Data Reading Algorithms for WSNs Railway Monitoring System

Fayez al Fayez*
f.alfayez@mu.edu.sa
Department of computer science and information,
College of science in Zulfi, Majmaah University
Al-Majmaah 11952, Saudi Arabia

Abstract
For the system railways, reading of data sent from sensors to the control unit is a very important issue. Therefore, the impact of data reding latency can cause an accident. In this paper, we propose some algorithms to solve the problem of data reading applied in the railway sensors. It is an information system and communication issue to read the maximum data sent from sensors to the control unit of railway system, hence conduct us to propose algorithms to solve it. The problem of maximizing the data reading is a very hard problem. The proposed algorithm in this paper gives more enhanced results comparing those from literature.

Keywords:
decision maker; railway; algorithm; information system

1. Introduction

Transportation has become one of the most important infrastructure countries as it serves people and drive economy. Recently, speed transportation and on time operations is a necessity. One of the most common used mean of the transportation is trains as its feature moving people and goods from one station to another (in some cases passing by several stations) with a predetermines scheduling and precise time. There are many surveillance systems that focus on assuring safety and timely arrival of train, one of them is sensor network surveillance. Authors of this work focuses on data scheduling of such systems that gather data by sensor from the railway track four instance, (Fire, Fog, Obstacle, Dust, ...); in order to study the effect of such data on the safety and Time of arrival of the specific Journey. Data will be gathered from railway track sensors monitoring systems and video sensor surveillance system in this regard. These dates will be transferred to the control unit of the system dedicated for the railway monitoring in order to examine and stored for further reference. Taking into account time constraints and computing limitations, the large data collected need to be transferred with consideration of resource imitation.

WSNs and the scheduling were studied in , authors determine an suitable metric based on sensor's data to assess the performance of WSN border crossing detection. Work in enhance railway system reliability by introducing a function that balance tracking and tracing real-time monitoring of railway vehicles and allows the utilization of the picked data in real time. Authors in also develop an intelligent mechanism in order to minimize the completion time to achieve improvement performance of institution and organization of Smart Cities Application.

In analytical approach was introduced to adopt an advance scheduling in order to exploit the suitable action for the occasion of irregular behavior. The developed approach can calculate the delay cost in a quantified way.

Work in investigate and classify nine researches focusing on railway and parallel computing. Research of justify the importance of parallel computing to enhance computing and data delivery and railway monitoring applications. The developed heuristic algorithms were examined and tested for parallel machine problem. To the authors best of knowledge, the heuristic methods never been applied before to the railway applications. The performance of developed heuristics algorithms will be examined.

Several recent works treated the railway tracking system , , and .

Authors in proposed for the first time the problem of the optimization of the read frequency of the sensors data sent by sensors to the center unit to make the intervention in time. In this latter work authors developed 6 heuristics to solve the problem, but all heuristics given in this work are based on the same method which is assume that there a separation between processes and the execution periods. The comparison between these heuristics given that result the heuristic based on the knap sack problem is the bets one.

The rest of this paper will be structured as follows; the first section will be devoted to the problem description. In section 3, the reading data algorithm will be presented and developed. In section 4 is reserved for the experimental results. In the end of the paper a conclusion and some future work will be given in section 5.

2. Problem Description

It is essential for rail surveillance system to detect irregularity and spot any irregular objects that on or close by rail tracks through train journey from departure point till

Manuscript received August 5, 2020
Manuscript revised August 20, 2020

arrival. Irregularity can be defined as one and not exclusive of the following: any type of obstacles or object on a rail tracks , animals, obstacles, turnout of a railway tracks , etc. The challenge arise is how to identify such barriers taking into account time factor, and reporting to control unit in order to send a personal to address the problem assuring safe journey without cancellation or face scheduling. This can be done after sensor collecting data and reporting to their control unit to analyze the gathered data. Let's denote N by the number of sensors; the sensor index will be denoted by i; S_i will be the sensor; t_s is a journey departure time; t_a is the journey arrival time; $freq$ is the frequency of sending the sensor's data (seconds). The summation of sensor collected data S_i when a train arrived at its final station is: $N_{rd} = (t_a - t_s) \times freq$. In such a system, sensor nodes transfer it's collected data to the control unit.

The control unit will process this data and compared it with the initial value recorded $H_0(S_i)$ for each sensor. $H_0(S_i)$ symbolize the starting ideal values before any change take place to the railway track. In order to detect any barriers or turnouts it is essential to compare the current value with the recorded value $H_0(S_i)$ and $H_t(S_i)$ at time t for the sensor (S_i). The aim of this comparison is to provide a difference between the starting values with the values gathered by sensor S_i at time t. The variation vector is denoted by $V_t^s(S_i)$. For every time t all reported variation values of $V_t^s(S_i)$ will be recorded in table DC (Data Central) in control unit database for analysis and a future comparison.

If the variation between the initial vector and the vector at time $\boldsymbol{t}$ is 0, then the control unit don't have any process to do. The problem persists if there a difference between vectors. At this time several processes will be executed. Each sensor putted on the railway will send several data to control unit. If the control unit detect an abnormal track (maintenance, sand, turnout, obstacle…). The total estimated time to call the concerned service related to the sensor $\boldsymbol{S_i}$ is denoted by $\boldsymbol{p_i}$. The data sent at the period $\boldsymbol{j}$.
The studied problem in this research work is based essentially on the problem proposed in .

The maximization of the read frequency is the objective of the paper. We propose an algorithm that give results more better that results in . Maximize the read frequency is equivalent to minimize the $\boldsymbol{C^j_{max}}$. $\boldsymbol{C^j_{max}}$ is the makspan given in the period $\boldsymbol{j}$.

Example 1:
Assume that we have 3 processors. Table 1 gives the processing times of all processes by sensors.

TABLE I. INSTANCE FOR EXAMPLE 1

$Sensor$	1	2	3	4	5	6	7
$j = 1$	30	16	20	40	0	0	0
$j = 2$	0	0	0	25	12	13	9
$j = 3$	0	0	0	0	0	15	42

Applying a given algorithm, we obtain the schedule illustrated in Figure 1.

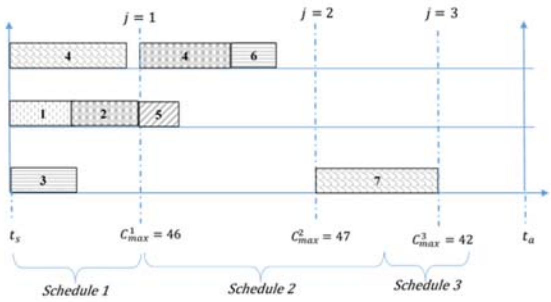

Figure 1. Scheduling for Example 1

The problem is how to find a schedule that minimize all C^j_{max} with $j \in \{1,2,3\}$. From Figure 1 above, we observe that to complete all processes we have to spend a global time denoted by $C^g_{max} = C^1_{max} + C^2_{max} + C^3_{max}$. For this example, $C^g_{max} = 135$.

If we find a schedule that minimize the $\boldsymbol{C^1_{max}}$ this will give a better solution than given in Figure 1. The enhanced solution is given in Figure 2.

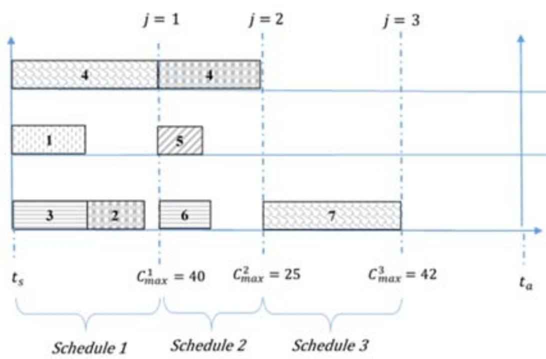

Figure 2. Enhanced algorithm

3. Reading Data Algorithm

As we introduce the studied problem in Example 1, the enhanced algorithm is obtained after resolution of different scheduling problem in each period time j.

For the first period, we apply an algorithm to obtain the first completion time, the second period we solve the problem to find the second solution and so on until reaching the time t_a. At each period j we have a fixed set of processes to be executed denoted by Ps_j. We denoted by $A()$ the function that apply a specific algorithm to solve the problem in the fixe period.

The algorithm to give the global completion time to ensure the data reading as given below.

We denoted by $SendData(j)$ the function that retrieve the data from sensors in period j.

Assume that $timer()$ is function return the current time.

	Algorithm 1: Calculus of global completion time
0	Initialize $j = 1;\ t = 0$;
1	**While** ($j \leq N_{rd}$ and $t < t_a$)
2	Call $SendData(j)$
3	Determine Ps_j
4	Apply $A(Ps_j)$
5	Calculate C^j_{max}
6	$j = j + 1$
7	$t = timer()$
	EndWhile
8	Calculate $C^g_{max} = \sum_{k=1}^{j-1} C^j_{max}$
9	Return C^g_{max}

In this paper, we propose a method based on algorithm that gives a better results comparing with algorithm developed in . So, the algorithm developed in the latter paper in literature review based on the separated periods and jobs method. Thus, we can't assign any processes until the previous period finish the execution.

In this paper we allow the assignment of processes received in period j to the available places kept in period $j - 1$. This method allows as to optimize available free spaces and minimizing the global completion time.

	Algorithm 2: Enhanced algorithm to calculate C^g_{max}
0	Initialize $j = 1;\ t = 0$;
1	Call $SendData(j)$
2	Determine Ps_j
3	Apply $A(Ps_j)$
4	Determine $Pr^1[\,]$
5	$j = 2$
6	**While** ($j \leq N_{rd}$ and $t < t_a$)
7	Call $SendData(j)$
8	Determine Ps_j
9	Apply $A_{enh}(Ps_j, Pr^{j-1}[\,])$
10	Determine $Pr^j[\,]$
11	$j = j + 1$
12	$t = timer()$
	EndWhile
13	Calculate C^g_{max}
14	Return C^g_{max}

We denoted by $A_{enh}()$ the function that apply a given scheduling algorithm with taking under consideration the assignment obtained in the previous period.

We denote by $Pr^j[\,]$ the table contained the availability of each processor after applying $A_{enh}(Ps_j, Pr^{j-1}[\,])$ for processes in period j.

The C^g_{max} is the completion time after finishing all execution. The Algorithm 2 represent the enhanced method to calculate the global completion time.

4. Experimental results

We give, in this section, the results obtained after application of the proposed algorithm.

In order to give the comparison between the best algorithm given in literature review in and our proposed algorithm, we coded them in C++. The computer used in all our experiments is an intel core i7 personal computer.

$N \in \{2,3,4,5,7,10,12,15\}$: 8 elements

$m \in \{2,4,6,8\}$: 4 elements.

For a fixed number of sensors N, the total number of received data N_{rd} is not independent and randomized. However, the number of received data in DC table is in dependence with the number of sensors. Indeed, if we have 5 sensors, we can't have $N_{rd} = 12$ or $N_{rd} = 13$ because all sensors send in the same time different data. So, the number of received data in DC table is a multiple of the number of sensors. Thus, we have M times of N data received.

Therefore: $N_{rd} = N \times M$, with M is positive integer which represent the multiplicator.

$M \in \{2,3,5,7,10,15,20,50\}$: 8 elements. So, for example if $N = 5$, N_{rd} can be a multiple of 5 and M as this set : $N_{rd} = \{10,15,25,35,50,75,100,250\}$.

So just for $N = 5$ the number of received data can be reach 500, this is just if we choose the max multiplicator 100. So the problem is very hard and with enormous data.

We generate 6 types of classes instances of the processing times. We denote by $U[\ ]$ the uniform distribution.

We propose 6 classes to assess the proposed algorithm as follows:

Class 1: The processing times is $U[1-5]$.

Class 2: The processing times is $U[1-10]$.

Class 3: The processing is $U[5-10]$.

Class 4: The processing times is $U[10-20]$.

Class 5: The processing times is $U[20-30]$.

Class 6: The processing times is $U[1-30]$.

For all classes given above, the experimental results prove that 100% of cases the proposed algorithm is better than the algorithm cited in literature review.

5. Conclusion

In this paper, we treated the algorithm optimization of the read frequency of the data sensors regarding the railway system. The studied problem is very hard to solve and must test several algorithms to compare results. In recent previous work the algorithm is based on the separation of the processes and the executions periods. In this paper, we propose a new method to solve the problem by given a new algorithm. This algorithm prove that the results is very remarkable in 100% of instances. This algorithm can be used to apply several algorithms used in the literature review to be improved.

Acknowledgment

The authors would like to thank the deanship of scientific research and Research Center for engineering and applied sciences, Majmaah University, Saudi Arabia, for their support and encouragement; Project No. R-1441-157.

Reference

[1] al Fayez, Fayez, Loai Kayed B Melhim, and Mahdi Jemmali. "Heuristics to Optimize the Reading of Railway Sensors Data." Paper presented at the 2019 6th International Conference on Control, Decision and Information Technologies (CoDIT), 2019.

[2] Bin, Sheng, and Gengxin Sun. "Optimal Energy Resources Allocation Method of Wireless Sensor Networks for Intelligent Railway Systems." *Sensors* 20, no. 2 (2020): 482.

[3] Burdett, Robert L, and Erhan Kozan. "Performance Profiling for Predictive Train Schedules." *Journal of Rail Transport Planning & Management* 4, no. 4 (2014): 98-114.

[4] Enshaei, Nastaran, Amin Hammad, and Farnoosh Naderkhani. "A Comprehensive Review on Advanced Maintenance Strategies for Smart Railways." In *Handbook of Research on Interdisciplinary Approaches to Decision Making for Sustainable Supply Chains*, 433-57: IGI Global, 2020.

[5] Hammoudeh, Mohammad, Fayez Al-Fayez, Huw Lloyd, Robert Newman, Bamidele Adebisi, Ahcène Bounceur, and Abdelrahman Abuarqoub. "A Wireless Sensor Network Border Monitoring System: Deployment Issues and Routing Protocols." *IEEE Sensors Journal* 17, no. 8 (2017): 2572-82.

[6] Lan, Jinhui, Yaoliang Jiang, Guoliang Fan, Dongyang Yu, and Qi Zhang. "Real-Time Automatic Obstacle Detection Method for Traffic Surveillance in Urban Traffic." *Journal of Signal Processing Systems* 82, no. 3 (2016): 357-71.

[7] Lu, Chen, and Yong Shen. "Research on Location Detection of Railway Track Spike Based on Neural Network." Paper presented at the IOP Conference Series: Materials Science and Engineering, 2020.

[8] Melhim, Loai Kayed B, Mahdi Jemmali, and Mafawez Alharbi. "Intelligent Real-Time Intervention System Applied in Smart City." Paper presented at the 2018 21st Saudi Computer Society National Computer Conference (NCC), 2018.

[9] Qi, Zhiquan, Yingjie Tian, and Yong Shi. "Efficient Railway Tracks Detection and Turnouts Recognition Method Using Hog Features." *Neural Computing and Applications* 23, no. 1 (2013): 245-54.

[10] Ulianov, Cristian, Paul Hyde, and Ramy Shaltout. "Railway Applications for Monitoring and Tracking Systems." In *Sustainable Rail Transport*, 77-91: Springer, 2018.

[11] Wiesmeyr, Christoph, Martin Litzenberger, Markus Waser, Adam Papp, Heinrich Garn, Günther Neunteufel, and Herbert Döller. "Real-Time Train Tracking from Distributed Acoustic Sensing Data." *Applied Sciences* 10, no. 2 (2020): 448.

[12] Wu, Qing, Maksym Spiryagin, Colin Cole, and Tim McSweeney. "Parallel Computing in Railway Research." *International Journal of Rail Transportation* (2018): 1-24.