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

6,900
Open access books available

185,000
International authors and editors

200M
Downloads

Our authors are among the

154
Countries delivered to

TOP 1%
most cited scientists

12.2%
Contributors from top 500 universities



WEB OF SCIENCE™

Selection of our books indexed in the Book Citation Index in Web of Science™ Core Collection (BKCI)

Interested in publishing with us?
Contact book.department@intechopen.com

Numbers displayed above are based on latest data collected.
For more information visit www.intechopen.com



Chapter 4

Tradeoffs Among Delay, Energy and Accuracy of Data Aggregation for Multi-View Multi-Robot Sensor Networks

Wuyungerile Li, Ziyuan Pan and Takashi Watanabe

Additional information is available at the end of the chapter

http://dx.doi.org/10.5772/48225

1. Introduction

Due to the recent development in micro mechanics, electronics and wireless communication technologies, Wireless Sensor Network (WSN) has been a hot issue for many applications like monitoring, detecting, remote control, life saving etc. However, along with the applications in different area, the ways of deploying the sensor nodes are different. In some harsh environmental detection, sensor nodes are always dropt into the target area by aircraft which may lead some unnecessary troubles, for example some nodes are out of communication range, some nodes are broken, and lack of flexible because of its immobile characters et al.

The Multi-Robot Sensor Network (MRSN) which is comprised of large numbers of small, simple and inexpensive wireless robots can solve the problems mentioned above. In MRSN, besides perceived sensors, the robot also can be set up digital camera or voice recording equipment, even video camera according to the applications' requirement. Hence, it can detect more detailed information like pictures, video or sound etc.

In this section, at first we introduce the MRSN and its applications. The open problems and one of the effective methods, data aggregation is presented. At last we show a preview of this chapter.

1.1. Wireless multi-robot sensor networks

In wireless MRSN, from a viewpoint of sensor network communication, each robot senses data and transmits data to the adjacent lower node. To collect all data at the sink, data are sent by relay nodes in a multi-hop manner. However, due to the mobility of the robot,

INTECH
open science | open minds

© 2012 Li et al., licensee InTech. This is an open access chapter distributed under the terms of the Creative Commons Attribution License (http://creativecommons.org/licenses/by/3.0), which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

sometimes robots are out of the network area so that break the network connectivity. Hence, how to keep the connectivity all the time is a crucial issue so that it already became a hot research topic (Mi et, al., 2010). Besides, the reliability and robustness as well as secure et al. are some other concerned research topics. With respect to communication techniques in MRSN, a network with the goal of search and rescue is described in (Reich & Sklar, 2006). In this paper, they proposed an entirely distributed gradient propagation (GP) algorithm. Each sensor in the paper independently executes the GP algorithm and broadcasts after some independent, randomly chosen time interval. The robots sensors estimate their target by "hot" values and "cold" values, where the "hot" values become the searching target. In (Sheng, et al. 2006), for reliability and robustness, a distributed biding algorithm was proposed for multiple robots in exploration tasks to address the problems caused by the limited communication range. In this algorithm, all the robots work asynchronously. There are three states for each robot that (1) sensing and mapping, (2) bidding and (3) traveling. A distributed algorithm that makes mobile robots in a multi-robot system aware of network connectivity was discussed in (Leyzx, et al., 2009). The basic idea is to take a "fixed" robot as the reference robot that keep in touch with at least one neighboring robot from which a communication path to the reference robot can be established.

1.2. Applications of multi-robot sensor networks

Due to its flexibility, operability, mobility and self-organization, the applications of MRSN has been increasing (Maxim & Gaurav, 2005), (Trigui S, et, al., 2012). Harsh environmental monitoring is the most popular application of MRSN, for example let wireless robots get into Amazon rainforests where it is very dangerous for human get inside or let them climb to Mount Everest where there is not enough Oxygen for human and covered by snow all over the mountain. In medical application, if the MRSN can help the nurses to do some simple task like checking body temperature and sending to a doctor, which would save much more labors in some countries those short of nurses. One of the most important utilizations of MRSN is that it can be used to detect nuclear radiation and to accomplish some other relevant tasks. The most recent example is the Fukushima nuclear leakage where if a MRSN was applied, it would have alleviated damage. Some other applications like outer space monitoring (space junk detection), industrial monitoring (quality control), disaster monitoring (forest fire detection), agriculture monitoring (soil moisture detection), traffic monitoring (intelligent transport system) etc have also much potential.

1.3. Open problems in multi-robot sensor networks

In MRSN, when an event occurs, multiple robots in the near area sense the event data and generate an abundance of sensed data; however, many of the data generated in the same area are highly redundant. Hence the transmission and relaying of all generated data caused a big waste of bandwidth and energy; it also causes data collision and congestion so that result in low efficiency of data gathering. On the other hand, similar to wireless sensor network, WRSN could not avoid the shortcoming of lack of continuous energy supplement.

One may say robot node can be equipped a large capacity battery, but its energy consumption is also large due to its big size, moving, detecting and transmitting etc. An energy harvesting algorithm (Eu et, al., 2010) is proposed for WSN. According to energy harvesting technique, a robot can absorbs solar energy from sunlight. However, how can the robot manage its task in the night? The vibrational energy get from the environment is too lees to trigger the robot. Therefore, saving energy is the most feasible way in WRSN.

For energy saving, decreasing the redundant data and sending a representative data of the detected area are the most considerable strategy. With a view of reducing the quantity of the transmitted data, the well-known scheme is data aggregation (Rajagopalan &Varshney, 2006). Since a sensor node in WSN waits for a period of time to collect extensive quantity of data to aggregate, data aggregation leads long transmission delay and low data accuracy. Some application, like medical and architectural utilization requires more accurate data while disaster relief requires receiving data as soon as possible. However, energy, delay and accuracy are trading off each other, one can not improve three of them at the same time. Hence, how to control the trade off of energy, delay and accuracy among different applications is the problem we will solve in our work.

1.4. Data aggregation in multi-robot sensor networks

We focus on data aggregation technology for collecting data in MRSN. Data aggregation (Rajagopalan & Varshney, 2006) is a process of aggregating the data from multiple robot sensors to eliminate redundant data and provide fused information to the base station. Considering from the point of data redundant, data aggregation can collect the most efficiency data. However, transmission delay and data accuracy are also important in many applications such as military application and architectural application. Hence trading off transmission delay, energy consumption and data accuracy is an important issue. There are several typical algorithms of data aggregations. PEGASIS (Lindsey & Raghavendra, 2001) is one of energy efficiency chain based data aggregation protocols that employs a greedy algorithm. The main idea of PEGASIS is forming a chain among the sensor nodes so that each node receives (or transmits) fused data from (to) the closest neighbors. The data gathered are sent from node to node, and all the sensor nodes take turns to be the leader for transmission to the Base Station. Data Funnelling (Petrović, et. al, 2003) is another scheme that sends a stream of data from a group of sensor readings to destination. Moreover, they proposed a compression method called “coding by ordering” to suppress some readings and encoding the values in the ordering of the remaining packets. On the other hand, LEACH (Heinzelman W., 2000) is one kind of energy saving schemes in which a small number of clusters are formed in a self-organized manner. A designated sensor node in each cluster collects and combines data from nodes in its cluster, then transmits the result to the BS. Directed Diffusion (Intanagonwiwat, et. al, 2000) is a kind of data centric routing protocols. The sink broadcasts an interest message to all the sensor nodes, and the nodes gather and transmit the sink-interested data to the sink. When the receiving data rate becomes low, the sink starts to attract other higher quality data.

Regarding the trade-offs, (Boulis, et. al, 2003) proposed an energy-accuracy tradeoffs algorithm for periodic data-aggregation which is a threshold-based scheme where the sensors compare their fused estimations to a threshold to make a decision of regarding transmission. Energy-latency tradeoffs algorithm (Yu at. el., 2004) is proposed for minimizing the overall energy consumption of the networks within a specific latency constraint where data aggregation is performed only after a node successfully collects data from all its children and its own local generated data. ADA (Adaptive Data Aggregation) (Chen et. al., 2008) is an adaptive data aggregation (ADA) for clustered wireless sensor networks. In ADA, sensed data are aggregated on two levels; one is aggregated at sensor nodes controlled by the reporting frequency (temporal reliability) of nodes; another is aggregated at cluster heads controlled by the aggregation ratio (spatial reliability). The reliability of observed data that is decided by the number of arrival data at sink node is compared with the reliability of desired data, which is decided by the application. According to comparison, nine characteristic regions and nine states are defined in which the eight states must change into the desired state through the calculating and adjusting of observed reliability.

Most of the previously mentioned works focus on energy saving and aggregate as much data as possible. As a result, they prolong the transmission delay. Many works aimed to achieve energy-delay trade off, however they still have shortcomings for example (Yu at, el., 2004) has long waiting time at nodes with less event data while the constant latency makes the networks very inflexible in (Galluccio L. & Palazzo S., 2009). A desired energy-delay tradeoff is achieved in (Ye Z. et al., 2008); however the algorithm ignored the issue of data accuracy. Energy-delay-accuracy tradeoffs in (Mirian F. & Sabaei M.) and (Chen et al., 2008) adapt to a situation that could be described by the following question: 'what is the average temperature of this area at this hour?' The algorithms did not consider delay and accuracy among nodes and data, which may lead to large data deviation as well as transmission delay in some other applications.

1.5. Preview of our work

In this paper, at first, we show the analyses of transmission delay, energy consumption and data accuracy of non-aggregation, full aggregation and partial data aggregation with Markovian chain. The analytical results show that non-aggregation consumes much energy and full aggregation causes long transmission delay; but the proposed partial aggregation can trade off total delay, energy consumption, and data accuracy between non-aggregation and full aggregation. Then we intensively discuss the tradeoffs among energy consumption, transmission delay and data accuracy with a Trade Off Index (TOI). We discuss the TOI under the different conditions of accuracy dominant, energy dominant, and delay dominant. By comparing the TOI value among non-aggregation, full aggregation and partial aggregation in different data generation rates, we obtain the best TOI. The results show that with small data generation rate, non-aggregation is the best TOI; with moderate data generation rate, the partial aggregation is the best TOI while the data generation rate is large, the full aggregation is the best TOI. At last a multi view multi robot sensor network is discussed and a User Dependent Multi-view Video Transmission (UDMVT) scheme is introduced.

2. Preliminary concepts

In this section, we will introduce network topology, network parameters and the definitions of network parameters, which will be helpful in understanding our work clearly.

2.1. Wireless MRSN network topology

Fig. 1 depicts tandem network topology of the MRSN, the most basic and simplest model, which enables us to make an analytic model. The results can be extensible to other topologies that are more complex. In such kind of network, all the robots deployed statically in a flat area and have same role. The robots are allocated omni-directional antenna for wireless communication and have the same transmission ranges. When a robot senses data, it transmits the data to the sink, if the data could not get to the sink by one hop; the robot sends the data to the sink by multi-hop way.

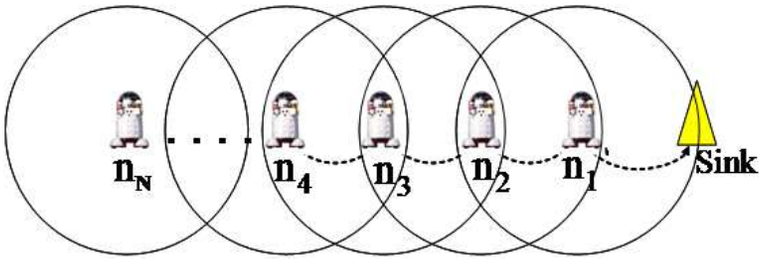


Figure 1. Tandem Multi-Robot Network

2.2. Definition

- n_i denotes the i-th node from the sink. N is a set of all nodes.
- n_{i+1} is called the adjacent upper node of , while n_{i-1} is the adjacent lower node of n_i.
- A set of nodes of $\{n_k \mid n_k \in \mid N\mid, k>i\}$ denotes the up-per nodes of n_i, while $\{n_k \mid n_k \in \mid N\mid, k<i\}$ denotes the lower nodes of n_i.
- Suffixes non, ful and par attached to terms mean non-aggregation, full aggregation and partial aggregation respectively.
- Arrival data denotes that the data come from adjacent upper nodes.
- Local generated data denotes that the data are generated at local nodes.
- Server: in our work, we assume that each node has a server to process data aggregation and data transmission.
- The MAC protocol used in this research is CSMA.
- The propagation delay between adjacent robots is negligible.

2.3. Aggregation factor

Here a robot aggregates its own generated data and received data from adjacent upper nodes before transmission. The sink does not participate in data aggregation. When data aggregation occurs at a robot node, the aggregation factor denotes the proportion of aggregated data size and local generated data size. It means that the aggregated data size is

AF times of the generated data size. AF=1 means that aggregated data have the same data size with generated data, and we assume there is one generated data at one time.

$$AF = \frac{\text{Aggregated data size}}{\text{Generated data size}} \tag{1}$$

2.4. Transmission delay

- Total delay *D(N)* shows a time interval between the instance when event E_{ij} occurred at robot n_n and the instance when the sink receives D_{ij} in *N* hops networks.
- Data transmission time τ' is defined as a time interval between the instance that data are transmitted from robot and the instance that the data are received at the adjacent lower robot.
- Channel waiting time $\tau^c(i)$: it is the time interval that data cannot utilize the channel.
- Event waiting time $\tau^e(i)$: In full aggregation, before a robot processes data aggregation, the arrival data have to wait for local generated data to be aggregated together, hence the waiting time of arrival data called event waiting time.

2.5. Energy consumption

Total energy consumption *E(N)* is defined to be the sum of energy consumption of an event data that is generated at node n_n and finally received by sink node in *N* hops networks.

2.6. Data accuracy

We define the data accuracy as the proportion of collected data at sink and the amount of sensed data at all the robots.

3. Data aggregations

In this part, we analyze and evaluate the data aggregation simply in terms of non-aggregation, full aggregation and partial data aggregation.

3.1. Non-aggregation

The arrival data are transmitted to the adjacent lower node immediately after having been received; data neither wait for local generated data nor aggregate with any other data. The analytical model of non-aggregation is shown as follows in figure 2.

In the analytical model of node n_i in fig. 2, the average arrival rate from the upper node is approximates to Poisson distribution. Generated data rate at a node is assumed to be Poisson distribution. The generated data and arrival data join the service queue and wait for transmission. There is one server for data transmission at each node. All data in the queue will be sent based on first in first out. λ_i'' is the data rate upon exiting the server at node n_i.

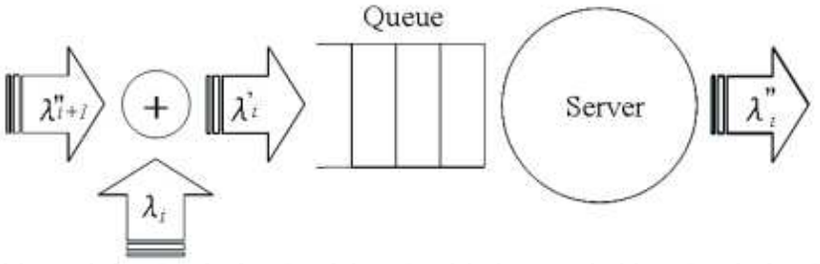


Figure 2. Analytical model of non-aggregation

- Arrival process to the queue

According to the analytic model, we find that the arrival rate to the queue is

$$\lambda_i' = \lambda_{i+1}'' + \lambda_i \tag{2}$$

Strictly speaking, arrival data from the upper node is not Poisson distribution. However, for the purpose of simplicity, we approximate the process as Poisson distribution. Since the arrival data rate and local generated data rate are independent Poisson distributions, the sum of the two is also a new Poisson distribution.

- Service process

In our network model, each node has one server. The ACK packet transmission time is not considered. Data aggregating time is very short and negligible. Therefore the service time is one hop data transmission time. In our work, data transmission rate is v_c and local generated data size is S_i. Therefore the service time for each generated data is:

$$\tau^1 = \frac{s_i}{v_c} \tag{3}$$

Since v_c and S_i are constant in non-aggregation, the service time for each data are fixed and constant.

From the above analysis we can determine that the queuing system approximates to M/D/1 model.

According to equation (4), the average data transmission time that we obtain at a node is:

$$\tau^1{}_{non}(i) = \frac{\tau^1}{(1\text{-}\lambda_i'\tau^1)} \tag{4}$$

According to queuing theory and equation (4), we determine the server waiting time as follows:

$$\tau_{non}^s(i) = \frac{\lambda_i'(\tau_{non}^1(i))^2}{2(1-\lambda_i'\tau_{non}^1(i))} \tag{5}$$

- Channel waiting time

Node n_i communicates with only one neighbor node at a time. If a neighbor node is transmitting data, node n_i has to wait until its neighbor node finishes transmission, due to the over hearing caused by the omni-directional antenna. This waiting time is defined as channel waiting time and is obtained the formulation as follows:

$$\tau_{non}^{c}(i) = 2\lambda_i' * \tau_{non}^{1}(i) * \tau_{non}^{1}(i) \tag{6}$$

3.1.1. Total delay

Total delay $D_{non}(N)$ is derived as follows where the number of hops from robot n_i to the sink is N.

$$D_{non}(N) = \sum_{i=1}^{N} (\tau_{non}^{1}(i) + \tau_{non}^{s}(i) + \tau_{non}^{c}(i)) \tag{7}$$

3.1.2. Energy consumption

Node n_i transmits data and relays arrival data from the upper nodes. Since the consumed energy is proportionate to the number of data transmissions, we can find the mean number of data $L^{Q}_{non}(i)$ in the service queue at node n_i according to Little's formula. The number of data in the queue waiting for data transmission is shown as follows:

$$L_{non}^{Q}(i) = \lambda_i' * T_{non}^{s}(i) \tag{8}$$

The λ'_i is arrival data rate at node ni and $T_{non}^{s}(i)$ is time duration from data joining the queue to data having been received by the next neighbor node at node ni, in case of non-aggregation, can be determined as follows:

$$T_{non}^{s}(i) = \tau_{non}^{1}(i) + \tau_{non}^{s}(i) + \tau_{non}^{c}(i) \tag{9}$$

According to equations (8) and (9), we obtain the whole energy consumption in N hops network as follows:

$$E_{non}(N) = \sum_{i=1}^{N} \mathrm{L}_{\mathrm{non}}^{\mathrm{Q}}(i) * (P_t + 2P_r) \tag{10}$$

Here P_t and P_r denote the energy consumption for transmitting and receiving data.

3.1.3. Data accuracy

In non-aggregation, data are not aggregated and the packet drop occurs with the transmitting in real system. However, for simplicity, we assume there is no packet drop and retransmission, all the generated data will get to the sink, thus the data accuracy approaches to 100%.

3.2. Full aggregation

We define the full aggregation that the arrival data are sent to an adjacent lower node only after having been aggregated with local generated data at nodes. It means data transmission occurs only after a new local data generated a node. Hence, the waiting time for data aggregating at a node is decided by the data generation rate of the node. When there is local generated data, the node aggregates all the arrival data with generated data then waits for transmission at server. Data after aggregation undergo the same procedure as non-aggregation to detect the server and the channel for further transmission.

The analytical model of full aggregation is shown in figue3. Before explaining the model, we introduce queue A, queue B and "G." Queue A denotes the arrival data queue at a node that is waiting for local generated data for data aggregation. Data in Queue B are waiting for server; when the server is idle, data are transmitted to a neighbor node. The "G" is assumed as a virtual gate between queue A and queue B. Immediately after local generated data aggregate with the arrival data in queue A, the gate opens and lets the aggregated data join queue B.

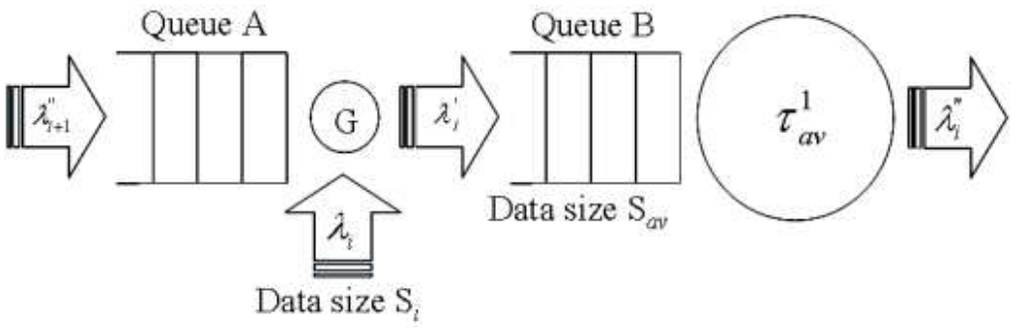


Figure 3. Analytical model of full aggregation

In full aggregation, the data join queue A with arrival rate of λ''_{i-1} and wait for new generated data. When an event occurs at local node, the node aggregates the generated data and all arrival data in queue A according to the aggregation factor Af. The size of aggregated data becomes S_{av} and the aggregated data join queue B with the rate of λ'_i to await further transmission. In full aggregation, the difference from non-aggregation is that we have to determine how long the arrival data wait for aggregation in queue A.

3.2.1. Event waiting time

To determine the event waiting time, we apply the state transition rate diagram. We describe the state transition rate diagram in fig. 4. The basic idea of the analysis is that data waiting in queue A for exponential distribution have an average of $1/2\lambda_i$. In the diagram, the state variable is the number of data waiting for an event.

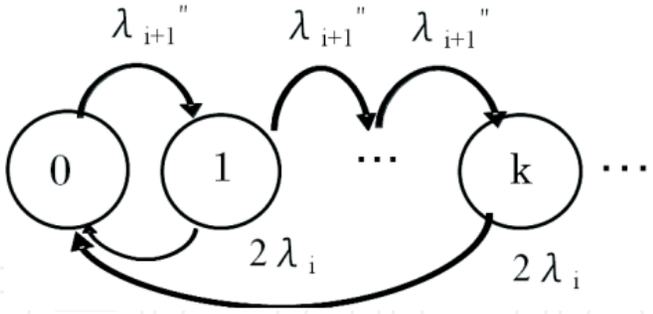


Figure 4. State transition rate diagrams of full aggregation

According to calculation of state probability distribution and Little's formula, we determine the event waiting time as shown below; more details please read (Li, et. al, 2010).

$$\tau_{ful}{}^{e}(i) = \frac{\lambda''_{i+1}}{2\lambda_i{}^2} \tag{11}$$

3.2.2. Total delay

From the definition of full aggregation we know that the arrival data join queue B only if there is new generated data at local node, hence the data arrival rate to queue B is equal to data generation rate at the node. The data generation rate abides by Poisson distribution; therefore the arrival data rate to queue B is Poisson distribution. Since the data arrival rate involves only one server and the data transmission time for server is fixed, according to queuing theory we model the queue by means of M/D/1 queue. Similar to non-aggregation, we determine the total delay $D_{ful}\ (N)$ of the network in full aggregation is consists of event waiting time in queue A, server waiting time in queue B, channel waiting time and data transmission time at server.

$$D_{ful}(N) = \sum_{i=1}^{N} (\tau^{e}_{ful}(i) + \tau^{s}_{ful}(i) + \tau^{c}_{ful}(i) + \tau^{1}_{ful}(i)) \tag{12}$$

3.2.3. Energy consumption

The energy consumption is proportional to the number of data transmissions. $L^{Q}_{ful}(i)$ is the number of data at a robot n_i. The time duration from data joining Queue B to data having been received by the next neighbor node is $T^{S}_{ful}(i)$ and it can be obtained as follows:

$$T^{s}_{ful}(i) = \tau^{s}_{ful}(i) + \tau^{c}_{ful}(i) + \tau^{1}_{ful}(i)) \tag{13}$$

According to Little's formula and equation (13), the amount of data in queue B is as follows:

$$L^{Q}_{ful}(i) = \lambda_i * T^{s}_{ful}(i) \tag{14}$$

Therefore, the whole energy consumption is determined as follows:

$$E_{ful}(N) = \sum_{i=1}^{N} L_{ful}^{Q}(i) * (p_t + 2p_r) \tag{15}$$

3.2.4. Data accuracy

In full aggregation, the aggregation factor Af=1. Thus, we can get the data accuracy in N hops transmission as follow:

$$A_c = \frac{1}{\mathrm{N}} \tag{16}$$

3.3. Partial data aggregation

According to previous analyses of non-aggregation and full aggregation, we find that non-aggregation sends all the generated data to sink node which results in large energy consumption. In case of full aggregation, the arrival data must wait for local generated data to aggregate, which causes the prolonged transmission delay and low data accuracy for the data that come from nodes far away from sink.

To minimize these two shortcomings, we propose a partial data aggregation. The main idea of partial aggregation is that nodes process data aggregation and transmit data only if a) if there are new local generated data at a node or b) after waiting a holding time at a node; the inverse of the holding time we call random pushing rate λ^{D_i}. The analytical model of partial aggregation is shown as follows.

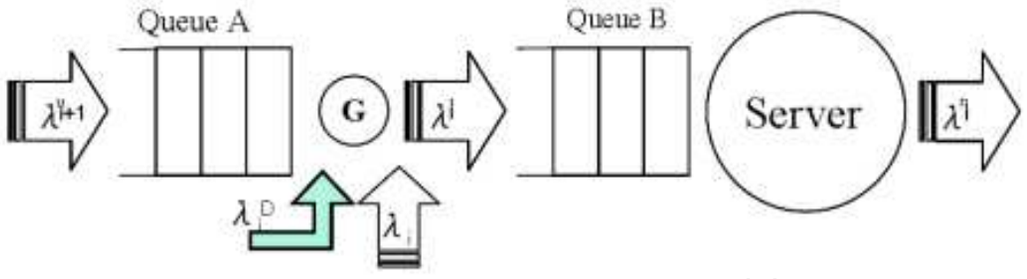


Figure 5. Analytical model of partial aggregation

For the purpose of simplifying our analytical model, we assume the arrival data rate from adjacent upper node is approximated to Poisson distribution and the arrival data join event waiting queue A in fig. 5. Data generation rate λ_i is assumed to be Poisson distribution. Random pushing rate is λ^{D_i} and assumed to be exponential distribution. If new generated data occur at a node or if holding time is over for arrival data, all the data are aggregated into one data, and the gate G opens and lets aggregated data join queue B. λ'_i is data arrival rate to queue B in which data are waiting for service (data transmission).

3.3.1. Event waiting time

Assume that a number of data are waiting for an event at robot n_i in queue; we describe the state transition diagram as shown in Fig.6.

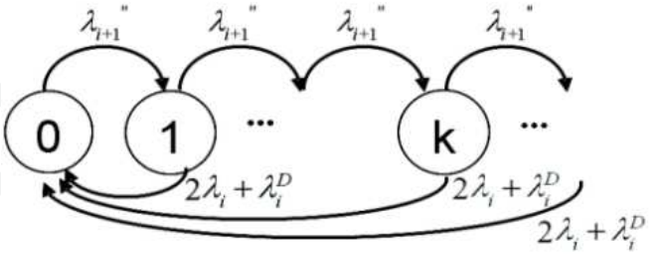


Figure 6. State transition rate diagram

Data are waiting in queue for the duration according to the exponential distribution of average $1/(2\lambda_i + \lambda_i^D)$. Similar to full aggregation, the event waiting time of partial aggregation can be determined as follows:

$$\tau_{par}^{e} = \frac{\lambda_{i+1}''}{(2\lambda_i + \lambda_i^D)^2} \tag{17}$$

3.3.2. Arrival process to Queue B

From the analytical process we find that arrival data rate λ'_i is decided by the random pushing rate and data generation rate at a node. To determine the formulation of λ'_i, we calculate the property distribution of λ_i and λ^{D}_i. We define that λ^{D}_i and λ_i are the independent distribution X and Y. Through proofing of the property Y is bigger than X, we determine the arrival process to Queue B as follows; the proof can be found in (Li, et. al, 2010).

$$\lambda_i' = \lambda_i + \frac{\lambda_{i+1}''}{\lambda_{i+1}'' + \lambda_i^D} * \lambda_i^D \tag{18}$$

3.3.3. Total delay

Since the data generation rate is Poisson distribution and the random pushing rate abides by exponential distribution, the data arrival rate to queue B approximates to Poisson distribution. Therefore, we can confirm that the queuing system approximates to M/D/1 model. With the same way of full aggregation, the server waiting time and channel waiting time can be determined easily. Therefore, the total delay of partial aggregation is as follows:

$$D_{par}(N) = \sum_{i=1}^{N} (\tau_{par}{}^{e}(i) + \tau_{par}{}^{c}(i) + \tau_{par}^{1}(i) + \tau_{par}^{s}(i)) \tag{19}$$

3.3.4. Total energy consumption

In the *N* hops transmission in partial aggregation, total energy consumption $E_{par}(N)$ is the sum of transmission energy consumption, reception energy consumption and overhearing energy consumption. P_t and P_r are energy required for transmitting or receiving a data. The period of time that aggregated data wait in a queue for transmission can be determined as follows:

$$T^s_{par}(i) = \tau^s_{par}(i) + \tau^c_{par}(i) + \tau^1_{par}(i) \tag{20}$$

According to Little's formula and equation (25), we determine the amount of data in queue B at node n_i as follows:

$$L^Q_{Par}(i) = \lambda'_i * T^s_{par}(i) \tag{21}$$

Accordingly, we determine the total energy consumption for the network as follows:

$$E_{par}(N) = \sum_{i=1}^{N} L^Q_{par}(i) * (P_t + 2P_r) \tag{22}$$

3.3.5. Data accuracy

The total generated data *Lpar(N)* in *N* hops network is obtained as follows:

$$L_{par}(N) = \sum_{i=1}^{N} \lambda_i * D_{par}(N) \tag{23}$$

The amount of data received by sink *Lpar(S)* is as follows:

$$L_{par}(S) = \lambda'_1 * D_{par}(N) \tag{24}$$

According to the definition and above equations, we determine the data accuracy as follows:

$$A_c = \frac{L_{par}(S)}{L_{par}(N)} * AF \tag{25}$$

3.4. Evaluation

Here we show the analytic results of the previous sections. The parameters are as below:

Transmission rate is 250[kbps], Data size is 4096 [bit], Energy consumption for data reception is 17.4 [mA] and for data transmission is 19.7 [mA]. In this section, we evaluate total delay, energy consumption and data accuracy when the aggregation factor is Af=1.

Fig. 7 to Fig. 9 show the total delay, energy consumption of whole network, robot energy consumption and data accuracy of five hops transmission where $\lambda_i=\lambda$. Partial-T1 and Partial-T2 are two sets of random pushing rate vectors in partial aggregation. We get the vectors randomly [1, 2, 3, 4, 5] and [5, 10, 15, 20, 25].

From figure 7, we find that when event generation rate is small, full aggregation has long transmission delay in comparison to non-aggregation. The reason for concaving up of delay of the full aggregation is that, when event generation rate is small, the received data has to wait for generated data longer duration. In addition at a robot near to the sink, the total delay increases because of the large waiting time due to the congestion around the sink. As long as total delay is concerned, non-aggregation is suitable for situation of small event generation rate. From the figure, we also find that the performances of partial-T1 and partial-T2 are between non-aggregation and full aggregation. If λ_i^D is infinite, it means fully non-aggregation, if λ_i^D is zero, it means fully aggregation.

Fig. 8 shows the energy consumption of the whole network. Obviously, non-aggregation consumes much more energy than full aggregation. Thus, full aggregation is suitable for energy consumption while non-aggregation is efficiency for transmission delay. The partial-T1 and partial-T2 has energy consumption between non-aggregation and full aggregation. In addition, the smaller random pushing rate vector set partial-T1 has less energy consumption than the set of partial-T2.

Fig. 9 shows the data accuracy of different data aggregation. From fig. 9, we find that the data accuracy of partial aggregation is between non-aggregation and full aggregation. The partial aggregation with the larger random pushing rate achieves higher data accuracy.

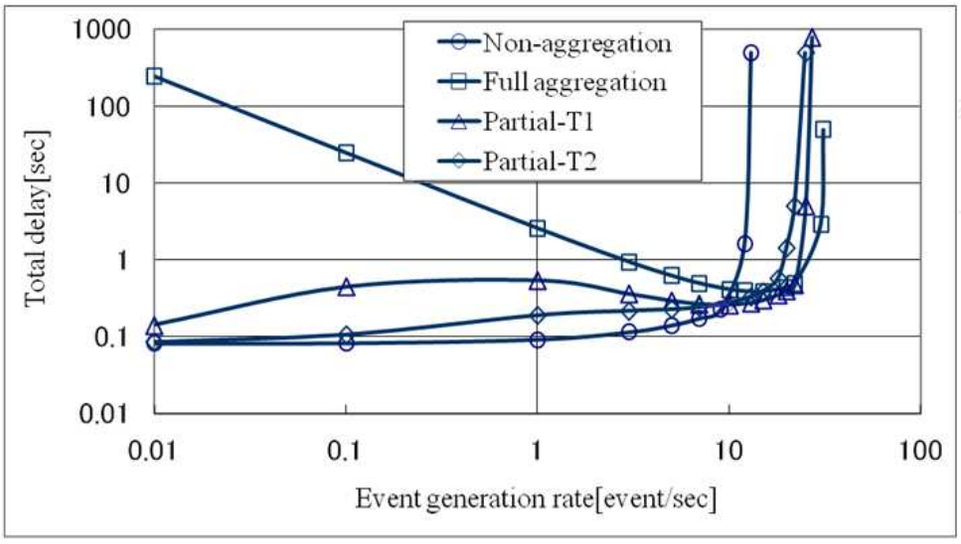


Figure 7. Total delay

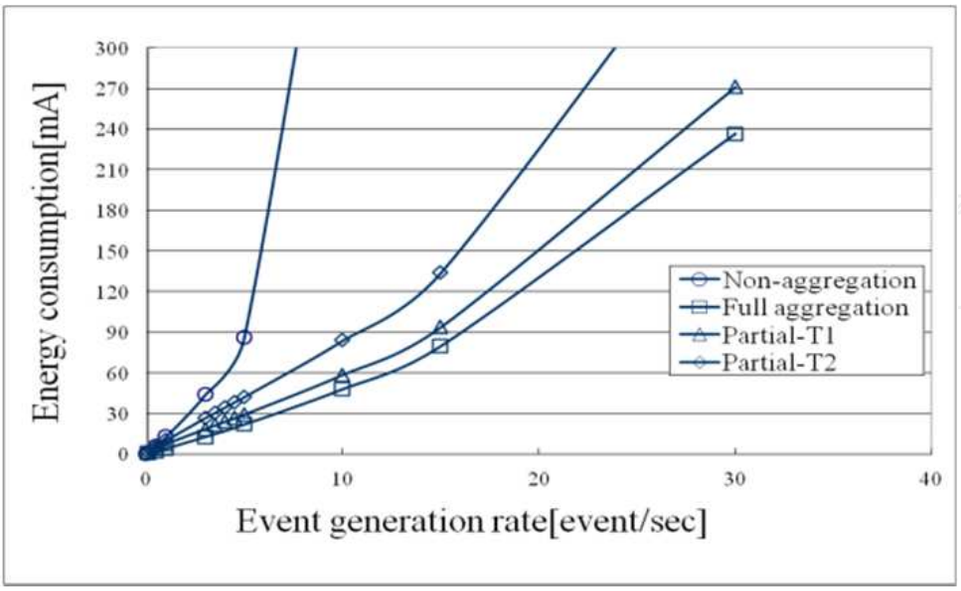


Figure 8. Energy consumption

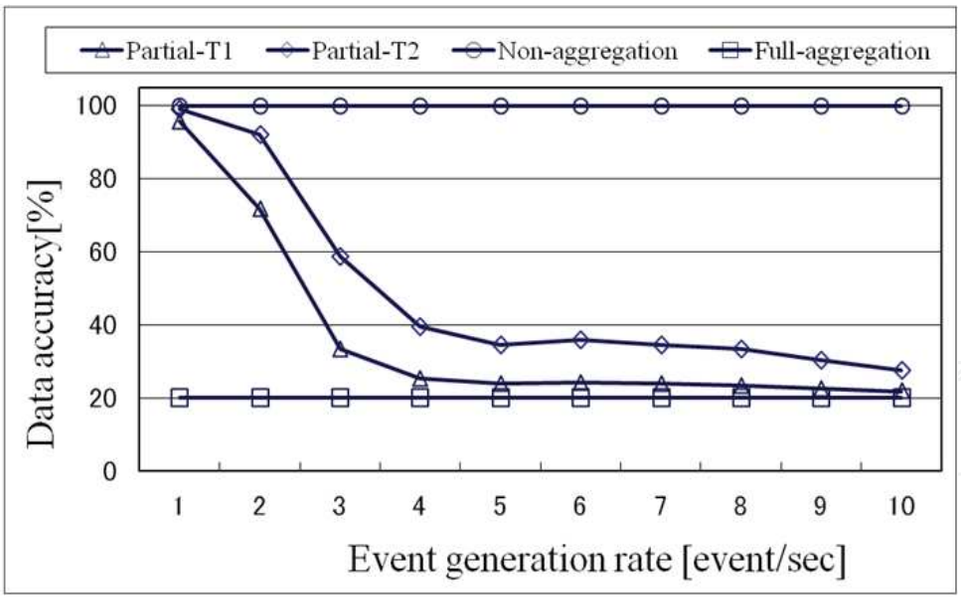


Figure 9. Data accuracy

From above evaluations we find that the partial aggregation with random pushing rate vectors can control the energy, delay and data accuracy between non-aggregation and full aggregation. Hence, one can achieve desired MSRN by controlling the random pushing rate.

4. Tradeoffs among accuracy, energy and delay

4.1. Trade off index TOI

Previous section clearly shows partial aggregation with random pushing rate λ_i^D can control the energy consumption, transmission delay and data accuracy. In MRSN, according to applications, delay taken to collect data, energy consumed by each sensor node for communication and data accuracy of the collected data are critical concerns and are in trade-off each other. Energy, delay and accuracy cannot reach full potential at the same time, but we can achieve the best possible tradeoff between them. To obtain the best trade-off value of practical application, we propose a Trade-Off Index (TOI). In the following subsections, we discuss energy, delay and accuracy of trade-offs in respect of TOI as criteria. Here E denotes total energy consumption, D denotes total delay, Ac denotes data accuracy. α, β, γ indicate the significance of accuracy, energy and delay and larger α, β, γ indicate more significance of energy, delay and accuracy. The smallest TOI value denotes the best data aggregation.

$$TOI = \frac{E^{\beta} * D^{\gamma}}{A_c^{\alpha}} \tag{26}$$

4.2. Applications of WSNs with different criteria

In MRSN, according to the different applications and objectives, we need different significances for transmission delay, energy consumption and data accuracy. Some application areas need to save energy because it is impossible to replace or recharge the battery. In some applications not only the energy is significant, but also the data freshness, such as in military monitoring and disaster monitoring; however data accuracy is most important in medical utilization and in quality control. According to real application, we formulate some of the applications according to the significances of energy, data accuracy and transmission delay in table 1.

Accuracy	Energy	Delay	Average event generation rate		
			month, day, hour	ten min, min …	sec, ms, µs …
L	S	S	Daily health care, Medical application, Survey, Home Automation, Industrial Applications		
S	L	S	Agriculture Application, Nature monitoring, River pollution detecting		
S	S	L	Outer space monitoring, Architectural, Flood detection, Meteorological system		

Table 1. Applications of MRSN

Here the "L" denotes large significance and "S" denotes small significance; the application is formed from left to right along a scale from smaller event generation rate to bigger generation rate. According to the table 1 we can decide the significant parameters of the application in order to perform our proposed TOI; we can achieve the best data aggregation corresponding to the applications.

4.3. Tradeoffs of different applications

In this section, we will investigate the tradeoffs among the applications of which data generation rate is in the range of 0.0001 to 100 events in per second, and here for corresponding to the event generation rate, we define the random pushing rate vectors as the same with event generation rate. We define the random pushing rate vectors as below:

As data generation rate of λ=0.0001, T= [0.0001, 0.0001, 0.0001, 0.0001, 0.0001],

As λ=0.001, set T= [0.001, 0.001, 0.001, 0.001, 0.001],
As λ= 0.01, set T= [0.01, 0.01, 0.01, 0.01, 0.01],
As λ= 0.1, set T= [0.1, 0.1, 0.1, 0.1, 0.1],
As λ= 1, set T= [1, 1, 1, 1, 1],
As λ= 10, set T= [10, 10, 10, 10, 10].

4.3.1. Accuracy significant networks

In accuracy significant utilization, we define α, β and γ as 2, 1, 1; however if the data accuracy is much more important than other two, we also can define α=3 or much larger. In this research, for simplify, we discuss none other but the case that significant parameter has the significance vector of 2 and the ordinary parameters are 1. According to TOI we can get the best result in fig. 10.

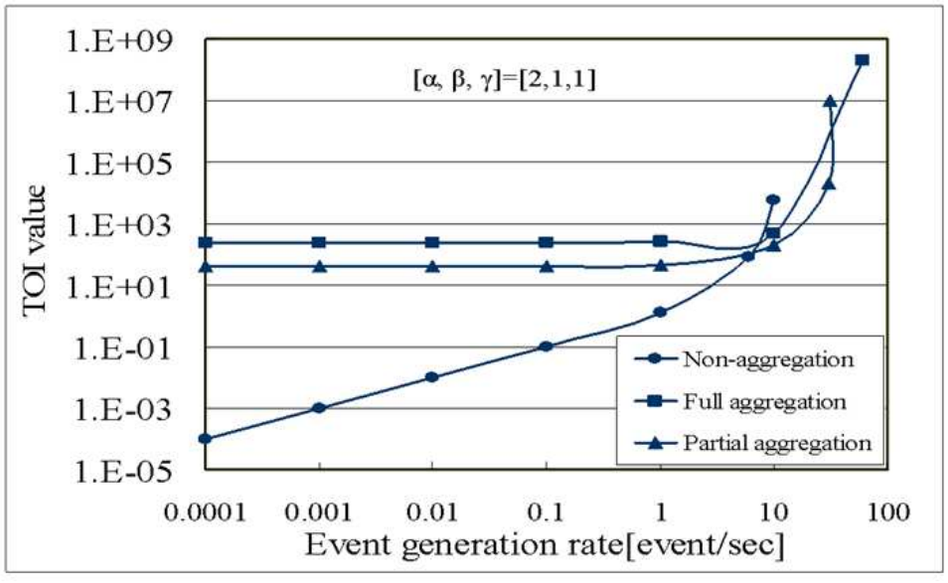


Figure 10. Tradeoffs of accuracy significant

From Fig. 10 we find that when the event generation rate is between 0.0001 and 4.0, non-aggregation is the best comparing with full and partial aggregation. When the data generation rate is between 4 and 30, the partial aggregation is the best, and the full aggregation is the best when data generation rate is larger than 30.

4.3.2. Energy significant networks

Here we discuss the case when energy is significant. The parameters are defined to be as below: α=1, β=2 and γ=1. According to proposed TOI we can get the best TOI values when data generation rate is from 0.0001 to 100.

Fig. 11 shows the result. We find from the figure that in the region of data generation rate between 0.0001 and 4.0, the non-aggregation is the best TOI. When the data generation rate is about 4-10, the figure shows that the partial aggregation is the best; the full aggregation is the best when event generation rate is larger than 10.

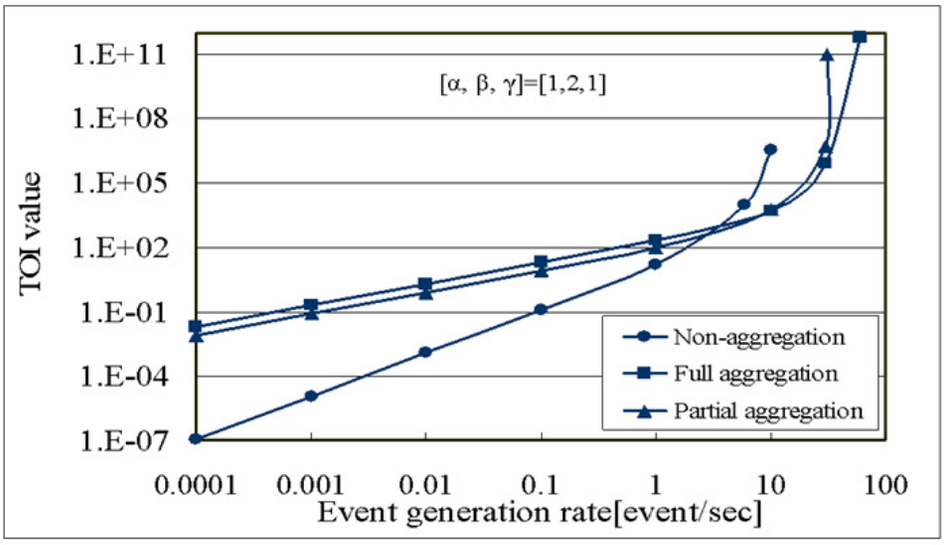


Figure 11. Tradeoffs of energy significant

4.3.3. Delay significant networks

In delay significant networks, α, β and γ is defined as 1, 1, 2, as shown in Fig. 12. From the figure we find that when event generation rate is between 0.0001 and 4.0, the non-aggregation is investigated to be the best TOI; and when the event generation rate is from 4 to 30, the partial aggregation is the best; the full aggregation is the best TOI when event generation rate is larger than 30.

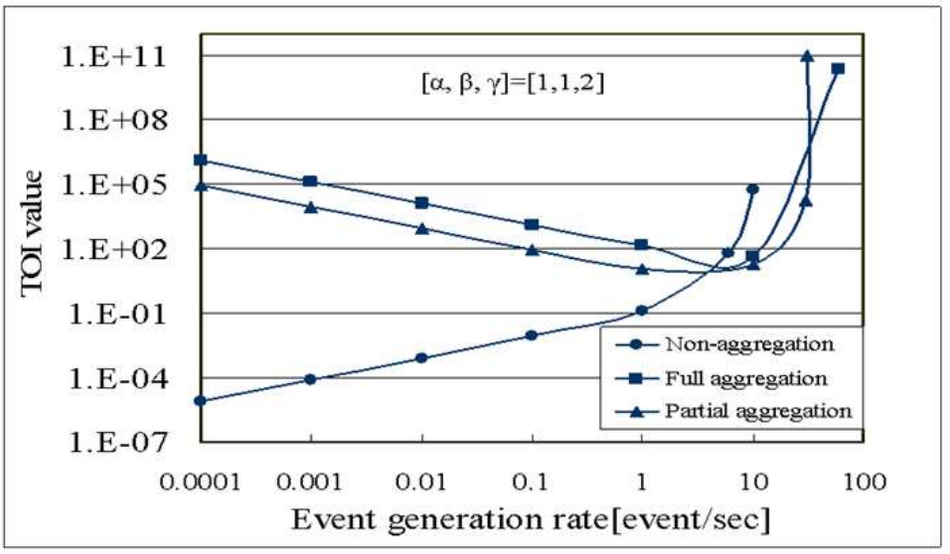


Figure 12. Tradeoffs of delay significant

4.3.4. Discussion

Let us summarize the data aggregation with best TOI according to different event generation rate in table 2. From the table we find that when event generation rate is small (0.0001-4.0, 6.0) the non-aggregation is the best TOI. Moreover, from the figures we find that in accuracy significant networks, the event data range of best TOI at non-aggregation is longer (0.0001-6.0) than any others. This is because, in non-aggregation, the data accuracy is 100%; and the other two have low data accuracy; when event generation rate is larger than 6, non-aggregation has very long delay because of the congestions around the sink node.

When event generation rate is moderate (4, 6-30), the partial aggregation is the best TOI except the case energy significance networks. In energy significance networks, the number of transmission in partial aggregation is much more than full aggregation, so the energy has great impact on partial aggregation with the significant of β=2. When data generation rate is large, due to the large number of transmission, the energy consumption is very high in non-aggregation and partial aggregation; therefore, the best TOI is the full aggregation in the networks with large event generation rate.

Best aggregation / Significance	Non-aggregation	Partial aggregation	Full aggregation
[α, β , γ]= [1, 1, 1]	0.0001-4.0 [event/sec]	4-30 [event/sec]	30-100 [event/sec]
[α, β , γ]=[2, 1, 1]	0.0001-6.0 [event/sec]	6-30 [event/sec]	30-100 [event/sec]
[α, β , γ]=[1, 2, 1]	0.0001-4.0 [event/sec]	4-10 [event/sec]	10-100 [event/sec]
[α, β , γ]=[1, 1, 2]	0.0001-4.0 [event/sec]	4-30 [event/sec]	30-100 [event /sec]

Table 2. The best data aggregation

5. Multi-view multi-robot sensor networks

As we mentioned in the introduction section, applications of the MRSN will be more advanced if multi-cameras are equipped on the robot nodes. The reason is quite similar to human with more eyes. From the point of application, multi-view MRSN can be applied in security system that will not miss a corner. In addition, in the medical application, the multi-view MRSN can accomplish some complex and long time operations. Meanwhile it can achieve more accurate and small cut operation; besides, multi-view MRSN has quick reaction for the vary vital signs and other monitored parameters of the patient.

5.1. Introduction of multi-view video and open problem

The developments of camera and display technologies make recording a single scene with multiple video sequences possible. These multi-view video sequences are taken by closely spaced cameras from different angles. Each video sequence in the multi-view video presents a unique viewpoint of this scene. Therefore, user can switch the viewpoint by playing different video sequences. When a robot is equipped with multi-cameras, it will bring the user who controls the robot a broad perspective. The operator also can switch his viewpoints by playing different video sequences. However, since the multi-view video consists of the video sequences captured by multiple cameras, the traffic of multi-view video is several times larger than conventional multimedia, which brings the dramatic increase in the bandwidth requirement. However, as multi-view video is taken from the same scene, a large amount of inter-view correlation is contained in the video. Therefore, compression transmission technologies are especially important for multi-view video streaming.

The state of the art in multi-view representations includes Multi-View Video Plus Depth (Merkle et, al., 2007), Ray-Space (Smolic, et, al., 2006) and Multi-view Video Coding (MVC) (Vetro, et, al., 2008), (Mueller, et, al., 2006). However, the research on Multi-View Video Plus Depth sequences (Merkle et, al., 2007) suggests that with the addition of depth maps and other auxiliary information, the bandwidth requirements could increase. MVC is issued as an amendment to H.264/MPEG-4 AVC (Vetro, et, al., 2008), (Mueller, et, al., 2006). It was reported that MVC makes more significant compression gains than simulcast coding in which each view is compressed independently. However, even with the MVC, transmission bitrates for multi-view video are still high: about 5 Mbps for 704 × 480, 30fps, and 8 camera sequences with MVC encoding (Kurutepe, et, al. 2007).

5.2. User dependent multi-view video transmission

5.2.1. Switching models

In order to reduce traffic for multi-view video transmission, we have analyzed which frames should be displayed when the viewpoint is switched. Our work mainly focuses on the successive motion model (Pan, et, al., 2011, 2011). In the successive motion model as shown

by Fig. 13, user is only able to switch to the neighboring views. In other words, if the multi-view video contains the views (1, 2… M), user is just able to switch from any view j to the view j', where max (1, j-1) ≤ j'≤ min(j+1, M). This kind of switching model is used in the applications such as free viewpoint TV and Remote Surgery System in which user's head is tracked to decide which views should be displayed.

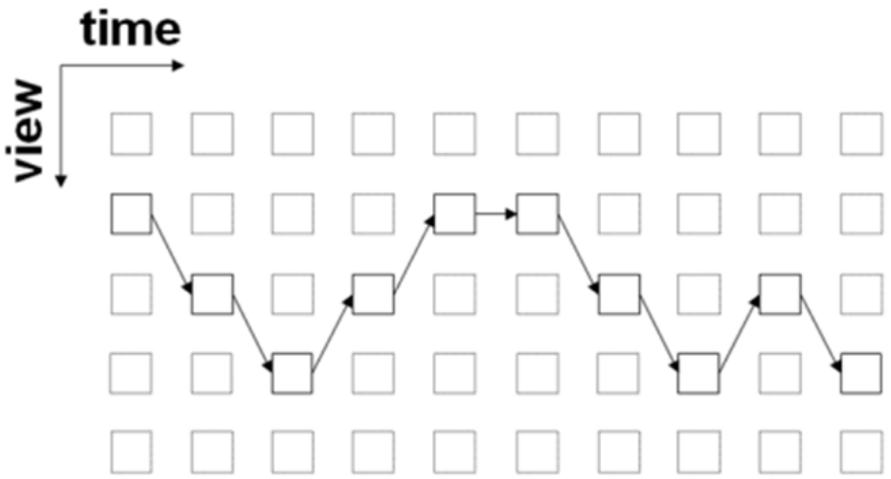


Figure 13. Switching models

5.2.2. User dependent multi-view video transmission (UDMVT)

In (Tanimoto, et. al., 2011), they developed two types of user interface for the Free Viewpoint TV. One showed one view according to the viewpoints given by user. With this type of user interface, the viewpoint of user can be switched by an eye/head-tracking system, moving the mouse of a PC or sliding the finger on the touch panel of a mobile player. In a real-time interactive multi-view video system (Lou, et, al., 2005), users can switch viewpoints by dragging the scroll bar to a different position. In the user interfaces of (Tanimoto, et. al., 2011) and (Lou, et, al., 2005), the changing of user's position, moving of mouse, sliding of finger and dragging of scroll bar are all successive motions. Since the switching models of these user interfaces are all successive motion models, it will take some time to switch from the current view to the neighboring view. For instance, in the head-tracking system, user needs to take some time to move from his current position to the next position for the new viewpoint. We call the speed with which user switch from one view to next view "switching speed." With different user and user interfaces, the switching speed is different. Even the same user may switch to a different switching speed each time.

In the successive motion model, which frames should be displayed when user starts to switch to the next view are decided by both the frame rate f (frame/s) of the multi-view video and the switching speed s (view/s) of user. Let k be the floor of the frame rate divided by switching speed: $k = \lfloor f/s \rfloor$. Fig. 14 presents the display of frames when k is 3, 2 and 1.

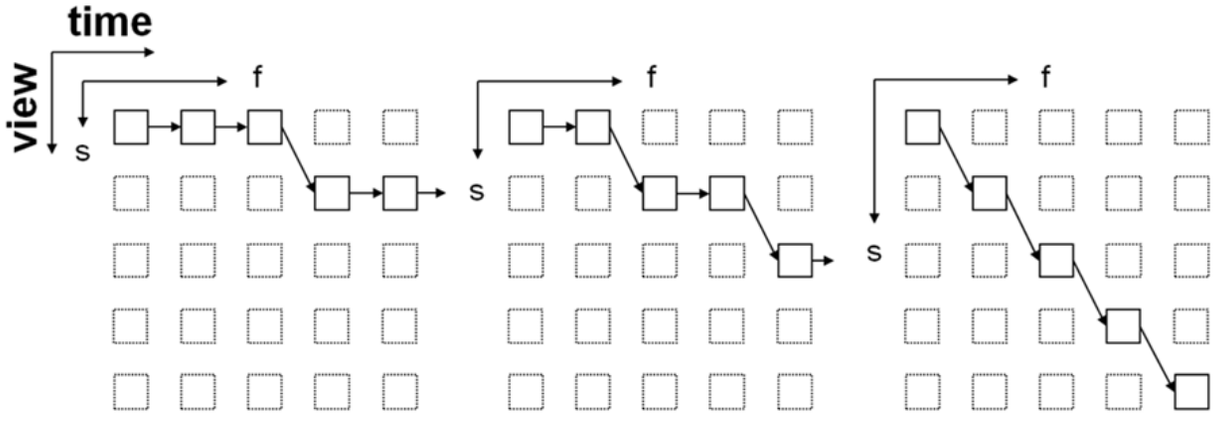


Figure 14. Multi-view video displays with different value of k.

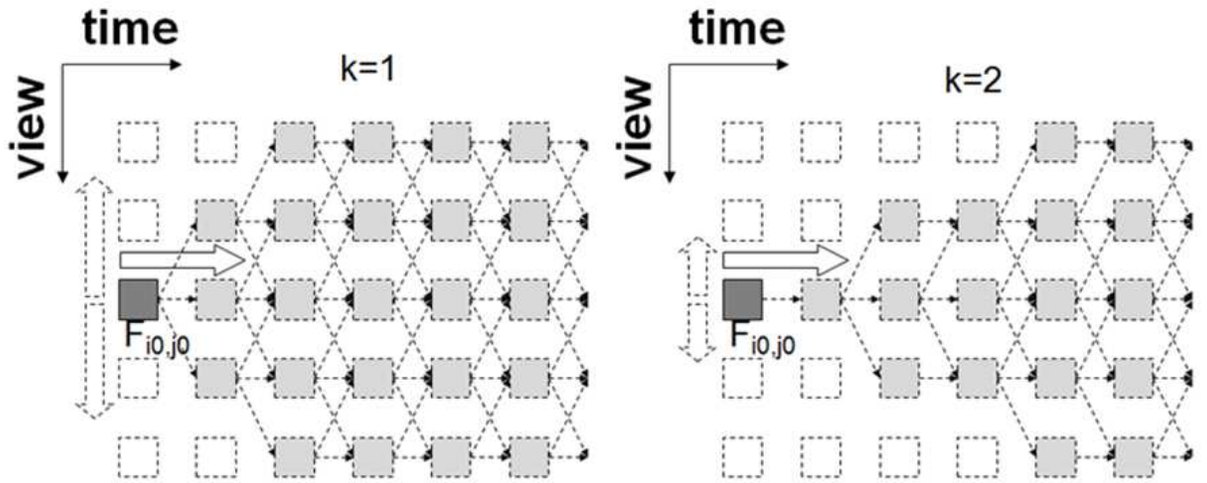
Assuming the frame rates are the same, different frames should be displayed by these three different switching speeds, although they are switching to the same direction. If switching slows down, more frames of the current view should be displayed before the display changes to the next view. Otherwise, less frames of the current view should be displayed. Therefore, k denotes the number of the frames should be displayed in the current view after user starts to switch and before the user reaches the position where display should change to the next view. In practice, the frame rate is about 25~30 (frame/s). The value of switching speed depends on the density of the views and the speed of user interface. However, the switching speed is usually much slower than the frame rate. When the switching speed is about 2~5 (view/s), the k is about 5~15 (frame/view). For simplicity, k=1 and k = 2 are selected as the examples in this paper. Let $F_{i,j}$ denote the frame of view j at time instant i. By the three-tuples N(p, f, s), it is able to predict a triangle area in which the frames are possible to be displayed in a subsequent period of time. p is the current position $F_{i0,j0}$. When the number of the views is M, R(t) is the set of frames that can be displayed at time instant t start from $F_{i0,j0}$. $F_{i,j'}$, R(t), in which:

$$i = i_0 + \lfloor f \times t \rfloor \tag{27}$$

$$j' \in \left[\max\left(1, j_0 - \lfloor s \times t \rfloor\right), \min\left(j_0 + \lfloor s \times t \rfloor, M\right) \right] \tag{28}$$

As the video continues to play, the frame at time instant i in (1) should be displayed starting from $F_{i0, j0}$. User can switch to the view $j_0 - \lfloor s \times t \rfloor$ or $j_0 + \lfloor s \times t \rfloor$ unless already at border view (view 1 or view M) during the period t. The user may also stop switching at any view before switching to view $j_0 - \lfloor s \times t \rfloor$ or $j_0 + \lfloor s \times t \rfloor$. Therefore, it is possible to display frames in view j' shown by (2). The triangles of frames are shown in Fig. 15 when k is 1 and 2, respectively. The frames in the triangle are called potential frames (PFs), which can be switched to and displayed. These frames should be encoded and transmitted. Those frames outside the

triangle are called redundant frames (RFs). It is impossible to display RFs no matter how the user switches the viewpoint start from the current position. UDMVT reduces the transmission bitrate for multi-view video transmission by transmitting only the PFs without RFs.



(a) k = 1 (b) k = 2

Figure 15. The triangles of the frames when k =1 (a) and k=2 (b). The M of the multi-view video is 5. Dotted line represents the possible display path.

When the length of the triangle is L, the number of RFs of the view j in the triangle is:

$$RFs(j) = \min\left(L, I(j)\right)$$

I(j) is:

$$I(j) = |j - j_0| \times k = |j - j_0| \times \lfloor f / s \rfloor$$

In I(j), $|j - j_0|$ is the distant between the view j and the current view j_0. The number of RFs in each triangle is:

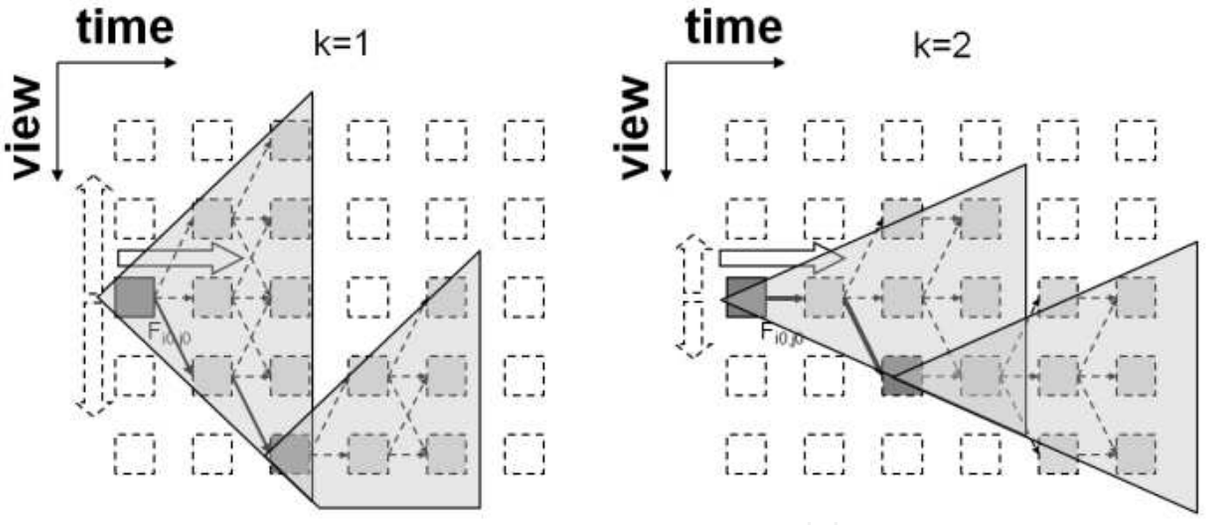
$$\sum_{j=1}^{M} RFs(j) = \sum_{j=1}^{M} \min\left(L, I(j)\right)$$

So the ratio of PFs to RFs is:

$$\frac{M \times L - \sum_{j=1}^{M} RFs(j)}{\sum_{j=1}^{M} RFs(j)} \tag{29}$$

From these expressions, it could be found that with the increase in the length L, the ratio of the PFs to RFs increases, which means that more frames should be encoded and transmitted. In other words, the triangle will be enlarged and finally all the frames at the same time instant are involved into the triangle, which is also shown in Fig. 15.

In order to overcome this problem, the N(p, f, s) should be fed back periodically, which is able to divide a large triangle into many smaller triangles as shown in Fig. 16. In the UDMVT, the N(p, f, s) is fed back periodically at the end of the triangle. The fed back N(p, f, s) from the end of the previous triangle is used to predict the next triangle. Therefore, only potential frames are transmitted each time and the transmission bitrate is reduced. N(f, p, s) should be detected at client and fed back periodically. At the server, N(p, f, s) is used to divide the frames into PFs and RFs. The transmission bitrate can be reduce by only transmitting the PFs and ignore the RFs. Although the transmission of RFs is unnecessary, encoding and transmitting the RFs can work as a kind of insurance against some special situations, such as the switching detection error.



(a) k = 1 (b) k = 2

Figure 16. The triangles of the potential frames. Dotted line represents the possible display path while solid line represents the actual display.

6. Conclusions and future work

6.1. Conclusion

In this paper, at first, we analyzed the conventional non-aggregation, full aggregation, and our proposed partial aggregation with Markovian chain. The analytical result showed that, conventional method suffers large energy consumption with the highest accuracy, while full aggregation suffers long transmission delay, with the least accuracy. However, our proposed partial aggregation has the energy, delay and data accuracy between non-aggregation and full aggregation. When the random pushing rate becomes larger, the partial aggregation tends to non-aggregation and it tends to full aggregation with large random pushing rate. Hence, we find that the partial aggregation can trade off energy, delay and accuracy according to different applications. Secondly, we discussed the tradeoffs among data accuracy, transmission delay and energy consumption with different significances according to different applications by proposing tradeoff index (TOI). From the results, we find that non-aggregation has the best TOI for low event generation rate, that the partial aggregation does for moderate event generation rate, and that the full aggregation does for large event generation rate. At last, we discussed multi-view multi-robot sensor network from the viewpoint of potential applications, existing schemes and our proposed UDMVT.

6.2. Future work

For the future work, at first, we will discuss the random pushing rate to adapt the various changes of data generation rate and information content. For example, in an MRSN, when it is of the state of affairs, nodes generate much more event data than in normal case, that means data generation rate becomes larger. In this case we should decrease the random pushing rate to control the amount of data transmission. On the other hand, from the view point of information entropy, if the self information of generated data is high, it means the generated data are rare generating data. However, when a node applies the normal data aggregation and aggregates the data with normal data, the aggregated data cannot reflect the real situation which may lead bad result. In this case, we can increase the random pushing rate to send high self information data immediately without data aggregation. When the self information of generated data decreases, we decrease random pushing rate to control the quantity of data transmission. Secondly, in wireless sensor network, data are transmitted to sink node by multi-hopping way, which causes the uneven energy consumption on nodes at different locations. Hence, to keep all nodes in the network having the same energy consumption is our another future work.

Author details

Wuyungerile Li, Ziyuan Pan and Takashi Watanabe
Shizuoka University, Japan

7. References

Boulis, A., Ganeriwal, S. & Srivastava, M. (2003), Aggregation in sensor networks: an energy–accuracy trade-off. IEEE 1st Int'l WKsp. Sensor Network Protocols and Applications, USA, May 2003.

Chen H., Mineno H. & Mizuno H. (2008). Adaptive data aggregation scheme in clustered wireless sensor networks. Computer Communications, Vol 31, No. 12, (September 2008) pp. 3579–3585

Cheung G., Ortega A. & Cheung N. (2009). Bandwidth-Efficient Interactive Multiview Live Video Streaming using Redundant Frame Structures. Proceedings of 2009 APSIPA Annual Summit and Conference, Sapporo, Japan, October 2009.

Eu Z., H. Tan & Seah W. (2010). Opportunistic Routing in Wireless Sensor Networks Powered by Ambient Energy Harvesting. Computer Networks (Elsevier), Vol. 54, No. 17, (December 2010), pp. 2943-2966

Galluccio L. & Palazzo S. (2009) End-to-end delay and network lifetime analysis in a wireless sensor network performing data aggregation. Proceedings of the 28th IEEE conference on Global telecommunications, Honolulu, Hawaii, November 2009

Heinzelman W. (2000) Application-Specific Protocol Architectures for Wireless Networks. Massachusetts Institute of Technology, June 2000, Available from http://www-mtl.mit.edu/researchgroups/icsystems/pubs/theses/wendi_phd_2000.pdf

Intanagonwiwat, C., Govindan, R. & Estrin, D. (2000). Directed Diffusion: A Scalable and Robust Communication Paradigm for sensor networks. Proceedings of ACM Mobicom'2000, pp. 55-67, 2000.

Kurutepe E., Civanlar M. & Tekalp A. (2007) Client-driven selective streaming of multi-view video for interactive 3DTV, Circuits and Systems for Video Technology, IEEE Transactions on, Vol. 17, No. 11. (29 October 2007), pp. 1558-1565

Leyzx V., Bouraqadiy N., Stinckwichz S., Morarux V. & Doniecy A. (2009) Making Networked Robots Connectivity-Aware. Proceedings of the 2009 IEEE international conference on Robotics and Automation, Kobe, Japan,

Lindsey S. & Raghavendra C (2002) PEGASIS: Power Efficient GAthering in Sensor Information Systems. Aerospace Conference Proceedings, 2002. IEEE In Aerospace Conference Proceedings, IEEE, Vol. 3 (2002), pp. 3-1125-3-1130

Li W., Bandai M. & Watanabe T. (2010) Trade offs Among Delay, Energy and Accuracy of Partial Data Aggregation in Wireless Sensor Networks. Proceedings of the 2010 24th IEEE International Conference on Advanced Information Networking and Applications, Perth, Australia, pp. 917 – 924, April 2010

Lou J., Cai H. & Li J. (2005). A RealTime Interactive MultiView Video System. Proceedings of ACM international conference on Multimedia, 2005

Maxim A. & Gaurav S. (2005) Sensor network-mediated multi-robot task allocation. The Third International Multi-Robot Systems Workshop, pp. 27-38, March, 2005.

Mirian F. & Sabaei M. (2009) A Delay and Accuracy Sensitive Data Aggregation Structure in Wireless Sensor Networks. 2009 International Conference on In-formation Management and Engineering, Kuala Lumpur, Malaysia, 2009

Mueller K., Merkle P., Schwarz H., Hinz T., Smolic A., Oelbaum T. & Wiegand T. (2006). Multi-view video coding based on H.264/AVC using hierarchical B-frames. Proceedings of the Picture Coding Symposium 2006, CD-ROM: Beijing, April 2006

Merkle P., Smolic A., Müller K. & Wiegand T. (2007) Multi-view video plus depth representation and coding," IEEE International Conference on Image Processing 2007, San Antonio, TX, Sep. 2007

Mi Z., Yang Y. & Liu G. (2010) Connectivity control of mobile multi-robot networks. Proceeding of 2010 IEEE/ASME International conference on advanced intelligent mechatronics, Montreal, Canada, Jul. 2010.

Pan Z., Ikuta Y., Bandai M. & Watanabe T. (2011). A User Dependent System for Multi-view Video Transmission. IEEE International Conference on Advanced Information Networking and Applications (AINA), pp.732-739, Singapore, Mar. 2011

Pan Z., Ikuta Y., Bandai M. & Watanabe T. (2011). User Dependent Scheme for Multi-view Video Transmission. IEEE International Conference on Communications (ICC) 2011, Kyoto of Japan, Jun. 2011

Petrović D., Shah R., Ramchandran K. & Rabaey J. (2003). Data Funneling: Routing with Aggregation and Compression for Wireless Sensor Networks, Proceedings of the First IEEE. 2003 IEEE International Workshop on Sensor Network protocols, pp. 156 – 162, May 2003

Reich J. & Sklar B. (2006). Robot-sensor networks for search and rescue. In IEEE International Workshop on Safety, Security and Rescue Robotics, Gaithersburg, MD, August 2006

Rajagopalan R. & Varshney P. (2006). Data-Aggregation Techniques in Sensor Networks: A Survey. IEEE Communication Surveys and Tutorials, pp. 48-63, Fourth Quarter 2006

Smolic A., Mueller K., Merkle P., Fehn C., Kauff P., Eisert P. & Wiegand T. (2006) 3D video and free viewpoint video technologies, applications and MPEG standards. Proceedings of IEEE International Conference on Multimedia and Expo, July 2006

Sheng W, Yang Q, Tan J & Xi N. (2006) Distributed multi-robot coordination in area exploration. Robotics and Autonomous Systems (2006), Vol. 54 No.12 pp. 945–955

Tanimoto M, Tehrani M, Fujii T & Yendo T (2011). Free-Viewpoint TV. IEEE Signal Processing Magazine, vol. 28, Issue 1, pp. 67-76, Jan. 2011

Trigui S., Koubaa A, Jemaa M, Chaari I & Al-Shalfan K (2012) Coordination in a Multi-Robot Surveillance Application using Wireless Sensor Network. 16th IEEE Mediterranian Electrotechnical Conference, Hammamet (Tunisia), 25-28 March, 2012.

Vetro A, Pandit P, Kimata H, Smolic A & Wang Y. (2008). Joint Draft 8.0 on Multi-view Video Coding. Joint Video Team, Doc. JVT-AB204, Jul. 2008

Ye Z., Abouzeid A. & Ai J. (2009) Optimal stochastic policies for distributed data aggregation in wireless sensor networks (Oct. 2008). IEEE/ACM Trans. on Networking, Vol. 17, No. 5 pp. 1494 - 1507

Yu Y., Krishnamachari B. & Prasanna V. (2004). Energy-Latency Tradeoffs for Data Gathering in Wireless Sensor Networks. IEEE INFOCOM 2004. Twenty-third AnnualJoint Conference of the IEEE Computer and Communications Societies, HongKong, March 2004

IntechOpen

IntechOpen