International Journal of Fuzzy Logic and Intelligent Systems, vol. 4, no. 3, December 2004 pp. 292-299

Benchmarking Evaluation of Web System Using Fuzzy Sets

Gu-Beom Jeong*, Doo-Ywan Kim, Chol-Hoi Her*****

Sangju University*, Catholic University of Daegu, Soungduk college***,**

jgb@sangju.ac.kr*, kimdy@cu.ac.kr, herch@hanmir.com***,**

ABSTRACT

For Web System composed of a variety of hardware, software and network system, there is a wide difference in price, function and capacity. In addition, it is important to select the most suitable system when web system is constructed as the claim of high-quality web service from user has increased. Data of bench-marking is usually employed when the system is selected. Evaluation with the result of bench-marking, however, is not easy at web situation composed of a variety of systems. In this paper, first integrated bench-marking of hardware, software and network system metric were determined, and the plan to effectively compare and evaluate the result of bench-marking by using fuzzy set were offered.

Key Words : Web system, Benchmarking, Functionality, Performance, Metric, Fuzzy sets

1. Introduction

Nowadays computer industry has been startlingly developed due to IT technology, and the field of computer application has been widely expanded. Web service through internet particularly yielded new business such as e-business and has become the foundation of global business, and computers owned by many governments, enterprises, university, research institute, individuals, etc. have come to share information through internet, and for our country the field of web service based on high-speed information communication network has been rapidly improved. This basis of web service is composed of a variety of client/server systems, databases, network systems, security systems, web application softwares, etc. as well as internet. Because web application system has diverse classes, much difference in price and performance as well as function, and need for high-quality web service is increasing, it is very important to select high-quality, high-reliable system[1,2].

Buyers are employing specification of system or outcome of benchmarking when purchasing the most suitable one of various systems, but it is not easy to understand information provided by enterprise, to comprise and evaluate it, and then to choose a suitable system. Particularly it is harder to perform benchmarking and to objectively evaluate the result in the environment where a variety of hardwares, softwares, and network systems are integrated. Because in this country there is no criterion of benchmarking for integrated environment, a great deal of time and effort is being spent in order to compare and evaluate analogous systems in buyer's position and in system seller's position to provide performance evaluation material requested by buyer.

For solving these problems, in this paper, metrics that can perform integrated benchmarking of hardware, software and network system is determined, and scheme that can evaluate the benchmarking result with fuzzy sets and POED (Performance Organization for Evaluation and Decision) is proposed. Evaluation method is to perform benchmarking with metrics of evaluation subject system, and to impose weight item by item according to the result. And through calculating and synthesizing fuzzy value for each item, compared and evaluated several systems.

2. Relate Research

2.1 Benchmarking

Benchmarking in a field of computer means procedure, tool, or program to measure performance of computer system, and is used as criterion mainly to compare and evaluate systems. These techniques propose exact evaluation for workload, and can be entire applied program, a part of performing program, or a comprehensive program[3-7]. Benchmarking for hardware and network mainly measures performance of system[8] and can measure six quality characteristics, functionality, reliability, usability, efficiency, maintainability and portability[9-10]. There are representative agencies for performance evaluation such as SPEC(Standard Performance Test Corporation), TPC(Transaction Processing Performance Council), VeriTest, etc in America. And there are many other enterprises for benchmarking and a variety of benchmarking programs for components of PC such as CPU, memory, board, etc. Therefore to sum up types of performance evaluation implemented for various benchmarking, they can be categorized as follows.

- Entire performance of computer system
- Performance of server system(CPU, memory, disk, video display, etc.)
- Performance capacity for a variety of application softwares
- Performance of database server or application server

Manuscript received Aug. 21, 2004; revised Dec. 6, 2004.

- Performance of network

2.2 Metrics

Performance metrics means measurement standard of benchmarking that is performed for test or evaluation of computer system, and in general is correlated with speed, accuracy, availability, etc. Examples of performance metrics for integrated benchmarking, based on types of performance evaluation, are as follows[1-8, 11-18].

- CPU: performance time, MIPS, MFLOPS, etc.
- Memory: access time, transfer rate, bandwidth, etc
- Cash memory: hit rate, access time, bandwidth, etc.
- Disk: access time, etc.
- Database: throughput, service request time, response time, etc.
- Server: throughput, response time, etc.
- Network: use rate, throughput, response time, etc.
- Stress Test: throughput, etc.

As software passes through unit test and integrated test, etc. in development process, evaluation of quality characteristics is continuously performed. Quality evaluation is mainly based on ISO/IEC 9126[9], and criterion for quality evaluation can be observed in "Standard for a Software Quality Metrics Methodology"[19]. Moreover in "Guide for Software Verification and Validation Plans"[20] standard guide for procedure of verification and validation of test result is presented. Because the quality can be reliable in case of software that pass through the procedure for evaluation and verification of this quality characteristic, in case of software benchmarking only functionality becomes the subject of evaluation.

Functionality of software specified in ISO/IEC 9126 evaluates attributes of a series of functions and features that meet obvious requirements of users, and evaluation items classify suitability, accuracy, interoperability, security, and compliance. Functional metrics for benchmarking evaluate whether or not function and capacity specified in product manual and user document are met exactly.

3. Evaluation of Integrated Benchmarking

3.1 Constitution of Integrated Benchmarking System

Because benchmarking of various web systems is accomplished compositely about hardware, software, and network systems, it is not easy to synthetically compare and evaluate function and performance only with result of benchmarking of each system. For this, evaluation method of integrated benchmarking is necessary, POED technique can be used effectively as synthetic evaluation method about benchmarking result [21-23]. POED technique, through imposing weight to small item composed of metrics according to degree of importance, and applying weight to benchmarking result made score, can automatically achieve totalization of big items categorized with functionality and performance characteristics, and as the result, can evaluate synthetically web system.

Proposed benchmarking system is based on POED technique, and applies fuzzy during evaluation. For this, first impose weight to evaluation of proposed system, and fuzzier the evaluation result benchmarked according to each item. Finally, evaluation system synthesizes weight and benchmarking result, and selects the most suitable system and proposes it to user after mplement of fuzzy operation.

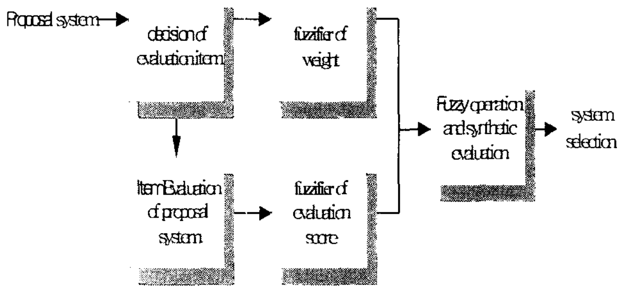


Figure 1. Architecture of benchmarking evaluation system

3.2 Measure Item and Standard of Evaluation Model for Integrated Benchmark

When constructing computer system, relative importance of hardware and network in entire system, due to expensive hardware, was big in past, but through continuous development of technology, the price of hardware and network, as compared with capacity, has lowered, and relative importance of software has comparatively increased. According to this, the importance of function evaluation of software as well as that of hardware or network, during capacity evaluation of system, has increased. Therefore, the evaluation module for this integrated benchmark is defined as table 1.

In table 1, the items of CPU, memory, cash memory, disk, back-up system, server, network system, database system and stress test are to set metrics about hardware, and suitability, accuracy, mutual operationality, security, and observance are to set metrics about software.

3.3 Performance of benchmarking and fuzzy number

Before performing benchmarking with evaluation items of table 1, first set up weight according to importance of evaluation items. Weight is imposed with interval value between 0 and 1 and can be set up according to system buyer's subjectivity or user's. At this time, high weigh is imposed to important item and low weight to low importance. For fuzzy operation for integrated benchmarking, map the evaluation result of each item into interval value between 0 and 1, language variable. In addition, set less than sup into 0 and more than inf into 1.

Evaluation of benchmark results with each item may be interpreted according to expert's opinion or test environment. Therefore, through performing objective evaluation by reducing subjective test result to a minimum, method to select the most suitable system is necessary.

The result value of evaluation of each item can be shown as follows.

$$U_{\mathcal{F}_i} = \{\mathcal{T}_1, \mathcal{T}_2, \cdots, \mathcal{T}_{55}\} \quad (1)$$

where, $\mathcal{T}_1, \mathcal{T}_2, \cdots, \mathcal{T}_{55}$ means result value of each item analyzed according to several test environments. Besides standard value about each item suitable to buyer is given as follows.

$$U_{\mathcal{S}_i} = \{\mathcal{S}_1, \mathcal{S}_2, \cdots, \mathcal{S}_{55}\} \quad (2)$$

Sup and inf about standard value is defined as follows, each standard value is set up with intermediate value between 0 and 1, and intermediateness of sup and inf is shown as fuzzy interval value between 0 and 1. And if the value is more than sup, it is, considered value out of boundary value, set up with 1, and if less than inf, 0.

Table 1. Measure item and standard for integrated benchmarking

class	item no	.measure item	metrics
CPU	1	*MIPS*	$\frac{instruction\ count}{CPU\ execution\ time \times 10^6}$
	2	*MFLOPS*	$\frac{number\ of\ floating-point\ operations\ in\ a\ program}{execution\ time \times 10^6}$
	3	*throughput*	*CPU clock cycles × clock cycle time*
	4	*price ratio*	*CPU price / MIPS or CPU price / MFLOPS*
Memory	5	*size ratio*	*proposed memory size / required memory size*
	6	*access time ratio*	*proposed access time / required access time*
	7	*transfer ratio*	*1 / memory cycle time*
	8	*bandwidth*	*data bus size*(byte) × *memory cycle time*(MHz)
	9	*price ratio*	*memory price / memory size*
Cash Memory	10	*size ratio*	*proposed cash size / required cash size*
	11	*hit ratio*	*page in cache / total page × 100*
	12	*access time ratio*	*proposed access time / required access time*
	13	*bandwidth*	*number of request per second × hit rate × average size of request data*
	14	*price ratio*	*cash memory price / cash memory size*
DISK	15	*size ratio*	*proposed disk size / required disk size*
	16	*access time*	*seek time + rotational latency time + transfer time*
	17	*price ratio*	*disk price / disk size*
Back-Up System	18	*backup ratio*	*back up data size / back up time*
	19	*recovery ratio*	*recovery data size / recovery time*
	20	*function*	*number of function*
	21	*capacity ratio*	*backup capacity / data total amount*
SEVER	22	*throughput*	*completed request number / unit time*
	23	*response time*	*latency(server) + communication time + processing time(client)*
	24	*price ratio*	*proposed price / required price*
Network System	25	*use ratio(* ρ *)*	$\rho = \frac{\lambda}{\mu C}$ (*messages*/ sec)
	26	*throughput(req/sec)*	*concurrent users × {mean response time(sec) + think time(sec)}*
	27	*response time*	$\frac{1/\mu}{1-\rho}$
	28	*service time*	$\frac{(message\ size + overhead_n \times 8)}{10^6 \times bandwidth_n}$

Table 1. Measure item and standard for integrated benchmarking(continue)

class	item no.	measure item	metrics
Network System	29	*degree of use*	$\sum_{messages\ j} \lambda_j \times service\ time\ \frac{j}{n}$
	30	*concurrent users*	*throughput* × {*response time* + *think time*}
	31	*price ratio*	*proposed price / required price*
Database System	32	*throughput*	*number of request / measure time(sec)*
	33	*service request time*	*total busy time of DB server*(sec) / *total search number*
	34	*degree of use*	*throughput* × *service request time*
	35	*response time*	$\frac{1/\mu}{1-\rho}$
	36	*price ratio*	*proposed price / required price*
stress test	37	*throughput*	*number of request / measure time(sec)*
suitability	38	*function information*	*function number offering in system / function number presented in document*
	39	*data information*	$\frac{embeded\ data\ type\ number\ offering\ in\ system}{embeded\ data\ type\ number\ presented\ in\ document}$
	40	*use environment manual*	*offer of an explanation of use environment or not*
	41	*completeness of function realization*	*number of function realized in system / number of function described in user document*
	42	*sufficiency of function*	*number of function realized in system / number of common function*
	43	*suitability of function*	*number of function realized in system / number of function in document*
	44	*boundary value*	*number of realized boundary item / number of boundary item in document*
	45	*boundary throughput*	$\frac{number\ of\ testcase\ success\ of\ each\ boundary\ value}{number\ of\ em\ of\ boundary\ value\ confirmation}$
accuracy	46	*accuracy of function realization*	*number of successful test-case by function / function number of evaluation item*
	47	*offer of accurate information of function realization*	*number of accurately described function / number of total function*
	48	*definiteness of function classification*	*number of definitely classified function / number of total function*
inter-operability	49	*connection possibility*	*number of possibility of real connection / number of connectable evaluation item*
	50	*degree of data exchange*	*number of data function presented in document / number of entire data exchange*
	51	*data convertibility*	*number of successful test-case by function / number of entire data exchange*
security	52	*offer of access control information*	*offer of information of access control or not*
	53	*access control possibility*	*number of successful test-case by function / number of total function for access control*
compliance	54	*offer function standard compliance*	*offer of function standard compliance information or not*
	55	*function standard compliance ratio*	*number of successful test-case by function / number of function standard item to evaluate*

$$INFs_i = \{b \mid \tilde{T}_i \geq b\} \quad (3)$$

$$I_{Lsize} = \frac{\tilde{s}_i - infs_i}{p} \quad (4)$$

$$I_{Usize} = \frac{sups_i - \tilde{s}_i}{p} \quad (5)$$

where I_{Lsize} is the interval range which is less than standard value, I_{Usize} is he interval range which is more than standard value, and p means interval number.

Therefore each membership function is defined as follows.

$$I_i = \{0.0/I_0, \cdots, 0.5/I_5, \cdots, 1.0/I_{10}\} \quad (6)$$

Correspond to formula (6) the result value of evaluation of each item and show them as fuzzy number. And for weight $\tilde{W}_i$, buyer impose suitable weight according to importance of evaluation item.

To show as fuzzy number the above result value $\tilde{F}_i$ and weight for each item on table 1 is table 2.

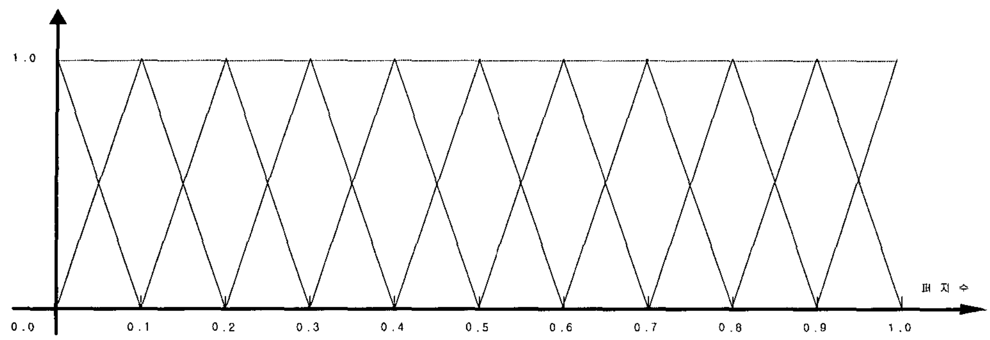


Figure 2. Interval value of membership function

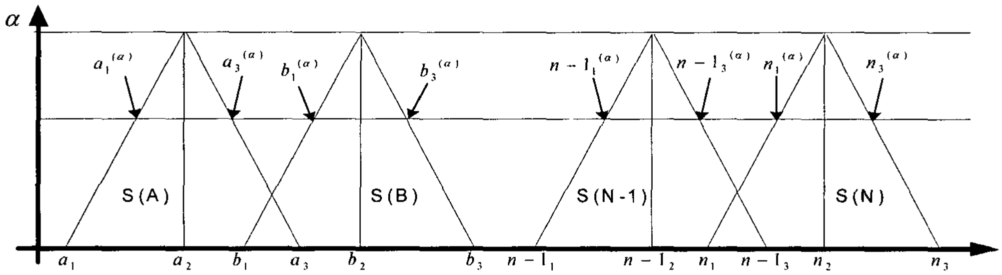


Figure 3. S(·) implementing $\alpha-cut$

Table 2. Weight and fuzzy number for each item

no.	item	system A	system B	⋯	system N
1	$\tilde{W}_1$	$\tilde{F}_{1A}$	$\tilde{F}_{1B}$	⋯	$\tilde{F}_{1N}$
2	$\tilde{W}_2$	$\tilde{F}_{2A}$	$\tilde{F}_{2B}$	⋯	$\tilde{F}_{2N}$
3	$\tilde{W}_3$	$\tilde{F}_{3A}$	$\tilde{F}_{3B}$	⋯	$\tilde{F}_{3B}$
⋮	⋮	⋮	⋮	⋯	⋮
55	$\tilde{W}_{55}$	$\tilde{F}_{55A}$	$\tilde{F}_{55B}$	⋯	$\tilde{F}_{55N}$

Entire fuzzy numbers of each system is shown as $S(\cdot)$.

$$S(A) = \tilde{W}_1 \otimes \tilde{F}_{1A} \oplus \tilde{W}_2 \otimes \tilde{F}_{2A} \oplus \cdots \oplus \tilde{W}_{55} \otimes \tilde{F}_{55A} \tag{7}$$

$$S(B) = \tilde{W}_1 \otimes \tilde{F}_{1B} \oplus \tilde{W}_2 \otimes \tilde{F}_{2B} \oplus \cdots \oplus \tilde{W}_{55} \otimes \tilde{F}_{55B} \tag{8}$$

$$\vdots$$

$$S(N) = \tilde{W}_1 \otimes \tilde{F}_{1N} \oplus \tilde{W}_2 \otimes \tilde{F}_{2N} \oplus \cdots \oplus \tilde{W}_{55} \otimes \tilde{F}_{55N} \tag{9}$$

where $S(A)$, $S(B)$, …, $S(N)$ is triangular fuzzy number.

The degree of suitability is $D_\lambda(A)$, $D_\lambda(B)$ …, $D_\lambda(N)$ and λ is the degree of suitability of system. Low value of λ means that the degree of suitability is high.

Figure 3, $S(\cdot)$ having performed $\alpha-cut$.

$$a_1^{(\alpha)} = \alpha(a_2 - a_1) + a_1 \tag{10}$$

$$a_3^{(\alpha)} = -\alpha(a_3 - a_2) + a_3 \tag{11}$$

$$b_1^{(\alpha)} = \alpha(b_2 - b_1) + b_1 \tag{12}$$

$$b_3^{(\alpha)} = -\alpha(b_3 - b_2) + b_3 \tag{13}$$

$$\vdots$$

$$n_1^{(\alpha)} = \alpha(n_2 - n_1) + n_1 \tag{14}$$

$$n_3^{(\alpha)} = -\alpha(n_3 - n_2) + n_3 \tag{15}$$

Where if $\alpha-cut$ is applied in order to get crisp value, $\alpha-cut$ of $S(A)$, $S(B)$, …, $S(N)$ can be shown as $[a_1^{(\alpha)}, a_3^{(\alpha)}], [b_1^{(\alpha)}, b_3^{(\alpha)}], \ldots, [n_1^{(\alpha)}, n_3^{(\alpha)}]\ \alpha \in [0, 1]$, and is as Figure 3.

Result of evaluation by each evaluation item can be pessimistically evaluated about certain system or optimistically by user's subjective weight. That is to say, because evaluation result of each system may be drawn by subjective, qualitative weight of user, it can be considered that uncertainty may be contained. Hence, we can evaluate as making change of number of suitability $\lambda (\lambda \in [0, 1])$ according to score $D_\alpha^\lambda(A)$, $D_\alpha^\lambda(B)$ setting the value of $\alpha-cut$ for each item. λ means the degree of 'optimism' of system evaluation. Small value of λ means to be high in the degree of 'optimism', high value of λ is to be low in the degree of 'optimism', and the value of fuzzy number for each system can be defined as formula (14), (15).

$$D_\alpha^\lambda(A) = \lambda a_1^{(\alpha)} + (1-\lambda) a_3^{(\alpha)} = P_A \tag{16}$$

$$D_\alpha^\lambda(B) = \lambda b_1^{(\alpha)} + (1-\lambda) b_3^{(\alpha)} = P_B \tag{17}$$

$$\vdots$$

$$D_\alpha^\lambda(N) = \lambda n_1^{(\alpha)} + (1-\lambda) n_3^{(\alpha)} = P_N \tag{18}$$

Evaluation score from each system for item shows that high occupancy ratio of a system for whole evaluation score is relatively superior, and it is shown as formular (19), (20), (21).

$$N_\lambda^\alpha(A) = \frac{P_A}{P_A + P_B + \cdots + P_N} \quad (19)$$

$$N_\lambda^\alpha(B) = \frac{P_B}{P_A + P_B + \cdots + P_N} \quad (20)$$

$$\vdots$$

$$N_\lambda^\alpha(N) = \frac{P_N}{P_A + P_B + \cdots + P_N} \quad (21)$$

Values of $N_\lambda^\alpha(A)$, $N_\lambda^\alpha(B)$, ..., $N_\lambda^\alpha(N)$) means the degree of suitability to select system, and biggest value means the most suitable system to buyer. Where $N_\lambda^\alpha(A)$, $N_\lambda^\alpha(B)$, ..., $N_\lambda^\alpha(N) \in [0, 1]$

4. System Evaluation and Result Analysis

Set up fuzzy number of each system about benchmarking result measured by each item and weight for each item in order to select the most suitable system to buyer among benchmarked systems. Here benchmarked result value and weight is shown as triangular fuzzy number. Weight of each item set up by fuzzy number and buyer is shown in table 3. And based on table 3, evaluation value of system can be obtained.

$$\begin{aligned} S(A) &= \tilde{W}_1 \otimes \tilde{r}_{1A} \oplus \tilde{W}_2 \otimes \tilde{r}_{2A} \oplus \cdots \\ &\quad \oplus \tilde{W}_{51} \otimes \tilde{r}_{51A} \oplus \tilde{W}_{55} \otimes \tilde{r}_{55A} \\ &= 0.8 \otimes 0.7 \oplus 0.9 \otimes 0.7 \oplus \cdots \\ &\quad \oplus 0.8 \otimes 0.8 \oplus 0.8 \otimes 0.9 \\ &= (0.7, 0.8, 0.9) \otimes (0.6, 0.7, 0.8) \\ &\quad \oplus (0.8, 0.9, 1.0) \otimes (0.6, 0.7, 0.8) \\ &\quad \oplus \cdots \\ &\quad \oplus (0.9, 1.0, 1.0) \otimes (0.6, 0.7, 0.8) \\ &\quad \oplus (0.6, 0.7, 0.8) \otimes (0.6, 0.7, 0.8) \qquad (22) \\ &= (26.82, 29.73, 33.60) \end{aligned}$$

$$\begin{aligned} S(B) &= \tilde{W}_1 \otimes \tilde{r}_{1B} \oplus \tilde{W}_2 \otimes \tilde{r}_{2B} \oplus \cdots \\ &\quad \oplus \tilde{W}_{54} \otimes \tilde{r}_{54B} \oplus \tilde{W}_{55} \otimes \tilde{r}_{55B} \\ &= 0.7 \otimes 0.7 \oplus 0.7 \otimes 0.7 \oplus \cdots \\ &\quad \oplus 1.0 \otimes 0.7 \oplus 0.8 \otimes 0.7 \\ &= (0.6, 0.7, 0.8) \otimes (0.6, 0.7, 0.8) \\ &\quad \oplus (0.6, 0.7, 0.8) \otimes (0.6, 0.7, 0.8) \qquad (23) \\ &\quad \oplus \cdots \\ &\quad \oplus (0.9, 1.0, 1.0) \otimes (0.6, 0.7, 0.8) \\ &\quad \oplus (0.7, 0.8, 0.9) \otimes (0.6, 0.7, 0.8) \\ &= (26.55, 30.46, 34.37) \end{aligned}$$

Table 3. fuzzy number of each system about benchmarking result and weight by each item

no.	weight	system A	system B	no.	weight	system A	system B
1	0.7	0.8	0.7	29	0.5	0.8	0.7
2	0.7	0.9	0.7	30	0.5	0.9	0.7
3	0.7	0.9	0.7	31	0.8	0.7	0.9
4	0.8	0.7	0.8	32	0.6	0.8	0.8
5	0.9	0.7	0.8	33	0.6	0.7	0.9
6	0.9	0.7	0.8	34	0.6	0.7	0.9
7	0.9	0.7	0.8	35	0.6	0.7	0.8
8	0.9	0.7	0.8	36	0.8	0.8	0.6
9	0.8	0.8	0.6	37	0.9	0.8	0.7
10	0.8	0.7	0.8	38	0.8	0.8	0.9
11	0.8	0.8	0.9	39	0.8	0.7	0.8
12	0.8	0.7	0.8	40	0.8	1.0	1.0
13	0.8	0.7	0.8	41	0.8	0.7	0.8
14	0.8	0.7	0.6	42	0.8	0.7	0.9
15	0.6	0.7	0.7	43	0.8	0.7	0.9
16	0.6	0.8	0.7	44	0.8	0.7	0.8
17	0.8	0.7	0.6	45	0.8	0.7	0.8
18	0.6	0.7	0.7	46	0.9	0.7	0.8
19	0.6	0.7	0.7	47	0.9	0.7	0.8
20	0.6	0.7	0.7	48	0.9	0.7	0.8
21	0.6	0.7	0.7	49	0.6	0.8	0.7
22	0.7	0.8	0.9	50	0.6	0.9	0.8
23	0.7	0.9	0.9	51	0.6	0.8	0.7
24	0.8	0.8	0.6	52	0.5	1.0	1.0
25	0.5	0.8	0.7	53	0.5	0.9	0.8
26	0.5	0.8	0.7	54	0.7	1.0	1.0
27	0.5	0.7	0.7	55	0.7	0.7	0.8
28	0.5	0.7	0.7				

In general, the value of $D_\alpha^\lambda(A)$, $D_\alpha^\lambda(B)$ when $\lambda = 0.5$ can be considered that weight set by user is evaluated appropriately, not optimistically nor pessimistically. Each evaluation value is $D_\alpha^\lambda(A) = 49.39$, $D_\alpha^\lambda(B) = 50.60$.

Evaluation score by each system based on value of $D_\alpha^\lambda(A)$, $D_\alpha^\lambda(B)$ show evaluation score ratio of each system to entire evaluation score. And when user's evaluation is very optimistic($\lambda = 0.1$), it is shown as table 4, and when the evaluation is very pessimist c($\lambda = 1.0$), it is shown as table 5.

Table. 4 When $\lambda = 0.1$, evaluation ratio

α \ system	system A	system B
0.0	0.4945	0.5055
0.1	0.4945	0.5055
0.2	0.4944	0.5056
0.3	0.4943	0.5057
0.4	0.4943	0.5057
0.5	0.4942	0.5058
0.6	0.4942	0.5058
0.7	0.4941	0.5059
0.8	0.4941	0.5059
0.9	0.4940	0.5060
1.0	0.4939	0.5061

Table 5. When $\lambda = 1.0$, evaluation ratio

α \ system	system A	system B
0.0	0.4930	0.5070
0.1	0.4931	0.5069
0.2	0.4932	0.5068
0.3	0.4933	0.5067
0.4	0.4934	0.5066
0.5	0.4935	0.5065
0.6	0.4936	0.5064
0.7	0.4937	0.5063
0.8	0.4938	0.5062
0.9	0.4939	0.5061
1.0	0.4939	0.5061

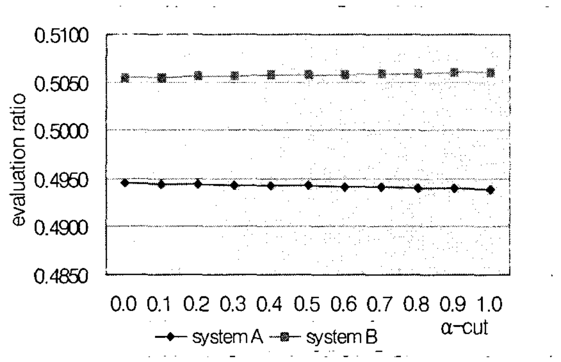


Figure 4. When is $\lambda = 0.1$, Evaluation ratio of system A and system B

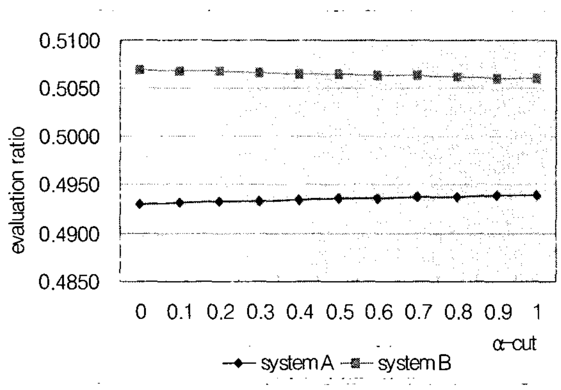


Figure 5. When is $\lambda = 1.0$, Evaluation ratio of system A and system B

In comparison of figure 5 and 6, it can be known that a suitable one of both systems is system B.

5. Conclusion

Benchmarking is program to measure function and performance of computer system, and is mainly being used as standard for comparison and evaluation between systems. Web system in which good quality of web service is increasingly required is composed of a various hardware, software and network system, and for selecting optimal system, function and performance of each system is evaluated. For this, benchmarking is employed. And relatively objective, exact evaluation material can be proposed. But for performing benchmarking effectively, first metrics should be defined, function and performance measurement should be achieved on the basis of this, and for decision making on benchmarking result synthetic evaluation method should be proposed.

This paper, for synthetically evaluating benchmarking, decided performance metrics of hardware, network and database system and functional metrics of software. Result value of benchmarking was mapped into value of linguistic variable through fuzzy set, and shown as fuzzy number by triangular fuzzy number. And through, according to importance of evaluation item, converting weight imposed by each metrics into fuzzy number and implementing result value of benchmarking and fuzzy operation, score was made. This totaled score can be used as material about benchmarking result for decision making. The result of proposed evaluation method for benchmark allows system buyer to easily use the result of benchmarking and to easily compare function and performance between systems through estimating system suitability with $\alpha - cut$..

6. Reference

[1] M. C. Paulk, B. Curtis, M. B. Chrissis, C. V. Beber, "Capability Maturity Model for Software Version 1.1," *Software Engineering Institude,* CMU/SEI-93 -TR-24, 1993, 2.

[2] 황선명, 강성기, "소프트웨어 품질 평가 메트릭스 분석," 대전대학교 산업정보연구소 연구논문, 제10권 2호, pp. 3-9, 1999.

[3] 김창민, "CMFstone:유닉스 기반한 합성 사용자 프로그래밍 벤치마크," 자연과학연구, 성결대학교 자연과학연구소, 제1집, pp. 17~39, 1996,

[4] L. Press, "Benchmarks for LAN Performance Evaluation," *Communications of the ACM,* Vol. 31, No. 8, pp. 1014-1017, 1988.

[5] R. H. Saavedra and A. J. Smith, "Analysis of Benchmark Characteristics and Benchmark Performance Prediction," *ACM Transactions on Computer systems,* Vol. 14, No. 4, pp. 344-384, November 1996.

[6] C. Williamson, R. Simmonds and M. Arlitt, "A case study of Web server benchmarking using parallel WAN emulation," *Performance Evaluation*, Vol. 49, pp. 111-127, 2002.

[7] R. eigenmann, Performance Evaluation and Benchmarking With realistic Applications, Massachusetts Institute of Technology, 2001.

[8] R. Jain, The Art of Computer Systems Performance Analysis, John Wiley & Sons, INC., 1991.

[9] ISO/IEC 9126, Information Technology-Software Quality Characteristics and metrics

[10] 양해술, "Embedded S/W의 품질평가 모델 개발 연구", TTA, 2002.

[11] SPEC, http://www.spec.org

[12] TPC, http://www.tpc.org

[13] VeriTest, http://www.veritest.com

[14] R. C. Wilson, Unix Test Tools and Benchmarks, Prentice Hall Inc., 1995.

[15] R. W. Hockney, The Science of Computer Benchmarking" Society for Industrial and applied Mathematics, 1996.

[16] J. P. Hayes, Computer Architecture and Organization, McGraw-Hill, 1978.

[17] W. SU. Stanley, Database Computers, McGraw- Hill, 1988.

[18] M. Schwartz, Telecommunication Networks - Protocols, Modeling and Analysis, Addison Wesley, 1987.

[19] TTA, "소프트웨어 검증 및 확인 계획서 지침," 정보통신단체표준 TTAS.IE- 1059, 2001.

[20] TTA, "소프트웨어 품질 메트릭 방법론 표준," 정보통신단체표준 TTA.IE-1061, 1999.

[21] 정보통신부, "소프트웨어 품질의 측정, 평점 및 심사를 위한 기술지원서," 한국전산망표준 KICS.IS-8402, 1994.

[22] 김두완, 허철회, 정구범, 정은영, 김재웅, "퍼지집합을 이용한 인터넷 응용 시스템 평가," 한국인터넷정보학회 추계학술발표대회 논문집, 제4권 2호, pp. 549-552, 2003.

[23] 김국보, "컴퓨터통합생산 시스템 구축을 위한 제안 시스템의 적합성 평가," 대구가톨릭대학교 대학원 박사학위 논문, 1997.

[24] M.M. Gupta, T. Yamakawa, Fuzzy Computing, Elsevier Science Pub., pp.500, 1988.

[25] 이광형, 오길록, 퍼지이론 및 응용(II권:응용), 홍릉과학출판사, 1992.

[26] S. M. Chen, "A new approach to handing fuzzy decision marking problems," *IEEE Trans. Syst.*, Man, Cybern., Vol. 18, No 6, pp.1012-1016, 198

Gu-Beom Jeong

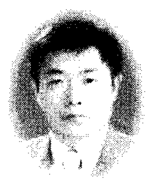
He is an associate professor at the department of computer science in Sangju National University, at SangJu in Korea since 1997. He received Ph.D degree form the Catholic University of Daegu 1999. His major research fields are artificial intelligent system, Fuzzy, Neural Network and rough set.

Phone : +82-54-530-5356

Doo-Ywan Kim

He received the B.S., Kyungnam University, Korea, in 1997, and M.S. Catholic University of Daegu Korea, in 1997. He received Ph.D degree form the Catholic University of Daegu 2003. He is an associate IT professor at the department of Faculty of Computer & Information Communication Engineering in Catholic University of Daegu, in Korea since 2004. His research interests are in MVL, Fuzzy, Neural Network, Rough set.

Phone : +82-53-580-2741



Chul Whoi Her

He was born October 3, 1958. He received the B.S. degree in Computer Science from Kwangwoon University, in Seoul, Korea, in 1984. and the M.S. degree in Computer Science from Myongji University, in Youngin, Korea, in 1987.

He received Ph.D degree form the Catholic University of Daegu 2002. He has been associated with Myongji University Computer Center from 1981 to 1997, in Youngin, Korea. Since 1997, he has been a faculty member of the Division of Computer Information at the SungDuk College, where he is currently a full-time lecture. His research interest are Artificial Intelligence, Agent, Fuzzy theory, Neural Network and Application. He is a member of KFIS, KISS, KIPS and CASE/EC.

Phone : +82-54-330-8731,
Fax : +82-54-330-8731