European Journal of Operational Research 271 (2018) 462–473

Contents lists available at ScienceDirect

European Journal of Operational Research

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

Discrete Optimization

Communication scheduling in data gathering networks of heterogeneous sensors with data compression: Algorithms and empirical experiments

Wenchang Luo[a,b], Boyuan Gu[b], Guohui Lin[b,*]

[a] Faculty of Science, Ningbo University, Ningbo, Zhejiang 315211, China
[b] Department of Computing Science, University of Alberta, Edmonton, Alberta T6G 2E8, Canada

A R T I C L E I N F O

Article history:
Received 12 July 2017
Accepted 22 May 2018
Available online 25 May 2018

Keywords:
Scheduling
Wireless sensor network
Data compression
Algorithm
Performance analysis

A B S T R A C T

We consider a communication scheduling problem to address data compression and data communication together, arising from the data gathering wireless sensor networks with data compression. In the problem, the deployed sensors are heterogeneous, in that the data compression ratios, in terms of size reduction, the compression time, and the compression costs, in terms of energy consumption, on different sensors are different. The bi-objective is to minimize the total compression cost and to minimize the total time to transfer all the data to the base station. The problem reduces to two mono-objective optimization problems in two separate ways: in the *original* problem a time bound is given and the mono-objective is to minimize the total compression cost, and in the *complementary* problem a global compression budget is given and the mono-objective is to minimize the makespan. We present a unified exact algorithm for both of them based on dynamic programming; this exact algorithm is then developed into a fully polynomial time approximation scheme for the complementary problem, and a dual fully polynomial time approximation scheme for the original problem. All these approximation algorithms have been implemented and extensive computational experiments show that they run fast and return the optimal solutions almost all the time.

1. Introduction

As an important way for collecting data, wireless sensor networks (WSNs) have found many applications in environment monitoring, surveillance and other areas (Akyildiz, Su, Sankarasubramaniam, & Cayirci, 2002). A data gathering WSN normally consists of a set of sensors for collecting data and a base station to which all the collected data should be transferred. Since these sensors have limited battery power, memory and processing capability (Akyildiz et al., 2002), it is important to utilize and manage the available resources effectively. Besides adopting specific communication protocols (Ergen & Varaiya, 2010; Kumar & Chauhan, 2011; Shi & Fapojuwo, 2010; Wu, Li, Liu, & Lou, 2010) to improve the network performance, one approach is to design effective scheduling algorithms for communication between the sensors and the base station (Alfieri, Bianco, Brandimarte, & Chiasserini, 2007; Berlińska, 2014; Choi & Robertazzi, 2008; Moges & Robertazzi, 2006; Rossi, Singh, & Sevaux, 2013), and another popular approach is to compress the data collected by the sensors, and thus to decrease the data sizes to shorten their communication time. For compression algorithm design and analysis on data gathering WSNs, we refer the readers to Kimura and Latifi (2005), Luo, Wu, Sun, and Chen (2009), Wang, Tang, Yin, and Li (2012), Xu, Wang, and Wang (2011) and Xiang, Luo, and Rosenberg (2013). We note that compressing data consumes a certain amount of energy and incurs a delay in the data transfer, and the extent of each of which depends on the specs of the sensor, the data size, and the compression algorithm.

To address all the data compression and the data communication issues together in an optimization model, Berlińska (2015) formulated a communication scheduling problem out of the data gathering WSNs with data compression, with its bi-objective to minimize the total compression cost and to minimize the total time (call the *makespan*) to transfer all the data. In such a scheduling problem, one is given a set of m *identical* (or *homogeneous*) sen-

* Corresponding author.
E-mail addresses: wenchang@ualberta.ca (W. Luo), bgu1@ualberta.ca (B. Gu), guohui@ualberta.ca (G. Lin).

https://doi.org/10.1016/j.ejor.2018.05.047
0377-2217/

sors $\mathcal{P} = \{P_1, P_2, \ldots, P_m\}$ and a single base station denoted as S; all the data collected by these sensors, either in the original form or in the compressed form, must be transferred to the base station non-preemptively one after another.

In this paper, we consider a more general scenario in which the deployed sensors are *heterogeneous*, in that the data compression ratios (in terms of size reduction), compression time, and compression costs (in terms of energy consumption) on different sensors are different.

The bi-objective communication scheduling problem reduces to a mono-objective optimization problem in the following ways: one may fix a bound on the makespan and seek to minimize the total compression cost, called the *original problem* and denoted as Compr-OptF; or fix a compression budget and seek to minimize the makespan, called the *complementary problem* and denoted as Compr-OptT.

When the deployed sensors are homogeneous, Berlińska (2015) proved that both Compr-OptF and Compr-OptT are NP-hard. In fact, the author proved the NP-hardness in two very special cases where the collected data has the same size for all sensors and data compression has no cost, and where transferring one unit of data has the same cost for all sensors and data compression is done instantly. Besides, the author proposed an exact algorithm to enumerate all possible subsets of collected data for data compression, in the *minimal change order* (Kreher & Stinson, 1998); the total running time is $O(m2^m)$. Two $O(m^2)$-time greedy heuristics were also proposed by the author, and examined through a series of computational experiments. In addition to the observation that the quality of the computed schedules depends on the experiment parameters, a major conclusion from the computational experiments is that, for Compr-OptF, the compression cost of the solution obtained by the heuristics is usually less than 1.5 times the optimum.

In this paper, the deployed sensors are *heterogeneous*. Realizing that deciding whether the Compr-OptF problem has a feasible solution is already NP-hard (Berlińska, 2015), we first present an $O(m^4/\epsilon)$-time bi-factor $(1+\epsilon, 2)$-approximation algorithm for the Compr-OptF problem, by invoking a *fully polynomial time approximation scheme* (FPTAS) for the *minimal knapsack* problem (Kellerer, Pferschy, & Pisinger, 2004) with an $\epsilon > 0$, where $(1+\epsilon)$ refers to the total compression cost and 2 refers to the makespan. We then present an $O(m^4/\epsilon)$-time 2-approximation algorithm for the Compr-OptT problem, by invoking an FPTAS for the *maximum knapsack* problem (Kellerer & Pferschy, 2004), with a specific $\epsilon > 0$. Using these two approximation algorithms, for the Compr-OptT problem we may compute an upper bound on the optimum makespan, denoted as T, and for the Compr-OptF problem we may compute an upper bound on the optimum compression cost, denoted as F. We then present a unified exact algorithm based on dynamic programming for both Compr-OptT and Compr-OptF problems, with an upper bound F on the optimum compression cost and an upper bound T on the optimum makespan, which are either given or computed. The running time of this dynamic programming algorithm is $O(mT^2F)$, which is pseudo-polynomial. Adopting a sparsing technique, this dynamic programming exact algorithm can be converted into an FPTAS for the Compr-OptT problem, and can be converted into a *dual* FPTAS (Hochbaum & Shmoys, 1987) for the Compr-OptF problem; both FPTAS have running time $O(m^4/\epsilon^3)$, when the (dual) worst-case approximation ratio is $(1+\epsilon)$. Lastly and most importantly, we implement all these approximation algorithms, exact algorithms, and FPTAS to examine their practical performance through computational experiments, and make comparisons against the exact algorithm and the heuristics proposed by Berlińska (2015), in both efficiency (the actual running time) and effectiveness (the quality of the computed solution, the total compression cost or the makespan).

Table 1
Numerical parameters associated with the data point P_i collected by the ith sensor.

Notation	Meaning
P_i	The data point collected by the ith sensor
α_i	The original size of P_i
β_i	The size of P_i if the data point is compressed
r_i	The required time for compressing the data point P_i
f_i	The cost (required energy) for compressing the data point P_i
C_i	The required time for transferring one unit of data from P_i to the base station

2. Problem definition

We consider a general scenario in which the deployed sensors are *heterogeneous*, in that the data compression ratios (in terms of size reduction), compression times, and compression costs (in terms of energy consumption) on different sensors are different.

Let the set of m heterogeneous sensors be denoted as $\mathcal{P} = \{P_1, P_2, \ldots, P_m\}$ and the single base station denoted as S. We call the data collected by the sensor P_i also as the *data point* P_i (Table 1). The original size of the data point P_i is α_i, for each $i = 1, 2, \ldots, m$; if the data point P_i is compressed, the size reduces to $\beta_i (\leq \alpha_i)$, the required compression time is r_i, and the compression cost is f_i, where all these parameters are non-negative integers. We assume without loss of generality that the sensor P_i has sufficient energy to compress its collected data, as otherwise it would be deprived of the option to compress the data. Consequently, the release time of the data point P_i, that is the time the data point is ready for transfer, is 0 if not compressed, or otherwise is r_i. We note that when the sensors are homogeneous as discussed in Berlińska (2015), the ratios β_i/α_i, r_i/α_i and f_i/α_i can be assumed sensor-independent, *i.e.*, they are constants for all sensors. For the data transfer time, the data point P_i needs time C_i for transferring one unit of its data to the base station S, where C_i is also assumed a non-negative integer, for each $i = 1, 2, \ldots, m$. In general, these C_i's are different due to the different specs of the sensors and the varying distances between the sensors and the base station.

The bi-objective communication scheduling problem reduces to two mono-objective optimization problems, formally, as follows:

Problem Compr-OptF: Given m heterogeneous sensors (data points) $\{P_i\}_{1 \leq i \leq m}$ and their parameters $(\alpha_i, \beta_i, r_i, f_i, C_i)_{1 \leq i \leq m}$, and a positive integer T representing the time bound, find a subset of data points to compress such that the total data compression cost is minimized and the data transfer completion time (that is, *makespan*) is no greater than T.

Problem Compr-OptT: Given m heterogeneous sensors (data points) $\{P_i\}_{1 \leq i \leq m}$ and their parameters $(\alpha_i, \beta_i, r_i, f_i, C_i)_{1 \leq i \leq m}$, and a positive integer F representing the compression cost budget, find a subset of data points to compress such that the makespan is minimized and the total data compression cost is within the budget F.

It is easy to observe that the central task in these optimization problems is to determine an optimal subset of data points to compress. In fact, when such a subset of data points to be compressed is determined, then obviously the compression cost is known, so is the minimum makespan. The latter conclusion holds because, when the subset is determined, the size and the release time of each data point are known and the remaining problem is to transfer all the data points to the base station one by one; by treating the data transfer phase as a single machine and the data points as the jobs to be processed on the single machine, the problem reduces to "*scheduling on a single machine with job release times to minimize the makespan*" (denoted as "$1|r_j|C_{\max}$" in three-field notation (Graham, Lawler, Lenstra, & Kan, 1979), where r_j is the release time of the job P_j), for which the *earliest release date* (ERD)