Front. Electr. Electron. Eng. China (2006) 4: 480–486
DOI 10.1007/s11460-006-0092-2

RESEARCH ARTICLE

LI Jie, WANG Ru-chuan, BIAN Zheng-ai

Mobile intelligent agent entity model towards QoS guarantee

Abstract Implementing a flexible configuration of the QoS parameter in a distributed computing network has become a problem due to the weak scalability of current approaches. In an effort to solve this problem, an inner basic model of an intelligent agent (IA) is presented. The IA functionality was extended by introducing a primarily mobile agent. A QoS guarantee scheme was subsequently designed and implemented based on the model as well. By utilizing the proposed scheme, the IA can sense, predict and configure the data flow traffic. Since the communicating ability was considered and provided, the competition among different devices could be eliminated effectively and the global traffic can be optimized. The results of the simulations have shown that the proposed model can provide a QoS guarantee.

Keywords intelligent agent, QoS

1 Introduction

Nowadays, the study of QoS has become one of the most important fields in the next generation of network research. A great prospect in the application is found in the research of intelligent agents because of its intelligence and mobility [1]. How to achieve the QoS using a mobile agent is the latest subject area in network research [2]. Currently, some researches on the management of an agent-based QoS are performed in the field, such as in Ref. [3], and a scheme in which the combination of the RSVP and the agent is presented, but using the scheme is not optimal for the co-natural expansibility in the RSVP. In Ref. [4], a kind of mobile agent for the configuration of the DiffServ is presented, and based on this scheme, the process of configuring the DiffServ is greatly simplified and the efficiency of the configuration is enhanced. In this scheme, the mobility of the intelligent agent is only used as a simple tool for configuration, which perfects the existing means for the configuration in the networks. In Ref. [5], a scheme on the crucial measurements of the QoS and the sensor for the performance applying the agent is presented. In the agent system, the preliminary assignment in the task is analyzed and the agent is classified into the application layer, the network layer and the link layer with a resource communication in every layer. In Ref. [6], the platform for the agent with the transparent support of the QoS is presented, where the agent is used as one middleware. The agent possesses both negotiating function and sensor function, but in these schemes, the agent is simply a kind of tool for negotiation and measurement and is not used to predicate the characteristics of the distribution in flux. In addition, in Ref. [7], the management with the synchronization in multimedia applying the mobile agent is presented. This scheme offers the negotiation function of the resource utilizing the KQML, and also the sensor function on management is presented utilizing the mobility of the agent on the server for multimedia. However, it is limited by the fact that there is not much discussion on the intelligence of the management in the QoS. Based on the deficiency of the above-mentioned schemes and some current researches on this field, it is discovered that the intelligence of the agent is not fully utilized in current research. However, this is indeed an important advantage of an intelligent agent. It is a necessary question on how to fully use the intelligence of the agent in the specific application.

A model for the intelligent agent is presented by combining the existing protocols for the QoS management in this paper, e.g., the strategy services, the DiffServ and the IntServ. The behavior of the data flow in the next stage can be effectively predicted by the agent based on the characteristics of the data flow, which is obtained by sensing the conversion flow in the application layer. The optimization of the global networks and the distribution of the tasks in the management are achieved by the exchanges of the

Translated from *Journal on Communications*, 2006, 27(4): 1–6 (in Chinese)

LI Jie, WANG Ru-chuan (✉), BIAN Zheng-ai
Department of Computer Science,
Nanjing University of Posts and Telecommunications,
Nanjing 210003, China
E-mail: wangrc@njupt.edu.cn

parameters of the QoS between the different agents based on the communication property and the negotiation in the performance of the QoS from the single node to the global networks. In the research process, an important question on how to obtain the characteristic of the flux applying with self-determination is analyzed in detail. In view of the periodicity of the aggregate macro-flow and the paroxysmal property in the micro-flow of the conversation, the agents between the two levels are classified. Different mathematical models are constructed to solve the concerned question. In addition, the engineering method for knowledge and the management are introduced in the agent to process the data on the flux and perform strategies based on the pre-established rules, and the intelligent property of the agent, i.e., the capacity of the self-reasoning and self-determination, is obtained by the establishment of the management for the QoS. Because it is not feasible in practice to have all the functionalities using one agent in distributed networks, the functionality must be divided to assure that the function of the different agents focuses on a specific part. The cooperation between the different sub-functionality is constructed at the same time. Based on the different function and the position, a different model for the agent is constructed and a specific classification is formed based on these agent models.

The remainder of the paper is organized as follows: the following section presents our approach in detail. Section 3 describes the mobile agent interaction scheme. Some experiments and comparisons are given in Sect. 4. We conclude this paper in Sect. 5.

2 Inner model of the intelligent agent

A basic model of the intelligent agent is presented in this section and the structure is described in Fig. 1. The intelligent agents presented in the following sections are all based on this model.

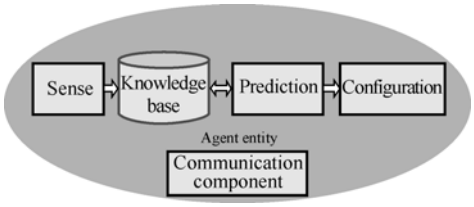

Fig. 1 The system structure for the intelligent agent

Figure 1 shows that the process of an intelligent agent is based on four basic steps: sense, learning, prediction and configuration.

1) The perception of the flow converts the data flow in the current time slot into a mathematic model. The data flow from the device is first monitored by the intelligent agent and then the data flow is classified based on certain rules. Lastly, the distribution characteristic for the flux in every time slot is acquired, which is described by a mathematic model and stored in a pattern database.

2) The history of data flow is analyzed by the intelligent agent based on the flow database and the distribution of the flow so that the next data flow can be predicted based on certain rules.

3) The agent completes the configuration via a strategy adapter, and the result in a previous prediction is converted to a corresponding strategy which is executed by the device.

3 Entity model of the intelligent QoS agent

3.1 The sensing parameters estimation of the exterior flux

Based on the detecting mode, the intelligent agent senses the flux passively, at the same time, determines the classification of the data in the interface. In a sampling time which is adjusted according to the actual flux in practice, the intelligent agent recognizes a process for the conversation according to the IP five elements group, i.e., the address of the source, the address of the destination, the port for the source, the port for the destination and the code number for the protocol. In the process, some characteristic parameters are obtained, such as the average speed in the sampling time, the maximum speed and the minimum speed in the sampling time and the maximum MTU value of the data flow in the sampling time, including the size of the packet. The process of the flux perception is described in Fig. 2. The intelligent agent classifies the data flow in the preliminary mode based on a protocol in the application layer which is loaded by the flux, and the data flow is directly classified into the queue of the application layer that it belongs to if the port adopted by the data packet is the standard port of the TCP or UDP. The intelligent agent recognizes the protocol in the application layer by obtaining the 500 bit of the TCP or UDP packet in head if the number of the port adopted by the data packet is dynamic. The object of the classification is to confirm the metric requirement, which is needed by the protocol of the application layer in QoS, according to the protocol of the application level itself, Then, every conversation flow is described using the $\boldsymbol{T}_{\text{spec}}$ vector according to the requirement of the metrics, several variables are included: the size b of the token-bucket, the speed r of the token, the maximum peak value p, the minimum overhead m and the maximum MTU value M, and the most important variables are the components of the (r, b, p) because they directly influence the queue delay of the data flow in the device, the dithering of the data flow and the packet loss rate. The speed of the update for the token-bucket must be enhanced in some tasks, which are sensitive to the time delay and dithering, such as the flux of the video and audio. In this way, the time delay of queue for the token-bucket is decreased. In addition, the size of the token-bucket can be reduced appropriately.

The MPEG video flow is sensitive to dithering and time delay. It also requires a higher bandwidth. The key to

describing the MPEG video flow is to assure an instantaneous time delay of the paroxysmal data and the loss rate of the data packet.

In Ref. [8], it is pointed out that the structure of the MPEG data flow is represented in the form of the group of picture (GoP). There are three different types of frames, i.e., I frame, B frame and P frame, and the paroxysmal data is mainly due to the I frame. The three types of frame in the MPEG data flow are passed in the group of the *MmNn*, where *m* is the interval of the I–P or P–P, *n* is the total of the frames, for example: *M*3*N*12 is expressed by: IBB-PBB-PBB-PBB.

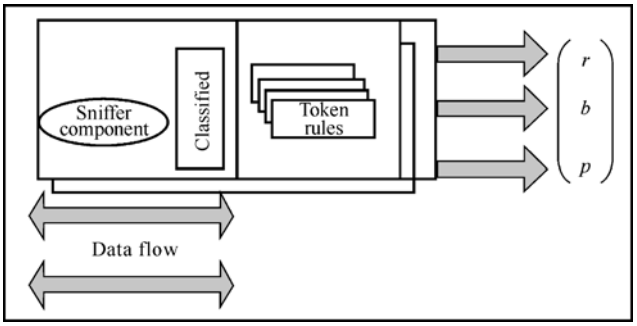

Fig. 2 The sense process of the flow

Thus, the parameter *r* in the speed of the token-bucket is:

$$r \geqslant \frac{R_{\mathrm{I}} + (n - \frac{n}{M})R_{\mathrm{B}} + (\frac{n}{M} - 1)R_{\mathrm{P}}}{n} \tag{1}$$

where R_{I}, R_{B} and R_{P} are the speed of the each frame.

In one period of the GoP, the durative time of the paroxysm in the total time is:

$$\tau = \frac{B/p}{n/f} \tag{2}$$

where *B* is the quantity of the paroxysmal data, *f* is the number of frames sent per second.

From Eqs. (1) and (2), the time delay of queue in the data flow is:

$$t_{\mathrm{delay}} \approx \frac{1}{f}[\tau n - 1] \tag{3}$$

In order to reduce the dithering caused by the volume of the paroxysmal data, the difference of the time delay between the two periods should be less than a pre-established threshold, i.e.:

$$\Delta t = \max\,(t_{\mathrm{delay1}} - t_{\mathrm{delay2}}) \leqslant \mathrm{Threshhold}(\Delta t_{\mathrm{delay}}) \tag{4}$$

It can be concluded based on the algorithm of the token-bucket that there is:

$$\Delta t = \frac{b_1}{p_1 - r_1} - \frac{b_2}{p_2 - r_2} \leqslant \mathrm{Threshhold}(\Delta t_{\mathrm{delay}}) \tag{5}$$

where $p_1 > r_1$ and $p_2 > r_2$, (r, b, p) in every period can meet the above-mentioned requirement, and this can assure that the value of the dithering is less than the pre-established threshold. In addition, to avoid the division and the re-grouping of the data packet, the size of the token-bucket should be larger than the MTU value, i.e., $b_{\min} \geqslant \mathrm{MTU}$

3.2 The analysis of the flux for micro-flow and the forecasting process

The result of the prediction is obtained by the agent through the statistical analysis of the historical data on the same application flow based on the micro-flow characteristic of the application protocols conversation, the method of data statistic is adopted and the instant speed of the data flow and the durative time of the flux are used as the metrics of the statistics. According to the assignment flow, the duration of the flux is classified according to the different random distribution, and here, the duration of the assignment flow is classified into the Poisson distribution and the logarithm distribution. Based on these, the process of the conversation flow is divided into several time slots and the distribution of the data flow in each time slot is a normal distribution. The mean value and the distribution of the swatch in each time slot are concluded by applying the data statistic on two components (the speed of the token-bucket and the size of the token-bucket) of every time slot $\boldsymbol{T}_{\mathrm{spec}}$, thus the speed of token-bucket and the variable of the token-bucket in every time slot are analyzed applying the estimation area.

Assuming that the speed of the token-bucket at the *j*th times sample of the *i*th time slot is r_{ij} for some application conversation, then the mean value and the distribution of the swatch of the *i*th time slot are $\bar{r}_i$ and S^2 in this conversation.

$$\bar{r}_i = \sum_{j=1}^{n} r_{ij} \tag{6}$$

$$S^2 = \frac{1}{n-1}[\sum_{j=1}^{n} r_{ij} - n\bar{r}_i]^2 \tag{7}$$

In the case of larger swatches, the mean value of the swatch $\bar{r}_i$ meets the following formula:

$$\frac{\bar{r}_i - \mu}{\frac{S}{\sqrt{n}}} \sim N(0,1) \tag{8}$$

Assuming that the level of the belief is $1-\alpha$, then the belief area of the mathematic expect in the time slot is:

$$\left(\bar{r}_i - \frac{\sigma_0}{\sqrt{n}} u_{1-\frac{\alpha}{2}}, \bar{r}_i + \frac{\sigma_0}{\sqrt{n}} u_{1-\frac{\alpha}{2}}\right) \tag{9}$$

Based on Eq. (9), we conclude that, in the *i*th time slot of this type of the data flow, the speed value of the token-bucket fluctuates within the area of the belief.

Applying a similar method, the estimated area of the mathematic expect for the depth of the token-bucket can be acquired, and after analyzing the depth and the speed of the token-bucket in every time slot by using the method of the

data statistic, the agent can adjust the flux of the data flow in the next time slot based on the duration of the conversation.

3.3 The aggregate process of the data flow applying the agent

The aggregate process describes a kind of data flow using a larger token-bucket, and this kind of data flow is formed in aggregating small token-buckets of several micro-flow conversation. The object of aggregating lies in the fact that when data flow is aggregated, it can be described as using only one token-bucket, and the overhead of the management is reduced because the devices need not manage every single conversation. The aggregation process is considered for every small token-bucket and there should not be too much of a loss on the metrics of the QoS after the aggregation. Furthermore, the aggregative flow can fully utilize the bandwidth resource at the inter-space between the paroxysmal flow-speed and the normal flow-speed of each small token-bucket.

Based on Ref. [9], every component of the aggregated time T_{spec} will perform the following operations:

$$r = \sum_{i=1}^{n} r_i \tag{10}$$

where r is the speed of the aggregative token-buckets, r_i is the component of the speed in the ith token-bucket.

$$B = \sum_{i=1}^{n} b_i - \beta \quad (\beta \geqslant 0) \tag{11}$$

where b is the size of the aggregate token-bucket, b_i is the component for size of the ith token-bucket, β is the factor for adjustment.

$$P = \sum_{i=1}^{n} p_i - \alpha \quad (\alpha \geqslant 0) \tag{12}$$

where p is the peak speed of the aggregated token-bucket, p_i is the component for the speed of the peak value of the ith token-bucket, α is the factor for the adjustment.

$$M = \max(M_1, M_2, \cdots, M_n) \tag{13}$$

where M is the size of the maximum packet in the new token-bucket, M_1, M_2, …, M_n are the sizes of the maximum packets in the n sub token-buckets. The factors α, β for the adjustment deserve to be noted from the analysis and their values are influenced by the rate of the time for the paroxysmal data to the total time. Assuming the existence of an independent data flow, the rate r of the time for paroxysmal data in every data flow to the total time is equal to the paroxysmal value p_i. In order to assure that the probability that the paroxysmal value of the aggregate flow is less than p is ρ, let $p(i)$ be the probability that the number of the data flow i occur at the same time, and when $p<mp_i$, the loss of the packet can occur at the port, there exists:

$$\rho = \sum_{i=1}^{m-1} p(i) \tag{14}$$

According to Eq. (14), the size of α is concluded by acquiring the value m.

$$\alpha = \sum_{i=1}^{n} p_i - mp_i \tag{15}$$

3.4 Analysis and prediction on the aggregate macro-flow

The key in the analysis on the aggregate macro-flow with the classification is to obtain the characteristic of a periodical distribution for the aggregate macro-flow in one period of time, the item reflecting the trend and the periods in granular time (one minute or one hour in general) are predicted based on the characteristic of the distribution. In view of Ref. [10], the forecasting algorithm of the neural net is introduced. For instance, the periods of the aggregate data flow is set to one day and then the variation of the flux is analyzed within that 24 hours, we define a swatch of learning and then analyze the item of the trend by the learning, and lastly, the prediction is performed. The reasons for selecting BP nets as the experimental nets in the paper are its simplicity and feasibility, and also because this net is a kind of learning net with a surveillance scheme, and is especially suited to the forecasting analysis of future trend.

There are 48 neural nodes at the input layer in BP nets in this paper, they map the fluxes in a continuous 48 hour period, and there are 24 neural nodes in the output layer, which represents the data flow for the next 24 hours. A critical level with 64 neural nodes is defined and at the same time, the time series are defined as follows:

$$\text{traffic}=\{d_1=(h_1, h_2, \cdots, h_{24});\ d_2=(h_1, h_2, \cdots, h_{24});\ \cdots\}$$

where d_n are a series of the data flow in the nth day, h_n is the flux of data flow in the nth hour.

The learning swatch S is defined as:

$$S=\{(d_1, d_2); (d_2, d_3); (d_3, d_4);\ \cdots\}$$

The corresponding swatch for the teacher and the swatch for the output are given respectively as:

$$T=\{d_3; d_4; d_5;\ \cdots\},\quad A=\{a_3; a_4; a_5;\ \cdots\}$$

Based on the training, the inner weights are adjusted based on the error between the layer of the output a_i in the nets and the teacher swatch t_i, the speed in the learning is defined as l_r=1.0, the weight is W and the initial value b for adjustment is a random number. The neural nets continuously compute the weight values of the nets and the changes of offset at the direction that is relative to the descending direction of the slope reflected by the error function, and the result of the computation is sent to every layer in the opposite direction and the weights of the inner neural nodes are adjusted, and eventually, the square sum of the error between the output of the learning swatch a_i and t_i achieves the minimum. After the learning of the nets, the next prediction for the flux can be performed, and that is the data flow in the periods i_{s+2} which can be predicted by the input $I=(i_s, i_{s+1})$.

4 Experiments

4.1 The terminal agent and the bottleneck agent

The flux sensor is one important terminal agent component and its main function include sampling, recognition, classification and statistics and token-bucket description. The sampling process consists of the preliminary classification and analysis on the reports of the data packets at the interface utilizing the analysis tool for the packet. The specific steps include: constructing a procedure for monitoring and computing the size of the packet, classifying and making the statistic on the data packet received per second by defining the timing task, classifying the data flow by utilizing the five IP elements group and make the statistics on the sizes of the different data flow within every time slot. After several sampling periods, the analysis program will collect the data flow in every sampling period and acquire the peak value and the mean value of the data flow. Simultaneous with the data analysis, the agent can determine the requirements on the metric of data flow QoS based on the characteristics of the data flow. Since the data flow is relatively steady in a prototype system (in order to simulate the characteristics of the video flow), the speed of the peak value is regarded as the optimal value in the bandwidth assignment of the steady data flow and at the same time, the agent will convert the speed of the peak value into the corresponding description of the token-bucket. When the agent finds that the difference between the state of the data flow and the state recorded is relatively larger, it will automatically send the bottleneck agent to detect the state of the nets by the iteration in every node.

There are several main attributes in the bottleneck agent, such as the bandwidth, dstHost, dstPort and seq. The former three designate the conditions of the filtering for the bottleneck agent in the configuration for the device and the latter designates the plan for the traverse; the bottleneck agent will visit the nets based on the perambulating plan. When the agent arrives at one node, the attribute of the instance will be converted into the configuration of the device by executing a prescribed operation.

4.2 The sensing experiment of the exterior flux

Utilizing the media server in the Windows 2000 operating system, we obtain the MPEG video flow and the sampling value is analyzed in the time slot within 10 ms and the data flow is described by a token-bucket for every time slot. The transmission time for the paroxysmal data in the total time is $\tau = 12/80 = 0.15$ and the number of frames per second is $f = 25$ fram/s. The maximum value of the queue time delay for the video flow is t_{max}= 250 ms and the maximum value of the dithering is Δt_{max}=10 ms, then the average speed of the token-bucket is r >315 940 bit/s, and in view of the speed of the peak value in the data flow with 344 600 bit/s, we define the range {315 940, 344 600} for the speed of the token-bucket. When the data flow MTU is at 1 500 byte, the size of the token-bucket is 1 500×8 bit.

The data flow in the experiment is analyzed and computed with the sampling mode and then the intelligent agent will obtain the flux of the video flow in the given period of time, and the detail can be referred to in Fig. 3. The speed of the token-bucket is analyzed in the statistic in the experiment and the result is described in Fig. 4. In the case where time delay is certain, the speed of the token-bucket is maintained on the whole and there is no frequent adjustment. This is suited to the establishment in advance. The result in the experiment proves that the scheme described in the paper can sense the exterior flux effectively.

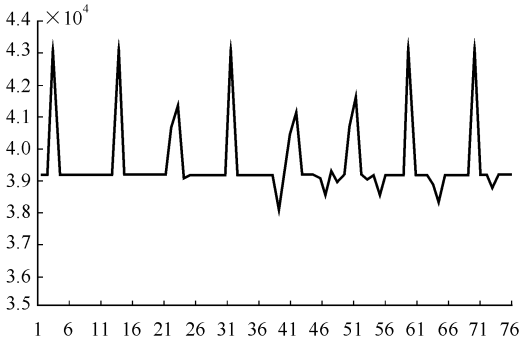

Fig. 3 The sampling of the video flow

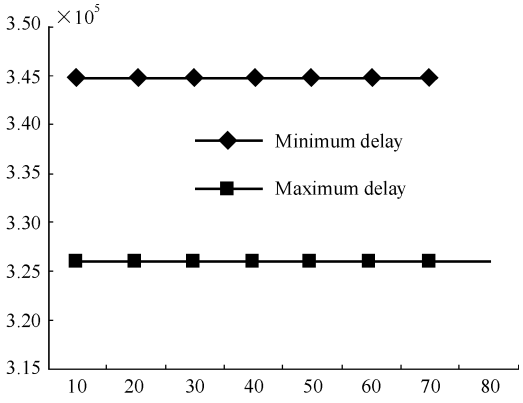

Fig. 4 The estimation of range for the token-bucket value

4.3 The predictive experiment for the flux

In the experiment, we gather the total of the TCP aggregate data flow per hour in the server for a period of 30 days in the lab, and the data gathered in a previous period of 20 days form the swatch of the training and the data in five days are used as the swatch for the simulation, and then data analysis is performed on the scheme. Since there exists a

relatively larger error when the swatch for the training is inputted directly, the swatch is preprocessed, that is, the operation of logarithm with parameter 10 is introduced to the data in the swatch and then a unitary process is applied, the maximum times in training is 60 000 and after the computation, the errors outputted are recorded in Table 1. The data in the table shows that the distribution of the aggregate data flow for certain periods can be approximated by using the neural nets method.

Table 1 The comparison between forecasted value and actual value

Actual value	Predictive value	Error
865	814.1	0.058 845
391	390.7	0.000 714
220	207.4	0.057 125
729	854.3	−0.17 192
334	398.5	−0.19 339
196	208.4	−0.06 351
914	788.9	0.136 827
437	372.3	0.147 892
221	195.5	0.115 247
960	994.4	−0.03 589
452	436.8	0.033 525
224	223.9	0.000 322

4.4 The instance validity

The data flow in different ports is produced in the prototype system, the former data flow is steady with the assumption that it requires a higher stability in the bandwidth, and the latter is a random paroxysmal data flow and it produces the data flow with peak values in a certain range when there is a lower requirement for stability in the bandwidth. A terminal agent is configured at the data reception device, the terminal agent monitors the state of the steady data flow and computes the requirement of the bandwidth for the steady data flow according to a pre-established strategy. When the data flow received is not consistent with the requirement of the bandwidth, a bottleneck agent will be generated and the bottleneck lookup procedure will be invoked. The transfer plan for the bottleneck agent is based on the address in the next hop, the bottleneck agent then performs the operation for preserving the bandwidth at the nodes of its route when it arrives at every node until the source node. After sending the bottleneck agent, the terminal agent will monitor the state of the data flow and when there is no data transmission for the data flow during a period of time, the terminal agent will set free the bandwidth resource preserved.

The changing state of the data flow which is monitored by the terminal agent is described in Fig. 5. It is discovered that at the time between 7 s and 9 s, the data flow reaches the bottom of the curve, and the reason for this is that the terminal agent has not passed the assignment to the bottleneck agent and at the time of about 9 s, the bottleneck agent fulfills its assignment, and the bandwidth of the data flow goes back to its normal state.

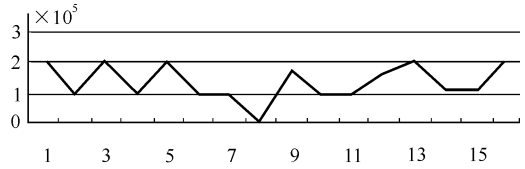

Fig. 5 The monitoring state of the data flow

5 Conclusions

Applying the intelligent agent into a QoS management is an important direction in current research. It presents many new requirements on the agent itself, such as the interaction of the agents, the cooperation between agents, etc. The object of the scheme in the paper concentrates on increasing utilization of the characteristics of the agent, the local information on the flux is processed at the different agents and the efficiency in controlling the QoS is enhanced by the communication between the agents; in addition, the knowledge database for resource management of the QoS is given a preliminary discussion and research. Since the distributed management of the QoS is a complicated system engineering, the configuration for the resource with the negotiation scheme need to be researched further in the future.

Acknowledgements The subject is sponsored by the National Natural Science Foundation of China (No. 60573141 and 70271050), the Natural Science Foundation of Jiangsu Province (No. BK2005146), High Technology Research Programme of Jiangsu Province (No. BG2004004, BG2005038 and BG2006001), High Technology Research Programme of Nanjing (No. 2006RZ105), Foundation of State Key Laboratory for Modern Communications (No. 9140C1101010603) and Key Laboratory of Information Technology processing of Jiangsu Province (No. kjs05001 and No. kjs06).

References

1. He Yan-xiang, Chen Zi-meng, Design and application of agent and multi-agent system, Wuhan: Publishing House of Wuhan University (in Chinese)
2. Li Ye-wen, Meng Luo-ming, Qi Feng, The Study and perspective of mobile agent applications in network management environment, Acta Electronic Sinica. 2002, 30(4): 564–569 (in Chinese)
3. Youssef S. M., Ismail M. A. et al., Integrating mobile agents and swarm optimization for efficient QoS management in dynamic programmable networks, Electrotechnical Conference, Roma, 2002: 358–363

4. Telma M., Stylianos G., Quality of service management in IP networks using mobile agent technology, Proceedings of Mobile Agents for Telecommunication Applications, Berlin, 2002: 193–205
5. Manuel G., Torsten B., Internet service monitoring with mobile agents, IEEE Network , 2002, 16(3): 22–29
6. Kalaiarul D., Martin C., Transparent QoS support of network applications using netlets, Proceedings of Mobile Agents for Telecommunication Applications, Berlin, 2002: 206–215
7. Wang Y., Cooperating intelligent mobile agent mechanism for distributed multimedia synchronization, IEEE International Conference on Multimedia and Expo, New York, 2000: 743–746
8. Fei Xue, Ben Yoo S. J., High-capacity multiservice optical label switching for the next-generation Internet, IEEE Communications Magazine, 2004, 42(5): 16–22
9. Yoo S. J. B., Fei Xue et al., High-performance optical-label switching packet routers and smart edge routers for the next-generation internet, IEEE Journal on Communications, 2003, 21(7): 1041–1051
10. Cheng Guang, Gong Jian, Seasonal neural network Model on internet traffic behavior, Mini & Micro Computer, 2002, 23(11): 1321–1324 (in Chinese)