Computers & Industrial Engineering 57 (2009) 114–124

Contents lists available at ScienceDirect

Computers & Industrial Engineering

journal homepage: www.elsevier.com/locate/caie

An auction-based dynamic bandwidth allocation with sensitivity in a wireless networked control system

Yodyium Tipsuwan [a,*], Suwatchai Kamonsantiroj [a], Jirat Srisabye [a], Prabhas Chongstitvattana [b]

[a] Intelligent Mechatronics Lab, Department of Computer Engineering, Faculty of Engineering, Kasetsart University, Bangkok, Thailand
[b] Department of Computer Engineering, Faculty of Engineering, Chulalongkorn University, Bangkok, Thailand

ARTICLE INFO

Article history:
Available online 17 September 2008

Keywords:
Network
Wireless
Control system
Auction
Game theory

ABSTRACT

Applications of control systems on wireless networks have been widely utilized due to their mobility. However, the performances of these networked control systems (NCS) could be degraded and become unstable by network-induced delays. Existing dynamic bandwidth allocation methods for NCS assign bandwidth to each system with respect to different priorities in an ascending order. However, these NCS may not be given bandwidths at equilibrium such that each of these NCS is satisfied with respect to bandwidth requests of other NCS. Therefore, some NCS may always consume most of given bandwidths, while others may never be given satisfied bandwidths. This paper proposes a dynamic bandwidth allocation methodology that controls bandwidths given to open-loop NCS to be at Nash equilibrium. In this paper, the average sensitivities of NCS are used in utility functions in order to evaluate the effects of network-induced delays for NCS. Then, a control center or an access point will use the proposed methodology to allocate bandwidths for all NCS based on Nash equilibrium. Simulations and experiments were setup from a set of DC motors controlled over a wireless network. Simulation and experimental results show good performances of the proposed methodology compared with three other methods.

1. Introduction

Control over a wireless network is a recently attractive research topic. A wireless network can enhance versatility and capability of a control system because the network enables mobility for a control system and eliminates wiring that is usually expensive to install and maintain. Applications of control systems on a wireless network have been widely utilized such as control and monitoring systems for industrial local area networks (Willig, 1997), automated factories (Jiang, 1998) and smart buildings (Seth, Lynch, & Tilbury, 2005).

In fact, a control system operating on a wireless network is a kind of a networked control systems (NCS) or a network-based control system (NBCS). This system contains a number of interconnected devices that exchange data among each other through a communication network. These data exchanges induce network delays, which have been known to degrade performances and destabilize control systems (Tipsuwan & Chow, 2003a, 2003b). Network delays of both wired and wireless NCS depend on several factors such as network scheduling protocols and network traffic conditions, e.g. network traffic, packet loss, and packet collision (Stalling, 2004). Although a wire-NCS has many advantages among a wireless NCS such as greater bandwidth capacity and reliability of data transfers, wireless control applications are still very attractive due to the greater mobility and cheaper cost.

Various approaches have been proposed to solve delay problems in NCS. In general, these approaches are grouped into two categories. The first category is control algorithm design, which focuses on developing a specific control algorithm to handle network delays on a NCS (Chan & Özgüner, 1995; Luck & Ray, 1990; Nilsson, Bernhardsson, & Wittenmark, 1998). The second category is managing and allocating network resources in order to satisfy the needs of NCS. For example, this second approach can be performed by a scheduling algorithm (Al-Hammouri, Branicky, Liberatore, & Phillips, 2006; Branicky, Phillips, & Zhang, 2002; Hong, 1995; Park, Kim, Kim, & Kwon, 2002; Walsh & Ye, 2001; Yepez, Marti, & Fuertes, 2003) and QoS (quality-of-services) (Abdelzaher, Atkins, & Shin, 2000; Tipsuwan & Chow, 2003a, 2003b) to manage bandwidth and network resources.

Traditionally, applications of allocation and scheduling techniques are based on static strategies such as round robin (RR) (Chaskar & Madhow, 2003), First-In First-Out (FIFO) and priority queuing (Forouzan, 2004). These static scheduling methods may not be able to satisfy an urgent need of a NCS. For example, a NCS, which is tracking a trajectory at a crucial point, may require more bandwidth than typical situations. In this case, a dynamic scheduling and allocation scheme could be a better alternative than static scheduling methods in order to adapt bandwidth and allocate network resources in real-time according to actual

* Corresponding author.
E-mail address: yyt@ku.ac.th (Y. Tipsuwan).

0360-8352/$ - see front matter © 2008 Elsevier Ltd. All rights reserved.
doi:10.1016/j.cie.2008.08.018

requirements of systems in the same network at a specific time frame.

To overcome the limitation of static scheduling, dynamic scheduling and allocation schemes that automatically adapt resource uses according to different situations have been developed. For example, in Yepez et al. (2003), the Large-Error-First (LEF) scheduling algorithm assigned bandwidth to NCS with different priorities based on errors obtained from plant states in real-time. The basic concept of LEF can be explained as follows: (i) if some plants are in the steady state, they will not receive actual resources, (ii) if some plants are in the transient state, the first priority will be assigned to this group, and (iii) the plant with the largest error is given the highest priority. Another dynamic scheduling approach in Walsh and Ye (2001) assigns priorities with respect to weighted errors. A system with a high error is given a high priority. Branicky et al. (2002) proposed a bandwidth allocation scheme by varying sampling periods of NCS based on a congestion level fed back from the network. In addition, Ye, Walsh, and Bushnell (2001) proposed a new MAC protocol P-CSMA/CA for a wireless networked control system based on three priority levels of mixed traffics of real-time control systems. However, most of these dynamic allocation approaches perform priority adjustment based on system errors or predefined priority levels. If some NCS always produce high errors, these NCS may always consume most of the given bandwidths. Therefore, there can be another group of NCS that can never consume bandwidths as they need. As a result, the overall system performance may be low. In addition, with the given bandwidths, some NCS may always be disappointed, whereas some NCS are always appreciated.

One possible approach to solve this problem is to apply an economic concept such as pricing and mechanism design (Semret, 1999) to control bandwidth allocation and resource management to an equilibrium point. In this study, a user obtains bandwidth based on a user's payment per time unit. Pricing enforces a user to considerably evaluate bandwidth requests with respect to its actual need, whereas mechanism design controls bandwidth allocation according to pricing. Some studies of pricing and mechanism design on scheduling and bandwidth allocation can be found in Johari and Tsitsiklis (2003), Lazar and Semet (1997), Maheswaran and Basar (2003), Marbarch and Berry (2002). The approaches in Marbarch and Berry (2002) and Johari and Tsitsiklis (2003) are centralized methods and these methods use an auction with Nash equilibrium to solve a resource allocation problem. On the contrary, Lazar and Semet (1997) and Maheswaran and Basar (2003) proposed decentralized methods. The study in Lazar and Semet (1997) used PSP (progressive second price) technique to provide truthful user reports. PSP is developed by a mechanism design method in order to obtain truthful user preferences. In this method, each user submits both demand quantity and price to pay. In Maheswaran and Basar (2003), each system can individually adapt its bandwidth to an equilibrium point after being given an allocated bandwidth and a price back from the auctioneer. By comparing these methods, we can observe that centralized methods derived based on Nash equilibrium is simple and do not require many data in correspondences between users and an auctioneer. Thus, these methods would be more suitable for allocating bandwidth resources on a very low bandwidth network. On the other hand, a decentralized method could be a better alternative if users do not want a control center or an auctioneer to know their private preferences. However, these methods will require more user data in communications among users and the control center.

In this paper, we introduce a methodology to apply an auction mechanism and game theory for dynamic bandwidth allocation on multiple open-loop wireless NCS. The main objective of the proposed methodology is to assign bandwidths at Nash equilibrium to NCS such that all NCS will be satisfied with their individual performances and the overall system performance. The auction method applied in this paper is a centralized approach, which is operated at a control center or an access point. Our approach can be more suitable for NCS operating on a network with a quite limited amount of bandwidth since a centralized approach does not require correspondences as many as in a decentralized approach. In order to evaluate the effects of network-induced delays to be utilized in the auction, average sensitivities of NCS are applied in utility and payoff functions. These utilities and payoff functions are then used in our methodology to compute and assign bandwidths for all NCS in the network. Three other methods are then compared with the proposed approach to illustrate the effectiveness of our methodology.

This paper is outlined as follows. Section 2 explains our system description. Section 3 describes our auction-based methodology for dynamic bandwidth allocation. Section 4 shows the comparision of bandwidth allocations using four methods including of equal bandwidth, allocation based on sensitivity, optimizing utilities, and our auction-based dynamic allocation. Section 5 shows the simulation and experimental setups. The results are shown in Section 6. Finally, Section 7 concludes the results of this study.

2. System description

The wireless NCS configuration used in this paper is shown in Fig. 1 and is described as follows.

2.1. Wireless network

The wireless network in our consideration is composed of an access point connecting with several NCS. Each NCS contains a pair of a control agent and an action agent. The access point is used to schedule and allocate bandwidth for data transmission among control agents and action agents based on requests from these NCS.

2.2. Control agents and action agents

A control agent is a main controller that controls a remote plant denoted as an action agent. In a general situation, a control agent sends a control command, which can be a set point to an action agent. The action agent then performs its task according to the control command and sends its status back to the control agent. A control agent has to report its bandwidth requirements to the access point for scheduling and bandwidth allocation periodically. Each action agent contains with a controller unit, a plant unit, and a sensor unit. The controller unit receives and converts the control command from the control agent to an actual signal to drive the plant unit.

Let us define T_S, T_P, T_A, T_T, and T_O as the sampling time period, the polling period, the allocation period, the transmission period, and the total operating period. The steps of bandwidth requirement submission are illustrated in Fig. 2 and are described as follows.

1. During T_P, the access point sends polling packets to control agents to ask for bandwidth required.
2. Each control agent receives a polling packet and sends its bandwidth requirement to the access point.
3. The access point completely collects bandwidth requirements from all control agents and allocates bandwidths during T_A for all control agents.
4. Each control agent starts to send its control command with respect to the given bandwidth and the scheduling algorithm used during T_T.

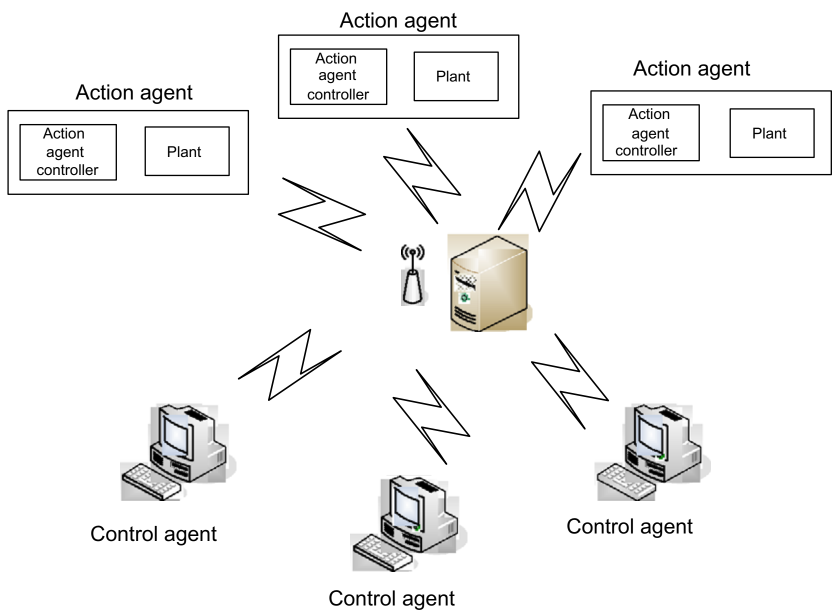

Fig. 1. NCS system configuration.

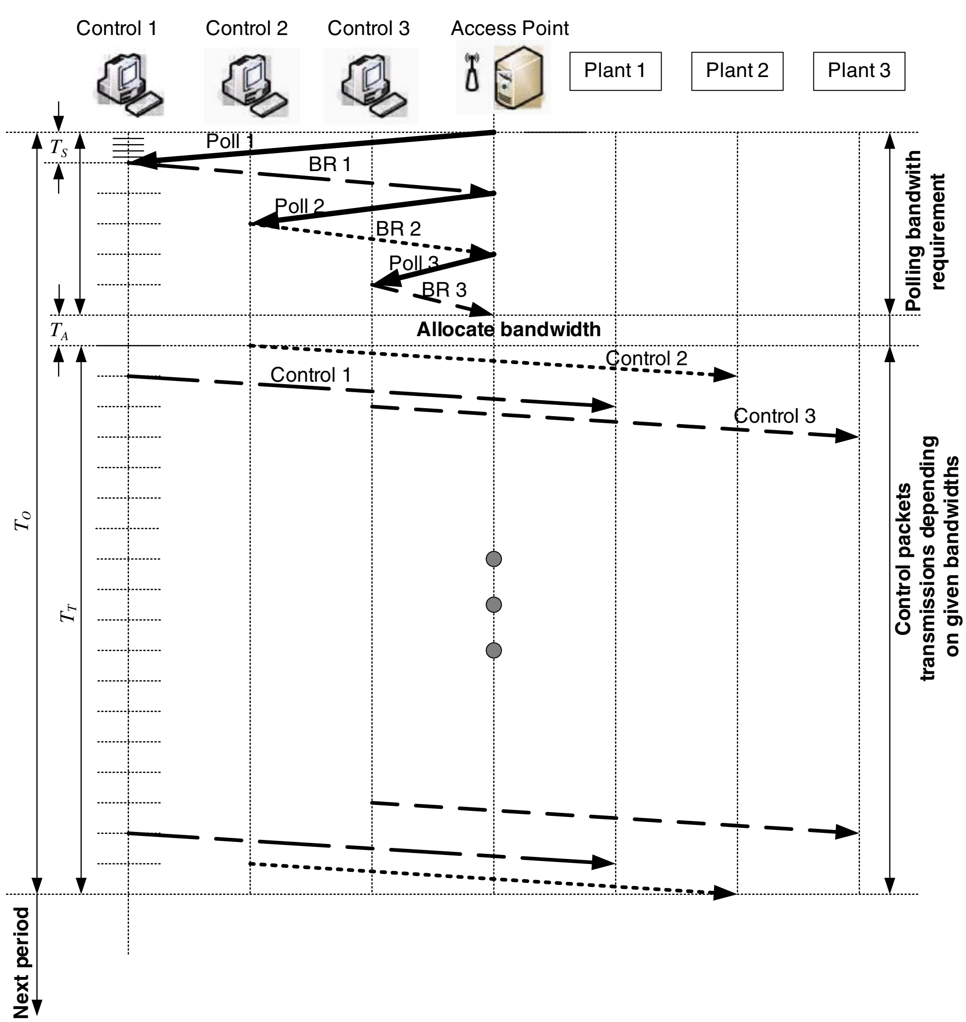

Fig. 2. Timing diagram of bandwidth allocation.

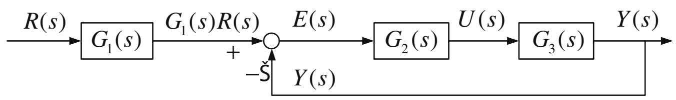

Fig. 3. System formulation.

In this paper, a control agent is assigned to be a set point generator, while an action agent is a PI controller connecting with a plant. During T_T, the control agent sends a set point to the action agent, and then the controller unit will control the plant to reach the set point. This configuration is widely applicable and used in industries such as an industrial mixer (Tunaboylu & Collins, 2004). The sampling time to send the set point reference will then be determined by our methodology. This operation can be mathematically formulated as shown in Fig. 3 and is described as follows.

The control command of the control agent, the action agent control, the plant output, and the error signal of this NCS system in frequency domain are defined as $R(s)$, $U(s)$, $Y(s)$, and $E(s)$, respectively. The transfer function for network time delays from the control agent to the action agent is defined by $G_1(s)$, which has an analytical form as

$$G_1(s) = e^{-\tau s}, \tag{1}$$

where τ is the delay from the control agent to the action agent. The action agent plant transfer function is defined as $G_3(s)$, where the PI controller $G_2(s)$ is described by

$$G_2(s) = \frac{K_P s + K_I}{s}, \tag{2}$$

where K_P and K_I are the proportional gain and the integral gain, respectively. The overall system transfer function $H(s)$ including the network delays becomes

$$H(s) = \frac{Y(s)}{R(s)} = G_1(s)\left(\frac{G_2(s)G_3(s)}{1 + G_2(s)G_3(s)}\right). \tag{3}$$

In this paper, three DC motors are used as plants in our study to demonstrate the proposed methodology. DC motors have been widely used for many industrial applications such as automobiles, robotic manipulators, manufacturing plants and conveyor belts. Thus, DC motor systems will be used to illustrate our methodology in wireless networked control systems through out this paper. The plant dynamics of the DC motor used in each system is described as follows

$$\mathbf{x}(t) = \begin{bmatrix} -\frac{R_a}{L_a} & -\frac{K_b}{L_a} \\ \frac{K_i}{J} & -\frac{B}{J} \end{bmatrix}\mathbf{x}(t) + \begin{bmatrix} \frac{1}{L_a} \\ 0 \end{bmatrix}\mathbf{u}(t), \quad \mathbf{y}(t) = x_1(t), \tag{4}$$

where x_1 is the speed of the motor (rad/s) and x_2 is the armature current (A). The parameters of the motor are shown in Table 1. By substituting parameters in Table 1 in (4), the transfer function of the DC motor can be described as follows:

$$G_3(s) = \frac{193,750,000}{(s + 329.38)(s + 14,889.01)}. \tag{5}$$

Table 1
DC motor parameters

Parameter	Description	Value
R_a	Armature resistance	0.753 Ω
L_a	Armature inductance	0.05 e−03 H
K_a	Back-EMF constant	13.0 e−3 V-s/rad
K_i	Torque constant	12.4 e−3 N-m/A
J	Moment of inertia	12.8 e−7 kg-m^2
B	Viscous-friction coefficient	2.0275 e−4 N-m/rad/s

3. Auction-based methodology for dynamic bandwidth allocation

3.1. Sensitivity of an action agent plant

There are several methods to evaluate an effect of network-induced delays on NCS such as checking stability region (Hong, 1995) or evaluating feedback preprocessor (FP) (Vanijirattikhan, Chow, & Tipsuwan, 2004). In this paper, the severity of the network-induced delay effect on an NCS is evaluated from the average sensitivity of the output change with respect to the reference change under an assumption that there is no packet loss and out-of-order packet. First, let us assume that during a transmission period T_T, the reference $r(t)$ of each agent k is periodically transmitted from the control agent through the network to the plant with the sampling time period T_S as shown in Fig. 4.

Let us define $t_{ij} = iT_O + (T_P + T_A) + jT_S$; $i = 0, 1, 2, \ldots$; $j = 0, 1, 2, \ldots, N$; $t_i = t_{i0}$, and $N = \lfloor T_T/T_S \rfloor$, where N is the total number of reference transmissions during T_T, $\lfloor a \rfloor$ is the largest integer that is less than or equal to a. At $t = t_{ij}$, the control agent sends the reference signal $r(t_{ij})$ to the action agent. For our following explanation, let us denote this newly generate reference at $t = t_{ij}$ as $r^{(n)}(t_{ij})$. Meanwhile, at $t = t_{ij}$, the action agent likely receives a past reference signal, which can be defined as $r^{(p)}(t_{ij}) \in \{r(t_{i-k,j-l})\}$; $k = 0, 1, 2, \ldots$; $l = 0, 1, 2, \ldots, N$; $k \le i, l \le j$. Let us denote $y^{(n)}(t_{ij})$ and $y^{(p)}(t_{ij})$ as the plant outputs from the reference $r^{(n)}(t_{ij})$ and $r^{(p)}(t_{ij})$, respectively. Loosely speaking, if the plant outputs $y^{(n)}(t_{ij})$ and $y^{(p)}(t_{ij})$ are very different compared to a small difference between $r^{(n)}(t_{ij})$ and $r^{(p)}(t_{ij})$, this output difference could imply that the network delay effect on an NCS is severe. On the other hand, if the difference between $y^{(n)}(t_{ij})$ and $y^{(p)}(t_{ij})$ is small even though the difference between $r^{(n)}(t_{ij})$ and $r^{(p)}(t_{ij})$ is large, this output difference could indicate a mild effect from the network delay. Of course, this evaluation has to be based on the same current state of the plant. Based on this concept, the severity of the network-induced delay effect on a NCS could be approximated from the following average sensitivity.

$$\bar{S}_{n,p}(t_i) = \frac{\sum_{j=0}^{N-1} S_{n,p}(t_{ij})}{N}, \tag{6}$$

where

$$S_{n,p}(t_{ij}) = \frac{(y^{(p)}(t_{ij}) - y^{(n)}(t_{ij}))/y^{(n)}(t_{ij})}{(r^{(p)}(t_{ij}) - r^{(n)}(t_{ij}))/r^{(n)}(t_{ij})} \tag{7}$$

In practice, the average sensitivity in (6) is pre-computed and stored in a look-up table along with network-induced delays applied. These data can be obtained by a simulation or an experiment. Although this average sensitivity seems to be quite heuristic, but this evaluation is quite easy to acquire and understand for a practical application. Nevertheless, other evaluations

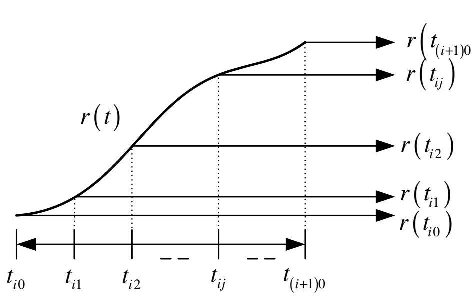

Fig. 4. Transmissions of the reference $r(t)$ during T_T.

could also be used as well depending on different kinds of applications, but these evaluations have to be suitable to be used to form a utility function described in the following section.

To illustrate a delay effect evaluation with the average sensitivity, a simulation was setup in MATLAB/SIMULINK V 7.0 by varying the bandwidth of a NCS in Section 2 in a single network link with the link capacity of 38,400 bps from 100% to 10% with the packet size of 24 bytes. This scenario results in the delay from 0.005 to 0.05 s. The reference signals used are in the following forms:

$$r(t_{ij}) = \frac{1}{(b-a)}(t_{ij} - t_{i0}) + c, \quad b > 0, \tag{8}$$

$$r(t_{ij}) = \begin{cases} 2\left(\frac{t_{ij}-t_{i0}-a}{b-a}\right)^2, & \text{if } t_{ij} - t_{i0} \leqslant \frac{a+b}{2}, \\ 1 - 2\left(\frac{b-t_{ij}+t_{i0}}{b-a}\right)^2, & \text{if } t_{ij} - t_{i0} > \frac{a+b}{2}, \end{cases} \tag{9}$$

By varying two parameters a and b to change shapes of references in (8) and (9), and applying different lengths of delays, the average sensitivities are changed as shown in Fig. 5.

As shown in Fig. 5, changing shapes of the references can affect the average sensitivity $\bar{S}_{n,p}(t_i)$. In this case, increasing the slope of a reference yields the higher average sensitivity. Noticeably, at points that have higher slopes, (9) yields a higher sensitivity than (8). In addition, a longer delay can increase the average sensitivity as well.

3.2. Utility function of NCS

In general, many agents operate on the same network may not, cooperate to each other to share a limited amount of network bandwidth. Typically, every agent requires as much as possible bandwidth. A bandwidth requirement depends on each agent's secret objective and may be not equal to other agent's requirements. Approaches in Walsh and Ye (2001) and Branicky et al. (2002) assign bandwidths depending on priority-based methods. On the other hand, if some NCS always produces high priorities, these NCS may always consume most of the given bandwidths. Therefore, there can be another group of NCS that cannot consume bandwidths as they need. As a result, the overall system performance may be low. In order to increase the total system performance, game theory could be considered.

Based on game theory, the satisfaction level of an agent with respect to the bandwidth granted according to the requirement of an agent can be represented by a utility function $U_k(x_k)$, where $x_k \in [0,1]$ is a ratio of bandwidth granted and $k = 1,\ldots,M$, is the index of an agent.

In this paper, we assume that $U_k(x_k)$ has the following properties as defined in Johari and Tsitsiklis (2003), Maheswaran and Basar (2003). (i) $U_k(x_k)$ is concave and continuously differentiable. (ii) $U'_k(x_k) = \frac{dU_k(x_k)}{dx_k} > 0$, where $U'_k(x_k)$ is denoted as the marginal utility of the agent k. (iii) $U''_k(x_k) \leq 0$.

The assumption 1 can guarantee that there exists a solution for Nash equilibrium. The assumption 2 indicates that $U_k(x_k)$ is an increasing function. Thus, an agent should be more satisfied when it is given a more ratio of bandwidth. The assumption 3 implies that the marginal utility is non-increasing. Therefore, if an agent is already given a high bandwidth, giving more bandwidth to this agent will not significantly increase its satisfaction level.

Several mathematical formulas can be applied as a utility function to describe a satisfaction level depending on different application requirements. In this paper, we define a utility function according to the average sensitivity during a transmission period $t \in [t_{i0}, t_{(i+1)0})$ described by

$$U_k(x_k) = \ln(1 + x_k \bar{S}^k_{n,p}(t_i)) \tag{10}$$

where $\bar{S}^k_{n,p}(t_i)$ is $\bar{S}_{n,p}(t_i)$ of agent k.

4. Bandwidth allocation methodology

In this paper, three other bandwidth allocation methods are used for performance comparisons: (i) equal bandwidth method which is a static bandwidth allocation, (ii) allocation based on sensitivity method, which is a dynamic bandwidth allocation with proportions of average sensitivities, and (iii) optimizing utility method, which is a dynamic bandwidth allocation to optimize the total utilities of all systems subjects to bandwidth ratios. These three methods will then be compared with the auction-based dynamic allocation method, which is the method proposed in this paper. These four bandwidth allocation methods are described as follows:

4.1. Equal bandwidth

Equal bandwidth allocation is a simple static bandwidth allocation method that can be quickly computed. In this method, the ratio of bandwidth given to agent k can be computed by

$$x_k = 1/M. \tag{11}$$

Nevertheless, this bandwidth allocation method can be ineffective since some agents may not receive enough bandwidth according to their demands, while others may receive bandwidths more than their needs.

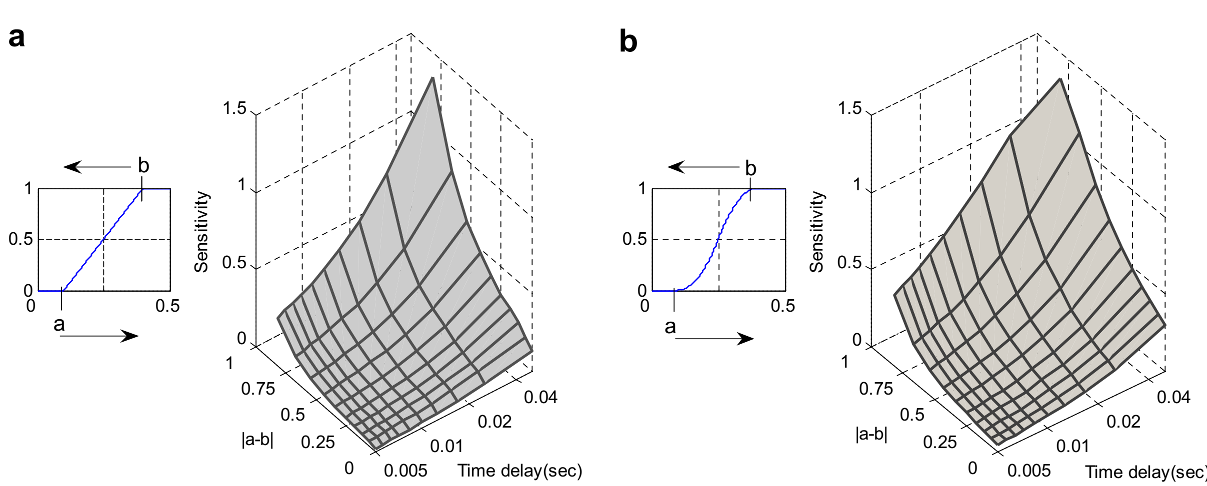

Fig. 5. Average sensitivity with respect to $|a - b|$ and network-induced delay; (a) using the reference in (8), (b) using the reference in (9).

4.2. Allocation based on sensitivity

This method dynamically allocates bandwidths to all agents based on its average sensitivity $\bar{S}_{\mathrm{n,p}}(t_i)$. Agents with high average sensitivities can be given bandwidths more than other agents with lower average sensitivities. The bandwidth ratio given to agent k is described by

$$x_k = \frac{\bar{S}^k_{\mathrm{n,p}}(t_i)}{\sum_{m=1}^{M} \bar{S}^m_{\mathrm{n,p}}(t_i)}. \tag{12}$$

For example, let sensitivities of three NCS, which are computed in advance, be represented in a vector $\bar{\mathbf{S}}_{\mathrm{n,p}}(t) = [\bar{S}^1_{\mathrm{n,p}}(t)\ \bar{S}^2_{\mathrm{n,p}}(t)\ \bar{S}^3_{\mathrm{n,p}}(t)]^{\mathrm{T}} = [1.50 \quad 1.83 \quad 1.29]^{\mathrm{T}}$. The result is $\mathbf{x} = [0.33\ 0.39\ 0.28]^{\mathrm{T}}$, where $\mathbf{x}$ is the allocated bandwidth vector. A disadvantage of this method is that each agent can lie by reporting a faked demand in order to gain more bandwidth.

4.3. Optimizing utilities

This method formulates the allocation problem as an optimization and uses the average sensitivities of agents as parameters similar to the allocation based on sensitivity method. The objective of this optimization is to maximize the total utility of all agents in the network as follows:

$$\max \sum_{k=1}^{M} U_k(x_k), \tag{13}$$

$$\text{subject to} \quad \sum_{k=1}^{M} x_k \le 1, \quad x_k \ge 0,\ k = 1, \ldots, M,$$

For example, if we use the same vector $\bar{\mathbf{S}}_{\mathrm{n,p}}(t) = [1.50 \quad 1.83 \quad 1.29]^{\mathrm{T}}$, the utility functions of three NCS can be represented in a vector $\mathbf{U} = [U_1 \quad U_2 \quad U_3]^{\mathrm{T}} = [\ln(1 + 1.50x_1)\ \ln(1 + 1.83x_2)\ \ln(1 + 1.29x_3)]^T$. Solving this optimization problem by Lagangian yields $\mathbf{x} = [0.33\ 0.45\ 0.22]$. Nevertheless, this method still has the same disadvantage as the allocation based on sensitivity method because there is no pricing mechanism to prevent the occurrence of selfish systems which choose $\bar{S}_{\mathrm{n,p}}(t)$ to maximize its own utility. The whole resources may be allocated to only one system.

4.4. Auction-based dynamic allocation

This is the proposed method. Since all agents have to compete for a limited bandwidth, interactions among these competing agents in the same network can be modeled and described in a game-theoretic framework. In this framework, these agents are considered to be the players of a game, where each agent has to pay an amount of money to gain bandwidth with respect to its requirement in an auction with other agents during the allocation period T_{A}. Our methodology requires that this auction is centrally managed by an access point. In general, typical procedures for an agent to bid for a network bandwidth in an auction are:

1. An agent submits a price according to its bandwidth requirement.
2. The access point allocates bandwidths and informs all agents about the bandwidths that will be given.
3. The agent pays for the given bandwidth.

Typically, every agent does not want to pay for an overpricing bandwidth. Therefore, in an auction, each agent has to predict other agents' plays and choose its own payment to maximize its own profit. In our methodology, each agent k is initially given the same amount of money w, which is the maximum price that an agent can pay. Each agent has a right to individually determine how to spend this money in an allocation period T_{A} and the amount of money will be reset to w at the beginning of the next allocation period. The price of bandwidth x_k defined as s_k is assumed to be calculated from

$$s_k = \lambda x_k, \tag{14}$$

where $s_k \in [0, \mathrm{w}]$ is the price strategy of bandwidth ratio given to agent k, λ is the unit price of bandwidth ratio. The price that is initially submitted by agent k for bidding in an auction may be different from price that the agent has to pay. The final price to pay is depended on the bandwidth allocated by the access point. After an allocation by the access point, s_k may be updated to another value. The appreciation of agent k according to the price to pay s_k can be evaluated from a payoff function described by

$$P_k(x_k) = \begin{cases} U_k(x_k) - s_k, & \text{if } s_k > 0, \\ U_k(0), & \text{if } s_k = 0, \end{cases} \tag{15}$$

Each agent has to individually determine its bid in order to achieve a satisfactory payoff. Noticeably, the payoff function in this framework enforces all agents not to request an over bandwidth. If an agent asks for a much larger bandwidth than its actual need, its utility will be deducted by a price to pay resulting in a low payoff. Therefore, to avoid this situation, each agent has not to overly spend its money.

Since all agents always want to maximize its own payoff, the access point has to allocate bandwidths such that all agents will not want to change their bids to acquire a higher bandwidth. A possible way to achieve this goal is to allocate bandwidths at Nash equilibrium. At Nash equilibrium, each agent will not try to increase or decrease its bid to gain more bandwidth because each agent may receive a less bandwidth with respect to other agents' bids. Nash equilibrium is defined as follows:

$$P_k(s^*_k; \mathbf{s}^*_{-k}) \geqslant P_i(s_k; \mathbf{s}^*_{-k}), \tag{16}$$

where $P_k(s_k; \mathbf{s}_{-k})$ is the payoff function of agent k given the other users' bids strategy $\mathbf{s}_{-k} = [s_1 \quad \ldots \quad s_{k-1}\ s_{k+1} \quad \ldots \quad s_{\mathrm{M}}]^{\mathrm{T}}$ and $s^*_k \in [0, w]$ is the best payment of agent k at Nash equilibrium, where agent k does not want to change its bid. The asterisk indicates that $\mathbf{s}^*_{-k}$ is the best bandwidth prices for other agents. The following theorem will be used to allocate bandwidths according to Nash equilibrium.

Theorem 1. *For each agent k, if the utility function $U_k(x_k)$ is concave and continuously differentiable, then the existence of a unique Nash equilibrium is guaranteed and satisfies $\sum_{k=1}^{M} S_k > 0$. Furthermore, $x^*_k = \frac{s^*_k}{\sum s^*_{-k}}$ is the unique solution and there exists a unique scalar unit price $\lambda = U'_k(x^*_k)(1 - x^*_k)$ to the following problem*

$$\max_{s_k} P_k(s_k; \mathbf{s}^*_{-k}) = \max_{s_k} \left(U_k \left(\frac{s_k}{\sum S^*_{-k}} \right) - s_k \right), \quad \text{subject to } \sum_{k=1}^{M} x_k \leqslant 1, \quad x_k \geqslant 0 \tag{17}$$

where $\sum s_{-k}$ is the summation of bandwidth prices of all agents except s_k, $\sum s^*_{-k}$ is the summation of the best bandwidth prices for other agents and the allocation rule is $x_k = \frac{s_k}{\sum s_{-k}}$. Initially, $\sum s_{-K}$ could be computed by predicting other agents' bids and will become $\sum s^*_{-K}$ at the equilibrium point. Therefore, several bidding iterations maybe required until all agents can reach the equilibrium point. Since there exists a unique scalar λ and a constraint $\sum_{k=1}^{M} x_k \leqslant 1$, the agent payment is $s_k = \lambda x_k$, and the vector $\mathbf{s} = \lambda \mathbf{x}$ results in a Nash equilibrium. The proof of this theorem can be found in Johari and Tsitsiklis (2003).

Nevertheless, if these bidding steps are normally performed like a generic auction mechanism in game theory, all agents will have to repeatedly adjust their bids until Nash equilibrium is obtained. These steps require some amount of time to reach Nash equilibrium.

Instead of finding Nash equilibrium by letting all agents adjust their bids in several iterations like other auction problems, our methodology uses the solution $x_k = s_k/\sum s_{-k}$ in (17), which already results in Nash equilibrium, as a given bandwidth for agent k. However, in this methodology, each agent k has to summit its utility function to the access point, which can be indirectly performed by submitting $\bar{S}^k_{\mathrm{n,p}}(t_i)$ instead. The access point will then use $\bar{S}^k_{\mathrm{n,p}}(t_i)$ to find the utility function for agent k by (10). Then, to obtain the solution x_k, the unique λ for (17) has to be found as follows:

- *Step 1:* The expression $s_k/\sum s^*_{-k}$ is a strictly concave function of s_k, $s_k \geqslant 0$ when $\sum s^*_{-k}$ is a fixed and $U_k(x_k)$ is a strictly increasing and concave function by Theorem 1. Therefore, $P_k(s_k;\mathbf{s}^*_{-k})$ is strictly concave in s_k; under the assumption that the utility is measured in the same monetary unit as s_k.
- *Step 2:* The first order necessary condition to find the maximal solution is

$$P'_k(s_k;\mathbf{s}^*_{-k}) = U'_k\left(\frac{s_k}{\sum s^*_{-k}}\right)\left(\frac{\sum s^*_{-k} - s_k}{\left(\sum s^*_{-k}\right)^2}\right) - 1 = 0, \tag{18}$$

After multiplying (18) by $\sum s^*_{-k}$, this equation can be rewritten as follows:

$$U'_k\left(\frac{s_k}{\sum s^*_{-k}}\right)\left(1 - \frac{s_k}{\sum s^*_{-k}}\right) = \sum s^*_{-k}, \tag{19}$$

From Step 1, $P_k(s_k;\mathbf{s}^*_{-k})$ is strictly concave and continuously differentiable in $s_k \geqslant 0$. Thus, s_k must be the unique maximal solution of $P_k(s_k;\mathbf{s}^*_{-k})$ over $s_k \geqslant 0$. Besides, the condition in (19) implies that s_k maximizes $P_k(s_k;\mathbf{s}^*_{-k})$ and the vector $\mathbf{s} = [s^*_k, s^*_{-k}]$ is unique Nash equilibrium. Therefore, there exists $x_k = s^*_k/\sum s^*_k$, which is also the unique.

- *Step 3:* Since there exists a unique vector $\mathbf{x}$, Lagrange method can be applied to find the scalar unit price λ subjects to $\sum_{k=1}^{M} x_k = 1,\ x_k \geqslant 0$ as follows:

$$L(\mathbf{x},\lambda) = P_k(x_k) - \lambda\left(\sum_{k=1}^{M} x_k - 1\right), \tag{20}$$

The first order necessary condition to find the maximal solution is

$$U'_k(x_k)(1 - x_k) = \lambda,\ x_k > 0, \tag{21}$$

where λ is the Lagrange multiplier and it refers to the scalar unit price. The constraint $\sum_{k=1}^{M} x_k = 1,\ x_k \geqslant 0$ since the total bandwidth ratio from all agents must be equal to 1. By using $U'_k(x_k)$ from every agent k and this constraint, λ can be obtained. Then, the bandwidth ratio vector $\mathbf{x}$ can found from $U'_k(x_k)$ and λ, which is unique.

Although $\mathbf{s} = \lambda\mathbf{x}$ can be found and is conceptually known as the solution of Nash equilibrium in theorem 1, $\mathbf{s}$ will not be practically used in our methodology. The access point can be thought of as it holds the money of all agents and deduces the money of each agent at every time that bandwidths are allocated. Therefore, each agent does not practically pay the money so that $\mathbf{s}$ is not used.

For example, if an average sensitivity vector is $\bar{\mathbf{S}}_{\mathrm{n,p}}(t) = [1.50\quad 1.83\quad 1.29]^{\mathrm{T}}$, the utilities of these three agents in this case are $U_1(x_1) = \ln(1 + 1.5x_1)$, $U_2(x_2) = \ln(1 + 1.83x_2)$, and $U_3(x_3) = \ln(1 + 1.29x_3)$, respectively. The bandwidth vector allocated by the access point in this case is $\mathbf{x}^* = [0.34\ 0.38\ 0.28]^{\mathrm{T}}$. However, the bandwidths computed here may be slightly changed by a scheduling policy of a scheduling algorithm used.

5. Simulation and experimental setup

5.1. Simulation setup

To illustrate the proposed methodology and verify its performance, MATLAB/SIMULINK V 7.0 was used to simulate dynamic bandwidth allocation. This simulation has three NCS, which consists of three control agents and three motor plants with three action agent controllers in a single link wireless network with the bandwidth of 38,400 bits/s. In this simulation, $T_{\mathrm{O}} = 0.5$ s, $T_{\mathrm{S}} = 0.005$ s, $T_{\mathrm{P}} = 0.03$ s, and the rest of T_{O} is T_{T}. The allocation period T_{A} is very short compared to T_{S} so that T_{A} can be assumed to be zero. The overall system configuration is shown in Fig. 6.

In this simulation, a reference signal $r_k(t)$ will then be delayed by τ_k according to the given bandwidth x_k and a scheduling algorithm used.

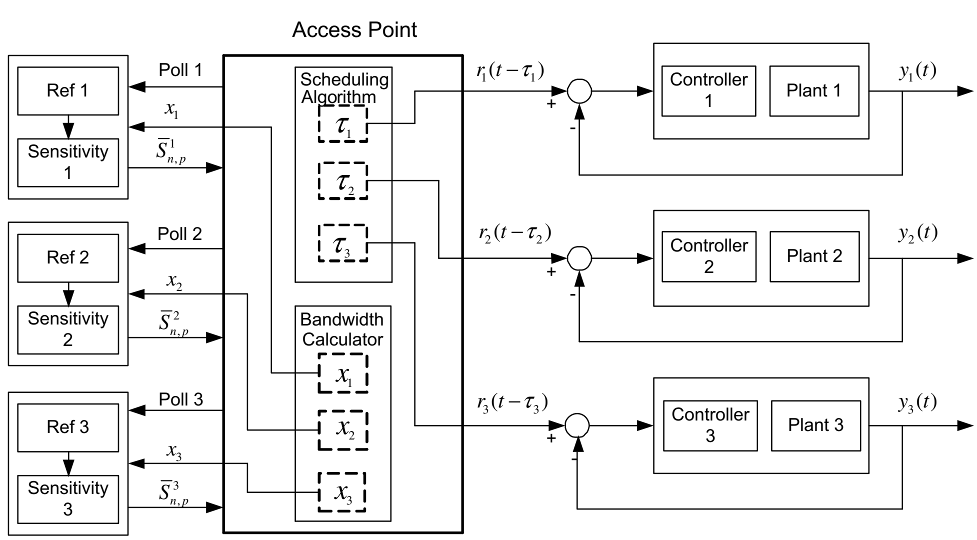

Fig. 6. Simulation setup.

5.2. Experimental setup

An experiment of bandwidth allocation using the proposed auction-based method is set up to investigate the performances of actual agents. The block diagram of the experiment and the actual experiment setup are shown in Fig. 7(a) and (b), respectively.

The access point is implemented based on a dsPIC30F4011 Microchip microcontroller, while each control agent is implemented on a dsPIC30F2010 Microchip microcontroller. Each action agent is also implemented on a dsPIC30F2010 microcontroller with the PI control algorithm. The DC motors used as plants in this experiment are Maxon A-max 226802. These motors are driven by L298 dual full-bridge motor drivers.

All wireless network communications are performed via TRF2.4 GHz wireless modules. The bit rate of the communication is assigned to be 38,400 bits/s. The packet format, which is used for requesting a bandwidth of each user and sending the control voltage signal, is shown in Fig. 8.

The first 14-byte of a packet is used as preamble part. The next 2-byte indicates the starting point of a packet following by the 2-byte address. The data size in this packet is 4 bytes. The packet ends with the 2-byte CRC for error checking.

5.3. Scheduling algorithm

Various scheduling algorithms can be used to assign bandwidths to agents. However, our main focus is not a design of a scheduling algorithm, but a dynamic bandwidth allocation. Thus, to illustrate our approach, a simple round-robin scheduling is then used in the access point to compare the control performance among different bandwidth allocation methods. To perform scheduling, the access point is assumed to have time slots for reference transmission of agents. The total number of time slots is defined as l. The size of slot times defined as $T_L = T_T/l$ depends on the performance of the access point. In this scheduling, each agent k will occupy a set of time slots indicated by L_k from the set of all unoccupied time slots denoted by L. For example, as

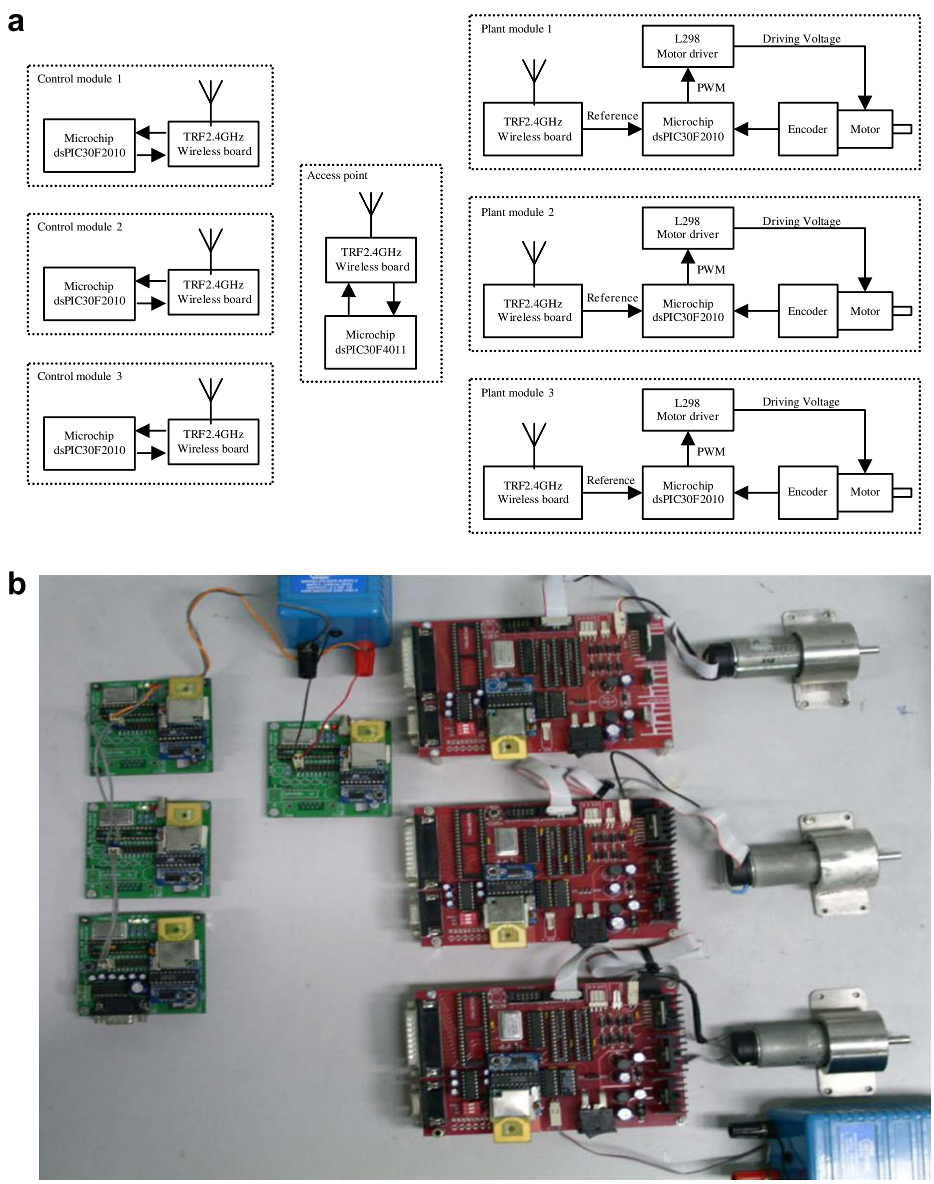

Fig. 7. Experimental setup: (a) block diagram, (b) actual experimental setup.

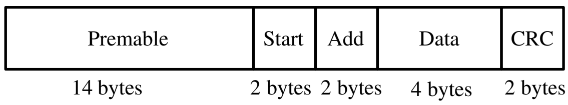

Fig. 8. Packet format.

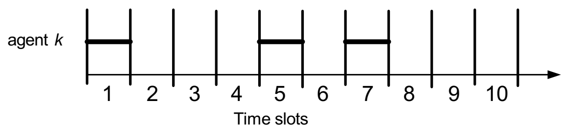

Fig. 9. Time slot occupation of agent k.

shown in Fig. 9, $L_k = \{1,5,7\}$ implies that agent k occupies the 1st, 5th, and 7th time slots from $|L| = 10$ previously unoccupied time slots. Thus, L becomes $L = \{2,3,4,6,8,9,10\}$.

The scheduling algorithm used is described as follows.

1. Sort all agents with respect to x_k in a descending order and reindex all agents with k. Thus, at this point, agent 1 has the highest bandwidth ratio.
2. Compute the number of time slots given to each agent k by $l_k = \lfloor l x_k \rfloor$.
3. Set the index $k = M$.
4. Initialize counter $a = \min p, p \in L$, and $c = 0$, and $L_k = \{\}$.
5. Compute the gap size among time slots of agent k by $h_k = \lceil 1/x_i \rceil$.
6. While $c \leqslant l_k$ and $b = a + ch_k \leqslant l$,
   a. If $b \notin L$,
      Update $b \leftarrow b + 1$.
      If $b > l$
         End the scheduling procedure because all time slots have been occupied.
      End If.
      Go back to 6a.
   End If.
   Add b to L_k, update $c \leftarrow c + 1$ and $L \leftarrow L - \{b\}$, and go back to 6.
   End While.
7. Update $k \leftarrow k - 1$.
8. If $k \neq 0$, go back to 4.

5.4. Performance measure

To compare performances of bandwidth allocation methods, the following overall performance measure at the allocation time t_i is used

$$J(t_i) = \sum_{k=1}^{M} V_k(t_i), \tag{22}$$

$$V_k(t_i) = \frac{1}{P} \sum_{p=0}^{P} |y_{\mathrm{nom},i}(t_p) - y_{\mathrm{act},i}(t_p)|, \tag{23}$$

where $t_\mathrm{p} = p\Delta t, t_i + (T_\mathrm{P} + T_\mathrm{A}) \le t_\mathrm{p} < t_{i+1}$ is the output measurement sampling time, p is the time index of output measurement, Δt is a time period, P is the total number of sampling points ($P = 100$), $y_{\mathrm{nom},i}(t_\mathrm{P})$ is the nominal plant output at time t_p measured from a plant without network-induced delays during reference transmissions, $y_{\mathrm{act},i}(t_\mathrm{P})$ is the plant output at time t_p measured from a plant using an allocation method in this paper. This performance measure is used to evaluate how effectively each allocation method can maintain the control system performance due to network resource sharing.

6. Simulation and experimental results

6.1. Average sensitivity and allocated bandwidth result

In our simulation, there are three time intervals, which are 0–0.5 s, 0.5–1 s, and 1–1.5 s used to illustrate the effectiveness of our methodology. These intervals are denoted as allocation periods 1, 2, and 3, respectively. Fig. 10(a)–(c) show different references used for agents 1 to 3 each allocation period. These references result in the average sensitivities as shown in Table 2. We have known that an average sensitivity depends on a reference signal. For example, during 0—0.5 s, agent 1 has two linear references according to (8). One has a=0.1 and b=0.3, while another has a=0.3 and b=0.5. These two linear references have $|a - b| = 0.2$. The average sensitivity of agent 1 is then obtained from the sensitivity surface shown in Fig. 5. Thus $\bar{S}^1_{\mathrm{n,p}}(t) = 1.97$. By using the same procedure for the agent 1, $\bar{S}^2_{\mathrm{n,p}}(t) = 2.64$ based on the reference that is composed of four sigmoid references, which has $|a - b| = 0.1$. Likewise, $\bar{S}^3_{\mathrm{n,p}}(t) = 1.39$ based on a linear reference signal with $|a - b| = 0.4$.

The allocated bandwidth in each allocation period of all agents are shown in Fig. 10(d)–(f). For example, during 0.5 s to 1 s, the average sensitivity vector is $\bar{S}_{\mathrm{n,p}}(t) = [1.97 \quad 2.64 \quad 1.39]$. Bandwidths allocated in the allocation period 1 by the four methods used for agent 1, 2 and 3 are [33 33 36 25], [33 44 61 49] and [34 23 3 26], respectively. These allocated bandwidth vectors of the motors are obtained by using (11)–(13), and (17), respectively.

6.2. Simulation performance measure

Differences of output measurements $y_{\mathrm{nom},i}(t_p) - y_{\mathrm{act},i}(t_p)$ are depicted in Fig. 10(g)–(i), whereas the individual performance measures $V_k(t_i)$ for all agents and the overall performances $J(t_i)$ are shown in Tables 3 and 4, respectively.

As shown in Fig. 10 and Table 3, individual performance measures increase when agents are given small numbers of bandwidths. The equal bandwidth allocation method always allocates an equal bandwidth to all agents. Therefore, some agents are given unnecessary bandwidth compared to others. For example, during 0.5–1 s, the plant of agent 3 has to be run at a constant speed of 1 rad/s. As indicated by its average sensitivity of 1.00 in Table 2, a small bandwidth is likely required to send the reference for the agent 3. As a result, the overall performance J in this case is quite bad.

With the allocation based on sensitivity method, an agent is given an allocated bandwidth with respect to its average sensitivity in Table 2 without considering the performances of other agents. Even though the individual performance of an agent in some periods may be high, the overall performance J compared to the other methods may not be optimal as shown in Table 4.

On the other hand, optimizing utilities of all agents can allocate bandwidths with respect to the overall performance of all agents. Although the overall performance is optimized, some agents may consume most of available bandwidth without caring some other agents. For Example, during 0–0.5 s, agent 2 requires high bandwidth due to a high average sensitivity of 2.64. As a result from this allocation method, agent 2 is given a very high bandwidth, while agent 3 with the average sensitivity of 1.39 is given a very low bandwidth as shown in Fig. 10(e)–(f).

By comparing with other allocation methods, the auction-based dynamic bandwidth allocation method gives the best overall performance since this method enforces bandwidth allocation to be at Nash equilibrium as shown in Table 4.

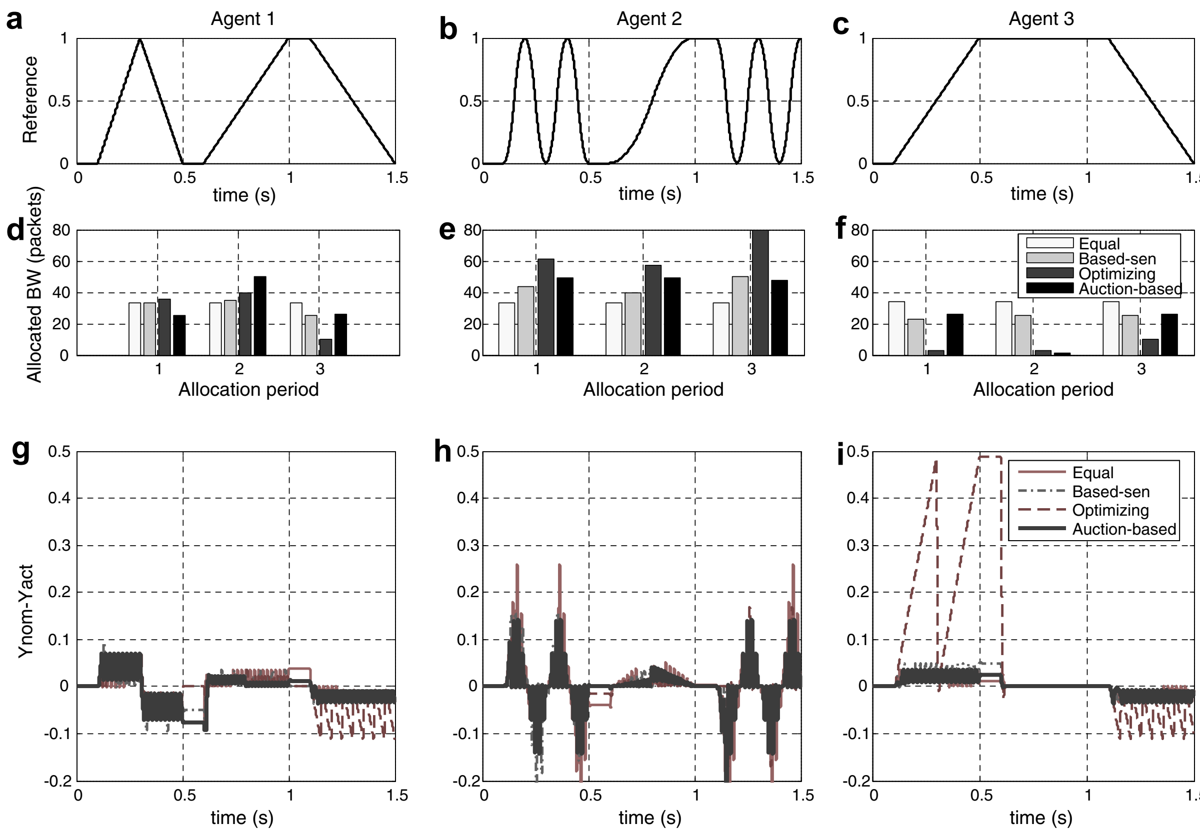

Fig. 10. Simulation result.

Table 2
Average sensitivities of agents

Agent	Average sensitivity		
	Period 1	Period 2	Period 3
1	1.97	1.39	1.39
2	2.64	1.58	2.64
3	1.39	1.00	1.39

Table 4
Overall performance measures $J(t_i)$ of agents 1–3 in three allocation periods

Allocation method	Period 1	Period 2	Period 3
Equal bandwidth allocation	0.1059	0.0305	0.0908
Allocation based on sensitivity	0.0982	0.0252	0.0729
Optimizing utilities	0.2820	0.0219	0.1264
Auction-based dynamic allocation	0.0918	0.0201	0.0728

6.3. Experimental performance measure

Fig. 11 shows the error between the nominal output and the actual motor output of all agents, while the overall performance J from four methods used in this paper are shown in Table 5.

As shown in Fig. 11 and Table 5, the overall performance J is better than the performance in the previous simulation due to some noises. However, the auction-based dynamic allocation still gives the best performances among other methods according to the simulation results.

Table 3
Individual performance measures $V_k(t_i)$ of agent 1 to 3 in 3 allocation periods

	A allocation method	Period 1	Period 2	Period 3
Agent 1	Equal bandwidth allocation	0.0313	0.0163	0.0160
	Allocation based on sensitivity	0.0305	0.0141	0.0184
	Optimizing utilities	0.0234	0.0111	0.0554
	Auction-based dynamic allocation	0.0369	0.0108	0.0175
Agent 2	Equal bandwidth allocation	0.0594	0.0134	0.0595
	Allocation based on sensitivity	0.0451	0.0106	0.0364
	Optimizing utilities	0.0186	0.0061	0.0156
	Auction-based dynamic allocation	0.0374	0.0091	0.0378
Agent 3	Equal bandwidth allocation	0.0152	0.0008	0.0153
	Allocation based on sensitivity	0.0226	0.0005	0.0181
	Optimizing utilities	0.2400	0.0047	0.0554
	Auction-based dynamic allocation	0.0175	0.0002	0.0175

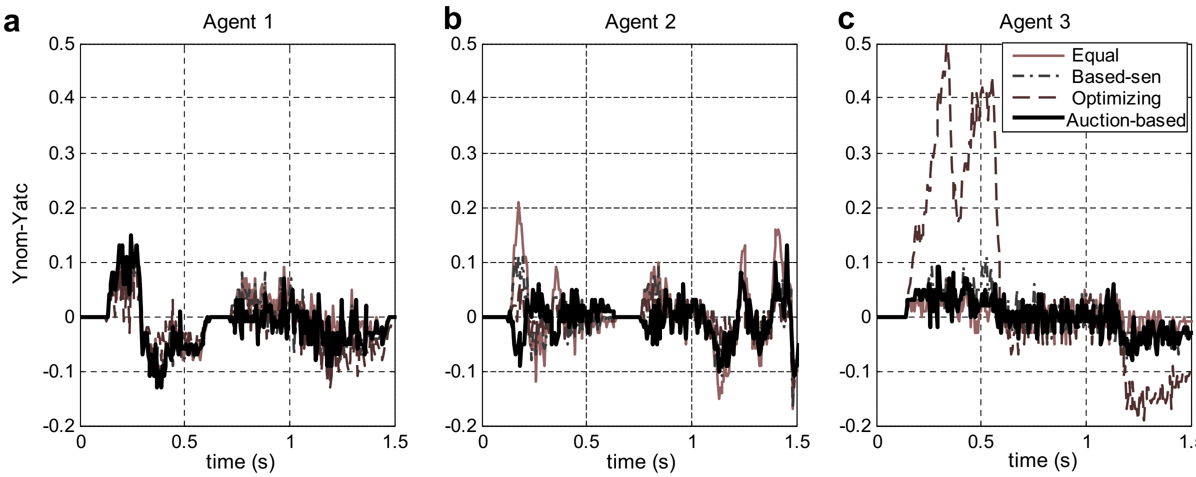

Fig. 11. Experimental result.

Table 5
Overall performance measures $J(t_i)$ of all agents in three allocation periods in the experiment

Allocation method	Period 1	Period 2	Period 3
Equal bandwidth allocation	0.1061	0.0650	0.1119
Allocation based on sensitivity	0.1107	0.0566	0.1111
Optimizing utilities	0.2598	0.0614	0.1873
Auction-based dynamic allocation	0.0999	0.0464	0.1094

7. Conclusion

In this paper, we proposed an auction-based dynamic allocation methodology to allocate bandwidth for open-loop NCS based on game theory. Average output sensitivities of NCS is used to indicate bandwidth requirement in a form of a utility function. The auction-based dynamic allocation methodology enforces all agents sharing the same network to have allocated bandwidths at Nash equilibrium. Pricing of the payoff function enforces all agents not to request an over bandwidth. Both simulation and experimental results have shown the effectiveness of the proposed methodology very well. Although the methodology in this paper was not implemented on a standard wireless protocol, it is still useful to be applied on some specific or custom configurations. Also, this methodology may be adapted to apply on closed-loop NCS with some modifications. This method can apply to other resource managements in which each agent requires different allocated resources. This methodology may be further improved, investigated, and adapted on several issues. For example, other delay effect evaluations rather than average sensitivities may be possibly applied. These additional issues could be studied in the future.

Acknowledgement

This research study is supported by The Thailand Research Fund (TRF) with the contract MRG 4780121.

References

Abdelzaher, T. F., Atkins, E. M., & Shin, K. G. (2000). QoS negotiation in real-time systems and its application to automated flight control. *IEEE Transactions on Computers, 49*(11), 1170–1183.
Al-Hammouri, A. T., Branicky, M. S., Liberatore, V., & Phillips, S. M. (2006). Decentralized and dynamic bandwidth allocation in networked control systems. In *IEEE Parallel and Distributed Processing Symposium*, Rhodes Island, Greece.
Branicky, M. S., Phillips, S. M., & Zhang, W. (2002). Scheduling and feedback co-design for networked control systems. In *IEEE CDC'02*, Las Vegas, USA.
Chan, H., & Özgüner, Ü. (1995). Closed-loop control of systems over a communication network with queues. *International Journal of Control, 62*(3), 493–510.
Chaskar, H. M., & Madhow, U. (2003). Fair scheduling with tunable latency: A round-robin approach. *IEEE/ACM Transactions on Networking, 11*(4), 592–601.
Forouzan, B. A. (2004). *Data communications and networking* (3rd ed.). Singapore: McGraw Hill.
Hong, S. H. (1995). Scheduling algorithm of data sampling times in the integrated communication and control systems. *IEEE Transactions on Control Systems Technology, 3*(2), 225–230.
Jiang, S. (1998). Wireless communications and a priority access protocol for multiple mobile terminals in factory automation. *IEEE Transactions on Robotics and Automation, 14*(1), 137–143.
Johari, R., & Tsitsiklis, J. (2003). Network resource allocation and a congestion game: single link case. In *IEEE CDC'03*, Hawaii, USA.
Lazar, A. A., & Semet, N. (1997). *Auction for network resource sharing*. CTR Technical Report, Telecommunication Research, Columbia University, New York, USA.
Luck, R., & Ray, A. (1990). An observer-based compensator for distributed delays. *Automatica, 26*(5), 903–908.
Maheswaran, R. T., & Basar, T. (2003). Equilibrium and negotiation in multiple resource auctions. In *IEEE CDC'03*, Hawaii, USA.
Marbarch, P., & Berry, R. (2002). Downlink resource allocation and pricing for wireless networks. In *IEEE INFOCOM*, New York, USA.
Nilsson, J., Bernhardsson, B., & Wittenmark, B. (1998). Stochastic analysis and control of real-time systems with random time delays. *Automatica, 34*(1), 57–64.
Park, H. S., Kim, Y. H., Kim, D. S., & Kwon, W. H. (2002). A scheduling method for network-based control systems. *IEEE Transactions on Control Systems Technology, 10*(3), 318–330.
Stalling, W. (2004). *Data and computer communications* (7th ed). Englewood Cliffs, NJ: Prentice Hall.
Semret, N. (1999). *Market mechanisms for network resource sharing*. Ph.D. thesis, Columbia University, New York, USA.
Seth, S., Lynch, J. P., & Tilbury, D. M. (2005). Wirelessly networked distributed controllers for real-time control of civil structures. In *American Control Conference*, Oregon, USA.
Tipsuwan, Y., & Chow, M.-Y. (2003a). Control methodologies in networked control systems. *Control Engineering Practice, 11*(10), 1099–1111.
Tipsuwan, Y., & Chow, M.-Y. (2003b). Gain adaptation of networked DC motor controllers based on QoS variations. *IEEE Transactions on Industrial Electronics, 50*(5), 936–943.
Tunaboylu, N. S., & Collins, E. R. (2004). Response analysis of an industrial mixer drive subjected to voltage sages Part II: Experimental design and results. In *IEEE International Conference on Harmonics and Quality of Power*, New York, USA.
Vanijirattikhan, R., Chow, M.-Y., & Tipsuwan, Y. (2004). Feedback preprocessed unmanned ground vehicle network-based controller characterization. In *IEEE IECON'04*, New York, USA.
Walsh, G. C., & Ye, H. (2001). Scheduling of networked control systems. *IEEE Control Systems Magazine, 21*(1), 57–65.
Willig, A. (1997). A MAC protocol and a scheduling approach as elements of a lower layers architecture in wireless industrial LANs. In *IEEE International Workshop on Factory Communication Systems*, Barcelona, Spain.
Ye, H., Walsh, G. C., & Bushnell, L. G. (2001). Real-time mixed-traffic wireless networks. *IEEE Transactions on Industrial Electronics, 48*(5), 883–890.
Yepez, J., Marti, P., & Fuertes, J. M. (2003). Control loop scheduling paradigm in distributed control systems. In *IEEE IECON'03*, Virginia, USA.