Research on and Realization of an Adaptive Clock Synchronization Algorithm for Distributed Systems

Jia Yang[1] and Piyan He[2]

[1] College of Communication Engineering, Chengdu University of Information Technology, Chengdu, 610225, China
[2] Institute of Aerospace Science and Technology, University of Electronic Science and Technology of China, Chengdu 610054, China
yangjia@cuit.edu.cn, he_py@126.com

Abstract. In distributed systems, the uncertainty of the clock drift and transmission delay is a problem that cannot be ignored, because it influences the precision of the clock synchronization directly. In this paper, considering this problem, an adaptive clock synchronization algorithm which aims the characteristics of enterprise distributed systems is proposed based on the passive algorithm. The adaptive algorithm can automatically determine the time interval of the two adjacent clock synchronization adjustments and choose the optimized measurement times and the clock adjustment value of each clock synchronization adjustment, so that, an optimized plan can be used to achieve the synchronization task. It is proved that the algorithm can inhibit effectively the influence of the uncertainty to meet the accuracy requirements of the clock synchronization in enterprise distributed systems. The algorithm has a good practical value.

Keywords: distributed systems, adaptive algorithm, clock synchronization, transmission delay.

1 Introduction

Clock synchronization is one of core technologies of distributed systems. Its task is to make sure the information, events and each node whose behavior associated with time have a global consistent reference. The key of promoting the precision of the clock synchronization is to offset or inhibit the influence of the uncertainty of the clock drift and transmission delay. So in this paper, according to characteristics of enterprise distributed systems, a network structure for the adaptive algorithm is built. On this basis, the algorithm is discussed how to resolve the influence of the uncertainty. At last, the realization and performance analysis of the algorithm are given to prove its effect.

2 Network Structure for the Adaptive Algorithm

These are three kinds of clock synchronization algorithms: centralized, master-slave, distributed. This paper aims enterprise distributed systems, such as distributed real-time

Y. Wang and T. Li (Eds.): Knowledge Engineering and Management, AISC 123, pp. 129–134.
springerlink.com

data motoring systems. So it pays attention to convenient management, high reliability, low cost, adaptation etc. Firstly considering the convenient management, the designer chooses the passive algorithm of the centralized mode that clients should be managed by at least one server and the server receives clock synchronization message from clients passively. Secondly, considering the high reliability, according to the number of clients, if the number is small, only two servers are needed. One of them manages clients and another as a backup. The two servers synchronize with the external GPS. If the number is large, it is considered to divide the whole area to several sub areas and each sub area can be managed by one server. All of the servers synchronize with external GPS. Once one of servers has fault, there is a server in the near sub area to replace it temporarily.

3 The Analysis of the Adaptive Clock Synchronization Algorithm

According to the passive algorithm, the communication process of the client and server is shown as Fig.1.

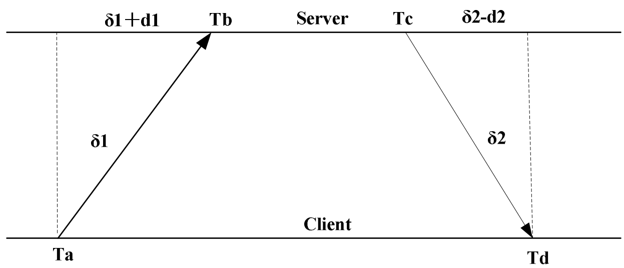

Fig. 1. The clock synchronization based on Client/Server model

Supposed: T_a, T_b, T_c, T_d separately are the time at which the client sends the clock synchronization request message, the time at which the server receives the request message, the time at which the server sends back the response message including the standard time, and the time at which the client receives the response message from the server. δ_1 is the delay from the client sending the request message to the server receiving the request message. δ_2 is the delay from the server sending the response message to the client receiving the response. d_1 is the clock deviation of the client relative to the server at T_a. d_2 is the clock deviation of the Client relative to the server at T_d. The key of the precision of the clock synchronization depends on the communication delay δ_1, δ_2 and the clock deviation d_1, $d2$. In theory, $\delta_1=\delta_2=\delta$, so formula (1) is shown as below:

$$\begin{cases} T_d - T_c = -d_2 + \delta . \\ T_b - T_a = d_1 + \delta . \end{cases} \tag{1}$$

For the computer clock drift is caused by the different frequency of the physical Oscillators in the side of the different nodes. For the specific system, with the advance of the timeline, the clock drift of the client relative to the server approximates a line [2], that $\Delta d = a\Delta t$. a is a constant that can be gotten in the experimental environment. In the actual measurement, because the time that the server processes the message is very shot, so Δt can be approximately considered as 2δ, that $\Delta d = 2a\delta$. So that according to formula (1), the formula (2) can be gotten. It is shown as below:

$$\delta = \frac{1}{2(1-a)}(T_b + T_d - T_a - T_c) \ . \tag{2}$$

In the actual measurement, because of involving many aspects, so the random error maybe brought to the measurement data. δ should be cumulated n times to calculate the average $\overline{\delta}$ shown as formula (3).

$$\overline{\delta} = \frac{1}{2(1-a)N}\sum_{k=1}^{N}(T_{bk} + T_{dk} - T_{ak} - T_{ck}) \ . \tag{3}$$

According to formula (3), the formula (4) can be gotten as below:

$$T_{jz} = T_c + \overline{\delta_n} \ . \tag{4}$$

Notes: *Tc* is the standard time of the server responding the client at the N time' measurement. T_{jz} is the client time after synchronization adjusting.

According to the analysis above, the synchronization focuses on the selection of the measurement times in each clock synchronization adjustment and the determination of the interval of the adjacent twice clock synchronization adjustment." **The mind of the adaptive algorithm is":** because of the random error, the standard error σ of the average is brought to analyze the jitter of measurement data of N times, shown as formula (5). The value of σ is smaller and the average is closer to the real data value, and it shows that the jitter of N times 'measurement is smaller.

$$\sigma = \sqrt{\frac{\sum_{i=1}^{n}(\delta_i - \overline{\delta})^2}{(n-1)}} \ . \tag{5}$$

Notes: δ_i is the communication delay of the client and server at the i time.

For the selection of the measurement times in each clock synchronization adjustment, the algorithm will set the upper limit MAX of the measurement times. MAX is the multiple of 10. Each clock synchronization adjustment will do MAX times measurement. When the measurement times N is cumulated to the multiple of 10, the average and average standard error of N times before will be calculated and compared with the standard error of the N-10 times before. The smaller value of the standard error, the measurement times and average corresponding with the standard error will be recorded. So the smallest standard error, the measurement times M and the average $\overline{\delta}$ corresponding with it will be gotten by comparing method. According to the formula (4), the $\overline{\delta}$ and Tc at the MAX time will be used to adjust the client's system time.

For the determination of the time interval of the two adjacent clock synchronization, the algorithm considers the clock drift of the client relative to the server approximates a line, that $\Delta d = a\Delta t$. So according to the actual network, we can set Te as the upper limit of the clock deviation, every one hour calculate the Δd and compared with Te, until $\Delta d = T_e$ .Then the client will send synchronization request message to do another clock synchronization adjustment. Considering the current life and precision of the clock, the time interval T_k when $\Delta d = T_e$ can be recorded. From now on, during a sometime, T_k can be used as the clock synchronization interval. For the higher precision, because of the aging of a clock, the time T_k should be measured periodically.

4 The Realization and Performance Analysis of the Algorithm

The software for the algorithm has two modules. One of them is synchronization module, another is performance analysis module. For the synchronization module, by using Client/Server mode in NTP (Network Time Protocol) clock synchronization measurement is done according to the Fig.2 as below.

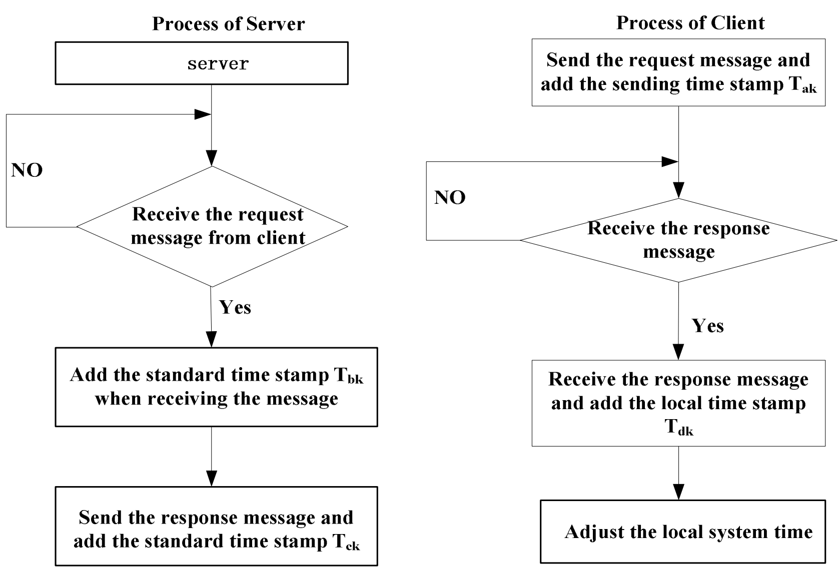

Fig. 2. Clock synchronization process

For getting the system time of the client and server, there are two kinds of plans to be provided. One is ‘QueryPerformanceCount’ time counter. It can be accurate to microsecond. If it needs a higher precision, and the client use the CPU above the level of Intel Pentium and including Intel Pentium. The timer class of RDTSC instruction: Ktimer class can be read the time stamp of the CPU main frequency. For the CPU whose frequency is above 1GHz, the system time can be accurate to nanosecond by using Ktimer class.

According to the design above, in some enterprise’s distributed systems, the adaptive algorithm is tested. ‘QueryPerformanceCount’ time counter is used to read the system time. The specific performance analysis is shown as Fig.3 and Fig.4.

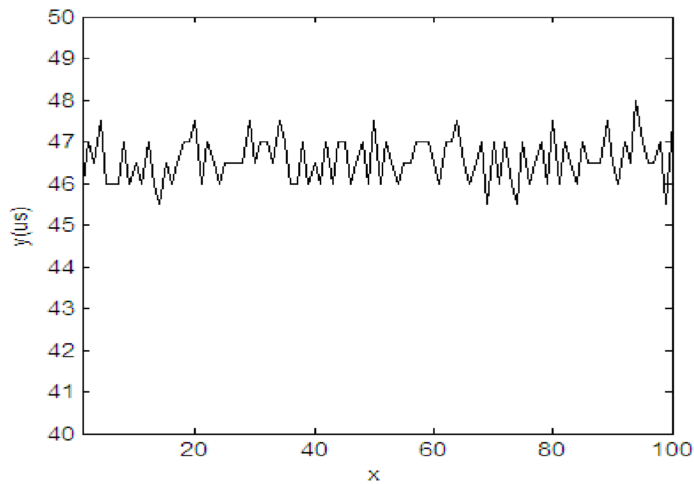

Fig. 3. The data of 100 times measurement in once clock synchronization adjustment

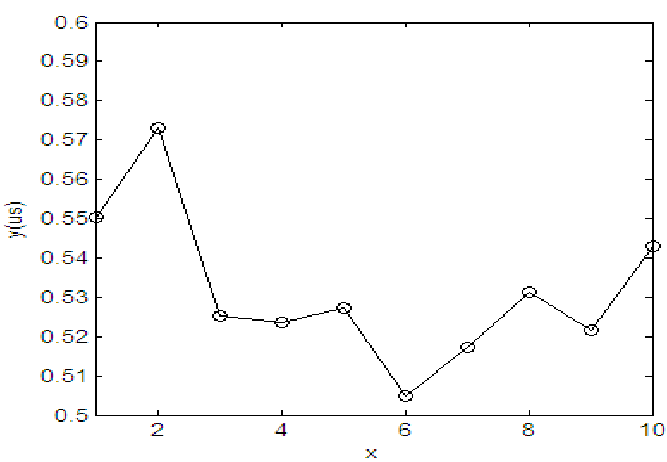

Fig. 4. The average standard error of $x \times 10$ times before

In Fig.3, x coordinate expresses the times of measurement in once clock synchronization adjustment. The times are 100. y coordinate is the communication delay δ between the client and server. Its unit is μs .

In Fig.4, based on the data in Fig.3, for easy watch, the data of 100 times is divided by 10 times as a unit. For example, *x=6* means 60 measurement times. y coordinate is standard error of average, its unit is μs .

In Fig.4, when *x=60*, the value of *y* is smallest. *y=0.5050*.So the algorithm will select the average of 60 times before to adjust the client clock. For the determination of the time interval between the two adjacent clock synchronization adjustments, the enterprise stipulates the clock deviation must be smaller than 10 ms, a is 2.1E-6 that is gotten in experimental environment. So, $\Delta t \cong 79\,\text{min}$. For easy measurement, the algorithm will select 70 minutes automatically to do the clock synchronization adjustment again.

5 Conclusions

The result and performance analysis of the adaptive algorithm show that the algorithm can choose the optimized plan to adjust the clock synchronization of the distributed systems through statistic learning. This algorithm has a strong ability to adapt the enterprises system.

References

1. Zhu, B., Yang, S.: Clock Synchronization Algorithm of Real-time Distributed Simulation Test Platform. Computer Engineering 24, 231–235 (2010)
2. Liu, L., Wang, X., et al.: High-Precision Clock Synchronization Algorithm for Distributed System and Its Realization. Journal of Tianjin University 8, 923–927 (2006)
3. Ning, W., Zhang, B., et al.: Statistical Mean Algorithm of Realizing Clock Synchronization Based on Distributed System. Computer Engineering 5, 47–50 (2005)
4. Qin, X., Zhai, Z., Liu, Z.: Research of Real-Time Requirement and System Time Synchronization About Distributed Experiment& Measurement. Measurement & Control Technology 2, 63–65 (2004)
5. Altuntas, B., Wysk, R.: A.: A Framework for Adaptive Synchronization of Distributed Simulation. In: Proceedings of the 2004 Winter Simulation Conference, Washington DC, USA, pp. 363–369 (2004)