Research Article

Game Theoretic Request Scheduling with Queue Priority in Video Sensor Networks

Jia Zhao, Jianfeng Guan, Changqiao Xu, and Wei Su

1 National Engineering Laboratory for Next Generation Internet Interconnection Devices, Beijing Jiaotong University, Beijing 100044, China
2 State Key Laboratory of Networking and Switching Technology, Beijing University of Posts and Telecommunications, Beijing, China
3 Institute of Sensing Technology and Business, Beijing University of Posts and Telecommunications, Wuxi, Jiangsu, China

Correspondence should be addressed to Jia Zhao; 1111004@bjtu.edu.cn

Received 1 November 2013; Accepted 8 February 2014; Published 30 March 2014

Academic Editor: Hongke Zhang

Copyright © 2014 Jia Zhao et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Video sensor networks have been widely used to monitor environment and report abnormality. Each node collects video data, select a head node, and transmit the data to the head, and then the head reports the data to the base station. A head has to process both normal and abnormal data-reporting requests from its nearby nodes. To achieve QoS of surveillance, previous request scheduling methods minimize the data transmission delay or blocking rate but no comprehensive way was studied in the literature. In this paper, we propose a game strategic request scheduling based on a queue priority model in which a handover mechanism ensures that the abnormal requests are processed in time. In the game, video sensors select their heads to decide the arriving rates of both normal and abnormal requests; the heads decide the probability of handing over the abnormal requests. At the Nash Equilibrium Point (NEP), the normal data requesters optimize mean delay, the abnormal data requesters optimize mean blocking rate, and the heads balance the request load on them. Numerical analysis shows that the game strategic scheduling outperforms other scheduling methods that consider single objective (minimum delay or minimum blocking rate).

1. Introduction

Video sensor networks act as an efficient and reliable way to monitor environment and discover emergency or accidents [1, 2]. Each node of a sensor network collects video data and transmits the data to a nearby head, which is also a sensor and selected within a cluster. The head processes the data-reporting requests from members of the cluster and transmits the data to a sink node or a base station. The monitoring quality of service (QoS) of a video sensor network depends on how the data-reporting requests are scheduled and processed [2]. In order to collect data in time, a scheduling mechanism is supposed to optimize a metric of QoS (e.g., delay or blocking rate).

When executing the task of surveillance, a video sensor node can provide two types of service, one of which is to collect the data of normal video stream, and the other one is to collect the data of accident identification. Compared with the normal data, the abnormal data has a strict requirement of real time in order that the emergency and accidents can be discovered in time. A sensor node may transmit either normal or abnormal data to heads, which have to process both kinds of requests. Accordingly, a scheduling mechanism has to be designed to ensure that regular monitoring tasks are accomplished and accident reports are not delayed.

Previous request scheduling mechanisms focused on optimizing a single QoS metric (e.g., delay or blocking rate) without considering different QoS demands of two kinds of requests. Some work deals with packet queue management to improve the reporting delay and the video distortion [3]. Some other work pays too much attention to the occasionally accident data to avoid disturbing the regular monitoring tasks [4, 5]. A comprehensive way to achieve QoS of both kinds of reporting requests has not been studied in the literature. Such request scheduling problem entails a solution to optimize the utility of both the normal data-reporting requesters and
abnormal data-reporting requesters. Game theory can be used where multiple participants interact with each other by their own strategies and pursue their respective optimal utility. When the participants arrive a Nash Equilibrium Point, they will not deviate from this solution because no more profit can be achieved by any of them [6, 7].

During the transmission of normal and abnormal data, there are totally three active participants: abnormal data-reporting requesters (ADR), normal data-reporting requesters (NDR), and cluster heads. They interact with each other. Both ADR and NDR have to choose the right heads to report their data because their choices will influence QoS of their monitoring tasks and the performance of the whole network. The cluster heads have to use cooperative way to optimize the cost and prolong the lifetime of the network. The three participants optimize their respective utility objective and converge to an equilibrium at which ADR and NDR make optimal head selection and heads use cooperation mechanism to lower and balance load on them.

In this paper, we propose a game theoretic request scheduling based on a handover mechanism between heads’ priority queues. ADR, NDR, and the heads act as three players in the game. NDR decides the head selection strategy to optimize mean request response delay. ADR selects their heads to optimize mean request blocking rate. The Cluster heads decide the handover strategy to optimize their load. We can get a Nash Equilibrium Point (NEP) at which each participant optimizes its own utility [6, 7]. Using NEP solution as the request scheduling, we satisfy the demand of service quality of both ADR and NDR as well as achieving load balance between heads. The main contributions of this paper include the following: (1) we analyze the interaction among the three active participants in request scheduling game and aim to converge their respective behaviors (strategies) to an optimal equilibrium; (2) from the view of data-reporting requesters, we use a comprehensive way to optimize utility of both kinds of data-reporting requesters rather than pay too much attention to a single QoS metric; (3) from the view of cluster heads, we propose a handover mechanism between their request queue with priority in order to balance their respective load and prolong the lifetime.

The remainder of the paper is organized as follows. Related work is introduced in Section 2. In Section 3, we illustrate the network and QoS metrics. In Section 4 we propose the request scheduling game and the NEP solution. Some properties of NEP are validated in Section 5. Evaluation of the scheduling is shown in Section 6. Section 7 concludes the paper.

2. Related Work

In video sensor networks, abnormal data-reporting requests are usually given the high priority for accident discovery. In the paper [4], the authors propose an energy-aware packet scheduling algorithm to minimize the power consumption and prolong the lifetime of the whole video sensor network. In order to minimize the video distortion, they use a priority strategy to let the high priority packets to be transmitted over high bandwidth paths. Their proposed algorithm can constrain the energy consumption by selectively dropping the least important video packets. Durnus et al. propose an event-based fairness scheme for fair resource allocation of the involving events [3]. Their scheme is a queue management to improve the video distortion and the reporting latency. In the paper [5], the authors propose a periodic time scheduling to maximize the quality of monitoring of stochastic events. This scheduling has to decide the proportion of the time of event-monitoring and distribute a sensor’s coverage time in order to achieve the proportion sharing and prolong the lifetime.

Cluster head or server selection schemes are studied in wired/wireless networks for efficient data transmission and QoS satisfaction [8–11]. Through redirection of a request dispatcher, distributed system assigns requests and balances loads on source servers. Much work pertains to system performance analysis with resource allocation method [12]. Different optimization schemes have been proposed to achieve QoS in data transmission [13–15]. Queue theory has been widely used in request scheduling [16]. In [12], the authors investigate price of anarchy of the distributed networks with multiple dispatchers. They model the request assignment by the dispatchers as a multiplayer noncooperative game, in which each dispatcher optimizes its own cost. Selfish server selection strategy of each dispatcher results in worse global cost of the whole network. Lower and upper boundaries of price of anarchy are given to illustrate that network performance degrades with increasing number of dispatchers. Authors in [17] study the job assignment problem in heterogeneous distributed server system. They propose an optimal static assignment to minimize the loads on the entire system. For the queue model, the author uses a scheduler to assign jobs (requests) to different servers. The scheduler is similar to the head selecting function in our request assignment game. This method can be deployed with handover mechanism to obtain global optimization of both normal and abnormal requests.

3. Priority Queue Model

3.1. Video Sensor Network Architecture. Cluster-based video sensor network architecture is composed of three types of nodes. As shown in Figure 1, each video sensor serves as a node to collect both normal and abnormal data from surrounding environment and transmits the data reports to their cluster heads. The head processes the data-reporting requests and transmits them to the sink node.

3.2. QoS Metrics for Both ADR and NDR. To give the QoS metrics based on the video sensor network, we detail the queue model of handover mechanism between cluster heads.

Let \( q_{jk}^{(i)} \) denote the probability that the head \( j \) hands over the abnormal data-reporting request to the head \( k \) at time \( i \). This sharing policy satisfies \( \sum_{k=1}^{n} q_{jk}^{(i)} = 1 \) for all the \( n \) heads. As shown in Figure 2, head 1 has a probability of \( q_1 \) to keep abnormal requests and \( (1 - q_1) \) to hand over the requests to head 2. Head 2 has a probability of \( q_2 \) to keep abnormal requests and \( (1 - q_2) \) to hand over the requests to sender 1.
For each head, let $q_1$ denote the sharing policies of Head 1. As shown in Figure 2, $q_1$ represents the proportion of abnormal requests kept and processed by head 1, and $1-q_1$ represents the proportion of abnormal requests handed over by head 1, and let $P_0$ denote the idle state probability of the head. As shown in Figure 2, we suppose that in the head 1’s request processor the request arrival interval and the service time both obey exponential distribution. Then we let the normal request arrival rate be $\lambda_1$ and the service rate $\mu_1$. The sharing policy for the two heads is $(q_1, q_2)$. $q_1$ means that in enough little interval the probability of keeping the abnormal requests in their own processor. $(1-q_2)$ represents the probability of receiving requests from the other heads. Hence, the queuing rate of abnormal request packets in head 1 would be $q_1 \beta_1 + (1-q_2) \beta_2$, while the arrival rates of both heads are $\beta_1$ and $\beta_2$. Then we can construct the state transition differential equations. In the stable condition, the equations are expressed as follows:

$$[\lambda_1 + q_1 \beta_1 + (1-q_2) \beta_2] P_0 = \mu_1 P_0$$

$$[\lambda_1 + \mu_1] P_{00} = [\lambda_1 + q_1 \beta_1 + (1-q_2) \beta_2] P_{00} + \mu_1 P_{10},$$

with the constraint of $P_0 + \sum_{s=0}^{\infty} P_{s0} = 1$.

By solving the equations, we can get the state probabilities as follows:

$$P_0 = \frac{\mu_1 - \lambda_1}{\mu_1 + \beta_2 + q_1 \beta_1 - q_2 \beta_2},$$

$$P_{s0} = \frac{\lambda_1}{\mu_1} \cdot \frac{(\mu_1 - \lambda_1)(\lambda_1 + \beta_2 + q_1 \beta_1 - q_2 \beta_2)}{\mu_1 + \beta_2 + q_1 \beta_1 - q_2 \beta_2}.$$  

For head 2, the state probabilities are expressed as follows:

$$P_0' = \frac{\mu_2 - \lambda_2}{\mu_2 + q_2 \beta_1 - q_1 \beta_1},$$

$$P_{s0}' = \frac{\lambda_2}{\mu_2} \cdot \frac{(\mu_2 - \lambda_2)(\lambda_2 + \beta_1 + q_2 \beta_2 - q_1 \beta_1)}{\mu_2 + \beta_1 + q_2 \beta_2 - q_1 \beta_1}.$$  

3.2.2. Blocking Rate for Abnormal Requests. Let $D_{1}^{(i)}$ and $D_{2}^{(i)}$ denote the blocking rate for abnormal requests at head 1 and head 2. Based on the results above, we can get the probability of delay as follows:

$$D_{1}^{(i)} = \frac{\lambda_1^{(i)} + \beta_1^{(i)} + q_1^{(i)} P_1^{(i)} - q_2^{(i)} P_2^{(i)}}{(\mu_1^{(i)} + \beta_2^{(i)} + q_1^{(i)} P_1^{(i)} - q_2^{(i)} P_2^{(i)})(\mu_1^{(i)} - \lambda_1^{(i)}),}$$

$$D_{2}^{(i)} = \frac{\lambda_2^{(i)} + \beta_2^{(i)} + q_1^{(i)} P_1^{(i)} - q_2^{(i)} P_2^{(i)}}{(\mu_2^{(i)} + \beta_1^{(i)} + q_1^{(i)} P_1^{(i)} - q_2^{(i)} P_2^{(i)})(\mu_2^{(i)} - \lambda_2^{(i)}).}$$

3.2.2. Blocking Rate for Abnormal Requests. Because the abnormal request packets can be regarded as the one processed by the cooperative heads, the blocking rate should be the average probability of abnormal requests being rejected by these heads. We use the sharing policy to figure out the condition probability.

As shown in Figure 2, the probability that a packet arrives at head 1 is expressed as $[1 - (1 - q_1^{(i)} q_2^{(i)})].$ The probability that a packet arrives at head 2 is $[1 - (1 - q_1^{(i)} q_2^{(i)})].$ The abnormal requests can be processed only when a head is idle. So the blocking rate $P_{c1}^{(i)}$ for abnormal requests at head 1 is expressed as follows:

$$P_{c1}^{(i)} = 1 - P_0 = \frac{\lambda_1^{(i)} + \beta_1^{(i)} + q_1^{(i)} P_1^{(i)} - q_2^{(i)} P_2^{(i)}}{\mu_1^{(i)} + \beta_2^{(i)} + q_1^{(i)} P_1^{(i)} - q_2^{(i)} P_2^{(i)}}.$$  

In the same way, we can get the blocking rate $P_{c2}^{(i)}$ at head 2 as follows:

$$P_{c2}^{(i)} = \frac{\lambda_2^{(i)} + \beta_2^{(i)} + q_1^{(i)} P_1^{(i)} - q_2^{(i)} P_2^{(i)}}{\mu_2^{(i)} + \beta_1^{(i)} + q_1^{(i)} P_1^{(i)} - q_2^{(i)} P_2^{(i)}}.$$
With condition probabilities, the total blocking rate $P_c^{(i)}$ at time $i$ can be expressed as follows:

$$P_c^{(i)} = \left[ 1 - (1 - q_1^{(i)}) q_2^{(i)} \right] p_1^{(i)} + \left[ 1 - (1 - q_2^{(i)}) q_1^{(i)} \right] p_2^{(i)}.$$  

(7)

4. Request Scheduling Game

In this section, we introduce the request scheduling game. By modeling the game as a 3-player noncooperative game, we obtain the NEP at which the three players of ADR, NDR, and cluster heads all optimize their own utilities with no motivation to deviate from the NEP.

4.1. The Three-Player Game. A multiplayer noncooperative game includes participants, policy for every player, and utility function for every player. The game is also a complete information static game in which every player knows the policy sets and the objective functions of others, but he does not know the policy choice of others. For the policy set, every player in this three-player game has infinite choices. We also suppose that every player in this game is selfish and rational. They choose the policy that can maximize their own utilities. The Nash Equilibrium Point is defined as the optimal policies chosen by all the players to optimize their own utility. At the NEP, no one has the motivation to deviate from this solution, because they will not gain more profit with other policies.

Consider a video sensor network with $n$ heads at $N$ different times. Let the total arrival rate of abnormal requests be a constant $\beta$. Let $\beta_j^{(i)}$ express the arrival rate of abnormal requests to head $j$ at time $i$. The policy set of ADR is $B^{(i)} = \{\beta_1^{(i)}, \beta_2^{(i)}, \ldots, \beta_n^{(i)}\}$ with the requirement as follows:

$$\sum_{i=1}^N \sum_{k=1}^n \beta_k^{(i)} = \beta.$$  

(8)

Since abnormal data is time-sensitive, an abnormal request packet will be dropped if it exceeds the acceptable delay threshold. Therefore, we use the blocking rate to represent the quality demand from ADR. The utility function of ADR is to minimize the average blocking rate.

Let $\lambda_k^{(i)}$ express the arrival rate of normal requests to the head $k$ at time $i$. The policy set of NDR is $A^{(i)} = \{\lambda_1^{(i)}, \lambda_2^{(i)}, \ldots, \lambda_n^{(i)}\}$ with the requirement as follows:

$$\sum_{i=1}^N \sum_{k=1}^n \lambda_k^{(i)} = \lambda.$$  

(9)

NDR queue in the high priority queue in a head. We use the average delay to express the objective function.

Two heads can communicate through free load detection message. By using this message a busy head can find the free load head nearby and pass on the requests to them since the abnormal requests are immediate rejection type. Through this sharing way, heads cooperate to balance the load. We define $q_{jk}^{(i)}$ as the probability that head $j$ passes on its own abnormal requests to head $k$. The policies for all the heads at time $i$ can be formatted with the matrix as follows:

$$C^{(i)} = \begin{bmatrix}
q_{11}^{(i)} & q_{12}^{(i)} & \cdots & q_{1n}^{(i)} \\
q_{21}^{(i)} & q_{22}^{(i)} & \cdots & q_{2n}^{(i)} \\
\vdots & \vdots & & \vdots \\
q_{n1}^{(i)} & q_{n2}^{(i)} & \cdots & q_{nn}^{(i)}
\end{bmatrix}.$$  

(10)

with the constraints of $\sum_{k=1}^n q_{jk}^{(i)} = 1, \ 1 \leq j \leq n, \ 1 \leq i \leq N$.

According to the definition of the NEP, given the utility function $f$, the policy set $(A^*, B^*, C^*) = (A^{(i)*}, \ldots, A^{(N)*}, B^{(i)*}, \ldots, B^{(N)*}, C^{(i)*}, \ldots, C^{(N)*})$ is a NEP if $f(A^*, B^*, C^*) \geq f(A, B, C)$.

4.2. The Game Theoretic Formulation. According to (8), we can get the probability that head $k$ at time $i$ shares at least one request. This probability is shown as follows:

$$Q_k^{(i)} = 1 - \prod_{j=1}^n (1 - q_{jk}^{(i)}).$$  

(11)

where $1 \leq k \leq n$.

Based on the definitions and constraints above, we obtain the utility objective for the three players in this game.

The ADR wants to minimize the blocking rate which is expressed as follows:

$$P_C^{(i)} = \sum_{k=1}^n Q_k^{(i)} \beta_k^{(i)} + \sum_{j=1}^n \beta_j^{(i)} q_{jk}^{(i)}.$$  

(12)

Then we get the average blocking rate in the whole networking by weighting $P_C$ with fraction of abnormal request arrival rate as follows:

$$\bar{P}_C = \frac{\sum_{i=1}^n \beta_i^{(i)} P_C^{(i)}}{\beta}.$$  

(13)

The NDR wants to minimize the response delay of the normal requests. The delay of head $k$ and the average delay of the whole networking are expressed as follows:

$$D_k^{(i)} = \frac{\lambda_k^{(i)} + \sum_{j=1}^n \beta_j^{(i)} q_{jk}^{(i)}}{\mu_k^{(i)} + \sum_{j=1}^n \beta_j^{(i)} q_{jk}^{(i)}} \left( \mu_k^{(i)} - \lambda_k^{(i)} \right).$$  

(14)

$$\bar{D} = \sum_{i=1}^N \sum_{k=1}^n Q_k^{(i)} \lambda_k^{(i)} / \lambda.$$  

(15)

With the queue intensity $L_k^{(i)} = (\lambda_k^{(i)} + \sum_{j=1}^n \beta_j^{(i)} q_{jk}^{(i)}) / \mu_k^{(i)}$ in head $k$, we get the load balancing index as follows:

$$E = \sum_{i=1}^N \sum_{k=1}^n (L_k^{(i)})^2.$$  

(16)
Each player in the game chooses policy to optimize its own utility objective function.

ADR solves the following problem.

Minimize

\[ L_{A}^{*}(A^{*}, B, C^{*}) = \frac{1}{\beta} \sum_{i=1}^{N} \sum_{j=1}^{n} \frac{P_{j}^{(i)}}{\beta} - u_{p} \left( \sum_{i=1}^{N} \sum_{k=1}^{n} \beta_{k}^{(i)} \right) - \sum_{i=1}^{N} \sum_{j=1}^{n} v_{p_{j}} P_{j}^{(i)}. \]  

(17)

With respect to constraints,

\[ \sum_{i=1}^{N} \sum_{j=1}^{n} \beta_{j}^{(i)} - \beta = 0, \quad \frac{v_{p_{j}}}{P_{j}^{(i)}} = 0, \quad 1 \leq i \leq N, \quad 1 \leq j \leq n. \]  

(18)

NDR solves the following problem.

Minimize

\[ L_{N}^{*}(A, B^{*}, C^{*}) = \frac{1}{\lambda} \sum_{i=1}^{N} \sum_{k=1}^{n} \lambda_{k}^{(i)} \cdot D_{k}^{(i)} - u_{D_{k}} \left( \sum_{i=1}^{N} \sum_{k=1}^{n} \lambda_{k}^{(i)} \right) - \sum_{i=1}^{N} \sum_{j=1}^{n} v_{D_{k}} \lambda_{k}^{(i)}. \]  

(19)

With respect to constraints,

\[ \sum_{i=1}^{N} \sum_{k=1}^{n} \lambda_{k}^{(i)} - \lambda = 0, \quad \frac{v_{D_{k}}}{\lambda_{k}^{(i)}} = 0, \quad 1 \leq i \leq N, \quad 1 \leq k \leq n. \]  

(20)

Cluster heads solve the following problem.

Minimize

\[ L_{H}^{*}(A^{*}, B^{*}, C) = \frac{1}{\alpha} \sum_{i=1}^{N} \sum_{k=1}^{n} (L_{k}^{(i)})^2 - \sum_{i=1}^{N} \sum_{l=1}^{n} L_{E_{k}} \left( \sum_{i=1}^{N} \sum_{j=1}^{n} d_{j k}^{(i)} \right) - \sum_{i=1}^{N} \sum_{k=1}^{n} \sum_{j=1}^{n} v_{E_{k}} d_{j k}^{(i)}. \]  

(21)

With respect to constraints,

\[ \sum_{k=1}^{n} d_{j k}^{(i)} - 1 = 0, \quad \frac{v_{E_{k}}}{d_{j k}^{(i)}} = 0, \quad 1 \leq i \leq N, \quad 1 \leq k, \quad j \leq n. \]  

(22)

To sum up, we can obtain the NEP solution from the following equation group:

\[ \frac{P_{C}^{(i)}}{\beta} + \frac{\sum_{j=1}^{n} P_{j}^{(i)}}{\beta} \cdot \frac{\partial P_{C}^{(i)}}{\partial P_{j}^{(i)}} - u_{p} - v_{p_{j}} = 0, \]

\[ \sum_{i=1}^{N} \sum_{k=1}^{n} \beta_{k}^{(i)} = 0, \]

\[ \frac{D_{k}^{(i)}}{\lambda} + \frac{\lambda_{k}^{(i)}}{\lambda} \cdot \frac{\partial D_{k}^{(i)}}{\partial \lambda_{k}^{(i)}} - u_{D_{k}} - v_{D_{k}} = 0, \]

\[ \sum_{i=1}^{N} \sum_{k=1}^{n} \lambda_{k}^{(i)} - \lambda = 0, \]

\[ \frac{v_{D_{k}}}{\lambda_{k}^{(i)}} = 0, \]

\[ 2L_{k}^{(i)} \cdot \frac{\partial L_{k}^{(i)}}{\partial \lambda_{k}^{(i)}} - u_{E_{k}} - v_{E_{k}} = 0, \]

\[ \sum_{k=1}^{n} d_{j k}^{(i)} - 1 = 0, \]

\[ \frac{v_{E_{k}}}{d_{j k}^{(i)}} = 0. \]

NEP solution is the optimal policy for each player to maximize their own utility. As rational player, no one has the motivation to deviate from this optimal policy choice.

5. Property Validation

In this section, we give some theorems and propositions with their proofs to reveal some properties of the NEP solution. We prove the existence and stability of NEP as well as the advantage of NEP and handover efficiency.

**Theorem 1.** A Nash Equilibrium Point exists in the request scheduling game.

*Proof. As defined in Section 3, we use A, B, and C, respectively, to represent the policy sets of NDR, ADR, and heads. So the whole policy space of all the three players will be S = A × B × C. Then we use (a, b, c) to represent a point in the space S. We can see that A, B, and C are all the subset of Hausdorff spaces. With the total arrival rate constraints (8) and (9) and the total sharing probability constraint (10) in Section 3, we can reason that the sets A, B, and C are all closed. We can also prove that A, B, and C are all convex sets (proof will be given in Appendix A).

Average delay \( \bar{D}(S) \), average blocking rate \( \bar{P}(S) \), and load balancing index \( \bar{E}(S) \) are the utility functions in this game. According to (13), (15), and (16), mapping \( \bar{D} : S \rightarrow R \), mapping \( \bar{P} : S \rightarrow R \), and mapping \( \bar{E} : S \rightarrow R \) are all continuous. For policy a in set A, delay a \( a \rightarrow \bar{D}(a, b, c) \) is...
pseudoconvex function on \( A \). In the same way, blocking rate \( b \rightarrow \overline{F}(a, b, c) \) and load balancing index \( c \rightarrow \overline{E}(a, b, c) \) are both pseudoconvex functions on \( B \) and \( C \) (proof will be given in Appendix B).

For NDR and its policy set \( A \), we define the set value mapping \( F_A : B \times C \rightarrow \mathbb{L}_0(A) \) with \( \forall(b, c) \in B \times C \) as follows:

\[
F_A(b, c) = \{ a^* \in A : \overline{D}(a^*, b, c) = \min_{a \in A} \overline{D}(a, b, c) \}.
\] (24)

For ADR and its policy set \( B \), we define the set value mapping \( F_B : A \times C \rightarrow \mathbb{L}_0(B) \) with \( \forall(a, c) \in A \times C \) as follows:

\[
F_B(a, c) = \{ b^* \in B : \overline{E}(a, b^*, c) = \min_{b \in B} \overline{E}(a, b, c) \}.
\] (25)

For heads and the policy set \( C \), we define the set value mapping \( F_C : A \times B \rightarrow \mathbb{L}_0(C) \) with \( \forall(a, b) \in A \times B \) as follows:

\[
F_C(a, b) = \{ c^* \in C : \overline{E}(a, b, c^*) = \min_{c \in C} \overline{E}(a, b, c) \}.
\] (26)

Given \( r = \inf_{(b,c)\in B \times C} \overline{D}(a,b,c) \), we can see that, for any \((b,c) \in B \times C, \overline{D}(a',b,c) \geq r \). Then we deduce that \( F_A(b,c) \) is the convex closed set on \( A \) and \( F_B : A 	imes C \rightarrow \mathbb{L}_0(A) \) is upper-continuous. For \( F_B \) and \( F_C \), we have the same conclusions. So we define the set value mapping \( F : S \rightarrow \mathbb{L}_0(S) \) for any \((a,b,c) \in S \), as follows:

\[
F(a,b,c) = F_A(b,c) \times F_B(a,c) \times F_C(a,b) .
\] (27)

Mapping \( F \) is upper-continuous with its convex closed value set \( F(a,b,c) \). Based on Fan-Glicksberg fixed point theorem [18], for existence of \((a^*,b^*,c^*) \in S \), we have \((a^*,b^*,c^*) \in F(a^*,b^*,c^*) \). To conclude, a NEP exists in this game.

Theorem 2. The NEP solution to the request scheduling game is stable.

Proof. Policy sets \( A, B, \) and \( C \) are all closed and convex subsets in Hausdorff spaces. Objective mappings \( a \rightarrow \overline{D}(a, b, c), b \rightarrow \overline{F}(a, b, c), \) and \( c \rightarrow \overline{E}(a, b, c) \) are all continuous and pseudoconvex functions. Let functional space be \( F = (\overline{D}, \overline{F}, \overline{E}) \). Set value mapping is a usco mapping on game space \( F \). Because policy spaces \( A, B, \) and \( C \) are all normed linear spaces (proof will be given in Appendix C), there is a dense complement set, in which any \( f \) belonging to \( F \) is general. In general game \( f, \) for any \((a,b,c) \in N(f), (a,b,c) \) is a general equilibrium point [19].

This theorem has practical significance in video sensor networks. In this game, when game function, such as average delay, changes slightly into another similar but different formulation, the new NEP will not vary greatly. A NEP represents the results of data-reporting request assignment and networking resource allocation. We know that accidents in the networking always happen without in-time response. As shown in (23), changes of processor in the head will lead to changes of the service rate \( \mu \). Then the whole game function of delay will change a little. Because of the stability of the NEP solution, even still following the proper policies, quality of service and performance of networking will not be affected too much. Hence, the NEP can give a smooth transition in some emergent accidents. It is meaningful to stable surveillance.

We will prove that the NEP solution and handover mechanism outperform other mechanisms in some difficult working conditions.

Proposition 3. Given the arrival rates for both ADR and NDR as \( \lambda_1 \) and \( \lambda_2 \), as well as the service rate \( \mu_1 \), the normal request response delay \( D_1(\gamma) \) is less than the delay \( W_1 \) in M/M/1 model, if the rates satisfy \( \lambda_1/\beta_1 \rightarrow \infty \).

Proof. We use the waiting time of a normal request packet to express the delay in M/M/1. Delay in M/M/1 and delay of the handover mechanism can be expressed as follows:

\[
W_1 = \frac{\lambda_1 + \beta_1}{\mu_1 (\mu_1 - \lambda_1 - \beta_1)^2} .
\]

\[
D_1(\gamma) = \frac{\lambda_1 + \beta_1}{\mu_1 + \beta_1} + \gamma \frac{\lambda_1}{\mu_1}.
\]

By using the condition \( \lambda_1/\beta_1 \rightarrow \infty \), we get the ratio of two delays. Then we use the high order infinitesimal of \( 1/\lambda_2 \) to obtain the limit of the ratio.

Consider

\[
\frac{D_1(\gamma)}{W_1} \rightarrow \frac{\mu_1 + O(1/\lambda_1^2)}{\mu_1 + \beta_1 + (1 - \beta_2) \lambda_2} < 1.
\]

So, we get \( D_1(\gamma) < W_1 \).

The limitation condition \( \lambda_1/\beta_1 \rightarrow \infty \) can be regarded as the situation that the number of normal requests is too much greater than abnormal requests. In this situation, normal data are the main load of networking comparing with abnormal data. It is necessary to pay attention to the quality demand of NDR. Proposition 3 indicates that by using handover the request delay is less than the delay in nondifferential M/M/1 model.

Proposition 4. On condition \( \beta_1/\lambda_1 \rightarrow \infty, \beta_2/\lambda_2 \rightarrow \infty \), handover mechanism performs like model M/M/I(1).

Proof. Low priority queue for abnormal requests is immediate rejection. Suppose that there are two heads and the probabilities of sharing are \( q_1 \) and \( q_2 \). According to (11), we get the weights \( Q_1 \) and \( Q_2 \), which mean that no abnormal request is handed over to head 1 or 2,

\[
Q_1 = 1 - (1 - q_1) q_2, \quad Q_2 = 1 - (1 - q_2) q_1.
\]

For M/M/I(1) model, weighted blocking rate for abnormal requests to the two heads are as follows:

\[
\overline{w} = Q_1 \cdot \frac{\lambda_1 + \beta_1}{\mu_1 + \lambda_1 + \beta_1} + Q_2 \cdot \frac{\lambda_2 + \beta_2}{\mu_2 + \lambda_2 + \beta_2}.
\] (31)
In the handover mechanism, weighted blocking rate for abnormal requests to the two heads are expressed as follows:

$$P_c^{(b)} = Q_1 \cdot \frac{\lambda_1 + q_1 \beta_1 + (1 - q_1) \beta_2}{\mu_1 + q_1 \beta_1 + (1 - q_1) \beta_2} + Q_2 \cdot \frac{\lambda_2 + q_2 \beta_2 + (1 - q_1) \beta_1}{\mu_2 + q_2 \beta_2 + (1 - q_1) \beta_1}$$  \hspace{1cm} (32)

when $\beta_1/\lambda_1 \to \infty, \beta_2/\lambda_2 \to \infty, 1/\beta_1 \to \infty, 1/\beta_2 \to \infty$, we get

$$\frac{P_c^{(b)}}{\Psi} = \frac{Q_1 + Q_2 + O(1/\beta_1) + O(1/\beta_2)}{Q_1 + Q_2 + O(1/\beta_1) + O(1/\beta_2)} \to 1. \hspace{1cm} (33)$$

The $M/M/1$ model is used to serve the requests, just because the queue model has to accommodate the condition $\beta_1/\lambda_1 \to \infty, \beta_2/\lambda_2 \to \infty$, which means that abnormal requests are much more than normal requests. Additionally, abnormal request has a delay threshold and will not wait for a long period of time. Since too little normal requests need to be processed, the main objective is to serve the ADR. $M/M/1(1)$ is one of proper models for abnormal requests. Proposition 4 indicates that handover mechanism performs as well as $M/M/1$ model when abnormal data imposes tremendous load on the networking.

**Proposition 5.** When $\mu_1/\mu_2 < 1, \beta_1/\beta_2 \to \infty$, LSG model has a smaller load balancing index than $M/M/1$ model.

**Proof.** Suppose that there are two heads to do handover. We use $M^2$ to express the balancing index in $M/M/1$, and $E^2$ to express the index between heads with handover

$$M^2 = \left(\frac{\lambda_1 + \beta_1}{\mu_1}\right)^2 + \left(\frac{\lambda_2 + \beta_2}{\mu_2}\right)^2 \quad \text{and} \quad E^2 = \left(\frac{\lambda_1 + q_1 \beta_1 + (1 - q_1) \beta_2}{\mu_1}\right)^2 + \left(\frac{\lambda_2 + q_2 \beta_2 + (1 - q_1) \beta_1}{\mu_2}\right)^2. \hspace{1cm} (34)$$

On the conditions $\mu_1/\mu_2 < 1, \beta_1/\beta_2 \to \infty, 1/\beta_1^2 \to 0$, the ratio can be limited with high order infinitesimal of $1/\beta_1^2$ as follows:

$$\frac{M^2}{E^2} = \frac{(1/\mu_1^2) + O(1/\beta_1^2)}{(q_1^2/\mu_1^2 + ((1 - q_1)^2/\mu_2^2) + O(1/\beta_1^2)).} \hspace{1cm} (35)$$

By using the condition and square inequality, we get

$$\frac{M^2}{E^2} \rightarrow \frac{\mu_1^2}{q_1^2 \mu_2^2 + (1 - q_1)^2 \mu_1^2} > \frac{1}{q_1^2 + (1 - q_1)^2} \geq 1. \hspace{1cm} (36)$$

The nodes of high process ability are supposed to serve more packets in accordance with their practical capacity. To analyze the condition $\mu_1/\mu_2 < 1, \beta_1/\beta_2 \to \infty$, we find that this is the situation in which the two heads may lose balance. Because the processor in head 2 is stronger than the one in head 1, head 2 is supposed to process more requests than head 1, yet, there are much more requests arriving at head 1 than head 2. This situation will test the adjustability of networking. Proposition 5 indicates that the handover model shows better adjustability than $M/M/1$.

**Proposition 6.** The NEP solution shows inefficiency as the number of heads increases on conditions as follows:

$$\lambda_1 = \lambda_2 = \cdots = \lambda_M, $$

$$\beta_1 = \beta_2 = \cdots = \beta_M, \hspace{1cm} (37)$$

$$\mu_1 = \mu_2 = \cdots = \mu_M.$$

**Proof.** By solving the equation group (23) to get the NEP, we have a load balancing constraint. According to the format of the balancing index and the square inequality, we have

$$\sum_{i=1}^{M} (L_i) \geq \frac{(\sum_{i=1}^{M} L_i)^2}{2} \hspace{1cm} (38)$$

with the equalization condition $L_1 = L_2 = \cdots = L_M$. Then we can figure out the sharing probability for each head in different cases.

When $M = n$, we get $q_{jk} = 1/n, 1 \leq j, k \leq n$.

When $M = n - 1$, we get $d_{rm} = 1/(n - 1), 1 \leq r, m \leq n - 1$.

We define the inefficiency indexes $F$ and $F'$ as follows:

$$F = n \cdot \left(1 - \frac{1}{n}\right)^n \cdot (1 - \lambda \Delta t), \hspace{1cm} (39)$$

$$F' = (n - 1) \cdot \left(1 - \frac{1}{n - 1}\right)^{n - 1} \cdot (1 - \lambda \Delta t),$$

where $(1 - \lambda \Delta t)$ is the probability of free load over infinitesimal period and $(1 - 1/n)^n$ is the probability that no abnormal requests are shared. Because $(n - 1)/n \cdot (1 - 1/n)^{n - 1} > (1 - 1/n)' \cdot (1 - 1/(n - 1))^{n - 1}$, we get

$$F > F'. \hspace{1cm} (40)$$

The inefficiency index can be regarded as the degree of wasting networking resource. It is the total probability that one node has to do some work but it does not . On the conditions in Proposition 6, every head in a LSG has the same work load and process ability. But in an infinitesimal period every head has a probability of free load by NDR but receive no abnormal request. It is a waste of networking resource. We figure out the total probability of inefficiency and find that the index gets larger as the head number increases. This proposition also has a practical significance. Because of the similar process ability and same work load, each head does not have motivation to share the abnormal requests. No share means inefficiency of handover. Therefore, we prefer some distinction in head selection.
6. Numerical Results

We use Matlab to evaluate the performance of our proposed handover mechanism and game theoretic scheduling method. We consider the single-hop case. It has 6 heads that respond to all the data-reporting requests from 30 sensor nodes. We set the 1st, 3rd, and 5th nodes to have capacity of 4.2 Mbps, and the 2nd, 4th, and 6th nodes to have capacity of 3.3 Mbps. This is because homogeneous nodes may cause the inefficiency which has been shown in Proposition 6. The ADR and NDR packet sizes are both 1 MB. Accordingly, the average service rates of the 1st and 2nd nodes are $\mu_1 = 25$ packets/min and $\mu_2 = 20$ packets/min. The 30 sensor nodes generate request packets in a Poisson distribution, and the average packet arrival rate $\lambda + \beta$ is set to be less than the total service rate $\mu$, thus keeping the networking stability. According to the capacities, the total service rate 135 requests/min is the upper bound of load. The networking utilization can be expressed as $\rho = \sum_{i=1}^{6} (\lambda_i + \beta_i) / \sum_{i=1}^{6} \mu_i$. We will evaluate the networking performance in different utilization values ranging over $0, 1$.

6.1. Handover Mechanism. By employing the game theoretic optimization to get the NEP solution, we record the probability changes with the ratio of normal request rate at head 1 to the rate at head 2. Probabilities $q_{12}^{(1)}$ and $q_{21}^{(1)}$ in vertical axis corresponding to ten values of $\lambda_1^{(1)}/\lambda_2^{(1)}$ in horizontal axis are shown in Figure 3(a). On the first half of horizontal axis, we have $\lambda_1^{(1)}/\lambda_2^{(1)} < 1$. Normal request arriving rate at head 1 is less than head 2, yet, head 1 has a capacity larger than head 2. It means that unbalance load will happen. To keep load balance, head 1 lowers the probability of passing abnormal request on to head 2. Meanwhile, head 2 raises the probability to pass on the requests. As shown in Figure 3, when the ratio reaches a value a little larger than 1, load balance is achieved. At the horizontal values larger than but close to 1, the system does not have an obvious tendency to lose balance. So, the two heads both experience a section of probability fluctuation across the horizontal axis. The section of fluctuation on the second half of horizon axis can be interpreted as the self-adjustment of heads to obtain the load balance again. When the ratio of is large enough, as shown in right section in Figure 3(a), heads have the definite orientation to change their probability of handing request over. Figure 3(b) shows more details of the adjustment and fluctuation by taking 100 ratio values into account. We can see the similar change tendency as Figure 3(a) on the first half of coordinate. We use the normal request ratio rather than abnormal request ratio, because large amount of normal requests is the main reason to make no free load head for the ADR; heads need to share abnormal requests frequently.

From Figure 3, we see the interaction between request arriving amount and heads’ policy based on handover. In different state of arriving rate, heads always use the NEP solution to alleviate load and keep balance through handover.

6.2. Performance Evaluation. Let $\lambda$ be the total normal request arriving rate. Let $\beta$ be the total abnormal request arriving rate. We record the simulation results in two cases of $\lambda/\beta = 2$ and $\lambda/\beta = 1$. So we get 9 values of $\lambda + \beta$ in 9 utilization values as a sequence as $13.5, 27, 40.5, 54, 67.5, 81, 94.5, 108, 121.5$ (requests/min).
In our proposed game theoretic request scheduling based on handover, we use the NEP from (23) to make head selection to optimize utility functions of three players in the game as we discussed in Section 3.

On condition of M/M/1, the solution can be figured out with a load balance constraint as follows:

\[ \frac{\lambda_1 + \beta_1}{\mu_1} = \frac{\lambda_2 + \beta_2}{\mu_2} = \cdots = \frac{\lambda_6 + \beta_6}{\mu_6}. \]  

(41)

In minimum delay selection method, based on M/M/1, optimization is to solve the problem

\[ \text{Min} \ \bar{W} = \sum_{i=1}^{6} \frac{\lambda_i}{\mu_i} \cdot \frac{\lambda_i + \beta_i}{(\mu_i + \lambda_i + \beta_i)}. \]  

(42)

In minimum blocking probability method based on M/M/1(1), optimization is to solve the problem

\[ \text{Min} \ \bar{q}_e = \sum_{i=1}^{6} \frac{\beta_i}{\mu_i} \cdot \frac{\lambda_i + \beta_i}{(\mu_i + \lambda_i + \beta_i)}. \]  

(43)

The NEP in our method contains three optimal solutions. Hence, we compare simulation results from three aspects. We compare our NEP with equal load and minimum delay solutions by delay simulation. For blocking rate, we show the comparison of our NEP with minimum blocking probability solution. In load index comparison, results of all the four optimizations will be illustrated.

6.2.1. Delay for NDR. Figure 4 shows three surfaces, which respectively represent the delay solutions of NEP, (minimum delay) MD [13], and (equal load) EL [14]. Each surface shows perform of an optimal delay method by giving delay value change with both ratio \( \lambda/\beta \) and networking utilization. For normal requests, NEP outperforms MD and EL. Details are shown in Figure 5. In Figure 5(a), when ratio \( \lambda/\beta = 2 \), we can see that NEP has lower delays than both of the other two methods when the networking utilization \( \rho < 0.6 \). The NEP performs better than minimum delay method in all utilization points except \( \rho = 0.9 \). The equal load method performs best only when \( \rho = 0.7 \), yet, we can see that all the three methods perform similarly in \( \rho = 0.7 \). In all the utilizations, average delays of both NEP and minimum delay are all lower than 0.15 min. When the ratio changes into \( \lambda/\beta = 1 \), Figure 5(b) shows the comparison. The NEP has lower delays than both the other two methods when \( \rho < 0.7 \) except \( \rho = 0.3 \). In particular in the point \( \rho = 0.9 \), the NEP has significant dominance over the other two methods.

As shown in Figures 4 and 5, the dominance of NEP is not only in the situation that the networking has relatively small utilizations, but also in the scenario that large amount of load put on the networking. We consider the reason from two aspects. Although NEP solution has the similar optimizing algorithm and constraints as the minimum delay method, the NEP is based on the handover mechanism to balance load. Another reason is that each head use high and low priority queues to process two kinds of packets. The head drops or hand over abnormal request packets when necessary to ensure these request are processed in time.

6.2.2. Blocking Rate for ADR. Minimum blocking probability method based on M/M/1(1) minimizes the average blocking rate. Figure 6 shows the respective surfaces of NEP and (minimum blocking rate) MB [15] on M/M/1(1). Blocking rate values change with ratio \( \lambda/\beta \) and networking utilizations. For abnormal requests, NEP outperforms MB on M/M/1(1). Results in Figure 7(a) are on condition \( \lambda/\beta = 2 \). We see that the NEP blocking rate is relatively stable and performs better than minimum blocking probability method when the networking utilizations are less than 0.6. As load increases further, the NEP blocking rate reaches high values. In Figure 7(b), we have total arriving rate ratio \( \lambda/\beta = 1 \). Although NEP still performs better at most utilization points in a range \( \rho < 0.5 \), the NEP blocking rate figure shows considerable fluctuation.

We draw two conclusions from the two figures. First, there is an optimal utilization range for blocking rate by NEP like \( \rho < 0.6 \) in Figure 7(a). In this range, live content delivery works well with a relatively low blocking probability. Second, performance fluctuation is mainly caused by the gradual reduction of ratio \( \lambda/\beta \). Increasing the number of both ADR and NDR do not lead to a fluctuation. When abnormal requests takes a considerable proportion of the total arriving rate, performance of blocking rate becomes unstable. In high utilization, high blocking rate for abnormal requests is reasonable. Rejection in time can let ADR find a new head as quickly as possible rather than wait a long period of time in the queue.

6.2.3. Load of Heads. According to (16), minimizing the index is a control and optimization process. we have

\[ \sum_{i=1}^{3} \sum_{k=1}^{2} \left( L_{ik}^{(j)} \right)^2 \geq \left( \frac{1}{2} \right) \cdot \left( \sum_{i=1}^{3} \sum_{k=1}^{2} \left( L_{ik}^{(j)} \right)^2 \right)^2. \]  

(44)
When expression on the right side of the inequality is a constant, we get the condition to equalize (44) as $L_1^{(1)} = L_2^{(1)} = L_1^{(2)} = L_2^{(2)} = L_1^{(3)} = L_2^{(3)}$.

The NEP load index is figured out with (16). For other methods, the index is expressed as $\sum_{i=1}^{6} ((\lambda_i + \beta_i)/\mu_i)^2$.

To evaluate the load balance of the four optimization methods, we let all the 6 heads have the same capacity of 4.2 Mbps. In addition, the total arriving rate $\lambda + \beta$ at each networking utilization point is a constant. As shown in Figure 8, all the four methods perform nearly the same in different utilizations and ratio $\lambda/\beta$. The results indicate that NEP satisfies QoS demands of both ADR and NDR without losing load balance.

7. Conclusion

In this paper, we propose a comprehensive way to satisfy the QoS demands of both normal and abnormal data-reporting requests in order to improve the accident discovery but not to interfere with the regular surveillance task in video sensor networks. The request scheduling game optimizes the utilities of both ADR and NDR and balances the load on cluster heads. A handover mechanism ensures that the abnormal requests are processed in time. Based on game theoretic analysis, we prove some properties of the NEP and the handover mechanism. Existence and stability of NEP ensure efficient optimization and good adjustability to some accidents in networks. Numerical analysis shows that the game strategic scheduling outperforms other scheduling methods that consider single objective (minimum delay or minimum blocking rate).

Appendices

A. Policy Set Convex Property Proof

A.1. Policy Set of the NDR Is a Convex Set. The sensor network contains $n$ heads. We use $\lambda_k^{(i)}$ to express the arrival rate of NDR to the head $k$ at time $i$. The policy set of NDR is $A^{(i)} =$
\begin{align*}
\{\lambda^{(1)}_1, \lambda^{(2)}_2, \ldots, \lambda^{(n)}_n\} \quad \text{with the requirement} \quad \sum_{i=1}^{N} \sum_{k=1}^{n} \lambda^{(i)}_k = \lambda.
\end{align*}
Then we get the policy matrix as follows:
\begin{equation}
A = \begin{bmatrix}
\lambda^{(1)}_1 & \lambda^{(1)}_2 & \cdots & \lambda^{(1)}_n \\
\vdots & & & \\
\lambda^{(N)}_1 & \lambda^{(N)}_2 & \cdots & \lambda^{(N)}_n
\end{bmatrix}.
\tag{A.1}
\end{equation}

For any policy \(a\), if \(a\) satisfies the constraint \(\sum_{i=1}^{N} \sum_{k=1}^{n} \lambda^{(i)}_k = \lambda\), we reason that \(a\) is in the policy set \(A\). For \(\forall k \in (0, 1)\) and any two policies \(a, b \in A\), we have
\begin{align*}
k \cdot a + (1-k) \cdot b &= k \cdot \begin{bmatrix}
a^{(1)}_1 & a^{(1)}_2 & \cdots & a^{(1)}_n \\
\vdots & & & \\
a^{(N)}_1 & a^{(N)}_2 & \cdots & a^{(N)}_n
\end{bmatrix} \\
& \quad + (1-k) \cdot \begin{bmatrix}
b^{(1)}_1 & b^{(1)}_2 & \cdots & b^{(1)}_n \\
\vdots & & & \\
b^{(N)}_1 & b^{(N)}_2 & \cdots & b^{(N)}_n
\end{bmatrix} \\
&= \begin{bmatrix}
ka^{(1)}_1 + (1-k) b^{(1)}_1 & ka^{(1)}_2 + (1-k) b^{(1)}_2 & \cdots & ka^{(1)}_n + (1-k) b^{(1)}_n \\
\vdots & & & \\
ka^{(N)}_1 + (1-k) b^{(N)}_1 & ka^{(N)}_2 + (1-k) b^{(N)}_2 & \cdots & ka^{(N)}_n + (1-k) b^{(N)}_n
\end{bmatrix} .
\tag{A.2}
\end{align*}

To sum up elements of a row, we have
\begin{equation}
\sum_{i=1}^{N} \sum_{j=1}^{n} \left(ka^{(i)}_j + (1-k) b^{(i)}_j\right) = \lambda.
\tag{A.3}
\end{equation}

Therefore, policy \(k \cdot a + (1-k) \cdot b\) is in set \(A\). \(A\) is a convex set.
A.2. ADR’s Policy Set Is a Convex Set. The policy set of ADR at time \( t \) is \( B(t) = \{\beta_1(t), \beta_2(t), \ldots, \beta_n(t)\} \) with the requirement \( \sum_{i=1}^{N} \sum_{j=1}^{n} \beta_j^{(i)} = \beta \). We get the policy matrix as follows:

\[
B = \begin{bmatrix}
\beta_1^{(1)} & \beta_2^{(1)} & \cdots & \beta_n^{(1)} \\
\vdots & \vdots & \ddots & \vdots \\
\beta_1^{(N)} & \beta_2^{(N)} & \cdots & \beta_n^{(N)} 
\end{bmatrix} \tag{A.4}
\]

For \( \forall k \in (0,1) \) and any two policies \( a, b \in B \), we have the same result proof as NDR. Then we have

\[
\sum_{i=1}^{N} \sum_{j=1}^{n} (k\alpha_j^{(i)} + (1-k)\beta_j^{(i)}) = \beta. \tag{A.5}
\]

Therefore, policy \( k \cdot a + (1-k) \cdot b \) is in set \( B \). \( B \) is a convex set.

A.3. Cluster Heads’ Policy Set Is a Convex Set. We have given the head policy matrix in (10). \( q_{jk}^{(i)} \) is the probability that the head \( j \) hands over its own abnormal requests to head \( k \). We give any two policies as a heads as follows:

\[
a = (C_a^{(1)}, C_a^{(2)}, \ldots, C_a^{(N)}), \quad b = (C_b^{(1)}, C_b^{(2)}, \ldots, C_b^{(N)}). \tag{A.6}
\]

For \( \forall k \in (0,1) \) and any two policies \( a, b \in A \), we have

\[
ka + (1-k)b = \left( (kC_a^{(1)} + (1-k)C_b^{(1)}), \ldots, (kC_a^{(N)} + (1-k)C_b^{(N)}) \right). \tag{A.7}
\]

At time \( i \), we have

\[
k \cdot C_a^{(i)} + (1-k) \cdot C_b^{(i)} = k \cdot
\begin{bmatrix}
a_{11}^{(i)} & a_{12}^{(i)} & \cdots & a_{1n}^{(i)} \\
a_{21}^{(i)} & a_{22}^{(i)} & \cdots & a_{2n}^{(i)} \\
\vdots & \vdots & \ddots & \vdots \\
a_{n1}^{(i)} & a_{n2}^{(i)} & \cdots & a_{nn}^{(i)} 
\end{bmatrix}
\begin{bmatrix}
b_{11}^{(i)} & b_{12}^{(i)} & \cdots & b_{1n}^{(i)} \\
b_{21}^{(i)} & b_{22}^{(i)} & \cdots & b_{2n}^{(i)} \\
\vdots & \vdots & \ddots & \vdots \\
b_{n1}^{(i)} & b_{n2}^{(i)} & \cdots & b_{nn}^{(i)} 
\end{bmatrix} + (1-k) \cdot
\begin{bmatrix}
k \cdot a_{11}^{(i)} + (1-k) \cdot b_{11}^{(i)} & \cdots & k \cdot a_{1n}^{(i)} + (1-k) \cdot b_{1n}^{(i)} \\
k \cdot a_{21}^{(i)} + (1-k) \cdot b_{21}^{(i)} & \cdots & k \cdot a_{2n}^{(i)} + (1-k) \cdot b_{2n}^{(i)} \\
\vdots & \vdots & \ddots & \vdots \\
k \cdot a_{n1}^{(i)} + (1-k) \cdot b_{n1}^{(i)} & \cdots & k \cdot a_{nn}^{(i)} + (1-k) \cdot b_{nn}^{(i)}
\end{bmatrix}. \tag{A.8}
\]

Then we have

\[
\sum_{j=1}^{n} (k \cdot a_{mj}^{(i)} + (1-k) \cdot b_{mj}^{(i)}) = 1. \tag{A.9}
\]

Therefore, policy \( k \cdot a + (1-k) \cdot b \) is in set \( C \). \( C \) is a convex set.

B. Utility Function Pseudoconvex Property Proof

NDR objective function is expressed as (15). We adjust the constants to simplify the equation as the following formats:

\[
D(\lambda_1, \lambda_2, \ldots, \lambda_n) = \frac{\lambda_1^2}{1 - \lambda_1} + \frac{\lambda_2^2}{1 - \lambda_2} + \cdots + \frac{\lambda_n^2}{1 - \lambda_n}. \tag{B.1}
\]

Because \( \lambda_i < 1 \), we have

\[
\frac{\partial D}{\partial \lambda_i} = \frac{\lambda_i (2 - \lambda_i)}{(1 - \lambda_i)^2} > 0, \quad \frac{\partial^2 D}{\partial \lambda_i^2} = \frac{1}{(1 - \lambda_i)^3} > 0. \tag{B.2}
\]

Function \( D \) is a convex function. A convex function must be a pseudoconvex.

ADR objective function is expressed as (13). We also adjust constants to get the expression

\[
P(\beta_1, \beta_2, \ldots, \beta_n) = \frac{\beta_1^2}{1 + \beta_1} + \frac{\beta_2^2}{1 + \beta_2} + \cdots + \frac{\beta_n^2}{1 + \beta_n}. \tag{B.3}
\]

Because \( \beta_i < 1 \), we have

\[
\frac{\partial P}{\partial \beta_i} = \frac{\beta_i (2 + \beta_i)}{(1 + \beta_i)^2} > 0, \quad \frac{\partial^2 P}{\partial \beta_i^2} = \frac{1}{(1 + \beta_i)^3} > 0. \tag{B.4}
\]

Function \( F \) is a convex function. It is also a pseudoconvex function.

Sender objective function is expressed as (16). By adjusting constants, we get the expression

\[
E = (q_1^{(1)})^2 + (q_2^{(1)})^2 + \cdots + (q_n^{(1)})^2. \tag{B.5}
\]

We also have

\[
\frac{\partial^2 E}{\partial^2 q_1^{(i)}} > 0, \quad \frac{\partial^2 E}{\partial^2 q_2^{(i)}} > 0. \tag{B.6}
\]

Function \( E \) is a convex function and pseudoconvex function.

C. Proof of Normed Linear Policy Space

C.1. NDR Policy Space. Let NDR policy space be \( A \). There is \( a = (\lambda_1, \lambda_2, \ldots, \lambda_n) \) and \( a \in A \). Total arriving rate is a constant \( \lambda \). We can see that \( A \) is a subspace in \( n \)-dimension Euclidean space. Hence, we can define the distance of \( a \) as follows:

\[
\|a\| = \sqrt{\left(\frac{\lambda_1}{\lambda}\right)^2 + \cdots + \left(\frac{\lambda_n}{\lambda}\right)^2}. \tag{C.1}
\]

We have

\[
\forall k \in R, \quad \|ka\| = |k| \|a\|, \tag{C.2}
\]

\[
\forall b \in A, \quad \|a + b\| \leq \|a\| + \|b\|. \tag{C.2}
\]

According to the definition of normed linear space, NDR policy space is a normed linear space.
C.2. ADR Policy Space. A policy choice $a = (\beta_1, \beta_2, \ldots, \beta_n)$ belongs to the policy space $B$. We define the distance as follows:

$$\|a\| = \sqrt{(\frac{\beta_1}{\beta})^2 + \cdots + (\frac{\beta_n}{\beta})^2}. \quad (C.3)$$

Following the same proof process as NDR policy above, ADR policy space is linear normed.

For each sender policy space, we give a choice the distance as

$$\|a\| = \sqrt{q_1^2 + \cdots + q_n^2}. \quad (C.4)$$

Conflict of Interests

The authors declare that there is no conflict of interests regarding the publication of this paper.

Acknowledgments

This work is supported by the National Basic Research Program of China (973 Program) under Grant 2013CB329101, partially supported by the National High-Tech Research and Development Program of China (863) under Grant no. 2011AA010701, by the National Natural Science Foundation of China (NSFC) under Grant nos. 61372112, 61003283, 2011AA010701, by the National High-Tech Research Development Program of China (863) under Grant no. 2011AA010701, by the National Basic Research Program of China (973 Program) under Grant 2013CB329101, partially supported by the National High-Tech Research and Development Program of China (863) under Grant no. 2011AA010701, by the National Natural Science Foundation of China (NSFC) under Grant nos. 61372112, 61003283, and 61001122, by Beijing Natural Science Foundation of China under Grant no. 1442037, and by the Jiangsu Natural Science Foundation of China under Grant no. BK2011I171.

References

[1] D. Wu, S. Ci, H. Luo, Y. Ye, and H. Wang, “Video surveillance over wireless sensor and actuator networks using active cameras,” IEEE Transactions on Automatic Control, vol. 56, no. 10, pp. 2467–2472, 2011.

[2] I. Akyildiz, T. Melodia, and K. Chowdhury, “Wireless multimedia sensor networks: a survey,” IEEE Wireless Communications, vol. 14, no. 6, pp. 32–39, 2007.

[3] Y. Durmus, A. Ozgovde, and C. Ersoy, “Distributed and online fair resource management in video surveillance sensor networks,” IEEE Transactions on Mobile Computing, vol. 11, no. 5, pp. 835–848, 2012.

[4] I. Politis, M. Tsagkaropoulos, and S. Kotsopoulos, “Optimizing video transmission over wireless multimedia sensor networks,” in Proceedings of the IEEE Global Telecommunications Conference (GLOBECOM ’08), pp. 117–122, New Orleans, La, USA, December 2008.

[5] D. Yau, N. K. Yip, C. Ma, N. S. V. Rao, and M. Shankar, “Quality of monitoring of stochastic events by periodic and proportional-share scheduling of sensor coverage,” ACM Transactions on Sensor Networks, vol. 7, no. 2, article 18, 2010.

[6] J. F. Nash, “Non-cooperative games,” The Annals of Mathematics, vol. 54, no. 2, pp. 286–295, 1951.

[7] J. F. Nash, “Equilibrium points in N-person games,” Proceedings of the National Academy of Sciences of the United States of America, vol. 36, no. 1, pp. 48–49, 1950.

[8] C. Xu, T. Liu, J. Guan, H. Zhang, and G. M. Muntean, “CMT-QA: quality-aware adaptive concurrent multipath data transfer in heterogeneous wireless networks,” IEEE Transactions on Mobile Computing, vol. 12, no. 11, pp. 2193–2205, 2013.

[9] C. Xu, F. Zhao, J. Guan, H. Zhang, and G. M. Muntean, “QoE-Driven user-Centric VoD services in urban multihomed P2P-based vehicular networks,” IEEE Transactions on Vehicular Technology, vol. 62, no. 5, pp. 2273–2289, 2013.

[10] C. Xu, G. M. Muntean, E. Fallon, and A. Hanley, “Distributed storage-assisted data-driven overlay network for P2P VoD services,” IEEE Transactions on Broadcasting, vol. 55, no. 1, pp. 1–10, 2009.

[11] C. Xu, E. Fallon, Y. Qiao, L. Zhong, and G. M. Muntean, “Performance evaluation of multimedia content distribution over multi-homed wireless networks,” IEEE Transactions on Broadcasting, vol. 57, no. 2, pp. 204–215, 2011.

[12] U. Ayesta, O. Brun, and B. J. Prabhu, “Price of anarchy in non-cooperative load balancing,” in Proceedings of the IEEE INFOCOM, pp. 1–5, San Diego, Calif, USA, March 2010.

[13] T. Wu and D. Starobinski, “A comparative analysis of server selection in content replication networks,” IEEE/ACM Transactions on Networking, vol. 16, no. 6, pp. 1461–1474, 2008.

[14] V. Cardellini, E. Casalicchio, M. Colajanni, and P. S. Yu, “The state of the art in locally distributed web-server systems,” ACM Computing Surveys, vol. 34, no. 2, pp. 263–311, 2002.

[15] K. S. Tang, K. T. Ko, S. Chan, and E. W. M. Wong, “Optimal file placement in VOD system using genetic algorithm,” IEEE Transactions on Industrial Electronics, vol. 48, no. 5, pp. 891–897, 2001.

[16] L. Levy and U. Yechiali, “Utilization of idle time in an M/G/1 queueing system,” Management Science, vol. 22, no. 2, pp. 202–211, 1975.

[17] S. Cho, J. Cho, and S. J. Shin, “Playback latency reduction for internet live video services in CDN-P2P hybrid architecture,” in Proceedings of the IEEE International Conference on Communications (ICC ’10), pp. 1–5, Cape Town, South Africa, May 2010.

[18] I. Glicksberg, “A further generalization of the Kakutani fixed point theorem with application to Nash equilibrium points,” Proceedings of the American Mathematical Society, vol. 3, no. 12, pp. 170–174, 1952.

[19] J. M. Grandmont, “Temporary general equilibrium theory,” Econometrica, vol. 45, no. 3, pp. 535–572, 1977.
Submit your manuscripts at
http://www.hindawi.com