고 w_j 는 사용자에게 의한 j 번째 속성의 가중치 값을 나타낸다. 각 속성에 부여된 가중치를 곱하여 모든 속성 값을 더한 후, 이 값이 최고가 되는 서비스가 선택된다. 만약 최댓값을 가지는 여러 서비스가 있을 경우, 그들 중에서 랜덤으로 하나를 선택한다.

3.2.3 전역 탐색 모델

합성 서비스의 QoS값을 최대화 하면서 전역 QoS 제약 조건에 따르는 구성 서비스들의 조합(combination)을 평가 기준으로, 유전자 알고리즘에서는 모집단에 있는 개체들을 대상으로 목적 함수(object function)를 적용하여 평가한다. 목적 함수치를 이용하여 각 개체마다 적합성 정도를 나타낼 수 있고, 높은 값을 가지는 개체(후보서비스)들은 다음 세대

에 선택되게 된다. 본 논문에서는 수식 (4)와 같은 동적 벌점함수(dynamic penalty function)을 적용하여, 제약조건에 만족하지 않는 경우 세대를 거듭할수록 벌점의 영향이 증가하도록 하였다[6]. 초기에는 벌점의 영향이 적도록 하여 불가능해가 최적해 탐색에 참여할 여지를 두었다가, 세대가 증가해갈수록 그 영향이 증가하도록 함으로써 최종적으로는 불가능한 해는 배제시킨다.

$$F(G) = \frac{w_1RT(G) + w_2LA(G)}{w_3LA(G) + w_4RE(G)} + \frac{C}{totalC} \times \frac{G}{maxG} \quad (4)$$

수식에서 속성에 대한 가중치 값($w_1 \sim w_4$)과 구성 서비스들의 QoS 합성 값을 그리고 G 는 현재 세대(current generation)를 $maxG$ 는 최대 세대 수를 의미 한다. 그리고 C 는 현재 세대에서 위반한 제약의 수를 $totalC$ 는 전체 제약의 수를 의미한다. 이 식을 통해 '불가능해로부터 얻어지는 정보의 활용'과 '불가능 해의 배제'라는 다소 상반되는 두 가지 요구 조건을 만족시킬 수 있다.

3.3 합성 서비스 선택 알고리즘

본 논문에서는 유전자 알고리즘에 지역 탐색방법을 결합하여, 합성 서비스의 전역 최적화를 보장하면서 유전자 알고리즘이 가지는 지역 최적점에서 미세조정 능력과 수렴시간이 오래 걸리는 단점들을 보완하였다. 그림 3은 서비스 선택 시 사용되는 QoS를 고려한 합성 서비스 선택 알고리즘을 나타내며, 그림 4는 그 내부에 적용된 지역 탐색 알고리즘을 나타낸다. [9]와 [10]의 논리를 적용하여 초기 향상된 모집단 ImproP(G) 구성을 위해 유전자 알고리즘 시작단계와 서비스 레벨에서의 적합성 향상을 위해 진화 연산 후 단계에서 지역 탐색(localSearch())을 적용하였다.

랜덤으로 생성한 초기 모집단을 지역 탐색알고리즘을 이용하여 향상된 모집단을 구하고, 목적함수 $F(G)$ 를 통해 적합성을 평가한다. 그리고 종료조건을 만족할 때 까지 7 ~ 13을 반복하게 된다. 여기서 평가된 적합 정도를 바탕으로 다음 세대로 넘겨줄 개체를 선택 한다. 지역 탐색 방법 적용으로 인해 수렴 속도를 가속화 하고, 이른 수렴을 할 가능성이 있는 교차 연산은 본 알고리즘에서 제외시켰으며, 선택된 개체에 돌연변이 연산 수행을 통해 개체의 다양성을 유지 하면서 탐색 공간을 넓혀 간다.

이를 통해 전역 근사 값을 찾고 그 다음에 지역 탐색을 통해 전역 근사 값 근처에서 근사값 미세조정을 하게 된다. 지역 탐색 알고리즘에서는 모집단에 있는 추상 서비스 각각에 대해서 추상 서비스에 바인딩 될 수 있는 후보 서비스 s_{ij} 를 대상으로 각 후보 서비스 마다 가중치(weight) 값을 계산한다. 만약 현재 있는 후보 서비스의 가중치가 초기 모집단에 속해있던 서비스의 가중치 보다 높을 경우 이 값을 교체한다.

```

1 G = 0 ;
2 randomly create initial population = P(G)
3 P(G).localSearch();
4 P(G)=ImproP(G);
5 evaluate (P(G=0)) using function  $F(G)$ ;
6 While isNotTerminated() do.
7     P(G) = SelectIndividuals()
8     mutate(P(G));
9     P(G).localSearch();
10    P(G=0)=ImproP(G);
11    evaluate(p(G)) using function  $F(G)$ ;
12    p(G+1)=buildNextGenetarionFrom(P(G));
13    G = G + 1;
14 end
    
```

그림 3. 합성 서비스 선택 알고리즘
Fig. 3. Selection algorithm for composition service

```

localSearch()
1 randomly generate a sequence of abstract
  web service  $A_1, A_2, \dots, A_n$  in P(G);
2 for  $A_i = A_1$  to  $A_n$  do
3   while each candidate web service  $s_{ij}$  of  $A_i$  do.
4     if G=0
5       S=Scaling( $s_{ij}$ );
6       W=Weight(S);
7     end if
8     check  $s_{ij}$  by in the decreasing order of R
9     if  $W >$  present weight value
10      replace the candidate web service;
11    end if
12  end while
13 end for
    
```

그림 4. 지역 탐색 알고리즘
Fig. 4. Local search algorithm

4. 실험 및 결과 분석

본 논문에서 제시한 QoS를 고려하여 서비스 선택을 위한 알고리즘 성능 측정을 위해, 유전자 알고리즘을 기반으로 한 서비스 선택 방법들과 비교 실험을 하였다. 비교 실험을 위한 시뮬레이션은 수치 해석 및 프로그래밍을 지원하는 공학용 소프트웨어 매틀랩(Matlab)[11] 환경에서 Genetic Algorithm Toolbox[12]를 이용하여 수행하였다. 그리고 실험 대상으로는 실제 웹상에 존재하는 서비스의 QoS 속성을 분석해 놓은 QWS(Quality Web Service)[13, 14, 15] 데이터 세트를 이용하였다. 그리고 다음과 가정을 토대로 합성 서비스 선택 알고리즘의 성능을 비교 분석 하였다.

- 합성 검색 단계에서 유사 기능을 제공하는 후보 서비스는 각 추상 서비스에 바인딩 될 수 있는 형태로 그룹화 되어 있다고 가정한다.
- 각 추상서비스 마다 대응 될 수 있는 후보서비스의 개수는 모두 같다고 가정한다.
- 복잡한 형태의 합성일지라도 [16]을 이용하면 순차적인 형태로 변경 가능할 수 있기 때문에, 순차적인 합성인 경우에 대해서만 다룬다.

실험은 세대가 지날수록 우수한 해를 갖는 2개의 개체를 다음 세대로 전달하는 방법인 엘리트 선택방법(Elitist Selection)을 사용하였고, 본 실험에서 사용한 실험 속성 값은 다음 표 2와 같다.

표 2. 실험 수행을 위한 실험 속성
Table 2. Experiment attributes conducting an experiment

실험 속성		실험 값
교차(crossover) 확률		0.8
변이(mutation) 확률		0.15
제약(constraint) 수		60
제약 위반 확률		0.3
QoS속성	응답시간(RT)	0.4
	처리시간(LA)	0.2
	가용성(AV)	0.2
	신뢰성(RE)	0.2

4.1 기본 유전자 알고리즘과의 비교

전역 최적화만을 수행하는 기본 유전자 알고리즘(기본 GA)과의 첫 번째 비교 실험에서는 추상서비스의 수를 10으로 고정하고, 각 추상 서비스에 대응할 수 있는 후보 서비스의 수를 5에서 40까지 5씩 증가시키면서 연산 시간을 측정하였다. 또한 알고리즘의 종료조건을 세대 수(100)만큼만 수행하도록 하였다.

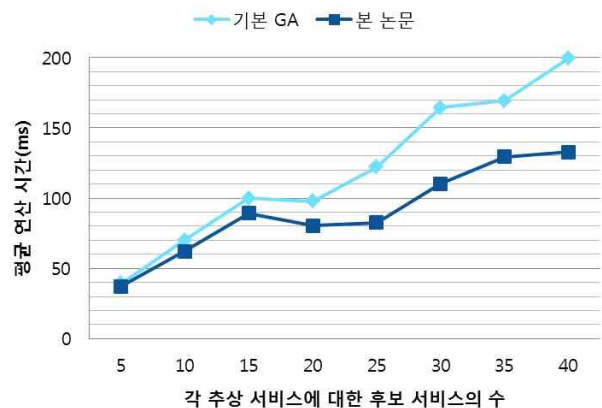


그림 5. 후보 서비스에 따른 연산시간
Fig. 5. Computational time as candidate services increases

첫 번째 비교에 대한 실험 결과는 그림 5와 같으며, 후보 서비스의 증가에 따른 평균 연산 시간을 보여준다. 기본 유전자 알고리즘(기본 GA)의 경우 전역에 걸쳐 합성될 서비스를 찾기 때문에 한 번 수행 시 오랜 연산시간을 필요로 하는데 반해, 본 논문에서는 지역 탐색 방법 적용을 통해 전역해 근처에서 높은 최적 해를 찾아 주기 때문에 기본 GA에 비해 짧은 연산 시간을 갖는다.

두 번째로 세대에 따른 적합성 평가를 위해 추상 서비스와 후보 서비스의 수를 각각 10으로 고정시키고, 적합성이 일정 세대 동안 향상되지 않을 때까지 진화연산을 반복 수행하도록 하였다. 그리고 세대를 거듭해 나감(진화)에 따라 0 ~ 450세대일 때의 적합성을 측정하였다. 그림 6은 세대에 따른 평균 적합성을 보여주며, 0세대인 경우 본 논문과 기본 GA와의 적합성의 차이가 평균 0.1정도 차이가 나는 것을 확인할 수 있다.

본 논문에서는 초기 집단에 대해 지역 탐색 적용하여 향상된 초기 집단을 구성하기 때문에 기본 유전자 알고리즘보다 높은 적합성의 결과를 얻을 수 있었다. 또한 기본적으로 유전자 알고리즘 자체가 초기집단의 상태가 그 이후의 세대의 개체에 영향을 미치기 때문에 전반적으로 기본 GA에 비해 빠른 시간에 높은 적합성을 보임을 알 수 있다.

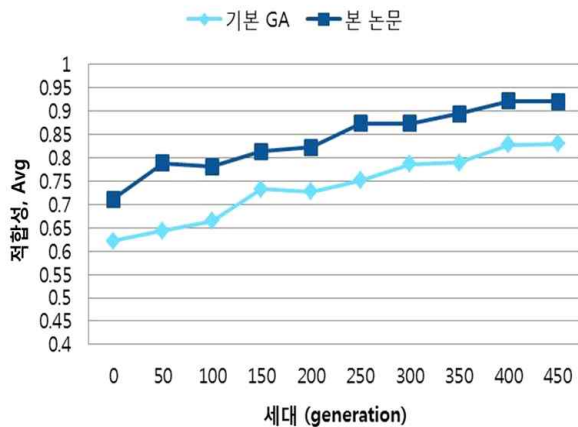


그림 6. 세대에 따른 평균 적합성
Fig. 6. Average fitness as generation increases

결과적으로 본 논문의 선택방법이 기본 GA에 비해 적은 반복 수행만으로 높은 적합성 결과를 얻을 수 있고, 빠른 시간 안에 정지조건을 만족하면서 수렴 가능성을 보인다고 할 수 있다. 따라서 기존 유전자 알고리즘에 지역 탐색 방법을 적용한 본 논문의 알고리즘이 순수 유전자 알고리즘을 사용했을 때 보다 높은 적합성과 적은 연산 시간을 가짐을 확인할 수 있었다.

4.2 다른 혼합 유전자 알고리즘과의 비교

두 번째 비교 실험에서는 본 논문과 유사한 혼합 유전자 알고리즘(Hybrid Genetic Algorithm: 하이브리드 GA)[17]과 서비스 수에 따른 연산 시간 및 적합성을 비교한다. 연산시

간 측정을 위해 추상서비스의 수를 10에서 100까지 10씩 증가 시키고, 후보 서비스를 30으로 고정하였으며, 추상서비스의 수 증가에 따른 평균 연산시간은 그림 7과 같다.

두 방법 모두 모집단에 포함되는 추상서비스의 수가 많아질수록 연산시간은 더 많이 소요되지만, 본 논문의 선택 방법이 서비스 각각에 대해 지역 최적화를 수행하는 하이브리드 GA에 비해 빠른 연산 시간을 보인다.

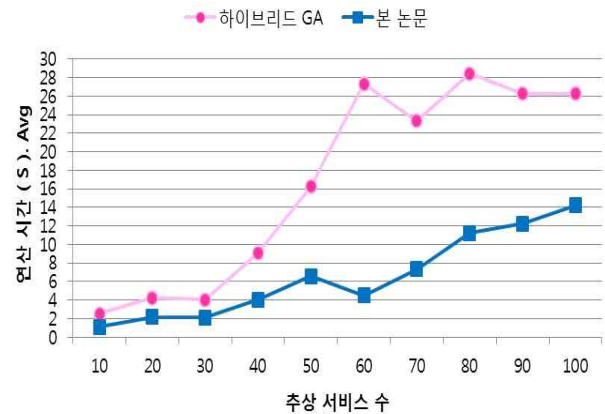


그림 7. 추상 서비스에 따른 연산시간
Fig. 7. Computational time as abstract services increases

이는 하이브리드 GA가 모든 서비스에 대해서 제약 조건 검사뿐만 아니라 적합성을 평가 및 교체하는 작업으로 많은 연산시간을 필요로 하기 때문에 단순히 지역 탐색만을 적용한 본 논문과 연산시간에서 차이를 보인다. 그리고 이때의 적합성은 다음 그림 8과 같다.

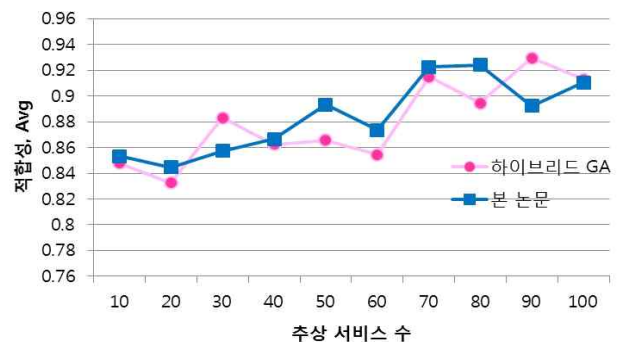


그림 8. 추상 서비스 변화에 따른 적합성
Fig. 8. Average fitness as abstract services increases

두 방법 모두 평균 0.85 정도의 높은 적합성 결과를 얻을 수 있음을 알 수 있다. 이는 사용자의 제약 조건을 만족시키면서 QoS 고려했을 때 85% 이상 적합한 합성 서비스를 찾았다는 것을 의미한다. 두 방법은 미세한 값이 차이난긴 하지만 확률적으로 수행된 결과이므로 유사한 결과를 낸다고 말할 수 있다. 결과적으로 유사 논문과 비교했을 때, 본 논문의 선택기법이 지역 탐색과정에서 연산 최소화로 인해 적은 연산시간으로 유사한 적합성을 가짐을 확인할 수 있었다.

5. 결 론

웹 서비스 합성에서 구성 서비스는 저마다 다른 QoS의 속성 값을 가지고 있으며, 합성 서비스의 QoS는 포함되는 후보 서비스의 QoS에 의해 결정 된다. 또한 합성 서비스는 사용자에게 의한 합성서비스 제약 조건을 충족 시켜야 한다. 본 논문에서는 이러한 선택 문제를 풀기 위해 QoS를 고려한 접근 방법을 통해 제약 조건을 만족하면서, 합성 서비스의 QoS 값을 최적화시키는 서비스 선택 방법을 제안 하였다. 이를 위해 전역 최적화 문제와 비선형 합성 문제에 효과적인 유전자 알고리즘에 지역 탐색 방법을 추가로 적용 하였다. 지역 탐색을 초기 모집단 구성에서의 적용과 변이 연산 후에 지역 탐색 방법을 적용하여, 합성 서비스의 레벨의 QoS 뿐만 아니라 단일 서비스 레벨에서의 QoS 향상시키고 빠른 수행 속도를 증가 시킬 수 있도록 하였다.

합성 서비스에서 지역 탐색 방법이 적용된 혼합 유전 알고리즘의 성능 분석을 위해서 기본 유전자 알고리즘과 다른 논문에서 제시한 혼합 유전자 알고리즘과의 시뮬레이션 결과를 비교하였다. 기본 유전자 알고리즘과의 비교를 통해 지역 탐색의 유무가 전체 서비스 선택의 연산 시간과 초기 적합성 및 전반적인 적합성 향상에 영향과, 다른 혼합 유전자 알고리즘과의 비교를 통해 비교한 기법 보다 빠른 시간 안에 유사하게 높은 적합성을 가짐을 확인 하였다. 본 논문의 서비스 선택 방법은 서비스 증가에 따른 연산시간의 변화가 크지 않기 때문에 기하급수적으로 많아지는 웹 서비스를 수용 가능할 것이며, 또한 정적인 합성 서비스 구성뿐만 아니라 동적 환경에서의 합성 서비스 구성에 있어서도 활용 가능할 것으로 보인다.

참 고 문 헌

[1] Mohammad A and Thomas R, "Combining global optimization with local selection for efficient QoS-aware service composition", Proceedings of the 18th international conference on World wide web, April 20-24, 2009.

[2] Ardagna D and Pernici B, "Global and local QoS guarantee in web service selection", In: Proc. of Business Process Management Workshops, pp.32 - 46, 2006.

[3] Cardellini V, Casalicchio E, Grassi V, Francesco LP, "Flow-based service selection for web service composition supporting multiple QoS classes", In: ICWS 2007. IEEE Intl. Conf. Web Services, pp.743 - 750, 2007.

[4] Zeng L, Benatallah B, Ngu AHH, Dumas, M, Kalagnanam, J. Chang, H.: "Qos-aware middleware for web services composition", IEEE Trans. Software Eng. 30, 311 - 327, 2004.

[5] Z.-J. Wang, Z.-Z. Liu, X.-F. Zhou and Y.-S. Lou, "An

approach for composite web service selection based on DGQoS", Int J Adv Manuf Technol, The International Journal of Advanced Manufacturing Technology, 56:1167-1179, 2011.

[6] Gerardo C, Penta MD, Esposito R, Villani ML "An approach for QoS-aware service composition based on genetic algorithms". In: Proceedings of the 2005 Conference on Genetic and Evolutionary Computation, pp.1069 - 1075, 2005.

[7] memetic algorithm, <http://ko.wikipedia.org/>

[8] H. C.-L and K. Yoon Multiple Criteria Decision Making. Lecture Notes in Economics and Mathematical Systems, Springer-Verlag, 1981.

[9] SeongGon Kong, InTeac Kim, DaeHee Park, ect, "An Introduction to Genetic Algorithms", Seoul: Green, 1996.

[10] Y. Ma and C. Zhang, "Quick Convergence of Genetic Algorithm for QoS-Driven Web Service Selection", Computer Networks, Vol.52, No.5, pp.1093-1104, 2008.

[11] Matlab, Web page. <http://www.mathworks.com/>

[12] Evolutionary Computation Research Team, Genetic Algorithm Toolbox, Web page. <http://www.shef.ac.uk/acse/research/ecrg/gat>

[13] E. Al-Masri and Q. H. Mahmoud. "Investigating web services on the world wide web", In Proceedings of the International World Wide Web Conference, 2008.

[14] E. Al-Masri and Q. H. Mahmoud. "Qos-based discovery and ranking of web services", In Proceedings of the IEEE International Conference on Computer Communications and Networks, 2007.

[15] E. Al-Masri and Q. H. Mahmoud. The qws dataset. Web page. <http://www.uoguelph.ca/~qmahmoud/qws/index.html/>.

[16] J. Cardoso, J. Miller, A. Sheth, and J. Arnold. "Quality of service for workflows and web service processes", Journal of Web Semantics, 1:281 - 308, 2004.

[17] Tang ML, Ai LF, "A hybrid genetic algorithm for the optimal constrained web service selection problem in web service composition", WCCI 2010 IEEE World Congress on Computational Intelligence, IEEE, pp.268-275, 30 Jul., 2010.



윤 경 아

e-mail : yoonka@sookmyung.ac.kr

2010년 제주대학교 컴퓨터공학과(학사)

2012년 숙명여자대학교 컴퓨터과학과

(석사)

관심분야: 그리드 컴퓨팅 환경, 시맨틱 웹 서비스, 웹 서비스, 지능형 시스템



김 윤 희

e-mail : yulan@sookmyung.ac.kr

1991년 숙명여자대학교 전산학과(학사)

1996년 Syracuse University 전산학과
(석사)

2000년 Syracuse University 전산학과
(박사)

1991년~1994년 한국전자통신연구소(ETRI) 연구원

2000년~2001년 Rochester Institute of Technology
컴퓨터공학과 조교수

2001년~2004년 숙명여자대학교 컴퓨터과학과 조교수

2004년~2009년 숙명여자대학교 컴퓨터과학과 부교수

2009년~현 재 숙명여자대학교 컴퓨터과학부 교수

관심분야: 그리드/클라우드 컴퓨팅, 오토노믹 컴퓨팅,
워크플로우 제어