

# A Delay-Based Dynamic Load Balancing Method and Its Stability Analysis and Simulation

Qingyang Meng<sup>1</sup>, Jianzhong Qiao<sup>1,2</sup>, Shukuan Lin<sup>1</sup>, Enze Wang<sup>1</sup>, and Peng Han<sup>1</sup>

<sup>1</sup> College of Information Science and Engineering,

Northeastern University, Shenyang, China

<sup>2</sup> Key Laboratory of Software System and Development

Generic Technology, Shenyang, China

neumqy@126.com

**Abstract.** Delay phenomenon commonly exists in most load balancing systems for parallel computing. It can cause some unstable oscillatory actions and intensely affect the performance of the load balancing system. In this case, a time delay feedback control model is presented to describe the dynamic load balancing system in parallel surrounding. By choosing proper Lyapunov-Krasovskii functionals and using Moon inequality and Schur complement lemma, the optimal control law is obtained not only for the different delay conditions but also in a scalable system scale. At last, the simulation experiments based on multi-threads proved the validity of the theory and method introduced.

## 1 Introduction

DLB (Dynamic Load Balancing) is one of the most significant problems in parallel computing. A good DLB method can substantially increase the efficiency of parallel computing and holds the particular advantages in solving irregular problems. The studies about this issue are various [1,2,3,4,5,6], but most of them do not consider the delay factors well. Against this background, some researchers are focusing on the DLB problem in the time delay surrounding. [7] proposed a linear delay DLB model and analyzed the relation between the delay and load balancing gain in a three-node system by using the frequency-domain stability theory. A time-varying nonlinear model was presented in [8,9]. They used the Lyapunov's second method to prove the stability of the proposed model and added some actual resource constraints to make the model more practical. Based on the model in [7], [10] used Finsler's Lemma and Jensen's inequality to find the delay-dependent stability condition in time-domain, so as to improve the scalability and use range of the load balancing method.

However, [7,8,9,10] only focused on the asymptotic stability. By contrast, the exponential stability obviously holds a quicker convergence rate and distinct advantages to find the optimal LBG(load balancing gain). Moreover, [7,8,9] did not give a specific relation among time delay, system scale and LBG in theory, and [10] lacked the validation of the introduced theory. In this case, a more practical time delay DLB model is proposed. Under some assumptions and definitions, the more optimal control laws are obtained in a different time delay and system scale by choosing proper Lyapunov-Krasovskii functionals. At last, a simulation method based on multi-threads is presented to verify the obtained theoretical results before.

The chapters are organized as: Sect. 2 introduces a time delay feedback control model of DLB system. Focusing on the model, some assumptions, definitions and modifications are presented later. In Sect. 3, the delay-dependent exponential stable condition is obtained by analyzing the modified model. Sect. 4 proposes a simulation frame based on multi-threads, and via comparing the simulation results with theoretical results, the estimated effects of proposed method are educed. Finally, our conclusions are given in Sect. 5.

## 2 The Mathematic Model

A DLB system for parallel computing is usually made up of several CEs(computing elements). Every CE holds its own CPU waiting queue, CPU processing rate and so on, and can communicate these local information with each other. According to the communicated information, CEs execute the DLB algorithm in a certain cycle, and when the transferred loads arrive to the target CEs, the one-time DLB is over. In the above-mentioned process, time delay may generate in three forms, namely CD (Communication Delay), TD(Transmission Delay) and LBD(Load Balancing Delay), where LBD is mainly caused by the execution cycle and spending of DLB in every CE. So based on the model in [7], a more practical time delay feedback control model is introduced as follows.

$$\frac{dx_i(t)}{dt} = q_i(t) - \frac{\lambda_i}{\sum_{l=1}^n \lambda_l} \sum_{j=1}^n q_j(t - \tau_{ji}) . \quad (1a)$$

$$q_i(t) = d_i(t) - u_i(t) + \sum_{j \neq i} \lfloor p_{ji} u_j(t - \eta_{ji}) \rfloor . \quad (1b)$$

$$u_i(t) = \lfloor k_i [x_i(t)]^+_{domain} \rfloor . \quad (1c)$$

where

1.  $n$  is the number of CEs with the condition of  $n \geq 2$ .
2. State variable  $x_i$  denotes the excess load of the  $i$ -th CE.
3. Input variable  $u_i$  denotes the transfer loads from the  $i$ -th CE to the other CEs due to the DLB algorithm in a unit time.
4.  $q_i$  is the queue length change rate of the  $i$ -th CE.
5.  $d_i(t) = \mu_i - \lambda_i$ ,  $\mu_i$  and  $\lambda_i$  are the generating rate and the processing rate of the  $i$ -th CE respectively.
6.  $\tau_{ji}$  and  $\eta_{ji}$  are the CD and TD from the  $j$ -th CE to the  $i$ -th CE respectively.
7.  $\sum_{j \neq i} p_{ji} u_j(t - \eta_{ji})$  denotes the excess loads transferred from the other CEs to the  $i$ -th CE due to the DLB mechanism. In which,  $p_{ji}$  is the LP(Load Partition) of the excess load from the  $j$ -th CE to the  $i$ -th CE. Generally, we have  $p_{ji} > 0$  and  $p_{ii} = 0$ .
8.  $k_i$  is the close-loop gain of the DLB system in a unit time.
9.  $[x_i]^+_{domain} = \begin{cases} x_i, & x_i \geq domain \\ 0, & x_i < domain \end{cases} .$

Compared with model in [7], model (1) considers  $d_i(t)$  as a external disturbance and supports heterogeneous resource. It's obvious that when  $\lambda_1 = \lambda_2 = \dots = \lambda_n = \lambda$ , model (1)

degenerates into homogeneous model. Meanwhile, because of using the time-domain analytical method, model (1) selects the excess load as state variable  $x_i$ . If model (1) is exponential stable in balanced point, we have  $\lim_{t \rightarrow t_b} x_i(t) = 0$ , this means that, the

system load is balanced in time  $t_b$ . Furthermore, because the load can not be divided infinitely and the CPU queue length should be non-negative (the detail about negative-queue phenomenon can be found in [8]), model (1) introduces the operations  $\lfloor \cdot \rfloor$  and  $\lceil \cdot \rceil_{domain}^+$  to keep a coherence with the actual environment. In which, the selection of *domain* can make a compromise between the balancing speed and balancing quality, and *domain* takes the default of  $\lceil n / 2 \rceil$  in this paper.

## 2.1 Dedicated and Non-dedicated CEs

In DLB system for parallel computing, there mainly exists two kinds of CEs, i.e. DCE(Dedicated CE) and NCE(Non-dedicated CE). The DCEs often service for a whole computing task, whose processing rate is relatively fixed and the new subtask is rarely generated due to the static partition of parallel task(such as MPICH [11], PVM [12]), so a DCE-oriented DLB system mainly focuses on the computing efficiency of a whole task. By contrast, the NCEs frequently service for many kinds of complete tasks at the same time, whose subtask generating rate and processing rate are often stochastic and nonlinear, so a NCE-oriented DLB system always pays more attention to the utilization rate of CEs. These conditions indicate that, when model (1) is DCE-oriented, we generally have  $\mu_i=0$  and the  $d_i(t)$  can be viewed as a linear disturbance. Meanwhile, if model (1) is NCE-oriented,  $d_i(t)$  may become the nonlinear disturbance for the DLB system.

## 2.2 Discussion about LP

In DLB system, the LP methods can be divided into two classes, namely CLP(Constant LP) and VLP(Variable LP), where CLP does not depend on the instant global information and is easy to implement. [7,8,10] discuss the selection of CLP, and based on these, a heterogeneous CLP is proposed as follows.

$$p_{ji} = \lambda_i / \sum_{s \neq j} \lambda_s. \quad (2)$$

However, by contrast, VLP often holds a better balancing speed, but its efficiency relies on the accuracy of the global information. Reference [13] proposed an efficient VLP method as follows.

$$p_{ji} = \begin{cases} x_i(t) / \sum_{s \in \Psi} x_s(t), & i \in \Psi. \\ 0, & otherwise. \end{cases} \quad (3)$$

where  $\Psi \triangleq \{i : x_i(t) < 0\}$ .

## 2.3 Some Assumptions, Definitions and Conversions

In order to analyze the proposed model more efficiently, some assumptions, definitions and conversions are introduced as follows.

**Assumption 1:** Model (1) is continuous linear.

Most systems in real environment are non-linear, but if neglecting some tiny nonlinear factors, we can establish an approximate linear model which can be easily analyzed. Assumption 1 indicates that, the discontinuous factor  $\lfloor \cdot \rfloor$  and saturation factor  $[ \cdot ]_{domain}^+$  will be neglected in the analyzing work.

**Assumption 2:**

$$d_i(t) = 0. \quad (4)$$

In Assumption 2,  $d_i(t)$  is viewed as linear disturbance approximately. Because the stability of linear system has no relation with the linear external disturbance [14], Assumption 2 has no effects on the stability analysis results. Furthermore, it also implies that the analytical model is more close to DCE-oriented DLB system under Assumption 2.

**Assumption 3:**

$$\tau_{ji} = \tau, \eta_{ji} = \eta, k_i = K. \quad (5)$$

In order to reveal the effects brought by time delay more clearly, Assumption 3 considers it as a constant. However, the time delay in DLB system is stochastic, but in most instances, it may be approximately described by the normal distribution [15]. In fact, from Assumption 3, CD and TD can be considered to obey normal distributions as  $N(\tau, \sigma_\tau^2)$  and  $N(\eta, \sigma_\eta^2)$ .

Note that in Assumption 3,  $K$  is the analytical gain for the DLB system, and it denotes the gain proportion in a unit interval. However, the actual gain  $K_a$  used in the actual algorithm represents the gain proportion during the time interval (i.e. LBD) between two approaching load balancing executions in each CE. The relation between  $K$  and  $K_a$  can be described by the following definition:

**Definition 1:**  $K_a = \int_0^{\Delta t} K dt = \Delta t * K . \quad (6)$

where  $\Delta t$  denotes the LBD in DLB system.

On the premise of Assumptions 1~3, substituting Eqs. (1b) and (1c) into Eq. (1a) and arrange it, we have

$$\dot{X}(t) = A_0 X(t) + A_1 X(t - \eta) + A_2 X(t - \tau) + A_3 X(t - \eta - \tau). \quad (7)$$

where

$$X(t) = \begin{bmatrix} x_1(t) \\ \vdots \\ x_n(t) \end{bmatrix}, F = \begin{bmatrix} f_1 & & & \\ & \ddots & & \\ & & f_n & \end{bmatrix}, P = \begin{bmatrix} 0 & p_{21} & \cdots & p_{n1} \\ p_{12} & \ddots & \ddots & \vdots \\ \vdots & \ddots & \ddots & p_{nn-1} \\ p_{1n} & \cdots & p_{n-1n} & 0 \end{bmatrix}, B = \begin{bmatrix} 0 & b_1 & \cdots & b_1 \\ b_2 & \ddots & \ddots & \vdots \\ \vdots & \ddots & \ddots & b_{n-1} \\ b_n & \cdots & b_n & 0 \end{bmatrix},$$

$$A_0 = -KF, A_1 = KFP, A_2 = KB, A_3 = -KBP, f_i = 1 - \frac{\lambda_i}{\sum_{r=1}^n \lambda_r}, b_i = \frac{\lambda_i}{\sum_{r=1}^n \lambda_r}.$$

### 3 Exponential Stability Analysis

Aiming at model (7), a delay-dependent exponential stable condition is proposed as follows.

**Theorem 1:** *The system (7) is exponential stable for all delays  $\eta, \tau > 0$  if there exists positive definite symmetric matrices  $Q, R_b, Z_b$ , symmetric matrices  $W_b, Y_b$  such that*

$$\begin{bmatrix} \Pi & -Y_1 + Q\bar{A}_1 & -Y_2 + Q\bar{A}_2 & -Y_3 + Q\bar{A}_3 & \eta\bar{A}_0^T Z_1 & \tau\bar{A}_1^T Z_2 & (\eta + \tau)\bar{A}_0^T Z_3 \\ * & -R_1 & 0 & 0 & \eta\bar{A}_1^T Z_1 & \tau\bar{A}_1^T Z_2 & (\eta + \tau)\bar{A}_1^T Z_3 \\ * & * & -R_2 & 0 & \eta\bar{A}_2^T Z_1 & \tau\bar{A}_2^T Z_2 & (\eta + \tau)\bar{A}_2^T Z_3 \\ * & * & * & -R_3 & \eta\bar{A}_3^T Z_1 & \tau\bar{A}_3^T Z_2 & (\eta + \tau)\bar{A}_3^T Z_3 \\ * & * & * & * & -\eta Z_1 & 0 & 0 \\ * & * & * & * & * & -\tau Z_2 & 0 \\ * & * & * & * & * & * & -(\eta + \tau)Z_3 \end{bmatrix} < 0. \quad (8a)$$

$$\begin{bmatrix} W_l & Y_l \\ * & Z_l \end{bmatrix} \geq 0. \quad (8b)$$

where  $\bar{A}_0 = \beta - KF, \bar{A}_1 = e^{\beta\eta} KFP, \bar{A}_2 = e^{\beta\tau} KB, \bar{A}_3 = -e^{\beta(\tau+\eta)} KBP, l \in \{1, 2, 3\}$ ,

$\Pi = (Q\bar{A}_0 + Y_1 + Y_2 + Y_3)^T + (Q\bar{A}_0 + Y_1 + Y_2 + Y_3) + \eta W_1 + \tau W_2 + (\tau + \eta) W_3 + R_1 + R_2 + R_3$ , \* denotes the transpose of symmetric block in matrix

**Proof:** Consider model (7), according to the definition of exponential stability in [16], transformation is introduced as follows.

$$\Phi(t) = e^{\beta t} X(t). \quad (9)$$

So model (7) is equivalent to

$$\dot{\Phi}(t) = \bar{A}_0 \Phi(t) + \bar{A}_1 \Phi(t - \eta) + \bar{A}_2 \Phi(t - \tau) + \bar{A}_3 \Phi(t - \eta - \tau). \quad (10)$$

And then, by using Newton-Leibnitz formula, system (10) is equivalent to

$$\dot{\Phi}(t) = (\bar{A}_0 + \bar{A}_1 + \bar{A}_2 + \bar{A}_3) \Phi(t) - \bar{A}_1 \int_{t-\eta}^t \dot{\Phi}(\alpha) d\alpha - \bar{A}_2 \int_{t-\tau}^t \dot{\Phi}(\alpha) d\alpha - \bar{A}_3 \int_{t-\eta-\tau}^t \dot{\Phi}(\alpha) d\alpha. \quad (11)$$

Consider system (10), construct Lyapunov-Krasovskii functionals as:

$$\begin{aligned} V(\Phi_t) &= \Phi(t)^T Q \Phi(t) + \int_{t-\eta}^t \Phi(\theta)^T R_1 \Phi(\theta) d\theta + \int_{t-\tau}^t \Phi(\theta)^T R_2 \Phi(\theta) d\theta + \int_{t-\eta-\tau}^t \Phi(\theta)^T R_3 \Phi(\theta) d\theta \\ &+ \int_{t-\eta}^t \int_\alpha^t \dot{\Phi}(\theta)^T Z_1 \dot{\Phi}(\theta) d\theta d\alpha + \int_{t-\tau}^t \int_\alpha^t \dot{\Phi}(\theta)^T Z_2 \dot{\Phi}(\theta) d\theta d\alpha + \int_{t-\eta-\tau}^t \int_\alpha^t \dot{\Phi}(\theta)^T Z_3 \dot{\Phi}(\theta) d\theta d\alpha. \end{aligned} \quad (12)$$

where  $\Phi_t = \Phi(t + \sigma), \sigma \in [-\tau - \eta, 0]$ .

Solve the time derivative of  $V(\Phi_t)$  along the trajectories of system (11) and use Moon Inequality [17], we have

$$\dot{V}(\Phi_t) \leq \xi(t)^T \Omega \xi(t), \begin{bmatrix} W_l & Y_l \\ * & Z_l \end{bmatrix} \geq 0. \quad (13)$$

where  $\xi(t)^T = [\Phi^T(t) \Phi^T(t-\eta) \Phi^T(t-\tau) \Phi^T(t-\eta-\tau)]$ ,  $\bar{\Pi} = (\bar{Q}\bar{A}_0 + Y_1 + Y_2 + Y_3)^T + (Q\bar{A}_0 + Y_1 + Y_2 + Y_3) + \eta W_1 + \tau W_2 + (\tau + \eta)W_3 + R_1 + R_2 + R_3 + \Xi_{00}$ ,  $\Xi_{ab} = \bar{A}_a^T (\eta Z_1 + \tau Z_2 + (\tau + \eta)Z_3) \bar{A}_b$ ,  $a, b \in \{0, 1, 2, 3\}$ .

$$\Omega = \begin{bmatrix} \bar{\Pi} & -Y_1 + Q\bar{A}_1 + \Xi_{01} & -Y_2 + Q\bar{A}_2 + \Xi_{02} & -Y_3 + Q\bar{A}_3 + \Xi_{03} \\ * & -R_1 + \Xi_{11} & \Xi_{12} & \Xi_{13} \\ * & * & -R_2 + \Xi_{22} & \Xi_{23} \\ * & * & * & -R_3 + \Xi_{33} \end{bmatrix},$$

According to Schur Complement theory [18],  $\Omega < 0$  is equivalent to system (8a), thereby,  $\dot{V}(\Phi_t) < 0$ , and the system (7) is exponential stable.

Based on Theorem 1, Corollary 1 is proposed as follows.

**Corollary 1:** Consider DLB system as model (7), the actual optimal LBG can be solved by following optimization problem.

$$\begin{aligned} \max \quad & \beta \\ \text{s.t. } & \Gamma < 0, \Theta \geq 0, K_a = \Delta t \sqrt[n]{|H| / |\bar{Q}|}. \end{aligned} \quad (14)$$

where

$$\Gamma = \begin{bmatrix} \bar{\Pi} & \tilde{A}_1 - \tilde{Y}_1 & \tilde{A}_2 - \tilde{Y}_2 & \tilde{A}_3 - \tilde{Y}_3 & \eta \tilde{A}_0^T & \tau \tilde{A}_0^T & (\eta + \tau) \tilde{A}_1^T & \bar{Q} & \bar{Q} & \bar{Q} & 0 & 0 & 0 \\ * & 0 & 0 & 0 & \eta \tilde{A}_1^T & \tau \tilde{A}_1^T & (\eta + \tau) \tilde{A}_1^T & 0 & 0 & 0 & \bar{Q} & 0 & 0 \\ * & * & 0 & 0 & \eta \tilde{A}_2^T & \tau \tilde{A}_2^T & (\eta + \tau) \tilde{A}_2^T & 0 & 0 & 0 & 0 & \bar{Q} & 0 \\ * & * & * & 0 & \eta \tilde{A}_3^T & \tau \tilde{A}_3^T & (\eta + \tau) \tilde{A}_3^T & 0 & 0 & 0 & 0 & 0 & \bar{Q} \\ * & * & * & * & -\eta \bar{Z}_1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ * & * & * & * & * & -\tau \bar{Z}_2 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ * & * & * & * & * & * & -(\eta + \tau) \bar{Z}_3 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ * & * & * & * & * & * & * & -\bar{R}_1 & 0 & 0 & 0 & 0 & 0 & 0 \\ * & * & * & * & * & * & * & * & -\bar{R}_2 & 0 & 0 & 0 & 0 & 0 \\ * & * & * & * & * & * & * & * & * & -\bar{R}_3 & 0 & 0 & 0 & 0 \\ * & * & * & * & * & * & * & * & * & * & \bar{R}_1 & 0 & 0 & 0 \\ * & * & * & * & * & * & * & * & * & * & * & \bar{R}_2 & 0 & 0 \\ * & * & * & * & * & * & * & * & * & * & * & * & \bar{R}_3 & 0 \end{bmatrix}, \Theta = \begin{bmatrix} \tilde{W}_l & \tilde{Y}_l \\ * & \bar{Z}_l \end{bmatrix},$$

$$\tilde{\Pi} = \tilde{A}_0 + \tilde{A}_0^T + \tilde{Y}_1 + \tilde{Y}_2 + \tilde{Y}_3 + \tilde{Y}_1^T + \tilde{Y}_2^T + \tilde{Y}_3^T + \eta \tilde{W}_1 + \tau \tilde{W}_2 + (\tau + \eta) \tilde{W}_3, \tilde{A}_0 = \beta \bar{Q} - FH, \tilde{A}_1 = e^{\beta t} FPH,$$

$$\tilde{A}_2 = e^{\beta t} BH, \tilde{A}_3 = -e^{\beta(\tau+\eta)} BPH, \bar{Q} = Q^{-1}, \bar{R}_1 = R_1^{-1}, \bar{Z}_1 = Z_1^{-1}, \tilde{W}_l = Q^{-1} W_l Q^{-1}, \tilde{Y}_l = Q^{-1} Y_l Q^{-1}, H = K \bar{Q},$$

and  $| \cdot |$  denotes the determinant of a matrix.

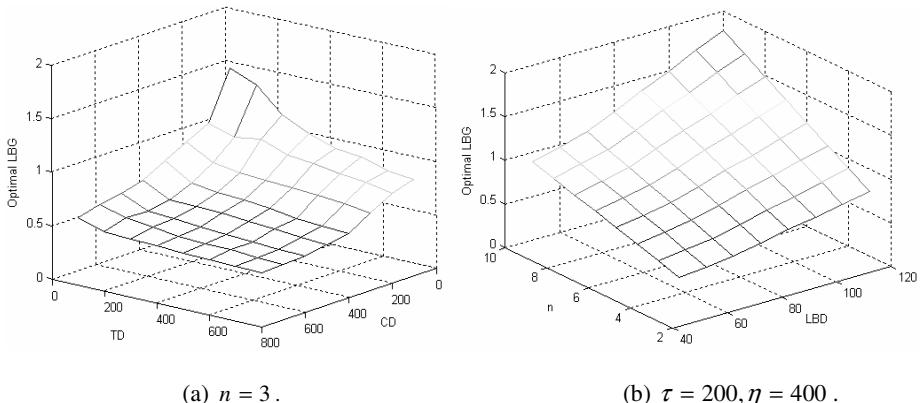
**Proof:** Select diagonal matrix  $\text{diag}\{Q^{-1}, Q^{-1}, Q^{-1}, Q^{-1}, Z_1^{-1}, Z_2^{-1}, Z_3^{-1}\}$  and  $\text{diag}\{Q^{-1}, Q^{-1}\}$  to left, right multiply (8a) and (8b) respectively. By using Schur Complement [18] and let  $\bar{Q} = \bar{Z}_l$ ,  $\Gamma < 0$  and  $\Theta \geq 0$  are obtained. And then, take determinant to both sides of the equation  $H = K \bar{Q}$ , we have  $K = \sqrt[n]{|H| / |\bar{Q}|}$ . At last, according to Definition 1 and the meaning of exponential stability [16], we can easily obtain Corollary 1.

## 4 Experimental Results

In this section, the numerical and simulative experiments mainly focus on the DEC-oriented DLB system, and based on the initial condition in [8], this paper selects the condition as:  $\Delta t=98\mu s$  (The mean of tests value in [8]),  $\lambda_1 = 0.250$ ,  $\lambda_2 = 0.500$ ,  $\lambda_3 = 1.000$ ,  $\lambda_4 = 0.167$ ,  $\lambda_5 = 0.333$ ,  $\lambda_6 = 0.200$ ,  $\lambda_7 = 0.125$ ,  $\lambda_8 = 0.143$ ,  $\lambda_9 = 0.333$ ,  $\lambda_{10} = 0.111$ . In which, the unit of  $\lambda_i$  is *Tasks Per  $\mu s$*  and we assume each task holds an approaching processing time in the DLB system. If not indicate distinguishingly, the experiments will choose the above-mentioned conditions acquiescently.

### 4.1 Theoretical Computation via LMI

By using LMI (Linear Matrix Inequality) tools, the theoretical results are solved according to Corollary 1, and the distribution of theoretical optimal LBG is illustrated as Fig. 1. We can easily obtain that the optimal LBG is proportional to  $n$  and LBD as well as inverse ratio to CD and TD.



**Fig. 1.** Distribution of theoretical optimal LBG in different conditions (Time unit:  $\mu s$ )

### 4.2 A Efficient Simulation Method Based on Multi-threads

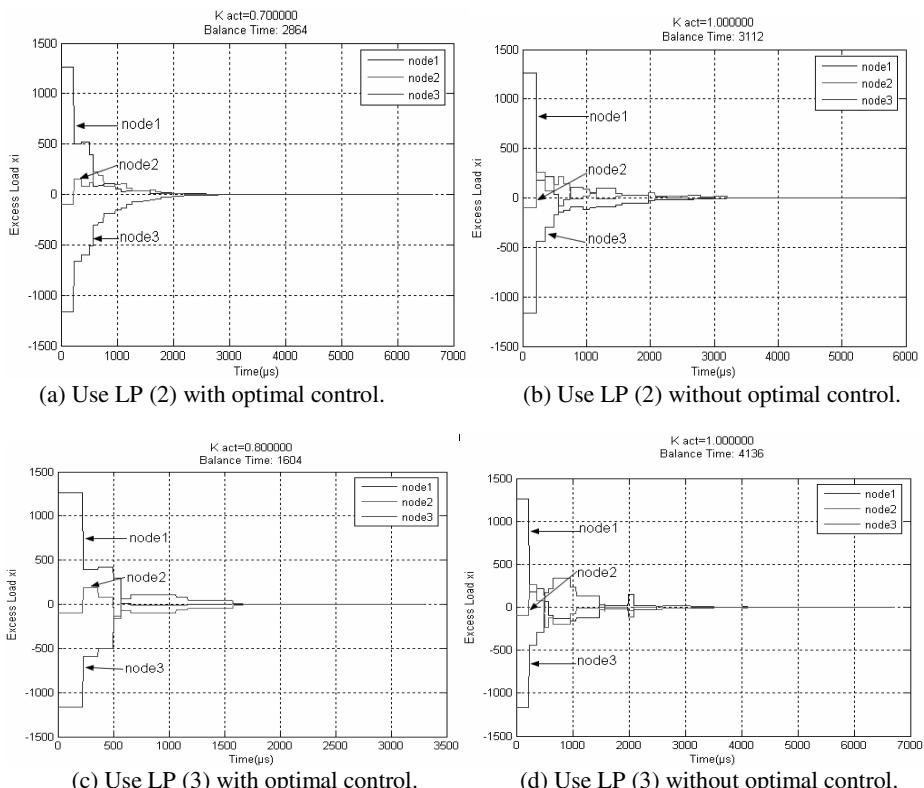
In order to describe the effects of time delay more clearly and efficiently, this section introduces a DLB simulation method based on multi-threads. This method mainly focuses on the impact of the time delay and neglects other factors such as the network protocol, topology, bandwidth and so on, thus, simplifies the simulation work of the DLB system. In fact, many kinds of impacting factors in DLB system can be described by the time delay, and the definition of the time delay has more generalized meanings in this paper. The specific simulation process according to model (1) can be described by following steps.

1. Creating  $n$  threads to simulate  $n$  CEs. Every CE thread maintains a table which contains the load information not only about itself but also about other CE threads.

2. Creating a global statistic thread which holds the highest priority to record the instant load information and make a change of generated and processed loads in each CE thread.
3. In each CE thread, the information about other CEs can be updated by a communication thread with a certain CD cycle.
4. The CE threads satisfied the conditions execute DLB algorithm with a certain LBD cycle according to their local information about others.
5. The CE threads executed the DLB algorithm update the load information of their own instantly and make a change of this transmission to others after a certain TD.

The simulation procedure is implemented in the win32 multi-threads environment, and time delays like CD, TD and LBD are simulated by the sleep function in OS. Note that in the multi-threads surrounding, some uncertain factors may appear in the process of context switch, but all these factors do not prevent us from revealing the effects of time delay in the DLB system. Moreover, the simulation time depends on the execution cycle of the statistic thread, so it is only a relative value in the DLB system.

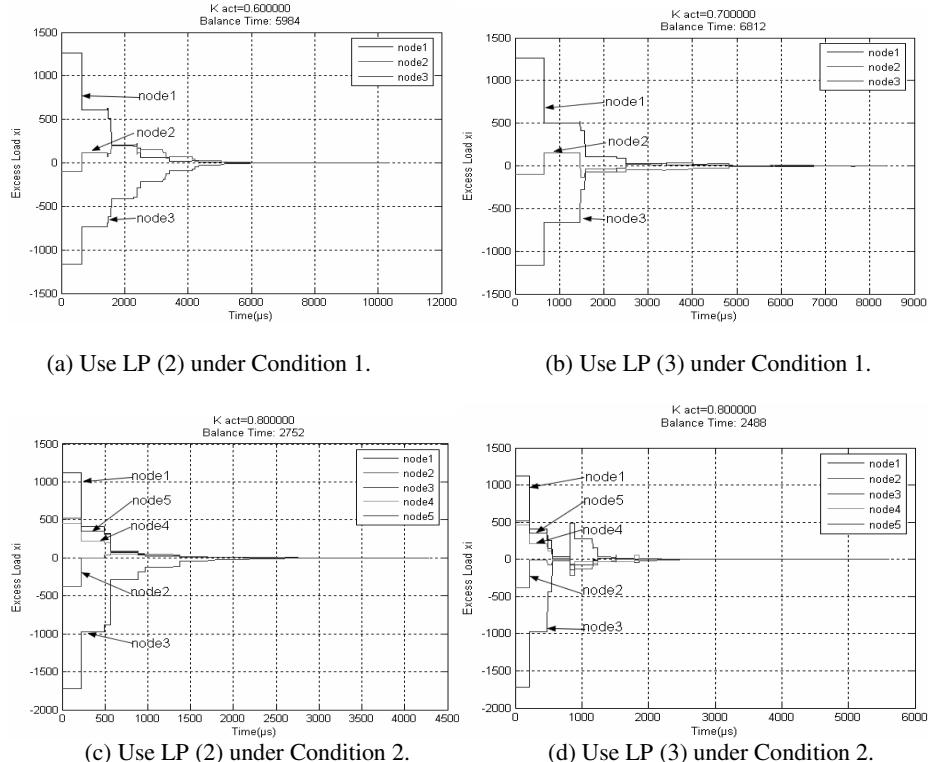
Based on the proposed simulation method, some experimental results are illustrated as follows.



**Fig. 2.** Simulation results when  $n = 3, \tau = 200\mu s, \eta = 400\mu s$

It can be seen in Fig.2, the impacts of optimal control respond well, and if all the excess loads migrated (i.e.  $K_a = 1.0$ ), there will be the excessive oscillatory phenomenon which intensely influences the performance of the DLB algorithm. Moreover, we can also see that the control effect under LP (3) is better than LP (2), that is because LP (3) uses more detailed and deterministic information in DLB system.

Note that the optimal  $K_a$  is obtained by selecting  $K_a = 0.1, 0.2, \dots, 1.0$ , so it is only an approximate value due to the coarse precision. Fig. 3 describes the optimal  $K_a$  obtain by using this method in different conditions.

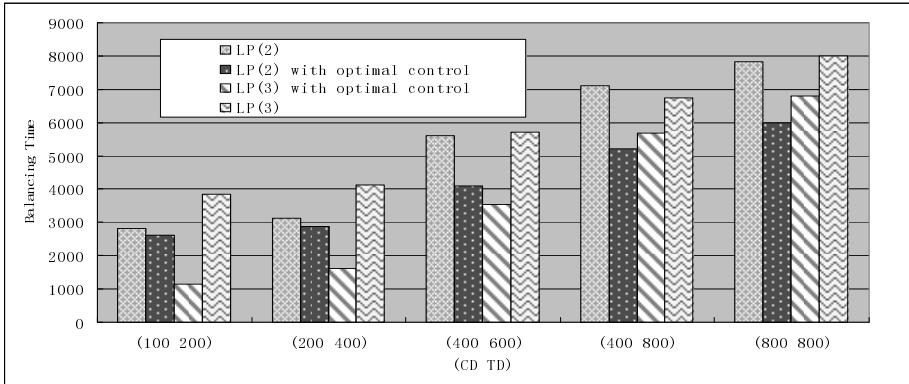


**Fig. 3.** The simulative optimal  $K_a$  in Condition 1 and 2 (Condition 1:  $n=3, \tau=800\mu s, \eta=800\mu s$ . Condition 2:  $n=5, \tau=200\mu s, \eta=400\mu s$ ).

### 4.3 Comparison

According to the simulation results, the specific simulation balancing time under different CD, TD and LP when  $n=3$  can be found in Fig. 4.

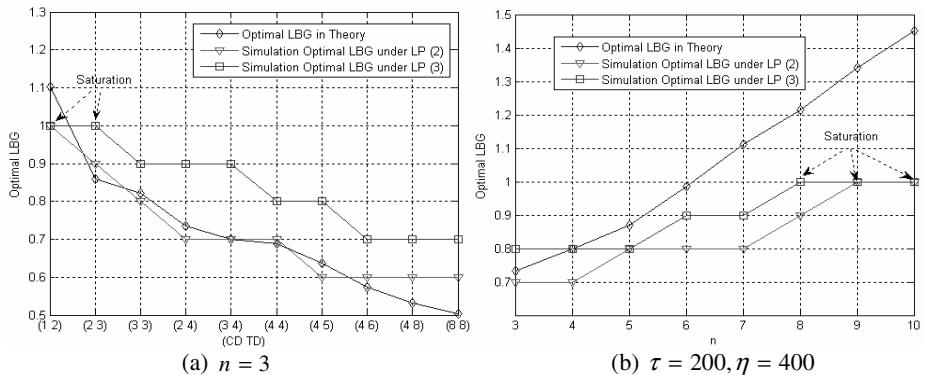
From Fig. 4, we can see that, when time delay is small, the effect of optimal control with LP (3) is better, but following the increasing time delay, the case is contrary. The reason lead to this phenomenon is that, compared with LP (2), LP (3) needs more deterministic information to keep its performance, in other words, the larger time delays bring more uncertain factors and thus influenced the effect of LP(3).



**Fig. 4.** Comparison of balancing time under LP (2) and (3) in different CD and TD when  $n=3$  (Time unit:  $\mu s$ )

And then, the comparison between theoretical and simulative results is described in Fig. 5.

From Fig.5, we can see that the change law and value of optimal  $K_a$  is close to the theoretical results, but following the increasing system scale, the estimated effects begin to get worse, and there exists an upper limit of saturation for optimal  $K_a$ . The reason lead to this phenomenon is that the saturated nonlinear factor which exists in model (1) is neglected during analytical work. In fact, if  $K_a > 1$ , that is equivalent to executing DLB algorithm in CEs where  $x_i < 0$ , this is conflict with the saturated operation as  $[x_i]_{Domain}^+$  and the LBG will lose effectiveness this time.



**Fig. 5.** Comparison between theoretical and simulative results (Time unit: hundred  $\mu s$ )

At last, the comparison results with other papers can be seen in Table 1.

Table 1 indicated the larger use range of the introduced theory and method, and compare with other simulation methods, the proposed method based on Multi-Threads can choose the time delay flexibly, support the scalable system scale conveniently and reflected the effect of LBD well, and thus reduces the simulation work intensely.

**Table 1.** Comparison results with other papers(S: Supported, NS: Not Supported, and “--” denotes that case is not mentioned )

	This paper	[7]	[8,9]	[10]	[13]
Heterogeneous Resource	S	NS	S	NS	S
Scalable System Scale	S	NS	--	S	NS
Time Delay Consideration	$\forall \eta, \tau$	$\eta = 2\tau$	--	$\forall \eta, \tau$	Stochastic
Validating Method	Multi-Threads	Simulink	Simulink	--	MC

## 5 Conclusion

Aiming at the DLB problem in parallel computing, a time delay model is proposed in a control theory view. Based on the analytical and simulative results, discussions focus on the effects of time delay in the DLB system, and the optimal control law is obtained in the different time delay, LP condition and system scale later. The compared experiments indicate that the time delay has important impacts on the performance of the DLB algorithm, and the theory and method proposed in this paper holds guiding effects on designing a more useful DLB algorithm in a delay surrounding.

Certainly, this paper also has some deficiencies, such as bad estimated effects for larger scale system and the NCE-oriented condition is not considered well etc. Corresponding improvements will be done in our future work.

## References

1. Barker, K., Chernikov, A., Chrisochoides, N.: A Load Balancing Framework for Adaptive and Asynchronous Applications. *IEEE Transaction on Parallel and Distributed Systems* 15(2), 183–192 (2004)
2. Rotaru, T., Nageli, H.: Dynamic Load balancing by Diffusion in Heterogeneous Systems. *Journal of Parallel and Distributed Computing* 64(4), 481–497 (2004)
3. Ghose, D., Kim, H.J., Kim, T.H.: Adaptive Divisible Load Scheduling Strategies for Workstation Clusters with Unknown Network Resources. *IEEE Transaction on Parallel and Distributed Systems* 16(10), 897–907 (2005)
4. Jeannot, E., Vernier, F.: A Practical Approach of Diffusion Load Balancing Algorithms. In: Nagel, W.E., Walter, W.V., Lehner, W. (eds.) *Euro-Par 2006*. LNCS, vol. 4128, pp. 211–221. Springer, Heidelberg (2006)
5. Rie, C., Wanka, R.: Periodic Load Balancing on the N-Cycle: Analytical and Experimental Evaluation. In: Kermarrec, A.-M., Boug  , L., Priol, T. (eds.) *Euro-Par 2007*. LNCS, vol. 4641, pp. 805–814. Springer, Heidelberg (2007)
6. Catalyurek, U.V., Boman, E.G., Devine, K.D.: A repartitioning hypergraph model for dynamic load balancing. *Journal of Parallel and Distributed Computing* 69(8), 711–724 (2009)
7. Abdallah, C.T., Alluri, N., Birdwell, J.D.: Linear Time Delay Model for Studying Load Balancing Instabilities in Parallel Computations. *International Journal of Systems Science* 34(10), 563–573 (2003)

8. Chiasson, J., Tang, Z., Ghanem, J.: The Effect of Time Delays on the Stability of Load Balancing Algorithms for Parallel Computations. *IEEE Transactions on Control Systems Technology* 13(6), 932–942 (2005)
9. Tang, Z., Birdwell, J.D., Chiasson, J.: Resource-Constrained Load Balancing Controller for a Parallel Database. *IEEE Transactions on Control Systems Technology* 16(4), 834–840 (2008)
10. Chiasson, J., Loiseau, J.J.: Application of Time Delay Systems. Springer, Heidelberg (2007)
11. Argonne National Lab.: MPICH-A Portable Implementation of MPI,  
<http://www.mcs.anl.gov/research/projects/mpi/mpich2>
12. Oak Ridge National Lab.: PVM (Parallel Virtual Machine),  
<http://www.csm.ornl.gov/pvm>
13. Dhakal, S., Hayat, M.M., Pezoa, J.E.: Dynamic Load Balancing in Distributed Systems in the Presence of Delays: A Regeneration-Theory Approach. *IEEE Transactions on Parallel and Distributed Systems* 18(4), 485–497 (2007)
14. Callier, F.M., Desoer, C.A.: Linear System Theory. Springer, Heidelberg (1991)
15. Carson, M., Santay, D.: NIST Net – A Linux-based Network Emulation Tool. *ACM SIGCOMM Computer Communication Review* 33(3), 111–126 (2003)
16. Liao, X., Chen, G., Sanchez, E.N.: Delay-dependent Exponential Stability Analysis of Delayed Neural Networks: an LMI Approach. *Neural Networks* 15, 855–866 (2002)
17. Moon, Y.S., Park, P., Kwon, W.H., Lee, Y.S.: Delay-dependent robust stabilization of uncertain state-delayed systems. *Int. J. of Control.* 74, 1447–1455 (2001)
18. Boyd, S., Ghaoui, L., Feron, E.I.: Linear matrix inequalities in system and control theory. SIAM, Philadelphia (1994)