UKACC International Conference on Control 2012
Cardiff, UK, 3-5 September 2012

A New Bandwidth Scheduling Method for Networked Learning Control

Lijun Xu*†, Kang Li†, Minrui Fei*, Dajun Du*
*Shanghai Key Laboratory of Power Station Automation Technology,
School of Mechatronical Engineering and Automation, Shanghai University, Shanghai, China
Email: ruby_mickey@sina.com mrfei@staff.shu.edu.cn
†School of Electronics, Electrical Engineering and Computer Science,
Queen's University Belfast, Belfast, BT9 5AH, UK
Email: k.li@qub.ac.uk

***Abstract*—In this paper, the optimal bandwidth allocation scheduling problem for two-layer networked learning control systems (NLCSs) is studied. In NLCS, multiple networked feedback control loops share a common communication channel and they compete to bid for available bandwidth. A non-cooperative game fairness model is first formulated, which takes into consideration of a number of factors, such as transmission data rate, control sampling strategy and scheduling pattern. Then, a novel two-layer hierarchical market competition algorithm (THMCA) is proposed. Two hierarchical population individuals are defined in the algorithm, namely the holding companies and the subsidiary companies which altogether form conglomerates. Market competitions among these conglomerates lead to the convergence to a monopoly at the end, resulting in an optimal solution of the above problem. The algorithm is shown to have a high convergence rate and the comparison simulation results on a NLCS with up to 100 subsystems have demonstrated the effectiveness of the proposed method.**

I. Introduction

A networked control system (NCS) [1] is defined as a feedback control system where the control loops are closed via a communication network. Due to its various advantages such as low cost installation, ease of maintenance and great flexibility, NCSs have been widely applied in manufacturing, aircraft and power systems etc. Most researches have focused on the traditional single-layer NCSs in the last decade. However, there exist many complex plants that are composed of a great number of subsystems and the traditional single-layer NCS architecture may not be applicable. Recently, a two-layer Networked Learning Control System (NLCS) architecture has been proposed [2]. In this architecture, the bottom layer is for real-time control, and local controllers communicate with the sensors and actuators attached to the plant through shared networks. The upper layer is used for complex learning and scheduling tasks.

In a real-time NLCS, limited network bandwidth along end-to-end paths [3] inevitably cause network-induced delays [4] or packet dropout which further deteriorates the system performance even causes instability. Therefore, an optimized scheduling for bandwidth allocation among all network links plays a key role in improving the quality of network service (QoS) and the performance of control system (PoC). This can be achieved generally through the following two steps. Firstly, bandwidth scheduling is designed to allocate the available bandwidth to control units to meet communication demand. A non-cooperative game (NG) [5] theory based rational competition mechanism can be employed, which emphasizes individual rational defined by Nash equilibrium (NE) and can simultaneously satisfy individual and collective requirements. Secondly, the exact solution to the bandwidth scheduling problem can be obtained by a complete enumeration.

This is however prohibitive due to its excessive computational time for real-time applications. To tackle this problem, several intelligent optimization methods can be adopted, for example genetic algorithm (GA) [6], particle swarm optimization (PSO) [7], clone evolutionary algorithm [8], bacterial foraging (BF) [9] and shuffled frog leaping algorithm (SFLA) [10] etc. Evolutionary algorithms (EAs), such as GA and PSO, are stochastic based search methods. GA is one of the early proposed evolutionary algorithms which has found many successful applications. However, it is computationally expensive, and the convergence cannot be guaranteed. Bacterial foraging (BF) algorithm is a feature selection method based on a heuristic search strategy with fast computing speed. But a large storage capacity is required to complete the computation. Shuffled frog leaping algorithm (SFLA) is a meta-heuristic optimization method which is based on observing, imitating, and modeling of the behavior of a group of frogs searching for the location where maximum amount of food is available. The SFLA combines the benefits of both the genetic-based memetic algorithm and the social behavior-based PSO algorithm. However SFLA suffers from the curse of dimensionality problem.

In this paper, a new integer-coded two-layer hierarchical market competition algorithm (THMCA) is proposed to efficiently solve the bandwidth scheduling problem in the NLCS applications. The rest of the paper is organized as follows. Section 2 introduces NLCS and the non-cooperative game scheduling scheme. A new two-layer hierarchical market competition algorithm (THMCA) is proposed in Section 3. Section 4 presents the comparative simulation results. Finally, a brief conclusion is given in Section 5.

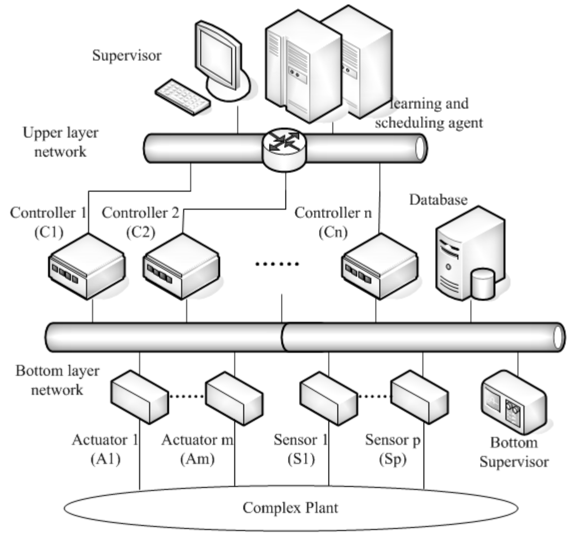

Fig. 1. A two-layer networked learning control system (NLCS)

II. Bandwidth non-cooperative game model for a networked learning control system

A. System Architecture

The two-layer networked learning control system architecture [2] is shown in Fig.1, where Ci, Si and Ai represent the i-th controller, sensor and actuator respectively. Local controllers are connected to the sensors and actuators, attached to the complex plant via the bottom layer communication network, typically some fieldbus dedicated to real-time control. Local controllers also communicate with the computer system, which functions as a learning and scheduling agent through the upper layer communication network. This network can be local area network (LAN), wide area network (WAN), or possibly the internet. Control and learning signals at two levels share the available network bandwidth with other consumers. This general architecture can be adopted in many industrial applications with distributed plants and units, such as the power systems.

B. Bandwidth non-cooperative game model

Suppose the set of subsystems is denoted as $L = \{\ell_i | 1 \le i \le n\}$, where n is the total number of subsystems in a NLCS. The network can only provide limited available bandwidth. Each subsystem aims to improve its own bandwidth utilization instead of the overall network performance. Therefore, all the subsystems in a NLCS form a non-cooperative game (NG). The NG model is a triple, namely $G(L, S, u_i)$, where S is the set of strategies including transmission data rate, allotted bandwidth and sampling period for each subsystem, that is, $S = \{s_i | s_i = (b_i, \delta_i, t_i), i \in L\}$, $b_i \in [\delta_i^{min}, U_d]$ is the pre-allocated data rate to the subsystem l_i, and t_i is the sampling period. Data rate set available for all subsystems in NLCS is given as $D = \{\delta_i | 1 \le i \le n\}$, $\delta_i \in [\delta_i^{min}, \delta_i^{max}]$. The minimum data rate δ_i^{min} is to guarantee the most basic work. Maximum bandwidth of the network is U_d and $\delta_i^{max} \le U_d$. u_i is the utility function of subsystem l_i which maps S to real numbers $u : S \to R$. If and only if $u_i(s_i^*) > u_i(s_i)$, the quality of s_i^* is better than s_i for subsystem l_i.

The available data rate δ_i satisfies the probability distribution $d_i(\delta_i)$ defined as follows [11].

$$d_i(\delta_i) = \begin{cases} 0 & \delta_i < \delta_i^{min} \\ (\Delta/\Delta_i)^c & \delta_i^{min} \le \delta_i \le \delta_i^{max} \\ 1 & \delta_i > \delta_i^{max} \end{cases} \tag{1}$$

where $\Delta = \delta_i - \delta_i^{min}$, $\Delta_i = \delta_i^{max} - \delta_i^{min}$ and $c \ge 1$ works as an empirical constant which can be adjusted for a specific subsystem.

Given the above definition, the mean value of occupied bandwidth for a subsystem is thus given as

$$Ed_i(\delta_i) = (c\delta_i^{max} + \delta_i^{min})/(c+1) \tag{2}$$

The utility function is defined as follows:

$$u_i(s_i) = \begin{cases} \frac{[\|e_i\|^2 - (\delta_i - Ed_i)^2 - \gamma_i t_i](\frac{b_i}{Ed_i} e)^{\mu}}{e^{\mu \frac{b_i}{Ed_i}} \|e_i\|^3} \\ if\ 0 \le b_i < Ed_i \\ [1 - (\frac{b_i - Ed_i}{Ed_i - \delta_i^{max}})^{\nu}] \frac{\|e_i\|^2 - (\delta_i - Ed_i)^2 - \gamma_i t_i}{\|e_i\|^3} \\ if\ Ed_i \le b_i \le \delta_i^{max} \end{cases} \tag{3}$$

Here, the rights law is used, including compounding data rate δ_i and sample period t_i, and γ_i is the weight coefficient for trade-off. $\|e_i\|$ is the maximum expectation deviation between optional data rate and the distribution of each subsystem, that is $\|e_i\| = \max(|\delta_i^{min} - Ed_i|, |\delta_i^{max} - Ed_i|)$. μ, ν are the empirical constants for specific subsystems to adjust the change rate of utility functions.

III. Two-layer hierarchical market competition algorithm (THMCA)

The two-layer hierarchical market competition algorithm (THMCA) is inspired by competitions among enterprises in economic activities. The THMCA begins with an initial population called perfect competition companies. Some of the best companies that have the best objective fitness function values are selected to be the holding companies. The rest become the subsidiary companies which are then divided among the holding companies based on their power. The power of a holding company is positively proportional to its fitness value. The holding companies and their subsidiary companies form different conglomerates. Then subsidiary companies move toward their relevant holding companies and the position of the holding companies will be updated if necessary. In the next step, the market competition among the conglomerates begins, and the weak conglomerates are eliminated. This is called monopolistic competition procedure. Then the oligopoly procedure begins. The market competition will gradually lead to an increase in the power of strong conglomerates (oligopolies) and a decrease in the power of weaker ones. Finally the weak conglomerates which could not to improve their performance will collapse. These competitions among the conglomerates will cause all the companies to converge to a state called monopoly where only one conglomerate exits in the market and all the other companies become subsidiary companies of this holding company.

To begin with, an initial population called perfect competition companies is created. In a D-dimensional problem, the position of the i-th company is defined as $Company_i = [x_{i,1}, x_{i,2}, \cdots, x_{i,D}]$, $i = 1, 2, \cdots, N_c$, where N_c is total number of competition companies.

The fitness function of the i-th company is defined as:

$$fit_i(Company_i) = fit_i(x_{i,1}, x_{i,2}, \cdots, x_{i,D}) \tag{4}$$

Then, the cost function of the i-th company can be defined as:

$$f_i = \frac{1}{fit_i(x_{i,1}, x_{i,2}, \cdots, x_{i,D})} \tag{5}$$

$N^{holding}$ of the most powerful competition companies are selected as the holding companies, which form different conglomerates. The remaining N^{sub} are the subsidiary companies of these holding companies. At the next step, the subsidiary companies must be divided among the holding companies based on their power. The initial number of subsidiary companies of a conglomerate is directly proportional to its power. Theoretically, the normalized cost of the n-th holding company can be defined as:

$$F_n^{holding} = \max_i\{f_i^{holding}\} - f_n^{holding} \tag{6}$$

where $f_n^{holding}$ is the cost of the n-th holding company.

The normalized power of each holding company is defined as:

$$P_n^{holding} = \|F_n^{holding} / \sum_{n=1}^{N^{holding}} F_n^{holding}\| \tag{7}$$

Then the initial number of subsidiary companies of a conglomerate becomes

$$N_n^{sub} = round\{P_n^{holding} \times N^{sub}, 0\} \tag{8}$$

where N_n^{sub} is the initial number of subsidiary companies of the n-th conglomerate. For each holding company, N_n^{sub} of the subsidiary companies are randomly selected and allocated. These subsidiary companies along with their holding company form the conglomerate.

Then the subsidiary companies start to move toward their relevant holding companies. The positions of the subsidiary companies of the n-th conglomerate are updated as follows:

$$SUB_{n,i} = sub_{n,i} + \frac{rand() \times \omega(holding_n - sub_{n,i})}{\cos\theta} \tag{9}$$

where $sub_{n,i}$ is the position of the i-th subsidiary company of the n-th holding company, $rand()$ is a random number between 0 and 1, ω is a weight factor, and $holding_n$ is the position of the n-th holding company. To search different points around the holding company, a random amount of deviation is added to the direction of movement. The movement of a subsidiary company toward its relevant holding company at its new direction θ is a random angle between $-\varepsilon$ and ε, where $\varepsilon > 0$ is the parameter that adjusts the deviation from the original direction.

However, a subsidiary company in a conglomerate may reach a position with higher fitness (or lower cost) than its holding company. In such case, the positions of the holding company will be replaced by the higher one. The rest will move toward the new position of the holding company.

The total power of a conglomerate depends on both the power of the holding company and the power of its subsidiary companies. But the holding company has larger weights. This total power is defined by the weighted cost of two hierarchical companies:

$$power_n^{cong} = \frac{1}{p_n^{cong}} = \frac{1}{f_n^{holding} + \tau\overline{f_{n,i}^{sub}}} \tag{10}$$

where p_n^{cong} is the total cost of the n-th conglomerate and $0 < \tau < 1$. Cost $\overline{f_{n,i}^{sub}}$ is the geometry mean of i-th subsidiary company in n-th conglomerate. In fact, τ represents the role of the subsidiary companies in determining the total power of a conglomerate.

The market competition among conglomerates begins and all the conglomerates try to take possession of the subsidiary companies of other conglomerates. This competition is modeled by picking some of the weakest subsidiary companies of the weakest conglomerates and making a competition among all conglomerates to possess these subsidiary companies. Each of the conglomerates will have a likelihood of taking possession of these subsidiary companies based on its total power; therefore, powerful conglomerates have greater chance to possess subsidiary companies. The possession probability of each conglomerate must be found. The normalized total cost of each conglomerate is calculated as:

$$P_n^{cong} = \max_i\{p_i^{cong}\} - p_n^{cong} \tag{11}$$

where P_n^{cong} is the normalized total cost of the n-th conglomerate.

The possession probability of each conglomerate is given by

$$pp_n^{cong} = \|P_n^{cong} / \sum_{n=1}^{N^{holding}} P_n^{cong}\| \tag{12}$$

where pp_n^{cong} is the possession probability of the n-th conglomerate. A vector is formed to divide the relevant subsidiary companies among the conglomerates:

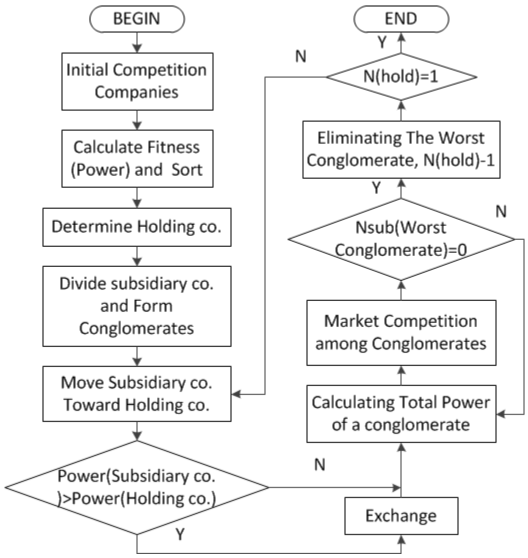

Fig. 2. Flowchart of two-layer hierarchical market competition algorithm

$$\mathbf{PP}^{cong} = [pp_1^{cong}, pp_2^{cong}, \cdots, pp_{N^{holding}}^{cong}] \tag{13}$$

Then a random vector with the same size is generated:

$$\mathbf{RAND} = [rand_1, rand_2, \cdots, rand_{N^{holding}}] \tag{14}$$

where $rand_1, rand_2, \cdots, rand_{N^{holding}}$ are randomly generated numbers between 0 and 1.

Finally, vector $\mathbf{Dif}$ is formed by subtracting $\mathbf{RAND}$ from $\mathbf{PP}^{cong}$:

$$\begin{aligned}\mathbf{Dif} &= \mathbf{PP}^{cong} - \mathbf{RAND} \\ &= \begin{pmatrix} PP_1^{cong} - rand_1, \\ PP_2^{cong} - rand_2, \\ \cdots \\ PP_{N^{holding}}^{cong} - rand_{N^{holding}} \end{pmatrix}^{\mathrm{T}}\end{aligned} \tag{15}$$

The mentioned subsidiary companies will be given to an conglomerate which has the maximum relevant index in $\mathbf{Dif}$ vector.

The powerless conglomerates will collapse in the market competition. Different criteria can be defined for collapse mechanism. In this paper, a conglomerate is assumed to be collapsed when it loses all of its subsidiary companies.

After the market competitions, all the conglomerates will collapse except the most powerful one and all companies under their possession become subsidiary companies of this conglomerate. All the subsidiary companies also have the same positions and the same fitness. In such case, the algorithm stops. The flowchart of the procedure is illustrated in Fig.2.

In the integer coded THMCA, the company position consists of a sequence of integer numbers, representing the sequence of the bandwidth allocation cycle duration of each subsystem during the scheduling horizon. A positive integer represents an increased bandwidth allocation, while a negative integer represents the bandwidth decrease of the subsystem.

TABLE I
RANGE OF THE OUTPUT DATA RATES OF SUBSYSTEMS FOR TWO DISTRIBUTIONS

Data Rate	DND		SD	
Subsystems	δ_i^{min}	δ_i^{max}	δ_i^{min}	δ_i^{max}
l_1	20.58	75.51	32	51
l_2	41.63	88.68	56	85
l_3	53.45	97.20	19	31
l_4	64.02	106.71	98	120
l_5	76.84	111.69	28	42
l_6	77.68	115.60	102	135
l_7	65.97	108.10	49	76
l_8	56.32	99.59	84	116
l_9	46.24	90.73	156	196
l_{10}	32.30	80.18	246	380

IV. SIMULATION STUDIES

To verify the effectiveness of the proposed bandwidth scheduling method, numerical simulations were carried out using MATLAB on an Intel(R) Core(TM)-i5-3.10-GHZ desktop personal computer (PC) with Windows 7 Enterprise.

The proposed THMCA was tested on NLCSs with Num = 10, 20, 50 and 100 subsystems. The required data for ten subsystems is given in table I, which shows the distributions of sending data rates for the first ten subsystems. Two different distributions, namely the Discrete Normal Distribution (DND) and Stochastic Distribution (SD) are generated to cover general cases. The simulation test system first has 10 subsystems. Initially, the network bandwidth is $U_d = 1000Kbps$. For NLCS with 20 subsystems, the data of the ten subsystems was duplicated and the total bandwidth available was multiplied by two. For the problem with more subsystems, the data was scaled appropriately.

The optimal THMCA parameters for ten subsystems which were chosen after several runs are given as $N^{holding} = 10$, $N^{sub} = 200$, $\tau = 0.2$, $\omega = 2$. For NLCS with more subsystems, the same parameters were utilized, except for $N^{holding}$ and N^{sub} which increase correspondingly. Another parameter to be selected is $\theta \in (45^\circ, 90^\circ)$, which adjusts the deviation from the original direction. These values were found suitable to produce good solutions in terms of the processing time and the quality of the solutions. The fitness function is set to be the utility function of NG, that is, $fit_i = u_i$.

For the comparison purpose among the THMCA and other optimization methods, all the simulation experiments used the same basic parameter settings.

The convergence of the algorithm for ten subsystems in NLCS with 50, 100, 150, 200, 250 and 300 initial competition companies with discrete nomal distribution data rate is shown in Fig.3. It is clear that the initial number of 200 competition companies is notably the best on both convergence (mean solution time) and the bandwidth consumption.

From the results of 10 simulation runs, it is found that the optimal solution can be obtained after 8-th to 10-th market competition interactions. Fig.4 shows the convergence of the

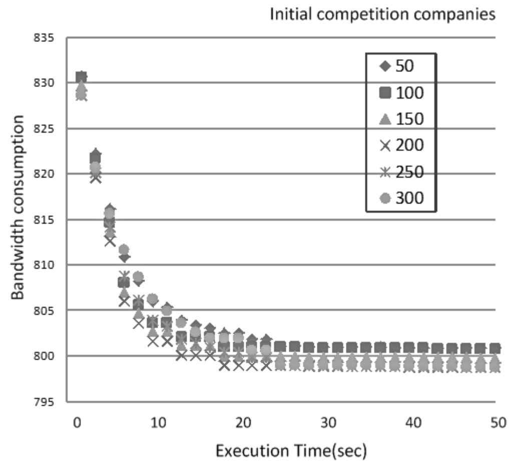

Fig. 3. Convergence characteristics of market competition algorithm with different initial competition companies in 10-subsystem NLCS

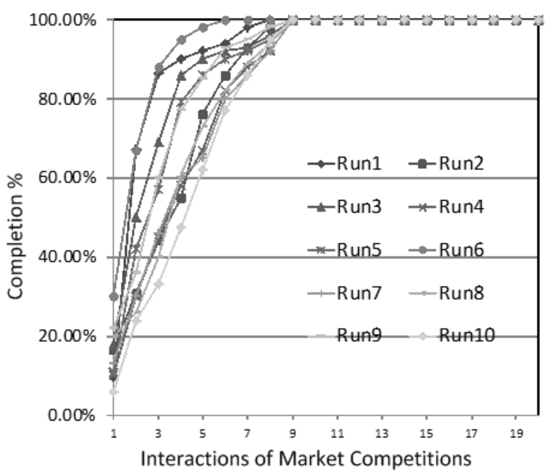

Fig. 4. 10-runs convergence of market competition algorithm in 10-subsystem NLCS

iteration for the 10-subsystem test system with stochastic distribution data rate. This indeed verifies the high convergence rate of the algorithm.

When the overall fitness value is stabilized, the nash equilibrium point is reached. For discrete normal distribution, this is: $\{b_1, b_2, \cdots, b_{10}\} = \{55.90,\ 68.12,\ 77.56,\ 93.37,\ 101.34,\ 105.82,\ 94.70,\ 84.18,\ 60.79,\ 56.84\}$. For stochastic distribution, this is $\{38,\ 78,\ 27{,}110,\ 36,\ 117,\ 62,\ 101,\ 107,\ 318\}$.

Figure 5 and 6 illustrate the proportion of statistical output data rates with lower limit, the reserved data rate and output data limit of the two distributions. It is evident that the game model based method is able to produce a fair network resource allocation in NLCS under different constraints. The overall performance of the system remained stable by effectively restricting large network resource utilization.

Figure 7 shows the ratio between the scheduled sampling period of subsystems and the original sampling period t_i^*/t_i, where the lower ratio means a better optimization scheme. The optimization results of the sampling period after the

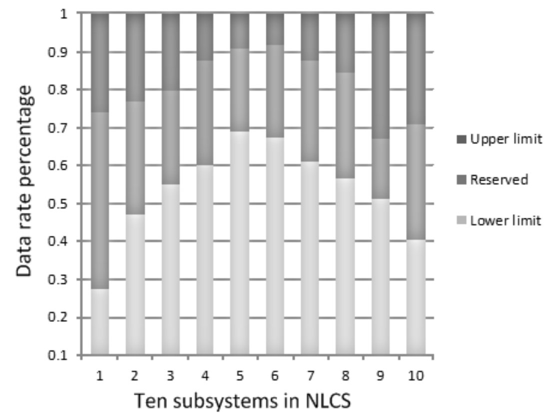

Fig. 5. Data rate percentage plot of DND in 10-subsystem NLCS

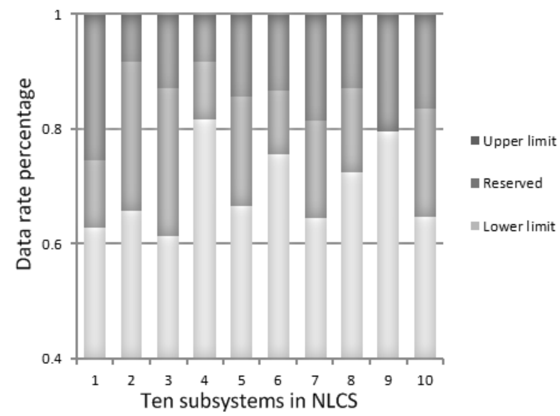

Fig. 6. Data rate percentage plot of SD in 10-subsystem NLCS

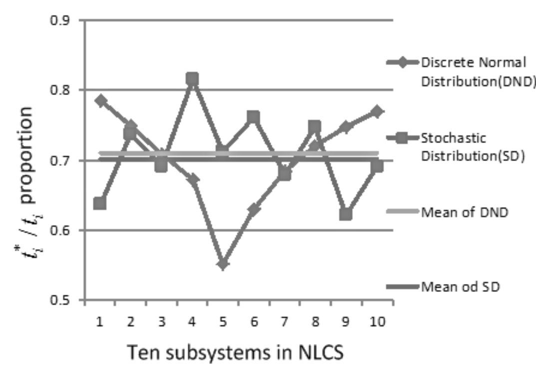

Fig. 7. t_i^*/t_i proportion plot in 10-subsystem NLCS

adjustment within the requirements of the two data rates distributions is also shown. Further, when the discrete normal distribution data rates was used, larger probability distribution leads to better optimization results, indicating that this method is adequate for meeting the requirements of most data rates.

The best bandwidth results obtained by THMCA are compared with those obtained by the shuffled frog leaping algorithm (SFLA) [11], BF [9], GA [6], hybrid quantum clone evolutionary algorithm (HQCE) [8], and quantum inspired

TABLE II
COMPARISON OF TOTAL BANDWIDTH CONSUMPTION(KBPS) WITH DIFFERENT NUMBER OF SUBSYSTEMS

Num	GA	HQCE	Q-PSO	BF	SLFA	THMCA
10	825	817	807	823	803	798
20	1643	1630	1611	1641	1601	1591
50	4105	4074	4026	4099	4002	3976
100	8209	8149	8053	8196	8005	7952

TABLE III
COMPARISON OF AVERAGE EXECUTION TIME(SEC) WITH DIFFERENT NUMBER OF SUBSYSTEMS

Num	GA	HQCE	Q-PSO	BF	SLFA	THMCA
10	30	24	22	28	21	17
20	82	65	61	75	58	46
50	214	169	158	196	147	119
100	1504	1189	1112	1376	1076	836

PSO (Q-PSO) [7] for NLCS with up to 100 subsystems in table II.

The execution time is also an important factor. Table III lists the execution time with different size of subsystem obtained by THMCA, SFLA, BF, GA, HQCE, and Q-PSO. It is obvious that the execution time of THMCA increases linearly with the size of the bandwidth scheduling problem. The overall execution time obtained by THMCA is less than that of other methods.

V. CONCLUSIONS

Resource allocation in a networked learning control system is a constrained nonlinear optimization problem. A fair non-cooperation game model has been proposed in this paper. Then, the resource allocation problem is transformed into a problem of settling the equilibrium point of the game model. A new optimization algorithm namely two-layer hierarchical market competition algorithm (THMCA) has been proposed to solve the problem effectively. The proposed method has been tested and compared with a few alternatives. Simulation results show that the computational time and bandwidth consumptions of THMCA are less than other algorithms such as SFLA, GA, BF, quantum-inspired PSO, and hybrid quantum clone evolutionary algorithm. However, the performance of the THMCA also depends on certain parameters selected. Future research includes developing more efficient algorithms and addressing uncertainties in the proposed non-cooperation game model.

ACKNOWLEDGMENT

This work was financially supported by National Natural Science Foundation of China under Grant No.61074032,61104089, Project of Science and Technology Commission of Shanghai Municipality under Grant No.10JC1405000, 11ZR1413100, and UK-China Science Bridge Project under RCUK grant EP/G042594/1.

REFERENCES

[1] T. Yang, *Networked control system: a brief survey*, IEE Proc.-Control Theory Appl, vol. 153, no. 4, pp. 403412, 2006.

[2] D. Du, M. Fei and K. Li, *A two-layer networked learning control system using actor-critic neural network*,Applied Mathematics and Computation, vol. 205, no. 1,pp.2636,2008.

[3] R. M. Salles and J. A. Barria, *Lexicographic maximin optimisation for fair bandwidth allocation in computer networks*, European Journal of Operational Research., vol. 185, no. 2, pp.778794,2008.

[4] W. Deng, K. Li, G. W. Irwin and M. Fei, *Identification and control of Hammerstein systems via wireless networks*, Int. J. of Systems Science, pp. 113, published online Feb 2012.

[5] A. Ganesh, K. Laevens and R. Steinberg, *Congestion Pricing and Noncooperative Games in Communication Networks*, Operations Research, vol. 55, no. 3, pp. 430438,2007.

[6] Q. Chen, Y. Zhong and X. Zhang, *A pseudo genetic algorithm*, Neural Comput & Applic, vol.19, no. 1, pp.77 83, 2010.

[7] J. Zhang, J. Wang and C. Yue, *Small Population-Based PSO for Short-Term Hydrothermal Scheduling*, Power Systems, IEEE Transactions on, vol. 27, no. 1, pp. 142152,2012.

[8] L. Xu and M. Fei, *A Hybrid Quantum Clone Evolutionary Algorithm-Based Scheduling Optimization in a Networked Learning Control System*, Chinese Control and Decision Conference (CCDC) 2010, pp.36323637,2010.

[9] M. Eslamian, S. H. Hosseinian and B. Vahidi, *Bacterial foragingbased solution to the unit-commitment problem*, IEEE Trans. on Power Syst., vol. 24, no. 3, pp. 14781488, 2009.

[10] J. Ebrahimi, S. H. Hosseinian and G. B. Gharehpetian, *Unit Commitment Problem Solution Using Shuffled Frog Leaping Algorithm*, IEEE Transactions on Power Systems, vol. 26, no. 2, pp.19,2011.

[11] L. Xu, M. Fei, T. Jia and T. Yang, *Bandwidth scheduling and optimization using non-cooperative game model-based shuffled frog leaping algorithm in a networked learning control system*, Neural Computing & Applications, published online September 2011.