We are IntechOpen, the world's leading publisher of Open Access books Built by scientists, for scientists

4

Predictive Control Applied to Networked Control Systems

Xunhe Yin[1,2], Shunli Zhao[1], Qingquan Cui[1,3] and Hong Zhang[4]
[1]*School of Electric and Information Engineering, Beijing Jiaotong University,*
[2]*School of Electrical and Information Engineering,*
University of Sydney, Sydney,
[3]*Yunnan Land and Resources Vocational College,*
Kunming,
[4]*Beijing Municipal Engineering Professional Design*
Institute Co.Ltd, Beijing,
[1,3,4]*China*
[2]*Australia*

1. Introduction

The researches of the networked control systems (NCSs) cover a broader, more complex technology, because that networked control systems relate to computer network, communication, control, and other interdisciplinary fields. Networked control systems have become one of the hot spots of international control areas in recent years. The networked control system theoretical research is far behind its application, so the networked control system theory study has important academic value and economic benefits at present.

NCSs performance is not only related with the control algorithms, but also the network environment and the scheduling algorithms. The purpose of network scheduling is to avoid network conflicts and congestion, accordingly reducing the network-induced delay, packet loss rate and so on, which can ensure the better network environment. If the case, where the data cannot be scheduled, appears in the network, the control algorithm has not fundamentally improved the performance of the system, thus only adjusting data transmission priorities and instants over the network by using the scheduling algorithms, in order to make the whole system to achieve the desired performance.

Along with the networked control system further research, people gradually realized that the scheduling performance must be taken into account when they research control algorithms, that is, considering the two aspects of scheduling and control synthetically. The joint design of both scheduling performance and control performance is concerned by the majority of researchers (Gaid M B et al., 2006a,2006b; Arzen K E et al., 2000). Therefore, NCSs resource scheduling algorithms, as well as scheduling and control co-design are the main research directions and research focus.

The generalized predictive control and the EDF (Earliest Deadline First) scheduling algorithm are adopted by the NCSs co-design in this chapter. The co-design method considers both the NCSs scheduling performance and control performance, and then the process of the general co-design method is also given. From the TrueTime simulation results based on NCSs with three loops of DC-motors, NCSs under co-design compared with NCSs without co-design, we can find that the former shows better control performance and scheduling performance, and a better anti-jamming ability and adaptive ability for network, so that the NCSs with co-design can guarantee to operate in an optimal state.

2. Brief review of Generalized Predictive Control

GPC (Generalized Predictive Control) algorithm is proposed by Clarke et al (Calrke & Mohtadi, 1989) in the 80s of last century, as a new class of predictive control algorithm. The algorithm is based on Controlled Auto-Regressive Integrated Moving Average (CARIMA) model, adopts an optimization of the long time indicators combined with the identification and self-correcting mechanism, shows strong robustness and has broad scope of application. The significance of GPC algorithm is that the algorithm can still get sub-optimal solution when mismatch or time-varying occurs in the controlled plant model, so it has strong robustness, but also can eliminate the static error of the system with using CARIMA model., The generalized predictive control, which is optimized control algorithms based on the prediction model, rolling optimization and online feedback correction, have distinct characteristics as a new type of control algorithms. (Wang et al., 1998; Guan & Zhou, 2008; Ding, 2008).

2.1 Prediction model

Refer to the generalized predictive control; the controlled plant is usually represented by the model of CARIMA:

$$Ay(k) = Bu(k-1) + C\frac{\xi(k)}{\Delta} \tag{1}$$

where $u(k)$ and $y(k)$ are control input and system output respectively, $\xi(k)$ is a white noise with zero mean and standard deviation σ^2, $\Delta = 1 - z^{-1}$ is a difference operator, $A = 1 + a_1 z^{-1} + \cdots + a_n z^{-n}$, $B = b_1 z^{-1} + \cdots + b_n z^{-n}$, $C = 1 + c_1 z^{-1} + \cdots + c_n z^{-n}$.

To simplify the inference process of the principle, without loss of generality, let C=1. To derive the optimization prediction value of $\boldsymbol{y(k+j)}$ after j steps, the Diophantine equation is considered firstly:

$$I = E_j(z^{-1})A(z^{-1})\Delta + z^{-j}F_j(z^{-1}) \tag{2}$$

where $E_j(z^{-1}) = e_{j,0} + e_{j,1}z^{-1} + \cdots + e_{j,j-1}z^{-j+1}$, $F_j(z^{-1}) = f_{j,0} + f_{j,1}z^{-1} + \cdots + f_{j,n}z^{-n}$ ，they are multinomial which are decided by the model parameter A and prediction length j, $e_{j,0}\cdots e_{j,j-1}$ and $f_{j,0}\cdots f_{j,j-1}$ are coefficients.

Using $E_j\Delta z^j$ to multiply both sides of (1), then combining (2), $y(k+j)$ is derived:

$$y(k+j) = E_jB\Delta u(k+j-1) + F_jy(k) + E_j\xi(k+j) \tag{3}$$

By the expressions E_j, can see that E_j ξ $(k+j)$ is an unknown noise starting from instant k^{th}, the output prediction value of the futurity j steps starting from instant k^{th} are derived after deleting the term $E_j\xi(k+j)$:

$$\hat{y}(k+j) = E_j B \Delta u(k+j-1) + F_j y(k) \tag{4}$$

Let $G_j = E_j B$ ，and $j = 1,2\cdots,N$ ，(4) can be written as matrix equation (5):

$$\hat{\boldsymbol{y}} = \boldsymbol{G\Delta u} + \boldsymbol{f} \tag{5}$$

where $\hat{\boldsymbol{y}} = [y(k+1)\ y(k+2)\ ...y(k+N)]$, $\boldsymbol{\Delta u} = [\Delta u(k)\ \Delta u(k+1)\ ...\Delta u(k+M-1)]$, N is the model time domain while M is the control time domain, $\boldsymbol{f} = [f_1(k)\ f_2(k)\ ...f_N(k)]^T$, $f_j(k) = z^{n-1}[\boldsymbol{G}_n - z^{1-n} g_{n,n-1} - \cdots - g_{n,0}]\Delta u(k) + F_j y(k), n = 1,2,\cdots N$,

$$\boldsymbol{G} = \begin{bmatrix} g_1 & 0 & \cdots & 0 \\ g_2 & g_1 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ g_p & g_{p-1} & \cdots & g_1 \\ \vdots & \vdots & \ddots & \vdots \\ g_N & g_{N-1} & \cdots & g_{N-M+1} \end{bmatrix}_{N\times M}$$

2.2 Rolling optimization

To enhance the robustness of the system, the quadratic performance index with output error and control increment weighting factors are adopted:

$$J = \sum_{j=N_0}^{P} [y(k+j) - y_r(k+j)]^2 + \sum_{j=1}^{M} [\lambda(j)\Delta u(k+j-1)]^2 \tag{6}$$

where N_0 is the minimum prediction horizon, and $N_0 \geq 1$, P is the maximum prediction horizon, M is the control horizon, that means the control value will not be changed after M steps, $\lambda(j)$, which is a constant λ in the general control systems, is the control increment weighting factor, but it will be adjusted in real time within the control process in the co-design of control and scheduling to ensure optimal control.

The optimal control law is as follow:

$$\boldsymbol{\Delta u}(k) = (\boldsymbol{G}^T \boldsymbol{G} + \lambda \boldsymbol{I})^{-1} \boldsymbol{G}^T [\boldsymbol{y}_r(k+1) - \boldsymbol{f}] \tag{7}$$

Then the incremental series of open loop control from instant k^{th} to instant ($k+M-1)^{th}$ is derived after expanding the formula (7):

$$\boldsymbol{\Delta u}(k+i-1) = \boldsymbol{d}_i^T [\boldsymbol{y}_r(k+1) - \boldsymbol{f}] \tag{8}$$

where $\boldsymbol{d}_i^T$ is the i^{th} increment of $(\boldsymbol{G}^T \boldsymbol{G} + \lambda \boldsymbol{I})^{-1} \boldsymbol{G}^T$, $\boldsymbol{d}_i^T = [d_{i1}\ d_{i1}\ \cdots d_{iP}]$.

In the real control systems, the first control variable will be used in every period. If the control increment $\boldsymbol{\Delta u}(k)$ of the current instant k^{th} is executed, the control increment after k^{th} will be recalculated in every period, that is equivalent to achieve a closed loop control strategy, then the first raw of $(\boldsymbol{G}^T \boldsymbol{G} + \lambda \boldsymbol{I})^{-1} \boldsymbol{G}^T$ is only necessary to recalculate. So the actual control action is denoted as (9):

$$\boldsymbol{u}(k) = \boldsymbol{u}(k-1) + \boldsymbol{d}_l{}^T[\boldsymbol{y}_r(k+1) - \boldsymbol{f}] \tag{9}$$

2.3 Feedback correction

To overcome the random disturbance, model error and slow time-varying effects, GPC maintains the principle of self-correction which is called the generalized correction, by constantly measuring the actual input and output, estimates the prediction model parameters on-line. Then the control law is corrected.

The plant model can be written as:

$$A\Delta y(k) = B\Delta u(k-1) + \xi(k)$$

Then we can attain $$\Delta y(k) = -(A-1)\Delta y(k) + B\Delta u(k-1) + \xi(k) \tag{10}$$

Model parameters and data parameters are expressed using vector respectively

$$\boldsymbol{\theta} = [a_1 \cdots a_n \vdots b_0 \cdots b_m] \tag{11}$$

$$\boldsymbol{\varphi} = [-\Delta y(k-1) \cdots - \Delta y(k-n) \vdots - \Delta u(k-1) \cdots - \Delta u(k-m+1)] \tag{12}$$

Then the above equation (10) can be written into the following form:

$$\Delta y(k) = \boldsymbol{\varphi}^T(k)\boldsymbol{\theta} + \xi(k) \tag{13}$$

The model parameters can be estimated by recursive least squares method with forgetting factor. The parameters of polynomial A, B are obtained by identification. $\boldsymbol{d}_l^T$ and $\boldsymbol{f}$ in control law of equation (9) can be recalculated, and that the optimal control $\boldsymbol{u}(k)$ is found.

2.4 Generalized predictive control performance parameters

Generalized predictive control performance parameters (Ding, 2008; Li, 2009) contain minimum prediction horizon N_0, maximum prediction horizon *P*, control horizon *M*, and control weighting factor λ.

1. Minimum prediction horizon N_0

When the plant delay *d* is known, then take $N_0 \geq d$. If $N_0 \geq d$, there are some output of $y(k+1),\cdots,\ y(k+P)$ without the impact from input $u(k)$, this will waste some computation time. When *d* is unknown or varying, generally let N_0 =1, that means the delay may be included in the polynomial $B(z^{-1})$.

2. Maximum prediction horizon *P*

In order to make the rolling optimization meaningfully, *P* should include the actual dynamical part of the plant. Generally to take *P* close to the rise time of the system, or to take *P* greater than the order of $B(z^{-1})$. In practice, it is recommended to use a larger *P*, and make it more than the delay part of the impulse response of the plant or the reverse part caused by the non-minimum phase, and covers the main dynamic response of the plant. The size of *P* has a great effect on the stability and rapidity of the system. If *P* is small, the dynamic performance is good, but with poor stability and robustness. If *P* is big, the robustness is good, but the dynamic performance is bad, so that system's real-time performance is reduced because of increasing of computing time. In the actual application, we can choose the one between the two values previously mentioned to make the closed-loop system not only with the desired robustness but also the required dynamic performance (rapidity) (Ding, 2008).

3. Control horizon *M*

This is an important parameter. Must $M \le P$, because that the optimal prediction output is affected by *P* control increment values at best. Generally, the *M* is smaller, the tracking performance is worse. To improve the tracking performance, increasing the control steps to improve the control ability for the system, but with the increase of *M*, the control sensitivity is improved while the stability and robustness is degraded. And when *M* increases, the dimension of the matrix and the calculation amount is increased; the real-time performance of the system is decreased, so *M* should be selected taking into account the rapidity and stability.

4. Control weighting factor λ

The effect of the control weighting factor is to limit the drastic change of the control increment, to reduce the large fluctuation to the controlled plant. The control stability is achieved by increasing λ while the control action is weakened (Li, 2009). To select small number λ generally, firstly let λ is 0 or a smaller number in practice. If the control system is steady but the control increment changes drastically, then can increase λ appropriately until the satisfactory control result is achieved.

3. EDF scheduling algorithm and network performance parameters

3.1 EDF scheduling algorithm

EDF scheduling algorithm is based on the length of the task assigned from deadline for the priority of the task: the task is nearer from the required deadline and will obtain the higher priority. EDF scheduling algorithm is a dynamic scheduling algorithm, the priority of the task is not fixed, but changes over time; that is, the priority of the task is uncertain. EDF scheduling algorithm also has the following advantages except the advantages of the general dynamic scheduling algorithm:

1. can effectively utilize the network bandwidth resources, and improve bandwidth utilization;
2. can effectively analyze schedulability of information that will be scheduled;
3. is relatively simple to achieve it, and the executed instructions is lessr in the nodes.

For *N* mutual independent real-time periodic tasks, when the EDF algorithm is used, the schedulability condition is that the total utilization of the tasks meets the following inequality:

$$U = \sum_{i=1}^{N} \frac{c_i}{T_i} \le 1 \tag{14}$$

where c_i is the task execution time, T_i is the task period. In NCSs, c_i is the data packet the sampling time, T_i is the data sampling period.

EDF scheduling algorithm can achieve high utilization from the point of resource utilization, and meet the conditions for more information needs under the same condition of resource, thus it will increase the utilization of resources. Furthermore, EDF is a dynamic scheduling algorithm, and it can dynamically adjust the priority of the message, and lets the limited resources make a more rational allocation under the case of heavy load of information, and makes some soft real-time scheduling system can achieve the desired performance under the condition of non-scheduling.

Suppose there are two concurrent real-time periodic tasks need to be addressed, the execution time of the two messages is 5ms, and the sampling periods are 8ms and 10ms respectively, and suppose the deadline for all information equal to their sampling period. The total utilization of the information is:

$$U = \frac{5}{8} + \frac{5}{10} = 1.125 > 1$$

By the schedulability conditions (14) of EDF, we know that EDF scheduling algorithm is not scheduled; in this case, co-design of scheduling and control is potential to research and solve this type of problem.

3.2 Network performance parameters

Network performance parameters include: network-induced delay, network bandwidth, network utilization, packet transmission time. The EDF scheduling algorithm is also related to the sampling period, priority, and deadline. The greater the network-induced delay is, the poorer is the network environment; data transmission queue and the latency are longer, whereas the contrary is the shorter. The network bandwidth is that the amount of information flows from one end to the other within the specified time, is the same as the data transfer rate, and network bandwidth is an important indicator for the measure of network usage. The network bandwidth is limited in a general way. When the data transmitted per unit time is greater than the amount of information of network bandwidth, network congestion will occur and network-induced delay is larger, thus impacting on the data in real time. The sampling period is an important parameter of network scheduling, but also associated to control performance of the system; the specific content will be described in the next section.

4. Co-design optimization method

4.1 Relationship between sampling period and control performances

In networked control system, which is a special class of digital control system, the feedback signal received by the controller is still periodic sampling data obtained from sensor, but these data transmitted over the network, rather than the point to point connection. The network can only be occupied by a task in certain instant, because that network resources are shared by multiple tasks; in other words, when one task is over the network, the other ones will wait until the network is free. In this case, the feedback signal sampling period and the required instant of feedback signal over network will jointly determine control system performance.

Although the controller requires sampling period as small as possible for getting feedback signal more timely, the smaller sampling period means the more times frequently need to send data in network, so that the conflict occurs easily between tasks, data transmission time will increase in the network, and even the loss of data may occur.

However, sampling period cannot too large in the network, because that larger sampling period can decrease the transmission time of the feedback signal in the network, but will not fully utilize network resources. Therefore, the appropriate sampling period must be selected in the practical design in order to meet both the control requirements and the data transmission stability in the network, and finding the best tradeoff point of sampling period to use of network resources as full as possible, thereby enhancing the control system performance (Li, 2009).

Fig.1 shows the relationship between the sampling period and control performance (Li et al., 2001), it clearly illustrates the effect of sampling period on continuous control system, digital control system and networked control system, the meanings of T_A, T_B and T_C are also defined.

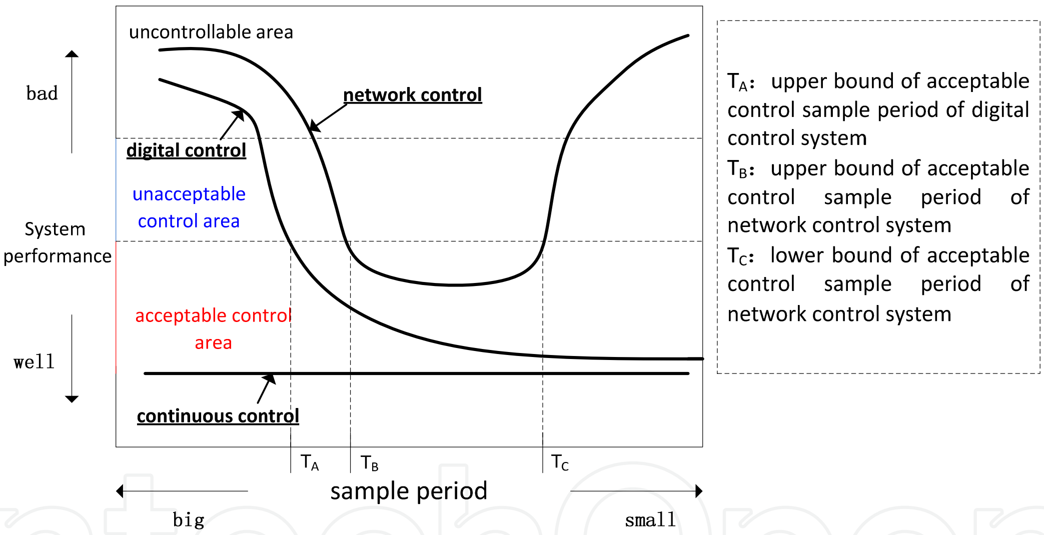
uncontrollable area
bad
network control
digital control
unacceptable control area
System performance
acceptable control area
well
continuous control
T_A
T_B
T_C
sample period
big
small
T_A: upper bound of acceptable control sample period of digital control system
T_B: upper bound of acceptable control sample period of network control system
T_C: lower bound of acceptable control sample period of network control system

Fig. 1. The impact of Sampling period on control system performance

By analyzing the impact of sampling period for the control system performance, we see that changing the sampling period is very important to the networked control system performance. According to the different requirements for loops of NCSs, it has great significance for improving the system performance by changing the network utility rate of each loop and further changing the sampling period of each loop.

4.2 Joint optimization of the sampling periods

In NCSs, sampling period has effect on both control and scheduling, the selection of sampling period in NCSs is different from the general computer control system. Considering both the control performance and network scheduling performance indicators to optimize the sampling period of NCSs is the main way to achieve the co-design of control and scheduling (Zhang & Kong, 2008).

In NCSs, in order to ensure the control performance of the plant, generally the smaller sampling period is needed, but the decreased sampling period can lead the increased transmission frequency of the packets, and increase the burden of the network scheduling, therefore, control and scheduling are contradictory for the requirements of sampling period. The sampling periods of sensors on each network node not only bound by the stability of the plant but also the network schedulability. The way to solve this problem is to compromise the control performance and scheduling performance under certain of constraint conditions, and then to achieve the overall optimal performance of NCSs (Guan & Zhou, 2008; Zhang & Kong, 2008).

1. The selection of the objective function

Sampling period is too large or too small can cause deterioration of the system output performance, therefore, to determine the optimal sampling period is very important for the co-design of control and scheduling in NCSs. From the perspective of control performance, the smaller the sampling period of NCSs is, the better is its performance; from the perspective of scheduling performance, it will have to limit the decrease of the sampling period due to network communication bandwidth limitations. Optimization problem of the sampling period can be attributed to obtain the minimum summation of each control loop performance index function (objective function) under the conditions that the network is scheduling and the system is stable.

Suppose the networked control system optimal objective function is $J_{\min}$, then

$$J_{\min} = \sum_{i=1}^{N} p_i J_i \tag{15}$$

where p_i is weight, the greater the priority weight value of the network system is, the more priority is the data transmission . J_i is the performance index function of loop *i*, *N* is the total number of control loops.

2. Scheduling constraints

In order to make control information of networked control system transmit over the network effectively, meet the real-time requirements of period and control tasks, network resources allocation and scheduling are necessary. It ensures the information of control tasks to complete the transfer within a certain period of time to ensure the timeliness of the data and improve the network utilization. In this chapter, single packet transmission of information is analyzed, and the scheduling is non-priority.

Different scheduling algorithms correspond to the different schedulability and sampling period constraints. Currently, the commonly used network scheduling algorithms are: static scheduling algorithm, dynamic scheduling, mixed scheduling algorithm, and so on.

For static scheduling algorithm, such as *RM* algorithm, the following scheduling constraints can be chosen (Guan & Zhou, 2008):

$$\frac{c_1}{T_1}+\frac{c_2}{T_2}+...+\frac{c_i}{T_i}+\frac{\bar{b}_{l,i}}{T_i}\leq i(2^{\frac{1}{i}}-1) \tag{16}$$

where T_i, c_i and $\bar{b}_{l,i}$ are the sampling period, transmission time and congestion time of i^{th} control loop respectively. $\bar{b}_{l,i}=\max\limits_{j=i+1,...,N} c_j$ is the congestion time of the worst time which means the current task is blocked by the low priority task.

For dynamic scheduling, such as EDF algorithm, the following scheduling constraints can be chosen (Pedreiras P & Almenida L, 2002):

$$U=\sum_{i=1}^{N}\frac{c_i}{T_i}\leq 1 \tag{17}$$

T_i, c_i are the sampling period and the data packet transmission time of i^{th} control loop respectively.

3. Stability conditions of the system

The upper limit of the sampling period of networked control systems with delay (Mayne et al.,2003) is:

$$\boldsymbol{T_{max}}=\frac{\boldsymbol{T_{bw}}}{20}-2\tau_{\boldsymbol{i}} \tag{18}$$

where T_{max} is the maximum value of the sampling period, ω_{bw} is the system bandwidth, T_{bw} is derived by ω_{bw}, τ_i is the network induce delay of loop i .

EDF scheduling algorithm is used in this chapter, the optimization process of the compromised sampling period of overall performance of the NCSs can be viewed as an optimization problem.

Objective function：

$$J_{\min}=\sum_{i=1}^{N}p_iJ_i$$

Constraint condition:

$$\boldsymbol{T_{max}}=\frac{\boldsymbol{T_{bw}}}{20}-2\tau_{\boldsymbol{i}}$$

$$U=\sum_{i=1}^{N}\frac{c_i}{T_i}\leq 1$$

The constraints of network performance and control performance are added in the problem above simultaneously. They ensure the system to run on a good performance under a certain extent.

However, the optimal design method takes into account the relatively simple elements of the networked control system, and the involved performance parameters are less. So adding more network scheduling parameters and system control parameters is necessary to optimize the design jointly. An optimization method of taking both scheduling performance and control performance is proposed for system optimization operation. The core idea of the proposed methods is to make the interaction between the two performance indicators of networked control system---network scheduling performance and control performance, which affect on the system stable and efficient operation, so as to ensure network performance and control performance in NCSs.

4.3 Joint optimization of predictive control parameters

The preferences of GPC can be considered from two aspects. For general process control, let N_0=1, *P* is the rise time of the plant, *M =1*, then the better control performance is achieved. For the higher performance requirements of the plant, such as the plant in NCS, needs a bigger *P* based on the actual environment. A large number of computer simulation studies (Mayne et al., 2003; Hu et al., 2000; Chen et al., 2003)have shown that *P* and λ are the two important parameters affecting GPC control performance. When *P* increases, the same as λ, the smaller λ and the bigger *P* will affect the stability of the close loop system. The increase of the two parameters λ and *P* will slow down the system response speed, on the contrary, *P* less than a certain value will result in the system overshoot and oscillation.

When network induce delay $\tau_i < T$ (T is the sampling period), based on the above analysis of control and network parameters affecting on NCSs performance, network environment parameters will be considered in the follows: network induce delay, network utilization and data packet transmission time. The optimal rules of prediction control parameters are determined by the following three equations of loop i :

$$M_i(k+1) = M_i(k) + [(\frac{\Delta\tau_i}{\tau_i} + \frac{\Delta U}{U})\omega_1] \tag{19a}$$

$$P_i(k+1) = P_i(k) + [(\frac{\Delta U}{U} - \frac{\Delta\tau_i}{\tau_i} + \frac{\Delta c_i}{c_i})\omega_2] \tag{19b}$$

$$\lambda_i(k+1) = \lambda_i(k) + [(\frac{\Delta\tau_i}{\tau_i} + \frac{\Delta U}{U})\omega_3] \tag{19c}$$

where $M_i(k)$ is the control domain of loop i at sampling instant k^{th}, $P_i(k)$ is the minimum prediction domain of loop i at sampling instant k^{th}, $\lambda_i(k)$ is the control coefficient of loop i at sampling instant k^{th}, $\{\omega_1, \omega_2, \omega_3\}$ is the quantization weight, U is the network utilization, τ_i is the network induce delay of loop i , c_i is the data transmission time of loop i , $\Delta\tau_i$ is the error change of network induce delay, Δc_i is the error change of transmission time, ΔU is the error change of network utilization.

As the control domain and the maximum prediction horizon are integers, the rounding of (19a) and (19b) is needed. That is the nearest integer value of the operating parameters (in actual MATLAB simulation, x is the parameter rounded: round(x)).

The role of quantization weight is quantificationally to convert the change values in parentheses of "round(x)" to the adjustment of parameters, in this section, the order of magnitude of prediction domain *P*, control domain M and control coefficient λ is adopted, for example, *M*=4, *P*=25, λ=0.2, the corresponding quantization weight are $\omega_1 = 1, \omega_2 = 10, \omega_3 = 0.1$.

This design, which considers factors of system control and network scheduling, will guarantee the optimization operation under the comprehensive performance of NCSs. From section 3.1, we can find that it is very important to improve the control performance of the whole system by dynamically change the network utilization in every loop and furthermore change the sampling period based on the different requirements in every loop. It adapts the system control in network environment and achieves the purpose of co-design by combined network scheduling parameters and changes the control parameters of prediction control algorithm reasonably.

4.4 General process of co-design methods

The general process of the co-design methods is (see Fig. 2):

1. Determine the plant and its parameters of NCSs.
2. Adopting GPC and EDF algorithm, defining the GPC control performance parameters and EDF scheduling parameters respectively.
3. According to the control parameters and scheduling parameters impact on system performance, design a reasonable optimization with balance between control performance and scheduling performance.
4. Use Truetime simulator to verify the system performance, then repeat the steps above if it has not meet the requirements.

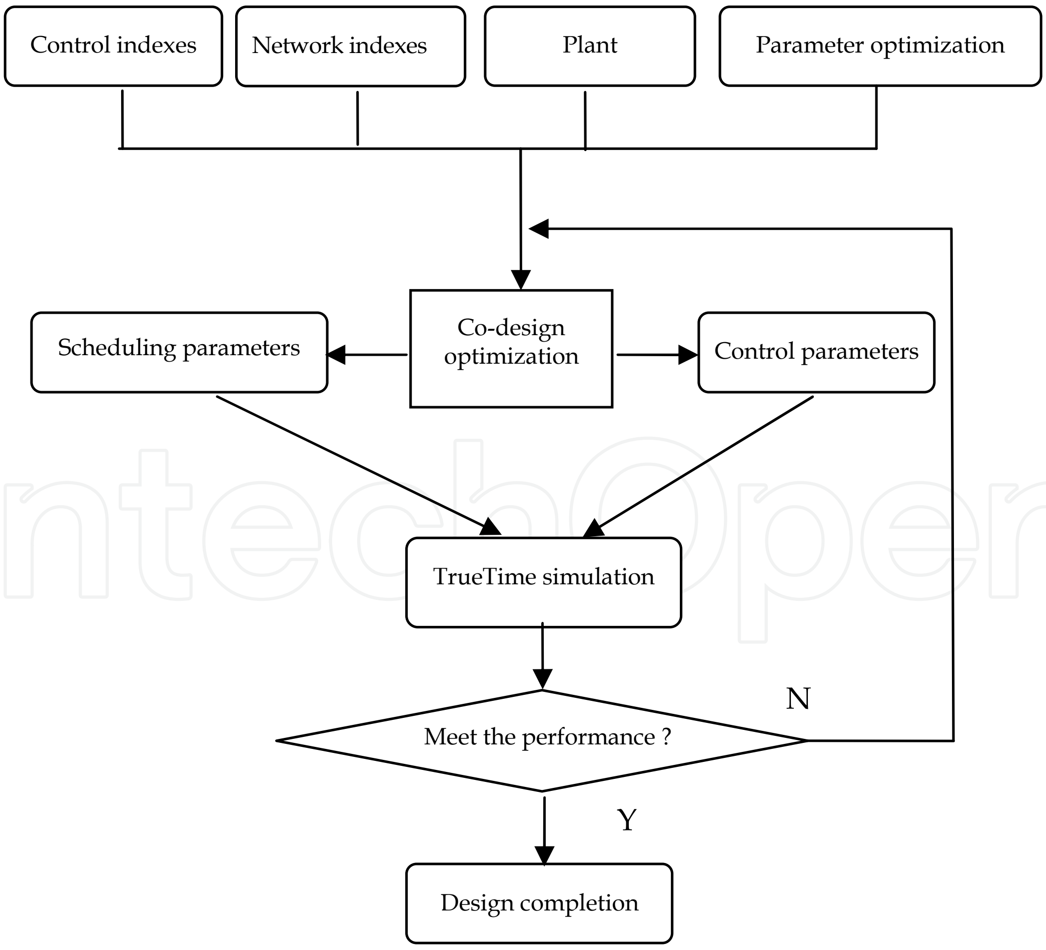

Fig. 2. General method of co-design of NCS scheduling and control

To facilitate the research of co-design, the algorithm proposed in this chapter can be extended to co-design of the other control and scheduling algorithms. And we can replace GPC with the other control algorithms and replace EDF with the other scheduling algorithms. The design idea and process are similar to the co-design algorithm presented in this chapter.

5. Simulation experiments

5.1 Simulation models and parameters' settings

In this chapter, NCS of three loops are used, the plants are the three DC (Direct Current) servo motors, and all the three loops have the same control architecture. The transfer function model of DC servo motor is:

$$G(s)=\frac{w(s)}{U_a(s)}=\frac{155.35}{s^2+12.46s+11.2} \tag{20}$$

The transfer function is converted into a state-space expression:

$$\begin{cases}\dot{\boldsymbol{x}}(t)=\boldsymbol{Ax}(t)+\boldsymbol{Bu}(t)\\ \boldsymbol{y}(t)=\boldsymbol{Cx}(t)\end{cases} \tag{21}$$

$$\boldsymbol{A}=\begin{bmatrix}-12.46 & -11.2\\ 0 & 1\end{bmatrix}，\boldsymbol{B}=\begin{bmatrix}1\\0\end{bmatrix}，\boldsymbol{C}=\begin{bmatrix}0 & 155.35\end{bmatrix}。$$

We can suppose that:

1. Sensor nodes use the time-driven, the output of the plant is periodically sampled, and sampling period is T .
2. Controller nodes and actuator nodes use event-driven.

At the sampling instant k^{th}, when the controller is event driven, after the outputs of the plant reach the controller nodes, they can be immediately calculated by the control algorithm and sent control signals, similarly, actuator nodes execute control commands at the instant of control signals arrived.

Let τ_k be the network induce delay, then

$$\tau_k=\tau_{sc}+\tau_{ca} \tag{22}$$

where τ_{sc} is the delay from sensor nodes to control nodes, τ_{ca} is the delay from control nodes to actuator nodes.

Suppose $\tau_k<T$, as the network induce delay exists in the system, the control input of the plant is piecewise constant values in a period, the control input which actuator received can be expressed by(23) (Zhang & Kong,2001):

$$\boldsymbol{v}(t)=\begin{cases}\boldsymbol{u}(k-1), & t_k<t\le t_k+\tau_k\\ \boldsymbol{u}(k), & t_k+\tau_k<t\le t_k+T\end{cases} \tag{23}$$

To discretize equation (22), and suppose the delay of NCS is stochastic, then

$$\begin{cases} \boldsymbol{x}(k+1) = \boldsymbol{A_d}\boldsymbol{x}(k) + \boldsymbol{\Gamma_0}\boldsymbol{u}(k) + \boldsymbol{\Gamma_1}\boldsymbol{u}(k-1) \\ \boldsymbol{y}(k) = \boldsymbol{C}\boldsymbol{x}(k) \end{cases} \tag{24}$$

where, $\boldsymbol{A_d} = e^{AT}$, $\boldsymbol{\Gamma_0} = \int_0^{T-\tau_k} e^{As}ds\boldsymbol{B}$, $\boldsymbol{\Gamma_1} = \int_{T-\tau_k}^{T} e^{As}ds\boldsymbol{B}$.

Then introducing the augmented state vector $\boldsymbol{z}(k) = [x_k^T \quad u_{k-1}^T]^T$, the above equation (24) can be rewritten as follows:

$$\begin{cases} \boldsymbol{z}(k+1) = \boldsymbol{\Phi_k}\boldsymbol{z}(k) + \boldsymbol{B_0}\boldsymbol{u}(k) \\ \boldsymbol{y}(k) = \boldsymbol{C_0}\boldsymbol{z}(k) \end{cases} \tag{25}$$

$$\boldsymbol{\Phi_k} = \begin{bmatrix} \boldsymbol{A_d} & \boldsymbol{\Gamma_1} \\ \boldsymbol{0} & \boldsymbol{0} \end{bmatrix} , \quad \boldsymbol{B_0} = \begin{bmatrix} \boldsymbol{\Gamma_0} \\ \boldsymbol{I} \end{bmatrix} , \quad \boldsymbol{C_0} = [\boldsymbol{C} \quad 0]$$

The initial sampling period $T = 10ms$, so the discretization model of DC servo motor is:

$$\begin{cases} \boldsymbol{x}(k+1) = \begin{bmatrix} 0.2625 & -0.629 \\ 0.0561 & 0.9618 \end{bmatrix} \boldsymbol{x}(k) + \begin{bmatrix} 0.0561 \\ 0.0034 \end{bmatrix} \boldsymbol{u}(k) \\ \boldsymbol{y}(k) = [0 \quad 155.35]\boldsymbol{x}(k) \end{cases} \tag{26}$$

The corresponding augmented matrix is:

$$\begin{cases} \boldsymbol{z}(k+1) = \begin{bmatrix} 0.2625 & -0.629 & 0.0561 \\ 0.0561 & 0.9618 & 0.0034 \\ 0 & 0 & 0 \end{bmatrix} \boldsymbol{z}(k) + \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} \boldsymbol{u}(k) \\ \boldsymbol{y}(k) = [0 \quad 155.35 \quad 0 \;]\boldsymbol{z}(k) \end{cases} \tag{27}$$

Convert the state space model of augment system to the CARIMA form:

$$\boldsymbol{y}(k) = 1.224\boldsymbol{y}(k-1) - 0.2878\boldsymbol{y}(k-2) + 0.5282\boldsymbol{u}(k-2) + 0.3503\boldsymbol{u}(k-3) \tag{28}$$

The simulation model structure of co-design of the networked control system with three loops is illustrated by Fig. 3. Controllers, actuators and sensors choose a Truetime kernel models respectively, the joint design optimization module in Fig.3 contains control parameter model and scheduling parameter model, and acts on the sensors and controllers of three loops, in order to optimize system operating parameters in real time.

The initial value of GPC control parameters: $M = 2$, $P = 20$, $\lambda = 0.1$, quantization weights: $\omega_1 = 1, \omega_2 = 10, \omega_3 = 0.1$; network parameters: CAN bus network, transmission rate is 800kbps, scheduling algorithm is EDF, reference input signal is step signal, amplitude is 500.

Loop1: initial sampling period $T_1 = 10ms$, size of data packet: 100bits, transmission time: $c_1 = 100 \times 8 / 800000 = 1ms$;

Loop2: initial sampling period $T_2 = 10ms$, size of data packet: 90bits, transmission time: $c_2 = 0.9ms$;

Loop3: initial sampling period $T_3 = 10ms$, size of data packet: 80bits, transmission time: $c_3 = 0.8ms$.

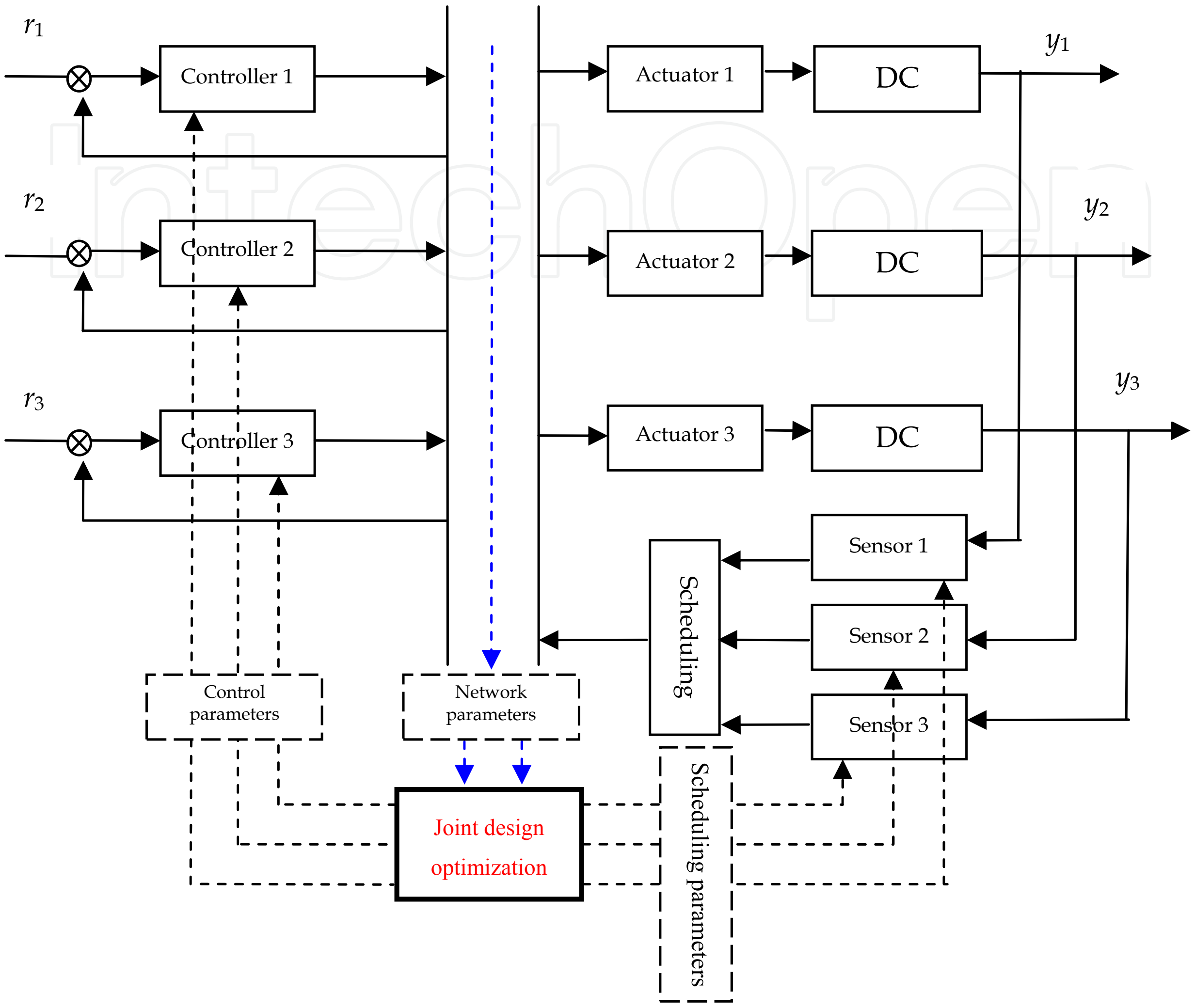

Fig. 3. Simulation framework of NCS with three loops

5.2 Simulation experimental results and their analyses

The following is comparison of joint design and no joint design, in order to facilitate comparison and analysis, defining as follows: "Co-design" expresses the simulation curve of joint design, while "N-Co-design" expresses the no joint design. Network induce delay can be achieved by delay parameter "exectime" in Truetime simulation. Node 1, 2 and 3 indicate the actuator, controller and sensor in loop 1 respectively; Node 4, 5 and 6 indicate the actuator, controller and sensor in loop 2 respectively; Node 7, 8 and 9 indicate the actuator, controller and sensor in loop 3 respectively.

Case 1: In the absence of interfering signals, and network induce delay is $\tau_k = 0ms$, under ideal conditions, the system response curves of both algorithms are shown in fig.4, where number 1, 2, 3 denote the three loops respectively.

From Fig. 4, in the situation of without interference and delay, the system response curves of Co-design and N-Co-design system response curves are basically consistency; they all show

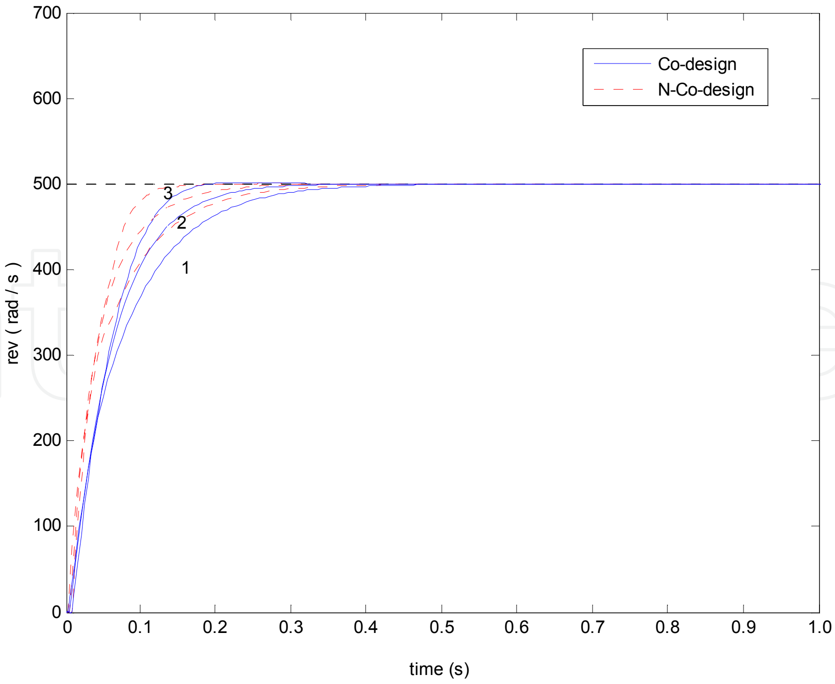

Fig. 4. The system response

the better performance. The system performance of N-Co-design is better than the Co-design one in terms of the small rise time and faster dynamic response. The main reason is the large amount of computation of GPC, and the system adds the amount of computation after considering Co-design, these all increase the complexity of the system and computation delay of network. So, in the ideal case, the N-Co-design system has the better performance.

Case 2: Interference signal network utility is 20%, and network induce delay is $\tau_k = 3ms$, τ_k is bounded by 0 and 1/2 of sampling period, that is 0~5ms. At this case, the network environment is relatively stable, network-induce delay is relatively small, interference signal occupied relatively small bandwidth.

Network scheduling timing diagrams of the two algorithms are shown as Fig. 5 and Fig. 6.

From the scheduling time diagrams of Co-design and N-Co-design (Fig.5 and Fig. 6), we can find that data transmission condition are better under two algorithms for loop1 and loop2, there are no data conflict and nonscheduled situation. But for loop3, compared with the co-design system, the N-Co-design shows the worse scheduling performance and more latency situations for data transmission and longer duration (longer than 7ms, sometimes), this greatly decreases the real-time of data transmission. The Co-design system shows the better performance: good real-time of data transmission, no latency situations for data, which corresponds to shorter adjustment time for loop3 in Fig.7. The system response curves are shown in Fig. 7.

Fig.7 shows that when the changes of network induce delay are relatively small, the response curves of co-design system and N-Co-design system are basically consistency, all three loops can guarantee the system performance. The system performance of N-Co-design is better than the Co-design one in terms of the small rise time and faster dynamic response.

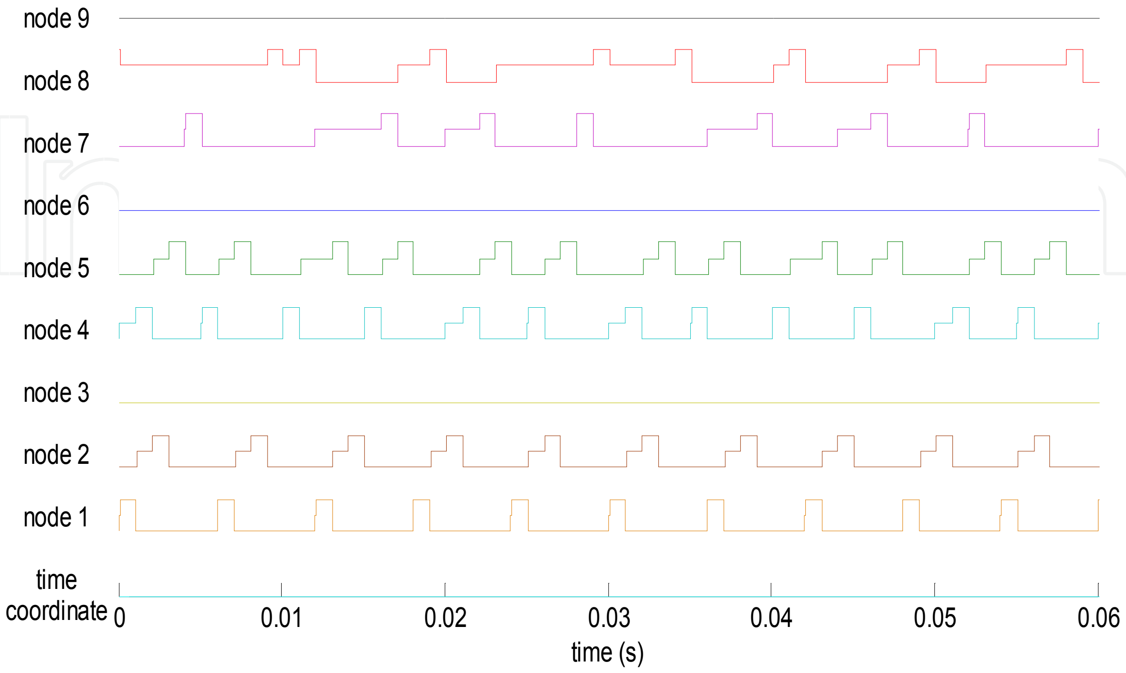

Fig. 5. The network scheduling time order chart of N-Co-design

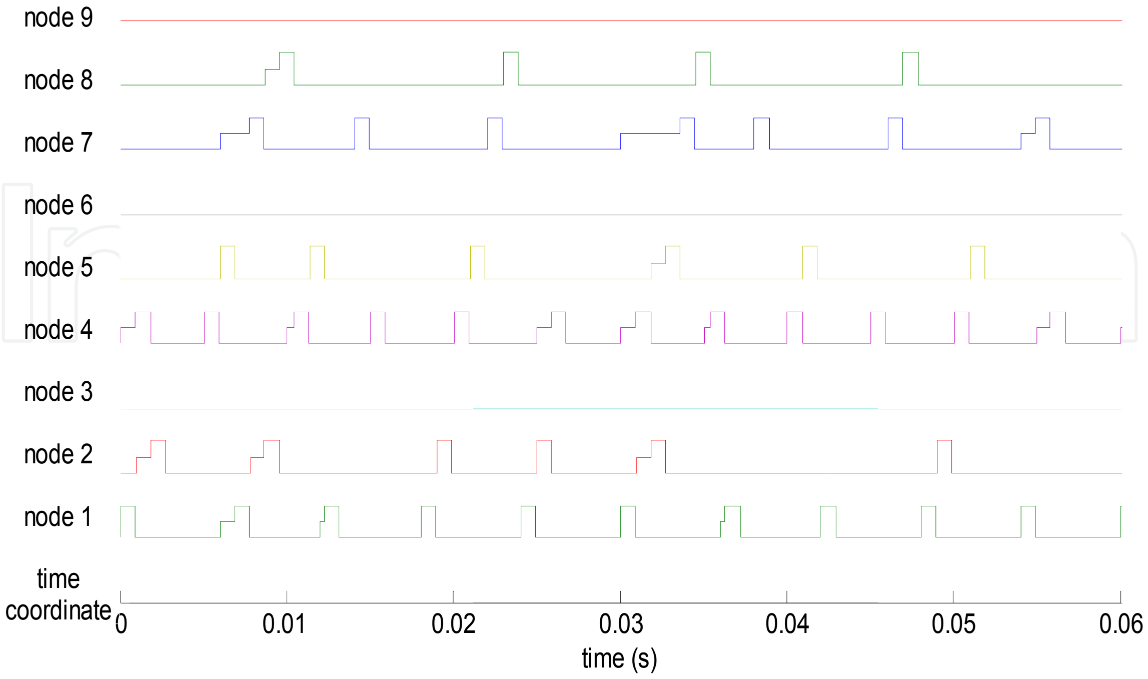

Fig. 6. The network scheduling time order chart of Co-design

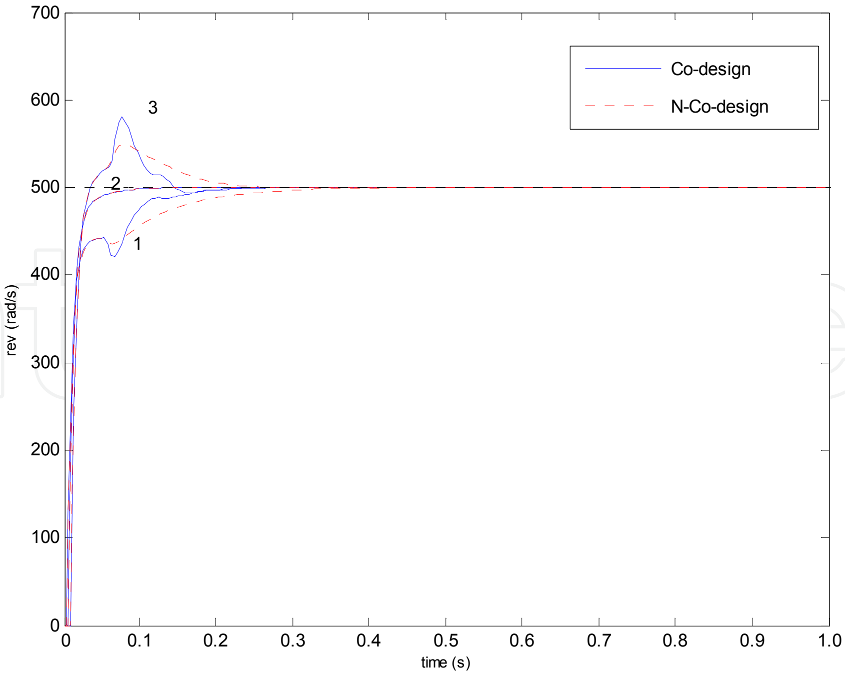

Fig. 7. The system response

The main reason is the large amount of computation of GPC, and the system adds the amount of computation after considering Co-design, these all increase the complexity of the system and computation delay of network. So, in smaller delay or less network load situations, the N-Co-design system has the better performance.

Case 3: Interference signal network utility is 40%, and network induce delay is $\tau_k = 8ms$, τ_k is smaller than the sampling period 10ms. At this case, the network environment is relatively worse, interference signal occupied relatively big bandwidth, network-induce delay is relatively big.

Network scheduling timing diagrams of the two algorithms are shown as Fig. 8 and Fig. 9.

From the two situations (Figure 8 and Figure 9) we can see that the data transmission condition of Co-design system is better than the N-Co-design one with all the three loops. Although there are no data conflictions and nonscheduled situation, the N-Co-design system shows the worse scheduling performance and more situations of latency data, which greatly affect the real-time data. This is bad for the real-time networked control system. In contrast, the Co-design system is better, latency data is the less, which can achieve the performance of effectiveness and real-time for the data transmission.

As shown in system response curves (Fig. 10) and scheduling timing diagrams (Fig. 8 and Fig. 9), when the network induce delay is bigger, the three loops of Co-design denote the better control and scheduling performance: better dynamic response, smaller overshoot, less fluctuation; scheduling performance guarantees the network induce delay no more than the sampling period, data transfer in an orderly manner, no nonscheduled situation. So, under the case of worse network environment and bigger network induce delay, the system with co-design expresses the better performance, while the worse performance of the system of N-Co-design. The main reason is the operation of control algorithm of Co-design with

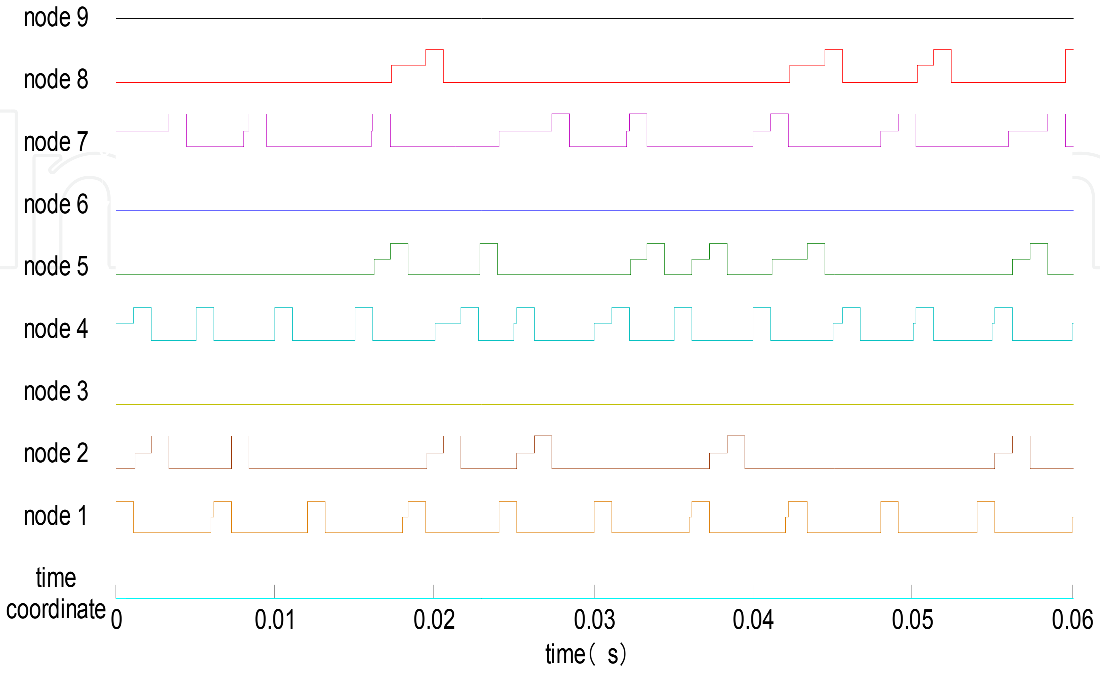

Fig. 8. The network scheduling time order chart of N-Co-design

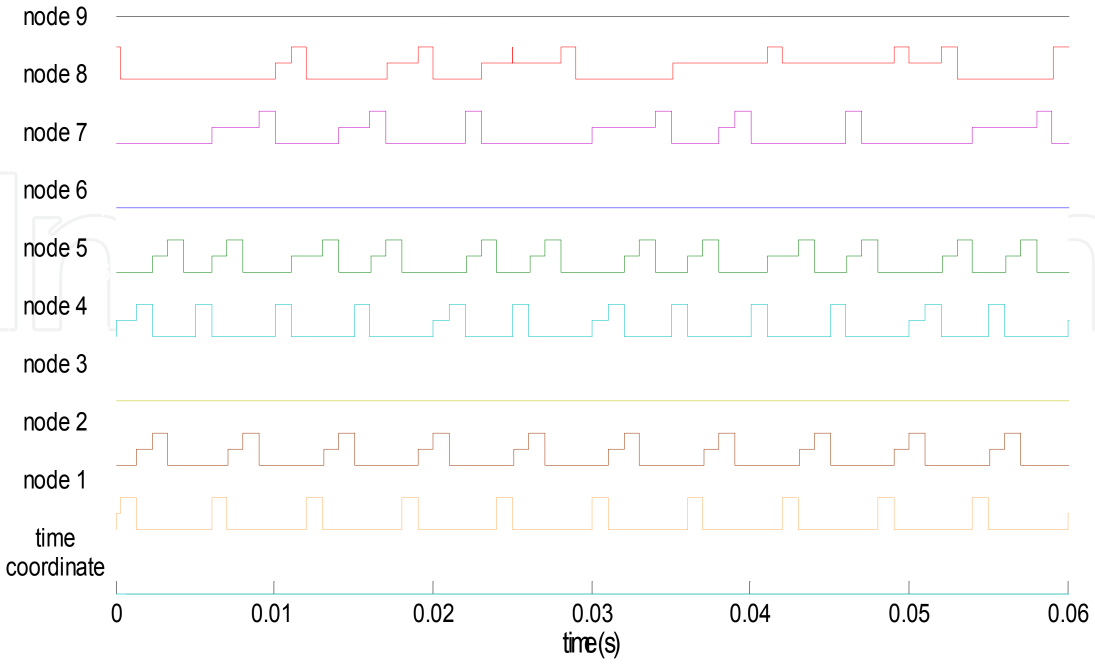

Fig. 9. The network scheduling time order chart of Co-design

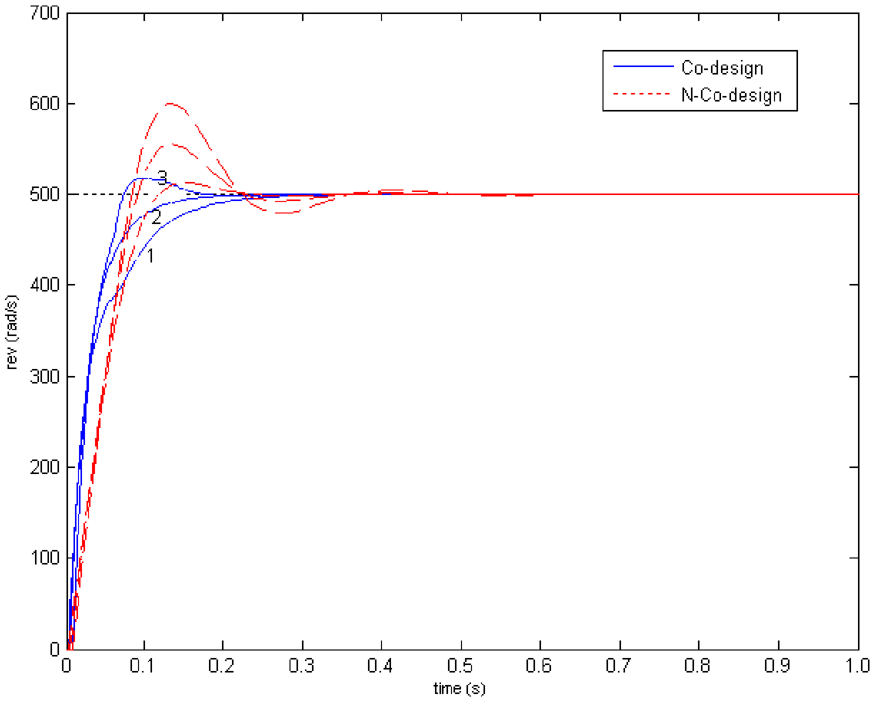

Fig. 10. The system response

considering the effect of network. When the network impact increases, the effect is decreased on the control algorithm.

Case 4: To illustrate the superiority and robustness of the designed algorithm, we add interference to the system at the instant t=0.5s, that is increasing the network load suddenly, the network utility of interference increases from 0 to 40%. The system response curves of the three loops with the two algorithms are shown as follows.

From the system response curves, we can see that the system of Co-design shows the better robustness and faster dynamic performance when increasing interference signal suddenly. In loop 1 (Fig. 11), the system pulse amplitude of Co-design is small, the rotational speed amplitude is 580rad/s (about 5400 cycles/min), the rotational speed amplitude of N-Co-design is nearly 620 rad/s; in loop 2 (Fig. 12), the system amplitude and dynamic response time increase compared to loop 1, but the both can guarantee the normal operation of system; but in loop 3 (Fig. 13), the system occurs bigger amplitude (nearly 660 rad/s) and longer fluctuation of N-Co-design system after adding interference signal, and also the slower dynamic response. The system of Co-design shows the better performance and guarantees the stable operation of system.

From the four cases above, we can conclude that under the condition of better network environment, the system performance of Co-design is worse than the one without Co-design, this is because the former adopts GPC algorithm, and GPC occupies the bigger calculation time, it further increases the complexity of the algorithm with joint design optimization. So, under the ideal and small delay condition, the system without Co-design is better, contrarily, the Co-design is better. When adding interference signal suddenly, the system with Co-design shows the better network anti-jamming capability and robustness.

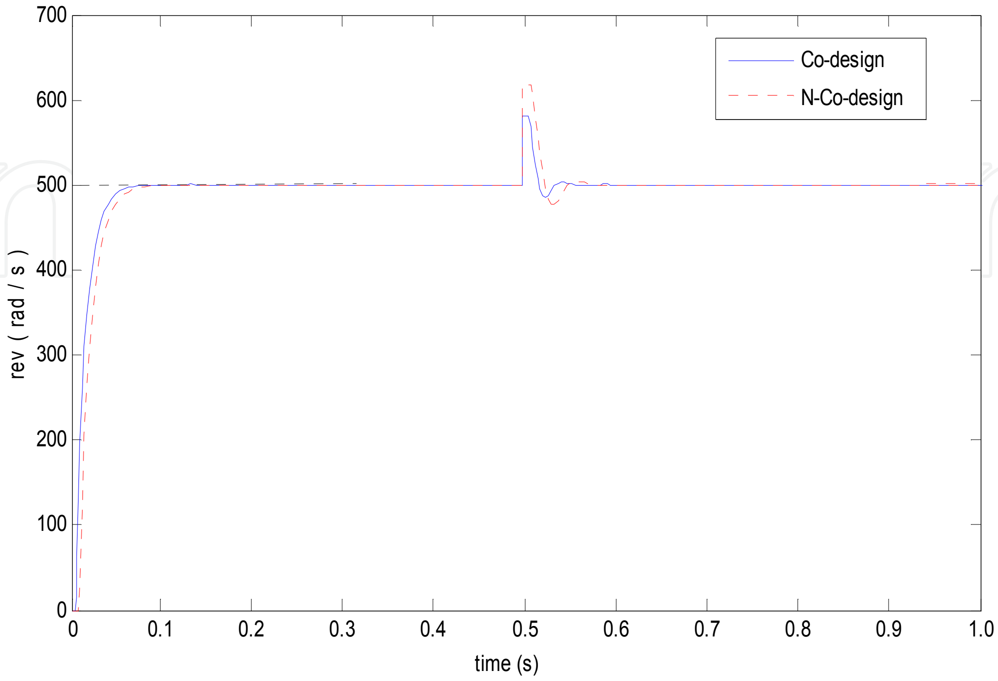

Fig. 11. The system response of Loop 1

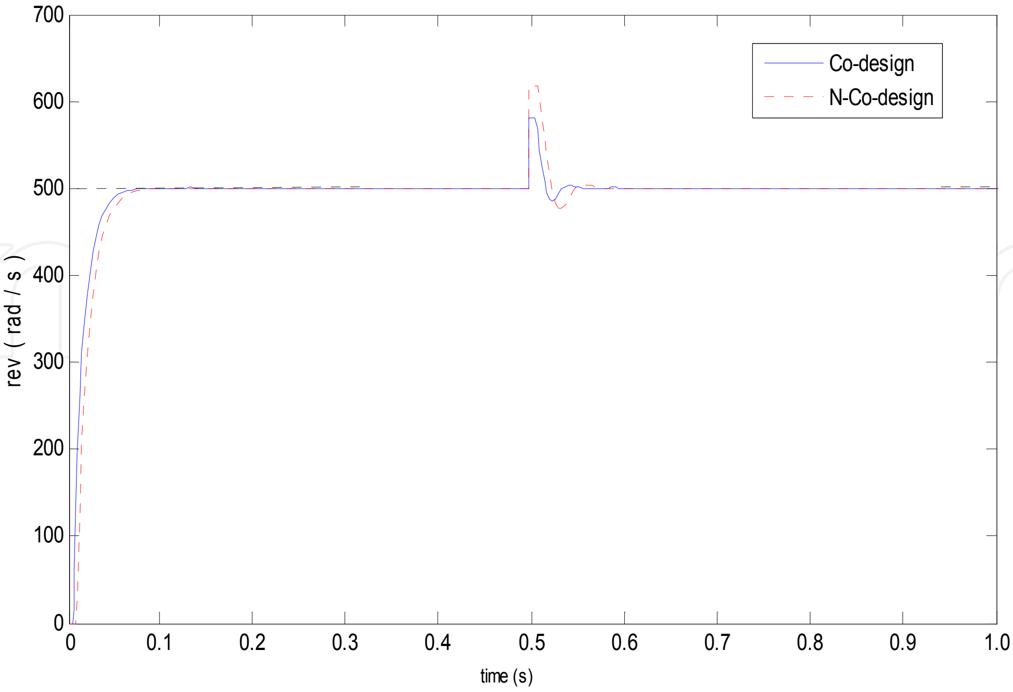

Fig. 12. The system response of Loop 2

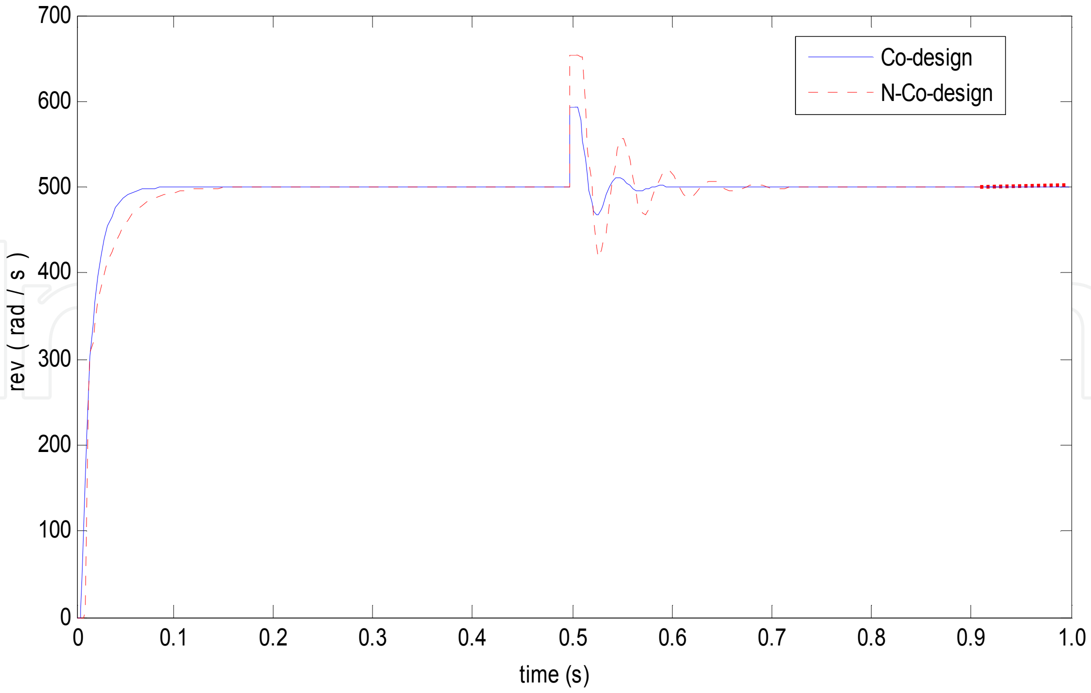

Fig. 13. The system response of Loop 3

6. Conclusion

First introducing the theory and parameters of GPC , then the EDF scheduling algorithm and parameter are presented. The co-design of control and scheduling is proposed after analyzing the relationship between predictive control parameters and scheduling parameters for a three-loop DC servo motor control system. By analyzing the effect on system performance by the control parameters and the scheduling parameters, a joint optimization method is designed considering the balance between control performance and scheduling performance. Finally this algorithm is validated by Truetime simulation, in the cases of big delay and bad environment, especially the presence of external interference, the co-design system shows the better performance, such as good robustness and anti-jamming capability.

7. Acknowledgment

This work is supported in part by National Natural Science Foundation of China (NSFC) under Grant No.60872012

8. References

Gaid M B,Cela A,Hamam Y. (2006). Optimal integrated control and scheduling of systems with communication constraints, *Proceedings of the Joint 44th IEEE Conference on Decision and Control and European Control Conference*, pp. 854-859, ISBN 0-7803-9567-0, Seville, Spain. December, 2005

Gaid M B, Cela A, Hamam Y. (2006). Optimal integrated control and scheduling of networked control systems with communication constraints: application to a car suspension system. *IEEE Transactions on Control Systems Technology,* Vol.14, No.4, (July 2006), pp.776-787, ISSN 1063-6536

Arzen K E, Cervin A, Eker J. (2000). An introduction to control and scheduling co-design. *Proceedings of the 39th IEEE Conference on Decision and Control*,pp. 4865-4870,ISSN 0191-2216, Sydney, Australia. Decmber, 2000

Calrke, Mohtadi. (1989). Properties of Generalized Predictive Control, *Automatic*, Vol.25, No.6,(November 1989), pp.859-875, ISSN 0005-1098

Wei Wang et al.(1998). *Generalized predictive control theory and its application*, Science Press, ISBN 978-7-030-06804-0, Beijing

Shouping Guan, Wei Zhou.(2008). *Networked control system and its application*, Publishing house of electronics industry, ISBN 978-7-121-06946-8, Beijing

Baocang Ding. (2008). *Predictive control theory and methods*, China machine press, ISBN 978-7-111-22898-1, Beijing

Bin Li.(2009). Study of fuzzy dynamic scheduling and variable sampling period algorithm of NCS. *Beijing Jiaotong University,*Vol.33, No.2, (April 2009),pp.98-102, ISSN 1673-0291

Lian Feng-li, Moyne James, Tilbury Dawn. (2001). Time Delay Modeling and Sampling Time Selection for Networked Control Systems, *Proceedings of the ASME Dynamic Systems and Control*, pp. 1265-1272, New York, USA. January, 2001

Xuelin Zhang, Feng Kong.(2008). Research on Scheduling of Networked Control Systems Based on CAN Bus, *Control and automation publication group,*ISSN 1008-0570

Pedreiras P, Almeida L. (2002). EDF Message Scheduling on Controller Area Network, *Computing & Control Engineering Journal*, Vol. 13, No. 4, (August 2002), pp. 163-170

Mayne, Rawlings, Ral and Scokaert. (2003). Constrained Model Predictive Control Stability and Optimality, *Automatic*,Vol.36, No.6, pp.789-814, ISSN 0005-1098

Yaohua Hu, Suwu Xu, Gongfu Xie et al. (2000). Robustness of General Prediction Control, *Journal of Dalian Maritime University*, Vol. 26, No. 1, (Feburary 2000), pp. 75-79, ISSN 1006-7736

Zengqiang, Chen, Xinhua Wang, Zhuzhi Yuan. (2003). GPC Control and Analysis of Stability for Hybrid Systems,*Computing Technology and Automation*, Vol. 22, No. 2,(June 2003), pp. 1-4, ISSN 1003-6199