

웹서비스 처리율 향상에 관한 연구

A Study on Throughput Elevation of Web

이성훈

백석대학교 정보통신학부

Lee Seong-hoon

Baekseok University

요약

웹서비스의 성능향상을 위해서는 서비스에 대한 추상적인 모델에 대한 설계 및 서비스 비용, 성능, 신뢰성 및 확장성 등의 서비스 품질 속성에 대한 정의 및 확장이 요구된다. 성능 측면에서 품질은 서비스 당 평균 응답 시간, 단위 시간 동안의 처리율이 포함된다. 본 연구에서는 웹 서비스 지원을 위해 단위 시간당 처리율을 개선하여 성능을 개선코자 하였다.

I. 서론

웹서비스의 보급에 따라 서비스 품질은 서비스의 사용자에게 중요한 선택 기준으로 등장하고 있으며, 웹서비스에 대한 품질에 대한 평가가 요구된다. OASIS를 중심으로 웹서비스 품질에 대한 평가를 위한 모델에 대한 표준이 진행되고 있으며 결과로서 WSQM TC가 운영되고 있다. WSDL은 웹서비스의 기능적인 정보만을 서비스 사용자에게 제공하고 있으며, 웹서비스의 성능이나 안정성에 대한 요소가 중요해짐에 따라 WSDL의 확장을 통해 서비스 명세 내에 타임아웃, 응답시간 등의 QoS에 대한 정보를 포함하고자 하는 연구가 진행되고 있다[1][2].

현재 웹서비스 표준은 네트워크 지연 및 제한된 대역폭으로 성능의 저하, 보안 공격에 따른 취약성, 웹서비스 및 응용의 결합, 네트워크 분리로 발생하는 결합에 따른 가용성 문제, 메시지 손실 및 중복 메시지에 따른 신뢰성의 제약점을 갖는다.

이를 해결하기 위해서는 서비스에 대한 추상적인 모델에 대한 설계 및 서비스 비용, 성능, 신뢰성 및 확장성 등의 서비스 품질 속성에 대한 정의 및 확장이 요구된다. 성능 측면에서 품질은 웹서비스가 얼마나 성능이

뛰어난지를 의미한다. 웹서비스 성능에는 서비스 당 평균 응답 시간, 단위 시간 동안의 처리율이 포함된다. 안정성 측면에서는 웹서비스의 이용 가능성, 접근 가능성, 신뢰성이 포함된다. 본 연구에서는 웹서비스에서의 서비스 제공을 위해 소요되는 통신비용을 감소시키고자 함을 목적으로 한다. 본 연구에서는 웹서비스 환경에서 각 시스템의 부하를 균등하게 유지하기 위한 부하분산 방법을 연구하였다.

II. 알고리즘

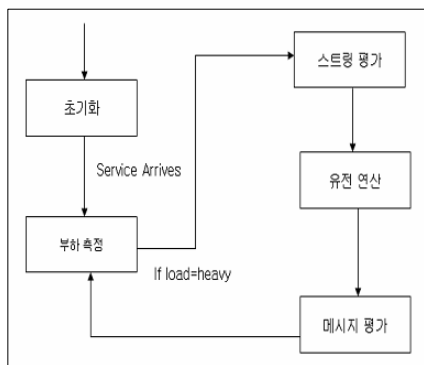
먼저 웹서비스 환경내의 프로세서들의 부하를 측정하기 위해 사용된 척도로는 큐에 들어있는 TASK 개수와 해당 TASK의 처리시간의 곱으로 정의하여 사용하였다. 다음으로 웹서비스를 제공하는 프로세서들에 대한 코딩방법으로, 웹서비스 시스템 내에 존재하는 프로세서의 개수가 n 개일 때 i 번째 프로세서는 p_i 로 표현되며 i 는 $0 \sim n-1$ 의 범위 값을 갖는다. 본 연구에서는 개체표현 방법으로 이진코딩 방법을 사용하며, 따라서 하나의 개체는 이진코드 벡터로 정의할 수 있으며 $\langle v_0, v_1, v_2, \dots, v_{n-1} \rangle$ 과 같이 표현할 수 있다. 각각의

v_i 값은 “0” 혹은 “1” 의 값을 갖는다.

웹서비스의 부하분산을 위하여 시스템 내 모든 프로세서의 부하상태를 3단계 - 저부하, 정상부하, 과부하 - 로 분리하며, 이들은 각 프로세서의 VQL로 결정된다. 3단계 부하상태를 유지하기 위해 2개의 임계값 즉, 상한(Tup) 및 하한(Tlow) 임계값을 이용하였다. 저부하 상태는 해당 프로세서의 VQL_i 값이 하한 임계값(Tlow) 보다 작은 경우이며, 과부하 상태는 VQL_i 값이 상한 임계값(Tup)보다 큰 경우에 해당된다. 그 외에는 정상부하 상태를 의미한다. 저부하 상태인 프로세서는 자신의 VQL 값이 적기 때문에 다른 과부하 상태인 프로세서가 가지고 있는 작업을 이전받을 수 있는 상태이며, 반대로 과부하 상태인 프로세서는 자신의 작업을 다른 프로세서로 이전해 줄 수 있는 상태이다. 한 집단에 포함된 각 스트링은 아래의 적합도 함수 -식 (1) -로 평가된다.

$$F_i = \left(\frac{1}{\alpha \times TMPT + \beta \times TMTT + \gamma \times TPTT} \right) \quad (1)$$

웹서비스 환경에서 부하분산은 과부하 프로세서에서의 서비스 요청에 대응하기 위해 저부하 프로세서로 서비스 요청을 이전하여 처리하는 것을 의미한다. 본 논문에서 제안하는 알고리즘은 아래 그림 1과 같이 5개의 모듈로 구성되며 이중 유전연산 모듈은 3개의 서브모듈로 구성된다. 5개 모듈은 초기화(), 부하측정(), 스트링평가(), 유전연산(), 메시지평가()이며 유전연산에서의 3개 서브모듈은 지역개선연산(), 선택(), 교배연산()이 포함된다. 이 같은 처리개요는 다음과 같다.



▶▶ 그림 1. 알고리즘 처리과정

초기화 모듈에서는 해당 집단 내 스트링들이 임의로 생성됨을 의미한다. 부하측정 모듈에서는 위에서 기술

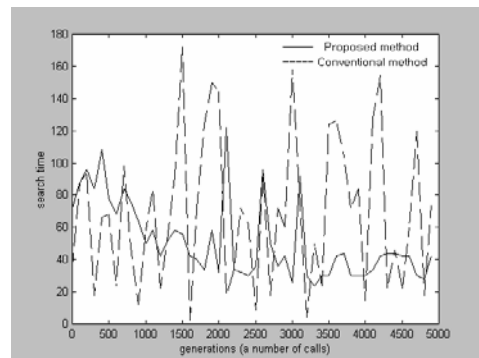
한 부하척도에 따라 해당 프로세서의 부하정도가 결정된다. 부하정도가 상한 임계값을 넘으면 다음 단계인 스트링 평가 모듈이 시행된다. 이 모듈에서는 각 스트링에 대해 식 (1)의 적합도함수에 의해 값이 결정되며 이어 유전연산 모듈이 적용되게 된다.

유전연산이 이루어진 후 각 스트링 중에서 최대값을 갖는 스트링이 선정되어 해당 프로세서에 메시지 전송이 이루어지게 된다.

III. 비교/분석

먼저, 연구에 대한 실험을 위해 다음과 같은 내용을 가정하였다. 먼저 웹서비스 환경에 놓여 있는 24개의 이질형 프로세서가 존재하며 각 프로세서로부터 다른 프로세서로의 메시지 전송 및 작업 전송시간은 2차원 배열로 표현하였다. 각 원소의 값은 난수 발생기로부터 생성되는 임의의 수로 하였다. 유전알고리즘의 적용 및 적합도 함수에 나타난 매개변수들에 대한 값은 α 는 0.5, β 는 0.15, γ 는 0.05로 하였고, 선택확률(Pc)은 0.7을, 돌연변이확률 (Pm)은 0.05, 집단 수는 30으로 하였다. 선택 및 돌연변이 확률은 많은 응용들에서 적절한 것으로 인용되고 있다[4]. 위에서 고려한 매개변수들을 고려하여 본 실험은 SUN 워크스테이션에서 이루어졌다.

다음 실험 결과는 위에서 기술된 파라메타 값을 기반으로 하여 작업 개수가 5000개일 때, 기존의 방법과 제안 알고리즘과의 비교 결과를 보인 것이다. 그림 2에서 알 수 있듯이 기존 방법은 임의의 기반임으로 인해 많은 요동을 보이는 반면에 제안된 알고리즘에서는 최적해로의 수렴성을 보이므로 요동성이 아주 미미하다.



▶▶ 그림 2. 실험 결과

IV. 결론

본 연구에서는 웹서비스에서의 자원 활용성을 높이고, 웹서비스를 제공하기 위해 필요한 협상시간을 줄임으로서 실제 서비스 제공을 위해 필요한 통신비용을 감소하기 위해 유전 알고리즘을 기반으로 하는 부하분산에 관한 문제를 다루었다. 이를 위해 필요한 구성요소들을 제시하였고, 이를 바탕으로 통신비용 감소를 위한 부하분산 알고리즘을 제시하였다.

■ 참고 문헌 ■

- [1] F. Curbera et. al., "Unraveling the Web Services Web: An Introduction to SOAP, WSDL, and UDDI," IEEE Internet Computing, Vol. 6, No. 2, pp. 86-93, March/April, 2002.
- [2] W3C, Web Services Description Language (WSDL) 1.1, 2001,
<http://www.w3c.org/TR/wsdl>.
- [3] T. Erl, Service-Oriented Architecture (SOA): Concepts, Technology, and Design, Prentice Hall, 2005.
- [4] J.Grefenstette, "Optimization of Control Parameters for Genetic Algorithms," IEEE Transactions on SMC, Vol. SMC-16, No. 1, pp. 122-128, January, 1986.